Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2020)
IEEE Xplore Part Number:CFP20K74-ART; ISBN: 978-1-7281-4876-2

Real-Time Object Detection for Visually Challenged People

Sunit Vaidya
Information Technology Department
Sardar Patel Institute of Technology
Mumbai, India
sunit.vaidya1@gmail.com

Naisha Shah
Information Technology Department
Sardar Patel Institute of Technology
Mumbai, India
naisha.shah14@gmail.com

Niti Shah
Information Technology Department
Sardar Patel Institute of Technology
Mumbai, India
nitishah50@gmail.com

Prof. Radha Shankarmani
Information Technology Department
Sardar Patel Institute of Technology
Mumbai, India
radha_shankarmani@spit.ac.in

Abstract— **One of the most important senses for a living is vision. Millions of people living in this world deal with visual impairment. These people encounter difficulties in navigating independently and safely, facing issues in accessing information and communication. The objective of the proposed work is to change the visual world into an audio world by notifying the blind people about the objects in their path. This will help visually impaired people to navigate independently without any external assistance just by using the real-time object detection system. The application uses image processing and machine learning techniques to determine real-time objects through the camera and inform blind people about the object and its location through the audio output. Inability to differentiate between objects has led to many limitations to the existing approach which includes less accuracy and low-performance results. The main objective of the proposed work is to provide good accuracy, best performance results and a viable option for the visually impaired people to make the world a better place for them.**

Keywords - Image Processing, Machine learning, Visually Impaired, Object Detection, YOLO.

I. INTRODUCTION

Blindness, as well as low vision, are conditions where people have a decreased ability to see and visualize the outside world. This reduces their mobility and productivity in completing daily tasks. Blind people usually depend on experience, smart sticks or some other people to help them in walking and avoiding obstacles. They do not have a sense of sight which makes them highly dependent on their memory. Also, they cannot be aware of sudden changes in the surroundings which makes it almost impossible to react to an instantaneous situation. Understanding any of the visual aspects like colour, orientation and depth of an object is not easy. Comprehending a three-dimensional object in a single go requires more time and effort than otherwise.

However, in the recent past, technology has made many advancements for visually impaired human beings. Hands-free devices work completely on the audio input of users. They do not require any visual or touch interaction which work as a boon for them. There are screen readers to help them read the screens on devices.

However, these devices are not enough to make the personal and professional life of sight impaired people easy. They only take audio input and when users want to understand the images of their surroundings or texts, these are not very helpful. Research is still going on how to make mobility as easy as possible without any hurdle or danger on the road. It can be equally helpful indoors where it is effortless to get a rough idea of the place around and search for any item. Hence, here a solution is proposed on how to make them confident while travelling or otherwise. Object identification and detection are done using a camera of Android phones as well as in a web application. As soon as the application launches, it starts capturing a live video stream as an input from the camera. The objects are detected in the frame of the camera along with its approximate position which is conveyed to users via audio output. This will help users gauge the location of the object and the direction he or she should be moving in.

The user will be able to cross the road easily as it will detect the signals and the vehicles coming along with their direction. Detection of lamp posts on the road as well as all other objects is also possible. Identification of small objects like pen, toothbrush, utensils and other such necessary objects can also be very useful to the visually impaired people to carry out their daily activities. This system is aimed at providing a robust and easy to use application to make daily activities of visually impaired people very convenient.

II. LITERATURE SURVEY

There doesn't exist much work related to Real-time object detection to assist visually impaired people. Some relevant works are given in this section.

Real-Time Objects Recognition Approach for Assisting Blind People [1]

This paper proposes a system that helps blind people in recognizing the object around them. They implemented it by using tools like GPS service, Ultrasonic sensors and cameras that are placed on a blind person's glasses. These tools are used to get necessary information around them. They have used a Real-Time approach for detecting the objects around them. The accuracy obtained is 90%. Hence it can be useful for us to detect the objects with that accuracy.

An Embedded Real-Time Object Detection and Measurement of its Size [2]

This system detects objects and their sizes in Real-time video streams using OpenCV software library, Raspberry Camera and Raspberry Pi 3 model. This system thus detects and measures the size but this paper doesn't provide the location of the object.

Object Detection and Identification for Blind People in Video Scene[3]

This system uses the Scale-Invariant Feature Transform algorithm to detect and populate necessary features of each frame in the video stream. They have worked for detection of key-points in fast frame video using SIFT. The key-points of the first frame video is matched with the large dataset of images to identify the object. If it contains the same object in the next frame they will not be detected. Audio file is created for each object detected to notify blind people about the object in video.

Detecting Real Time Object Along with the Moving Direction for Visually Impaired People [4]

In this system, Real-time object detection is performed for visually impaired people. In addition to that the users are also notified of which directions should they move in, in order to avoid any obstructions in their path. As directions are very important for visually impaired people like in escalators, whether it's moving upwards or downwards, Up-stair, down-stair, this proposed system works in any condition except in case of direct sunlight.

III. PROPOSED WORK

A. Description

Android Studio is the most preferred Android app development environment because of its Simple app acceptance procedure, Hardware independence, supporting programming languages and for other mobile app solutions and resources.

Python is the most preferred programming language for developing machine learning models and rapid app development because of its simple and concise codes, extensive selection of libraries and frameworks and platform independence.

OpenCV is a computer vision and machine learning software library used for image processing. OpenCV is the most preferred library for Real-time applications because of its improved computational efficiency and distinguished set of libraries.

TensorFlow is a machine learning framework which provides high performance numerical computations using a combination of machine learning, deep learning and other efficient algorithms. It is preferred due to its better computational graph visualizations, seamless performance, great debugging methods, scalability and pipelining.

Darknet is an open source neural network framework that supports CPU and GPU computation. Darkflow is a network building which allows TensorFlow to build networks using .cfg files and pre-trained weights.

The following are the steps for navigation through the app:

- Open the application on a smartphone.
- On launching the application, the device camera is launched automatically.
- The camera then captures real-time view.
- The real time view is then processed with OpenCV on pressing the button showing 'Start/Stop Yolo'.
- At each instant, detected objects are shown with boxes enclosing the objects, labels and their confidence scores.
- The detected objects and absolute location of the objects w.r.t. to frame are then notified to visually impaired people through voice output.

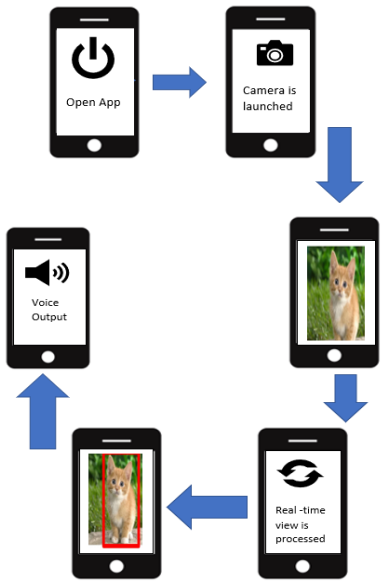


Figure1: Flow Diagram of the project in an Android app

The following are the steps for navigation through the web application:

- Open the web application on a computer or a laptop with a functioning webcam.
- The machine learning model will be invoked by pressing the button 'Start Yolo' which will turn switch-on the webcam.
- The webcam will process the video in real-time
- At every instant, the objects detected in each frame will be indicated in boxes, labels with their respective confidence scores.
- The objects detected will be conveyed to the users through a voice output indicating which object is detected and its absolute location in the frame captured by the webcam.

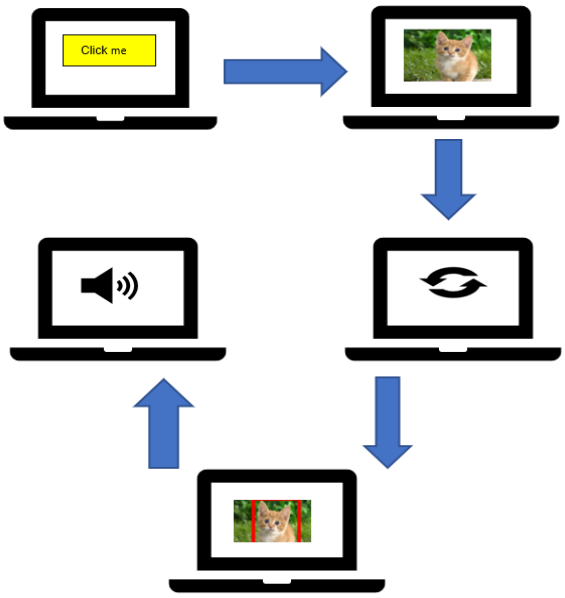


Figure 2: Flow Diagram of the project in a Web application

B. Object Detection Algorithm

For efficient implementation, selection of algorithms is the most crucial part. Thus, different object detection algorithms like R-CNN, Fast R-CNN and YOLO are compared.

R-CNN [5] uses a region based proposed method. R-CNN does not take the whole image, instead, it takes the part of the image that has a higher chance of containing the object. The training time of the network is very large. Moreover, it cannot be used in real time as its speed is very slow, it takes 47 seconds for each image to get detected.

The speed and accuracy of Fast R-CNN [6] are better than R-CNN [5]. In fast R-CNN[5], there is no need to feed 2000 regions every time to the convolution layer; instead, it is passed only once per image which provides a convolutional feature map.

In this system, the YOLO (You Only Look Once) algorithm is used which is the best fit for Real-time detection applications. YOLO, when compared to different detection algorithms, works differently. The YOLO algorithm takes the entire image(frame) in a single instance and processes it. The feature that makes YOLO stand out from other algorithms is its excellent processing speed where it can process 45 frames per second.

Algorithm	Speed	
R-CNN	.05FPS	20s/img
Fast R-CNN	.5FPS	2 s/img
Faster R-CNN	.7FPS	140 ms/img
YOLO	45FPS	22 ms/img

Figure 3: Comparison between different object detection algorithms.

YOLO Algorithm

On investigating YOLOv1[7], YOLOv2[8], YOLOv3[9], evaluated that in terms of computational speed, YOLOv3 is faster and more accurate compared to others. YOLOv3 dataset has 80 classes and 80000 objects. In one image more than 80 different objects can be detected by YOLOv3, bringing the error rate down drastically. In Table I, the input image is predicted in YOLO by dividing it into SxS grid cells. Fixed number of boundary boxes is predicted by each grid cell. The number of boundary boxes predicted by YOLOv3 is more as compared to YOLOv1 and YOLOv2. It is 10 times the number of boundary boxes predicted by YOLOv2. Moreover, YOLOv3 uses a thin-sized boundary box unlike other versions of YOLO which use thick boundary boxes for detection. In Web applications, the YOLOv3 dataset is used. High performance is required for an image to be processed. Therefore, a dataset with a minimum number of objects is needed to run YOLOv3 on any mobile phone device without compromising on the accuracy of detection. Thus, the Tiny Yolo dataset is used for running the object detection algorithm on an android application.

TABLE I.

Yolo Model	Grid cells	Boxes Predicted
YOLOv1	7x7 grid	98 Boxes
YOLOv2	13x13 grid	845 Boxes
YOLOv3	13x13 grid	10x No.of boxes of YOLOV2

IV. IMPLEMENTATION

1) Predicting Bounding Box:

The bounding box can be represented using four descriptors:

- Centre of a bounding box (b_x ,b_y)
- Width (b_w)
- Height (b_y)
- Value c is corresponding to a class of an object (i.e. Person, cell phone, cup, bicycle, etc.)
- Predicted value p_c is a probability of an object in the bounding box.[10]

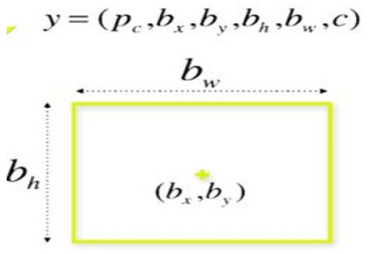


Figure 5: Boundary Box (Indicated for each object)

2) Class Prediction:

For the detection of objects, YOLOv3 uses multi-label classification. Softmax function relies on the theory that classes are mutually exclusive. For eg., when classes like Man and Person in a dataset, the assumption made above fails. Hence, Softmax function is not used in YOLOv3, instead, it simply uses independent logistic classifiers and threshold values to predict multiple labels for an object. During training, the class predictions are done using the binary cross-entropy method instead of the mean square error approach. By avoiding Softmax function the complexity is also reduced.

3) Predictions Across Scales

The predictions are made at three different scales. Each prediction is composed of a boundary box with a value 4, objectness with a value 1 and 80 in the class score as it consists of 80 objects.

i.e. N x N x [3 x (4+ 1+ 80)] predictions.

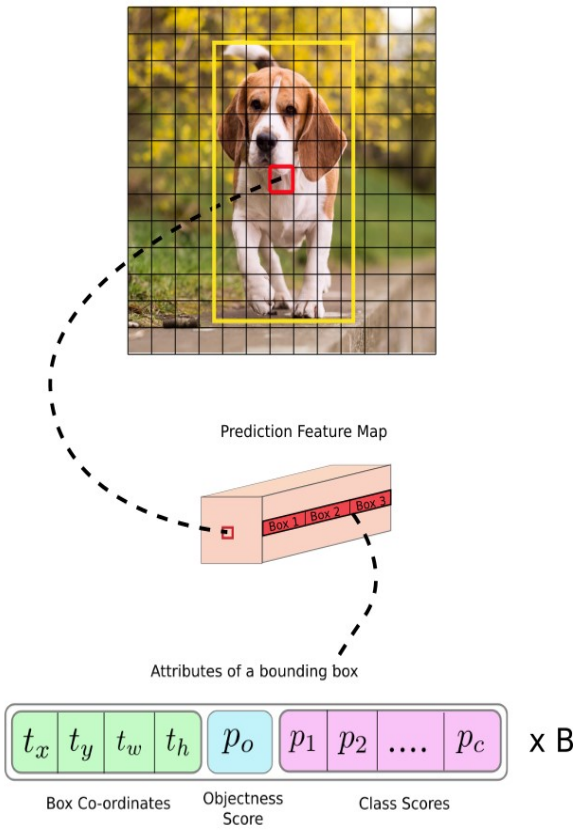


Figure 6: Prediction of an object

4) Feature Extraction

For Feature extraction, YOLOv2[8] uses Darknet-19 which contains only 19 convolutional layers. A new network with a greater number of layers than YOLOv2 is used by YOLOv3[9] i.e. Darknet- 53 with 53 layers is used for feature extraction. Darknet-53 consists of residual networks, same as in ResNet. Darknet 53 is composed of 3x3 and 1x1 filters. Darknet-53 is 2x times faster than ResNet-152. The FPO per second (floating point operations) is the highest of the darknet. Because of this, the Graphical processing unit (GPU) is utilized properly by the Darknet making it faster and efficient.[12]

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

Figure 7: Darknet53 [12]

In this system, the android app uses only the camera, audio output and storage modules of each device to perform the necessary operations. The GPS location data of the mobile device is not used in the application. There is real time object detection performed, where all the objects in the particular frame are detected and indicated using boxes with their labels and confidence scores. As soon as an object comes into the frame, the machine learning model identifies the object. The name of the object, and its absolute position w.r.t. the frame is conveyed to the user via voice output.

Since the processing time between object recognition and audio output is in milliseconds, the user will receive the notification almost immediately without any significant delay.

V. RESULTS

The experimental results prove that the proposed work can help visually impaired people by notifying them of their surrounding objects and absolute location. Experiments were conducted in many different settings and almost all the objects that were present at that time were detected and notified to the user. The average computational time required for detection was 2000 ms. By conducting these experiments, the computational time varies according to the number of objects present is observed. Computational time increases as the number of objects present increases. More than 75% of the objects are detected and recognized accurately at a time.

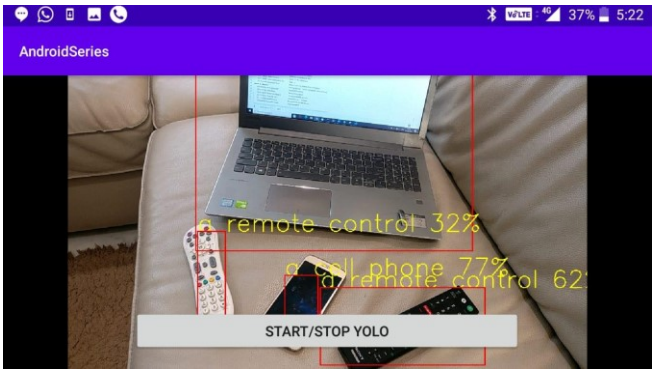
The datasets (Tiny YOLOv3 and YOLOv3) which are used in the proposed solution works very efficiently on the Android app as well as Web App respectively. As shown in Fig. 8, the mAP value for Tiny YOLOv3 is higher than

YOLOv3 indicating that it has better precision in detecting objects in its frame. In small object detection, Tiny YOLOv3 is more efficient in detection compared to YOLOv3 due to its low average loss and high mean average precision.

Currently, there are a total of 80 classes in the Tiny YOLOv3 dataset consisting of a person, traffic light, cow, frisbee, bottle, orange, bed, oven, toothbrush, bicycle, motorbike, cat and dog to name a few and are working on increasing the number of classes detected in the dataset in the future.

Neural Network	Input Resolution	Iterations	Avg loss	Average IoU(%)	mAP (%)
Tiny YOLO v3	416 x 416	42000	0.1983	45.58%	61.19%
Tiny YOLO v3	608 x 608	21700	0.3469	46.38%	61.30%
Tiny YOLO v3	832 x 832	55200	0.2311	48.68%	56.78%
YOLO v3	416 x 416	19800	0.1945	0.15%	0.25%
YOLO v3	608 x 608	2900	0.71	42.62%	23.47%
YOLO v3	832 x 832	5600	0.3324	38.78%	41.21%

Figure 8: Performance Summary for different networks



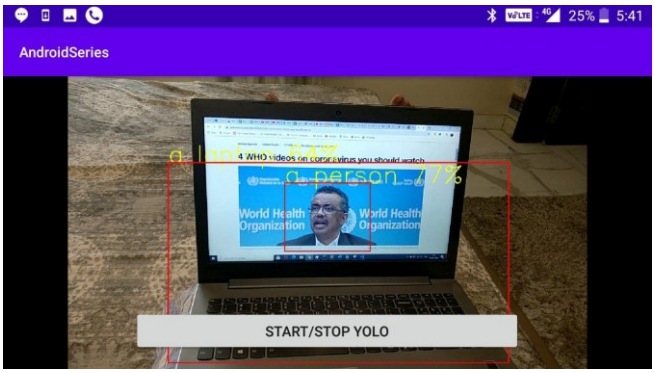


Figure 9: Object Detection on Android Application

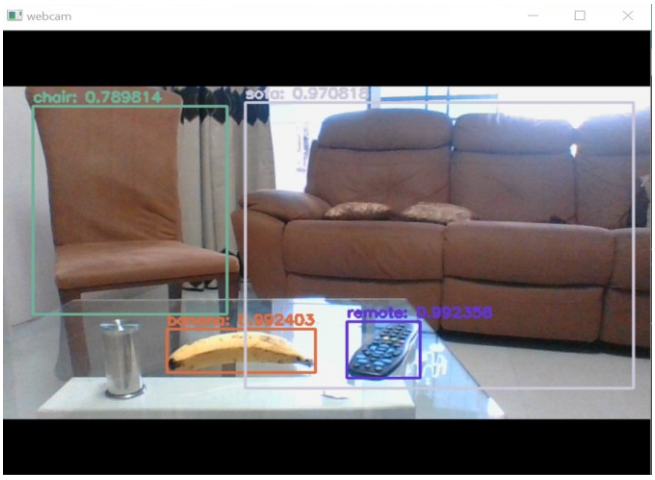


Figure 10: Object Detection on Web Application

VI. CONCLUSION AND FUTURE WORKS

The application has a very simple and easily navigable User Interface that suits the visually impaired users. As soon as the application is launched, the camera will start capturing the real time video. As soon as the user presses a button, the server-side backend algorithm will start processing it and notify the user accordingly as output audio. The Yolo algorithm can be stopped by pressing the same button again. This is how objects around the blind people and their positions are detected and conveyed to them via an audio output using the YOLOv3-tiny algorithm. The software does not require an internet connection for its running and hence, it does not have any such dependency.

The model achieved a maximum accuracy of 85.5% in mobile phones and 89 % in web applications. Though in most of the cases our model works accurately in detecting the different objects it may not work well in cases where the object is too close to the camera or is not a part of the trained dataset. The objects should not be too close to the camera frame and should be placed at a distance more than the focal length of the lens. This algorithm has a low Mean Average Precision for cases where the object is very far and too small to be captured. When the camera is moving at a speed faster than the detection speed of the YOLO algorithm i.e. if the camera is capturing more than 45 frames per second, the model will lag and accuracy of detection will reduce. This includes the scenario when the camera is shaking as that too increases the frames per second (FPS). However, the presence of sound makes no difference on the detection as the microphone module of the mobile device is

not used, so they do not create any difference to the streaming.

The dataset of YOLO weights used for this model is trained for only 80 different types of objects. There are some features like pothole detection, notifying the user in which direction they should move to avoid the obstacle, etc. can be implemented for better usability of the model. However, many new objects can still be added into the dataset.[11] The positions of the objects only have 3 different criteria for height and width which gives 9 different position possibilities. This can be further improved to give a more accurate positioning of the object in the future stages. If the objects are hidden by obstacles in front of them, then they are not captured in the camera and not detected. This will also be considered in future stages. The accuracy of detection in darkness should also be improved. The distance of the object from the camera is also a feature that can be incorporated in the next stage. The text boards or signs that come on the way can also be read out to the user for identification of hoardings or areas. An additional module of the exact location of the user and navigating the user along with detection of the objects on the way can also be added for better convenience and usability.

VII. REFERENCES

[1] Jamal S. Zraqou, Wissam M. AlKhadour and Mohammad Z. Siam Real-Time Objects Recognition Approach for Assisting Blind People 2017 International Journal of Current Engineering and Technology

[2] Naswan Adnman OTHMAN,Mehmet Umut SALUR,Mehmet KARAKÖSE,İlhan AYDIN An Embedded Real-Time Object_ Detection and Measurement of its Size International Conference on Artificial Intelligence and Data Processing (IDAP)2018 , At Turkey

[3] Hanen Jabnoun1, Faouzi Benzarti1, Hamid Amiri Object Detection and Identification for Blind People in Video Scene 2015 15th International Conference on Intelligent Systems Design and Applications (ISDA)

[4] Aniqua Nusrat Zereen, Sonia Corraya Detecting real time object along with the moving direction for visually impaired people 2016 2nd International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE)

[5] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition

[6] Ross Girshick. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision.

[7] Sik-Ho Tsang YOLOv1 — You Only Look Once (Object Detection) https://towardsdatascience.com/yolov1-you-only-look-once-object-detection-e1f3ffec8a89

[8] Rui Li , Jun Yang Improved YOLOv2 Object Detection Model 2018 6th International Conference on Multimedia Computing and Systems (ICMCS)

[9] YOLOv3 explained https://medium.com/analytics-vidhya/yolo-v3-theory-explained-33100f6d193

[10] Joseph Redmon , Santosh Divvala, Ross Girshick, Ali Farhadi You Only Look Once: Unified, Real-Time Object Detection

[11] Anitha.J, Subalaxmi.A, Vijayalakshmi.G Real Time Object Detection for Visually Challenged Persons International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-8 June, 2019

[12] A. Vidyavani , K. Dheeraj, M. Rama Mohan Reddy, KH. Naveen Kumar Object Detection Method Based on YOLOv3 using Deep Learning Networks ISSN: 2278-3075, Volume-9 Issue-1, November 2019