ISSN 2278-3091
Volume 10, No.3, May - June 2021
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse181032021.pdf
https://doi.org/10.30534/ijatcse/2021/181032021

Object Detection and Identification

Prinsi Patel[1], Barkha Bhavsar[2]
[1]Researcher, LDRP Institute of Technology & Research,gandhinagar-382015, Gujarat,India
ssistant Professor, LDRP Institute of Technology & Research, gandhinagar-382015,Gujarat, India

ABSTRACT

Object Detection systems have been growing in the last few years for various applications. Since the hardware can not detect the smallest objects. Many algorithms are used for object detection like Yolo, R-CNN, Fast R-CNN, Faster R-CNN, etc. object detection using YOLO is faster than other algorithms and the YOLO scans the whole image completely at one time. Object detection, which is based on Convolutional Neural Networks (CNNs) and it's based on classification and localization. An object is detected by extracting the features of an object like the color of the object, the texture of the object or shape, or some other features. Then based on these features, objects are classified into many classes and each class is assigned a label. When we subsequently provide an image to the model, it will output many objects it detects, the location of a bounding box that contains every object with their label and score indicates the confidence. Text-To-Speech (TTS) conversion is a computer-based system that requires for the label are converted text-to-speech. The main motive is that the smallest amount of objects can be detected object and labeling the object with voice for real-time object detection. The final model architecture proposed is more accurate and provides the fast result of object detection with voice as compared to previous researches.

Key words: Object Detection, Object Recognition, Text-to-Speech Convert, You Only Look Once(YOLO),CNN, R-CNN.

1. INTRODUCTION

In previous research, there are various algorithm to the detected object with their label. Object detection is the combination of image classification and object localization. In image classification is used for classify or predict the class of specifying the object in an image. In image classification main goal is accurately to identify the feature of an image. In object localization is locate the object on an image with the boundary box. Object detection is highly capable to deal with multi-class classification and localization.

Object detection is a technology that detects the semantic objects of an digital images. object detection is a computer vision technique that allows us to identify and locate objects in an image and accurately labeling with voice and text[1]. With this kind of image identification and object localization, object detection can be used to count objects in a scene and determine and track their precise locations. We can then convert the annotated text into voice responses and give the basic positions of the objects.

Object detection can be broken down into machine learning-based approaches and deep learning-based approaches for object detection and recognition,[1] such as Support vector machine (SVM), Convolutional Neural Networks (CNNs), Regional Convolutional Neural Networks (R-CNNs), You Only Look Once (YOLO) model etc., Since machines cannot detect the objects in an image instantly like humans, it is really necessary for the algorithms to be fast and accurate and to detect the objects in real-time object detection and recognition.

In real world there are many object detection systems available and they are providing such an accurate result too. By this survey, we are trying to detect the smallest amount of object of accuracy and label with voice. Text-To-Speech (TTS) conversion is a computer- based system that divide the two module image processing and voice processing module. In image processing module, optimal character recognition (OCR) has convert the .jpg to .txt format. OCR has recognize the character automatically. In voice processing module has convert .txt to speech.

2. RELATED WORK

In table1 shows the summary of related work.

Table 1: Related work

Paper	Method -Architecture –Datasets	Limitation	Output
Real-Time Object Detection with Yolo[2]	You only look once (YOLO)	The algorithm is simple to build and can be trained directly on a complete image. Region proposal strategies limit the classifier to a particular region.	YOLO accesses to the entire image in predicting boundaries. And also it predicts fewer false positives in background areas. Comparing to other classifier

			algorithms this algorithm is much more efficient and fastest algorithm to use in real time.
You Only Look Once: Unified, Real-Time Object Detection[3]	You only look once(YOLO), Convolution neural network(CNN), PASCAL VOC	YOLO imposes strong spatial constraints on bounding box predictions since each grid cell only predicts two boxes and can only have one class. This spatial constraint limits the number of nearby objects that our model can predict.	single neural network predicts bounding boxes and class probabilities directly from full images in one evaluation. It can be optimized end-to-end directly on detection performance.
Object Detection and Tracking using Tensor Flow[4]	OpenCV, TensorFLOW, CNN, Common object in context(COCO)	This method we will be using Tensor Flow and OpenCV library and CNN algorithm will be used and we will be labelling the detected layers with accuracy being checked at the same time.	It utilize tensorflow to join information from different sources and our joint improvement strategy to prepare all the while on COCO. The dataset measure hole between recognition also characterization.
Implementation of Text to Speech Conversion[5]	Text-To-Speech (TTS), Optical Character Recognition (OCR)	OCR system is implemented for the recognition of capital English character A to Z and number 0 to 9. Each character is recognized at one time.	The recognized character is saved as text in notepad file. In this work a text-to-speech conversion system that can get the text through image and directly input in the computer then speech through that text using MATLAB. It is cost effective user friendly image to speech conversion system

3. PROPOSED METHOD

First, consider the overview od datasets used and workflow of proposed system will be introduced and then overview of datasets used.

3.1 Datasets

Common Object in Context(COCO) is a large-scale object detection, segmentation, and captioning dataset. COCO has several features Object segmentation , Recognition in context330K images (>200K labeled),1.5 million object instances,80 object categories.

This dataset is used for multiple challenges: caption generation, object detection, key point detection and object segmentation. We focus on the COCO object detection challenge consisting in localizing the objects in an image with bounding boxes and categorizing each one of them between 80 categories. The dataset changes each year but usually is composed of more than 1,20,000 images for training and validation, and more than 40,000 images for testing.

3.2 Proposed system

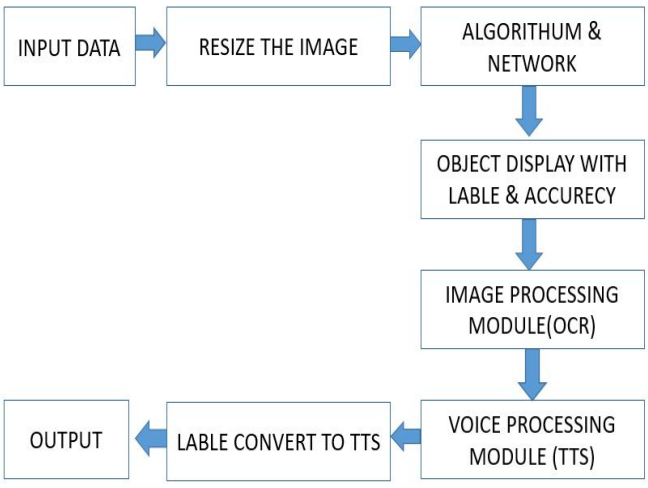


Figure 1: Workflow of proposed system

In figure 1 shows the proposed system workflow.

Input data: We will be using our webcam to feed images at 30 frames-per-second to this trained model and we can set it to only process every other frame to speed things up.

Model: The model here is the You Only Look Once (YOLO) algorithm that runs through a variation of an extremely complex Convolutional Neural Network architecture called the Darknet. Even though we are using a more enhanced and complex YOLO v3 model, I will explain the original YOLO algorithm. Also, the python **cv2** package has a method to setup Darknet from our configurations in the yolov3.cfg file.

Training data: The model is trained with the Common Object in context (COCO) dataset. You can explore the images that they labeled in the link, it's pretty cool.

API: The class prediction of the objects detected in every frame will be a string e.g. "cat". We will also obtain the coordinates of the objects in the image and append the position "top"/"mid"/"bottom" & "left"/"center"/"right" to the class prediction "cat". We can then send the text description to the Google Text-to-Speech API using the **gTTS** package.

Output: We will also obtain the coordinates of the bounding box of every object detected in our frames, overlay the boxes on the objects detected with label and voice.

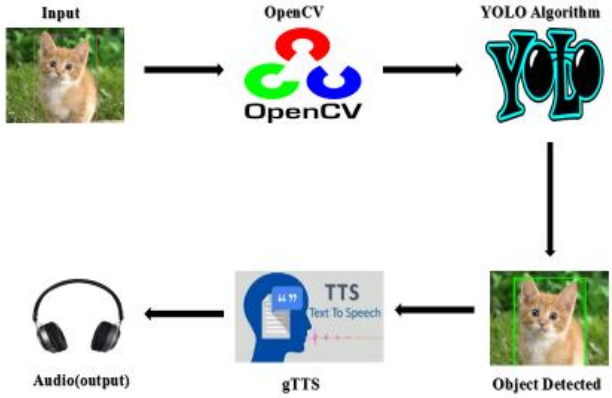


Figure 2: Proposed system

In Figure 2 shows the Proposed system steps followed:

1. First of all we will be using our webcam/image to capture the image as a input data.
2. Input image are resize the according to the network architecture.

In Figure 3 shows the Convolutional neural network to scale back the spatial dimension to 7x7 with 1024 output channels at every location. Convolution neural network has 24 convolutional layers followed by 2 fully-connected layers[2]. Reduction layers with 1x1 filters followed by 3x3 convolutional layers replace the initial inception modules. Most of convolution layer are pretrained using Imagenet Datasets. By using two fully connected layers it performs a linear regression to create a (7, 7, 2) bounding box prediction. Finally, a prediction is made by considering the high confidence score of a box. Convolution network check the every grid separately and marks the label which has an object in it and also mark its boundary boxes.

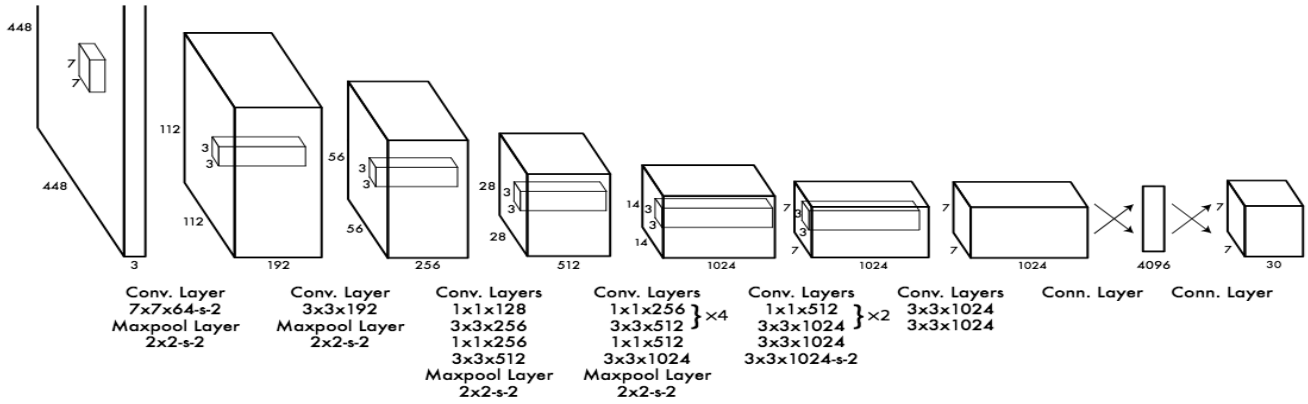


Figure 3: Convolution neural network[2]

3. Apply the YOLO algorithm for detect the multiple object using the trained datasets COCO.
4. The output become a object are detected with label and confidence score.
5. Detected label consider as a image for text-to-speech device. label are pass through the text-to-speech device.

In Figure 4 shows the TTS module has contain 2 parts, first is image processing module(OCR) and second is voice processing module(TTS). In first is image processing module, where OCR converts .jpg to .txt form. Second is voice processing module which converts .txt to speech.

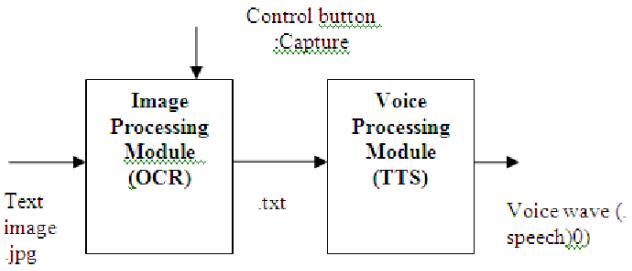


Figure 4: Block diagram of TTS[5]

6. The final output shows the objects are detected with their label and confidence score with voice.

3.3 Confidence Score

To evaluate the performance of an object detection. Normally, we focus on the accuracy (mean average precision, mAP) and consecutive images' average process rate of objects detection per second (Frames per second, FPS). In terms of accuracy, there are many different approaches used to evaluate the accuracy of a model or an algorithm for object detection, but mAP is the primary one. In terms of speed, the FPS is the standard one. Before analyzing it, we, at first, have to understand some basic concepts such as confidence score, IoU, precision, recall and so on for accuracy, and the FPS for the speed of performance.
A basic concept of mAP for an object detector's accuracy evaluation

Confidence score: The reflects the probability that an anchor box contains an object. It is usually predicted by a classifier.

Each grid predicts the boundary boxes with the confidence score. It shows the how accurate the predicted object. We define our confidence score:

$$C = P\,r\,(Object) * IOU^{Truth\ Pridict}$$

There is no object in grid the confidence score should be 0 to 1. If there in object in grid the confidence score should be equal to IOU between ground truth bounding box and predicted bounding box.

Ground truth bounding box (Bgt): represents the desired output of an algorithm on an input, for example, the hand labeled bounding box from the testing set that specify where the objects are in the image.

Predicted bounding box (Bp): represents a rectangle region generated from model detector that indicates the location of the object predicted.

Intersection over union (IoU): an evaluation metric used to measure the area encompassed by both the ground-truth bounding box (Bgt) the predicted bounding box (Bp). In figure 5 shows the equation of IOU.

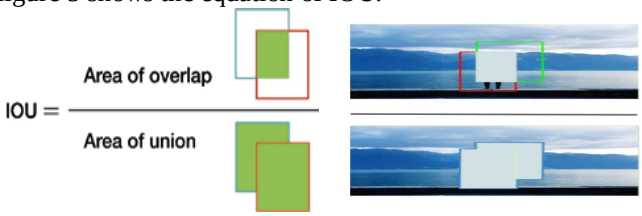


Figure 5: understand the IOU equation[13]

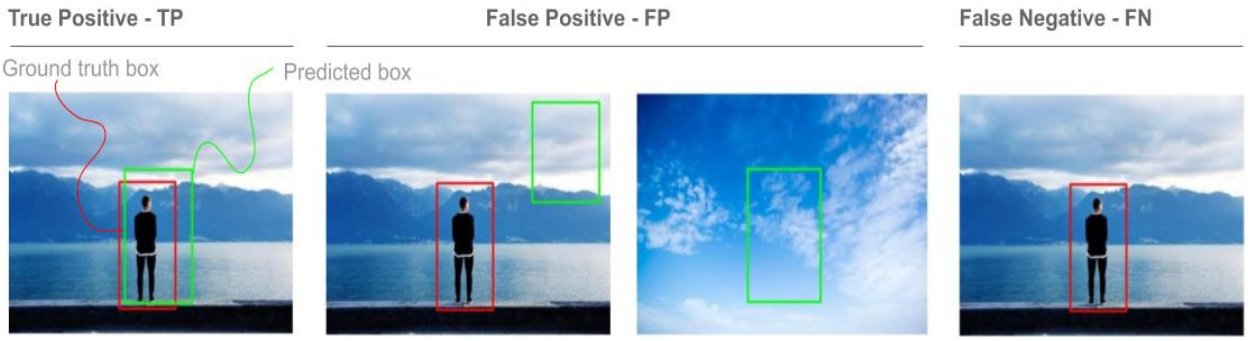


Figure 6: Example of ground truth box and predict box[13]

The figure 6 is an example of detection a person in an image. Based on this image we can have a basic understanding of IoU.

Threshold: we predefine a threshold of IoU (for instance, 0.5) in classifying whether the prediction is a true positive or a false positive.
True Positive: A true positive test result is one that detects the condition when the condition is present.
True Negative: A true negative test result is one that does not detect the condition when the condition is absent.

False positive (FP): A false positive test result is one that detects the condition when the condition is absent.
False Negative (FN): A false negative test result is one that does not detect the condition when the condition is present.

4. IMPLEMENTATION

The main aim of the proposed system is smallest object are detected and the detected object is convert to text to speech. the whole implementation is done in python programming language.

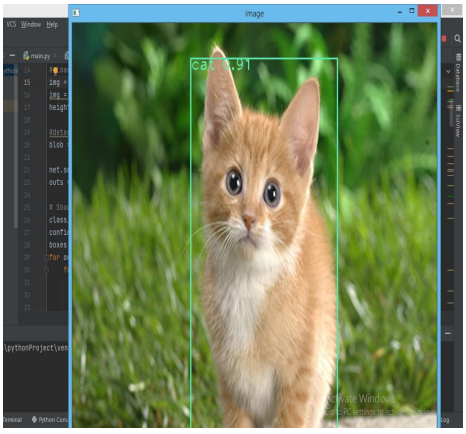
1. Input data image or webcam. camera starts capturing frames with the rate of 45 frames per second to the algorithm.

(a)

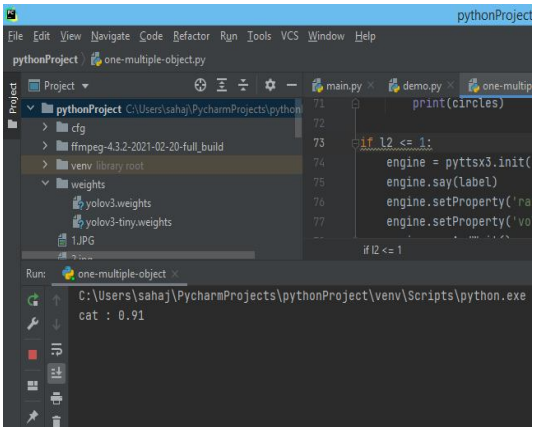
(b)

Figure 7: Input data (a)single object (b) multiple object

2. Resize the image according to network architecture and YOLO Algorithm for object detection and used OpenCV python library.



(a)



(b)

Figure 8: (a) object are detected (b)console result of detected object

3. Object is detected of an image the apply for text-to speech conversion
   - If on object is detect then directly label are convert text –to-speech.

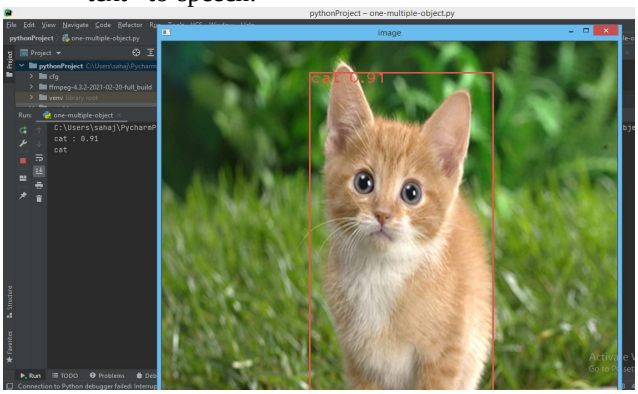


Figure 9: Detected object is convert text-to-speech(TTS)

- If multiple on an image:

- Select a particular object

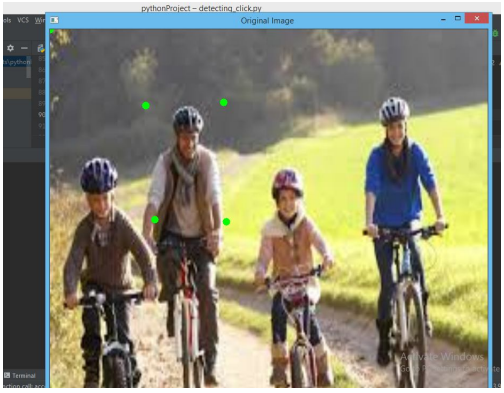


Figure 10: select particular object which has convert TTS

- Object is cropped and assign the label

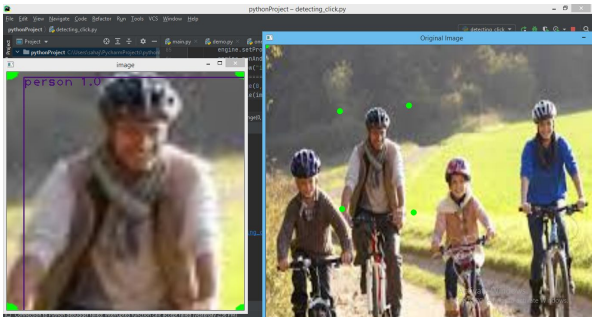


Figure 11: Particulate object are detected with the label

- Label is converted text-to-speech

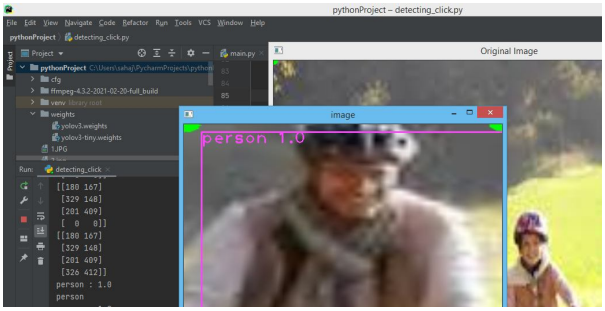


Figure 12: Particular object are convert text-to-speech(TTS)

In Real time object detection:

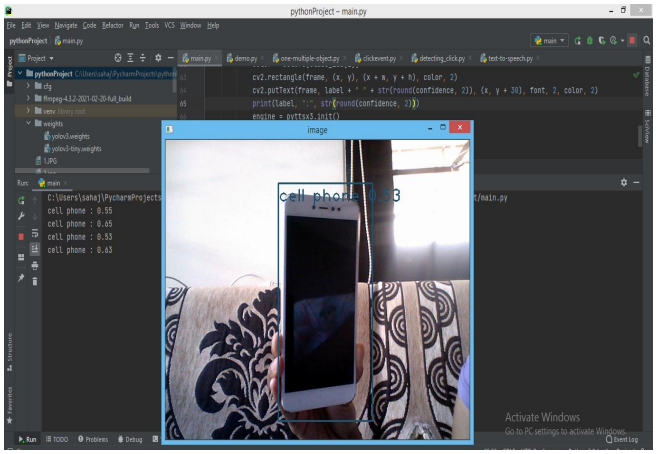
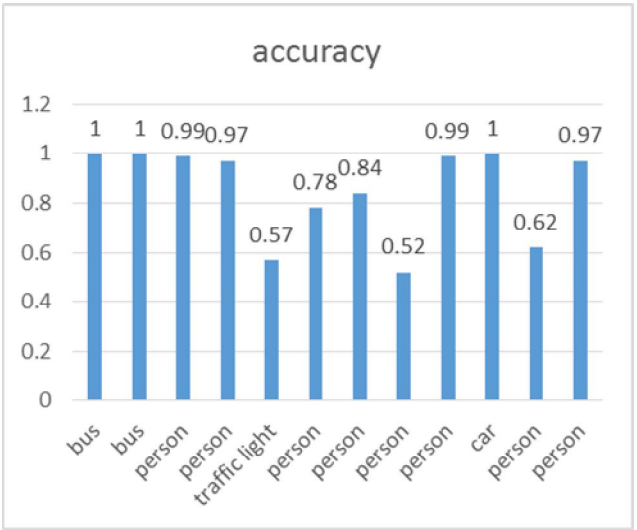


Figure 13: real time capture the object cell phone is detected with the label and accuracy along with voice

5. COMPARISION AND DISCUSSION

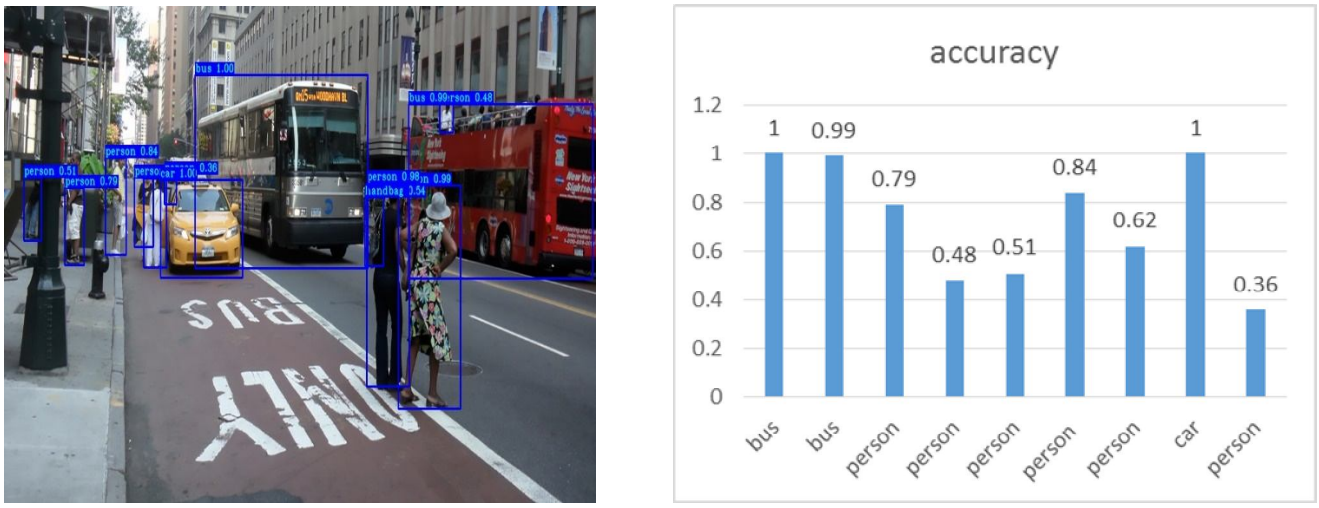
5.1 Comparison of openCV-YOLO and tensorflow-YOLO

(a)



(b)

Figure 14: (a) proposed system output using OpenCV-YOLO (b) proposed system graph of detected object



(a) (b)

Figure 15: (a) proposed system output using Tensorflow-YOLO (b) proposed system graph of detected object

In figure 14 has shows the 13 object is detected with label and accuracy in the image and figure 15 shows the only 9 object is detected in the image.

5.2 Loss function in YOLO

It is used to correctness of the center and boundary box of each prediction. In YOLO predicts multiple bounding boxes per grid cell. To compute the loss for the true positive, we only want one of them to be **responsible** for the object. For this purpose, we select the one with the highest IoU with the ground truth. The loss function defined as follow:

Classification Loss + Localization Loss + Confidence Loss

Classification Loss: If an object is detected in image, the classification loss at each cell is the squared error of the class conditional probabilities for each class[14]:

$$\sum_{i=0}^{s2} 1^{\text{obj}}_{i} \sum_{c \in classes} (p_i(c) - \widehat{p_i}(c))^2$$

where

1^{obj}_{i} = 1 if object appears in cell I, otherwise 0.

$\widehat{p_i}(c)$ denotes the conditional class probability for class c in cell i.

Localization Loss: It measures the errors in the predicted boundary box with locations and sizes. We only count the box responsible for detecting the object[14]:

$$\lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1^{\text{obj}}_{ij} \left[(x_i - \hat{x}_i)^2\right]$$

$$+ \lambda_{coord} \sum_{i\ 0}^{s^2} \sum_{j\ 0}^{B} 1^{\text{obj}}_{ij} \left[\left(\sqrt{w_i} - \sqrt{\widehat{w_i}}\right)^2 + \left(\sqrt{h_i} - \sqrt{\widehat{h_i}}\right)^2\right]$$

Where

1^{obj}_{ij} = 1 if the j th in the boundary box in cell i is a responsible for detecting the object, otherwise 0.

λ_{coord} increase the weight for the loss in the boundary box cooridinates.

Confidence Loss: If an object is detected in the box, the confidence loss (measuring the objectness of the box) is[14]:

$$\sum_{i=0}^{s^2} \sum_{j=0}^{B} 1^{\text{obj}}_{ij} \left[(c_i - \hat{c}_i)^2\right]$$

Where

$\widehat{c_i}$ is the box confidence score of the box j in cell i.

1^{obj}_{ij} = 1 if the j th in the boundary box in cell i is a responsible for detecting the object, otherwise 0.

If an object is not detected in the box, the confidence loss is:[14]

$$\lambda_{noobj} \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1^{\text{noobj}}_{ij} \left[(c_i - \hat{c}_i)^2\right]$$

Where

1^{noobj}_{ij} is a complement of 1^{obj}_{i}

$\widehat{c_i}$ is the box confidence score of the box j in cell i.

λ_{noobj} Weight down the loss when detecting background.

6. CONCLUSION

In this paper, we proposed about YOLO algorithm for the detecting objects using a convolution network layer. It is conclude the accurate results of object detection using YOLO are high compare to others. Object detection with YOLO library which take less time for object detection and highly accurate and also the label are convert text-to-speech(TTS) conversion is fast. In section 2 shows the YOLO algorithm is best for object detection and YOLO has entire image in a single instance and high predict the boundary box co-ordinates and class probabilities of the boxes. The comparison of OpenCV-YOLO and Tensorflow-YOLO shows the proposed system are better than existing system.

ACKNOWLEDGEMENT

We would like to thank the anonymous reviewers for their valuable and insightful comments. We believe their comments significantly improved the quality of this manuscript.

The research activities described in this paper were funded by LDRP Institute of Technology and Research, Gandhinagar, Carloman Systems, Ahmedabad, Gujarat, India.

REFERENCES

1. https://www.researchgate.net/publication/337464355_OBJECT_DETECTION_AND_IDENTIFICATION_A_Project_Report
2. Geethapriya. S, N. Duraimurugan, S.P. Chokkalingam **Real-Time Object Detection with Yolo**, *International Journal of Engineering and Advanced Technology (IJEAT)*, Volume-8, Issue-3S, February 2019
3. Joseph Redmon, Santosh Divvala, Ross Girshick, **You Only Look Once: Unified, Real-Time Object Detection**, *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779-788.
4. R. Sujeetha, Vaibhav Mishra **Object Detection and Tracking using Tensor Flow**, *International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878*, Volume-8, Issue-1, May 2019
5. Chaw Su Thu Thu, Theingi Zin **Implementation of Text to Speech Conversion**, *International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 3 Issue 3*, March – 2014
6. S. Venkateswarlu , D. B. K. Kamesh , J. K. R. Sastry and Radhika Rani **Text to Speech Conversion**, *Indian Journal of Science and Technology, Vol 9(38)*, DOI: 10.17485/ijst/2016/v9i38/102967, October 2016
7. Moonsik Kang **Object Detection System for the Blind with Voice Command and Guidance**, *IEIE Transactions on Smart Processing and Computing*, vol. 8, no. 5, October 2019
8. YOLO Juan Du1,**Understanding of Object Detection Based on CNN Family**, *New Research, and Development Center of Hisense*, Qingdao 266071, China.
9. Fushikida, Katsunobu; Mitome, Yukio; Inoue, Yuji, **A Text to Speech Synthesizer for the Personal Computer**, *IEEE Transactions on vol.CE-28, no.3*, pp.250-256, Aug. 1982 ICCV 2009
10. https://towardsdatascience.com/getting-started-with-coco-dataset-82def99fa0b8
11. https://medium.com/zylapp/review-of-deep-learning-algorithms-for-object-detection-c1f3d437b852
12. https://www.ijeat.org/wp-content/uploads/papers/v8i3S/C11240283S19.pdf
13. https://manalelaidouni.github.io/manalelaidouni.github.io/Evaluating-Object-Detection-Models-Guide-to-Performance-Metrics.html
14. https://jonathan-hui.medium.com/real-time-object-detection-with-yolo-yolov2-28b1b93e2088