Real Time Object Detection with Audio Feedback using Yolo vs. Yolo_v3

Mansi Mahendru[1], Sanjay Kumar Dubey[2]

[1,2] Department of Computer Science and Engineering,
Amity University Uttar Pradesh,Sec-125, Noida (U.P), India

{[1]mansi1997mahendru@gmail.com, [2]skdubey1@amity.edu}

Abstract— **Object recognition is one of the challenging application of computer vision, which has been widely applied in many areas for e.g. autonomous cars, Robotics, Security tracking, Guiding Visually Impaired Peoples etc. With the rapid development of deep learning many algorithms were improving the relationship between video analysis and image understanding. All these algorithms work differently with their network architecture but with the same aim of detecting multiple objects within complex image. Absence of vision impairment restraint the movement of the person in an unfamiliar place and hence it is very essential to take help from our technologies and trained them to guide blind peoples whenever they need. This paper proposes a system that will detect every possible day to day multiple objects on the other hand promt a voice to alert person about the near as well as farthest objects around them. In this paper system is developed using two different algorithms i.e. Yolo and Yolo_v3 and tested under same criterias to measure the accuracy and performance. In Yolo Tensor flow _SSD _Mobile Net model and in Yolo_v3 Dark net model is used. To get the audio Feedback gTTS (Google Text to Speech), python library used to convert statements into audio speech. To play the audio pygame python module is used. Testing of both the algorithms is done on MS-COCO Dataset consist of more than 200 K images. Both the algorithms are analysed using webcam in various situations to measure accuracy of the algorithm in every possibility.**

Keywords— Tensor flow, SSD, Yolo, Yolo_v3, gTTS, Deep Learning.

I. INTRODUCTION

Humans almost by birth are trained by their parents to categorize between various objects as children self is one object. Human Visual System is very accurate and precise that can handle multi-tasks even with less conscious mind. When there is large data then we need more accurate system to correctly recognize and localize multiple objects simultaneously. Here machines comes into existence, we can train our computers with the help of better algorithms to detect multiple objects within the image with high accuracy and preciseness. Object Detection is the most challenging application of computer vision as it require complete understanding of images. In other words object tracker tries to find the presence of object within multiple frames and assigns labels to each object [1]. There might be many problems faced by the tracker in terms of complex image, Loss of information and transformation of 3D world into 2 D image. To achieve good accuracy in object detection we should not only focus on classifying objects but also on locating the positions of different objects that may vary

image to image [2]. It is very important to develop the most effective real time object tracking algorithm which is a challenging task. Deep learning since 2012 is working in these kinds of problems and has revolutionized the domain of computer vision. This paper aims to test the performance of both the algorithms in different situations in real time using webcam and is made primarily for the visually impaired peoples. Blind peoples have to rely on someone who can guide them or on their physical touch which is sometimes very risky also. Daily navigation of blind peoples in unfamiliar environments could be the frighten task without the help of some intelligent systems. They key concern behind this contribution is to investigate the possibility of expanding the counts of objects at one go to expand the support given to the visually impaired peoples. Some common limitations of the previous techniques is less accuracy, complexity in scene, lightening etc. To overcome all those challenges two algorithms are analyzed on all possible grounds and from every perspective to achieve good accuracy.

II. RELATED WORK

In recent years many algorithms are developed by many researchers. Both machine learning and deep learning approaches work in this application of computer vision. This section outlines the journey of the different techniques used by the researchers in their study since 2012. Histograms of Gradient Descent (HOG) use SVM algorithm for detecting objects in real time [3]. It is a feature detector used to extract meaningful information from the image ignoring the background image. This algorithm works very effectively in detecting human and textual data. To improve the performance in more general situations various deep learning approaches were also used by many researchers in their work. Recently convolutional neural network (CNN) based methods were demonstrated to achieve real time object detection. For e.g. Region of proposals network (RCNN) [4]. RCNN instead of using full image only looks at the portion where the probability of having object is high. It extracts 2000 regions of every image and ignore rest of the part and takes 45 seconds to process every new image. Due to this selective search property RCNN works very slowly and sometimes ignore the important part of the image. Recently many improvements were made on RCNN for e.g. Fast RCNN [5] and Faster RCNN [6]. After this comes the YOLO family on which our research work depends. RCNN are

generally more accurate but YOLO algorithms are much faster and more effective to work in real time detection .You only look once (YOLO) [7] works on full image by dividing the input image into SXS grid and predicting bounding boxes and confidence scores for every grid. Second Version of YOLO algorithm i.e. YOLO_V2 [8] comes with some improvements in terms of improving accuracy and making it faster than YOLO algorithm. YOLO_V2 uses a batch normalization concept which improves the precision by 2% than original YOLO algorithm, it also uses a concept of anchor boxes as used in region of proposals method which make YOLO free from all guesses on bounding boxes. Then came the latest and the third version of YOLO algorithm i.e. YOLO_V3 [9] which is slighter bigger and more accurate than YOLO algorithm.YOLO_V3 replaces the mutual exclusive concept to multi label classification i.e. it makes 3 predictions at each levels. It shows excellent performance in detecting small objects. After doing above analysis it is concluded that Yolo family is better than RCNN family. In Yolo family there are many improvements done since 2015 from Yolo to Yolo_v3. The main aim of this study is to compare the original and the latest version i.e. YOLO and YOLO_V3 algorithm with audio feedback that can help blind peoples to recognize all kind of objects near them. As humans can see outside world by using their brains and eyes and can easily recognize every objects but this capability is lost for visually impaired peoples [10].

III. DATASET

Whenever we are working on any object detection algorithm the two main characteristics that we are looking at is detection and localization. Detecting any object has to state whether object belongs to a particular class or not. Localization refers to the bounding box around every object as location of object may vary for every image. Using Challenging datasets helps to set the benchmark for comparing the performance of different algorithms in same application. For testing the performance of algorithms for our problem statement we have used Microsoft common objects in context (MS COCO) [11]. COCO as the name suggest has taken all its images from the common scenes in context with the objects and can be downloaded from its original website [12]. There are total 330K images with 91 categories out of which 82 categories are labelled. The COCO dataset has fewer categories but more number of instances on particular object which makes machine learn more accurately. COCO dataset deals with small objects in very effective manner.

IV. COMAPARATIVE ANALYSIS BETWEEN YOLO AND YOLO_V3

A brief comparison between Yolo and Yolo_v3 on various factors are shown in Table 1. The below comparison covers the important criterias on which both the algorithms are dissimilar from each other. The whole table is designed for fast review of algorithms cited from [7] [9] [17] [18] [19] [20].

TABLE I. Comparison between Yolo and Yolo_v3

S.NO	CRITERIA	YOLO	YOLO_V3
1	No of Layers	Yolo has 24 convolutional layers followed by 2 fully connected layers	Yolo_v3 has 106 convolutional layers (53 original layers and 53 detection layers are added)
2	Shape of Detection Kernel	(S, S, B x 5+C)	1 x 1 x (B x (5+C))
3	No. of Predictions	Yolo gives only one prediction at the last layer by applying 1x1 detection kernel on a feature map.	Yolo_v3 gives predictions at three scales by applying 1 x 1 detection kernel at three different positions and sizes.
4	Detection of Smaller Objects	Yolo is not good in detecting small objects	Yolo_v3 preserve the features which helps in detecting small objects from the complex images.
5	Choice of Anchor Boxes	No. of anchor boxes is not fixed it makes use of K-means algorithm to generate anchors	Yolo_v3 make use of 9 anchor boxes i.e. 3 for each scale
6	Softmaxing	Yolo make use of softmax to select the object with maximum confidence score.	Yolo_v3 replaces softmax concept by performing multi label classification of objects.
7	Classification loss	It uses mean square error (SSE) for calculating classification loss.	It uses binary cross entropy for calculating loss.
8	Class Prediction	It uses Softmax function to calculate probability of object falling under particular category by converting scores that sum up to one.	Class Predictions and Object Confidence are Predicted through Logistic Regression.
9	Mean average Precision	23.7 %	55.3 %
10	Speed	Low	Medium
11	Accuracy	Medium	High

V. METHODOLOGY

Under this section the important features of Yolo and Yolo_v3 algorithm are described. The Framework of both the algorithm in detecting multiple objects with audio feedback is shown with the help of diagram which gives clear understanding of actual flow of every process.

A. Real Time Object Detection with Audio Feedback using Yolo Algorithm

YOLO as the name suggest "You Only Look Once" is the first algorithm after selective search approach that uses single neural network to the image and then divide the image into SXS grids and create the bounding box by assigning the confidence score and class label to each object. Each grid cell is predicting (x, y, w, h) and confidence score for every object. Confidence score

measures how accurately the object is present inside the bounding box. The value of all (x, y, w, h) are between 0 and 1. YOLO prediction has a shape of (S, S, BX5+C) [13]. Network Architecture of Yolo algorithm with counting of each convolutional layer can be studied from [14] every grid has 20 class conditional probability. Class conditional probability tells the probability of object presence in the cell.

The above description covers basic features of YOLO algorithm, for deep understanding refer [7]. In developing Object detection using YOLO with voice response we have used tensor flow with SSD _Mobile net model. Single Shot Detector (SSD) [15]. Tensor flow is a google open source machine framework which is used to detect real world objects in a frame. Tensor flow is a unique model that is used according to the need of the user. This library of python uses many architectures for e.g. SSD (Single hot Detector), R-CNN, Faster RCNN etc. these play very important role in speed and accuracy of model. In Building our model we have used tensor flow_SSD _Mobile Net model. SSD as the name suggest detect the class and position of the object in same step. Mobile Net is a convolutional feature extractor used to extract high level features of the images. Once the class and position of multiple objects is detected by feeding the dataset in the YOLO architecture using Tensor flow model the text output is converted into speech using "gTTS".gTTS [16] is the module that save voice text to mp3 using google text to speech (TTS) API. Next we have used "pygame" open source library of python that helps to play the audio feedback. Workflow of YOLO with audio feedback is shown in "*Fig. 1*,"
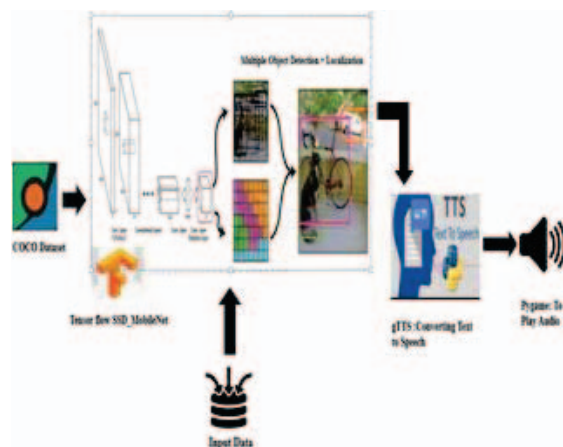

Fig. 1. Workflow of YOLO with Audio Feedback

B. Real Time Object Detection with Audio Feedback using Yolo_v3 Algorithm

You only look once is one fastest object detection algorithm when we need real time detection. Recently Third version of the YOLO algorithm i.e. YOLO_V3 came out, it is super-fast and more accurate than Single shot detector (SSD) [16]. YOLO_V3 has originally 53 layers of Dark net but for the detection 53 layers are added, giving us total 106 layers. Improved architecture of YOLO_V3 can be studied from [18]. The most interesting feature of YOLO_V3 algorithm is that it makes detections at three different scales at three different sizes and at three different places in the network. The shape of detection kernel is 1x1x (B x (5+C)) where C is the total classes i.e. 80 for COCO dataset, B is the number of bounding boxes around the objects. So the kernel size of YOLO_V3 is 1 x 1 x 255. For deep understanding of YOLO_v3 algorithm refer [17]. We have used a much deeper network i.e. Dark Net 53 which consist of 53 convolutional layers. After feeding the input image in the YOLO_V3 architecture multiple objects are classified and assigned a class labels. The resulting output is send to python module gTTS to convert text to speech and pygame to play the audio as shown in "*Fig. 2*,"
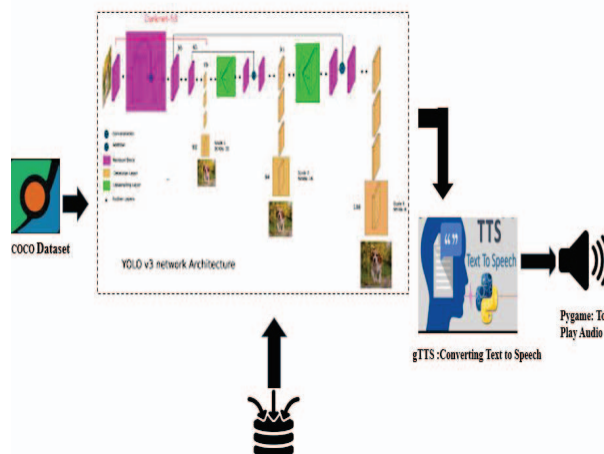

Fig. 2. Workflow of YOLO_V3 with Audio Feedback

VI. EXPRIMENTAL RESULTS AND ANALYSIS

In this section different evaluation metrics were used to evaluate the effectiveness and adaptability of the algorithm. To evaluate the performance precision and recall and the inference time were used. To calculate precision and recall value TP, FP, TN, and FN are calculated using threshold value. The IOU value is taken as 0.5 i.e. if the value of IOU ≥ 0.5 then the detection is true otherwise it is false.

$$Precision = TP/ (TP+FP) \qquad (1)$$
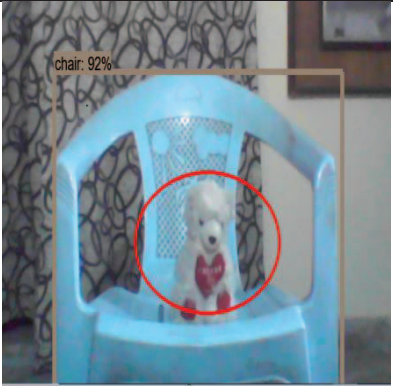$$Recall = TP/ (TP+FN) \qquad (2)$$

Apart from precision and recall value inference time i.e. total time taken by the algorithm to detect objects is calculated to measure the speed. The experiments were done in various situations i.e. for single object detection, multiple object detection, distant object detection and video object detection as shown below. All the experiments were done in real time using webcam.

A. Single Object Output

With single object test yolo and yolo_v3 both gives good accuracy but yolo takes more time in detecting object and even ignore front object as shown in Table 2.

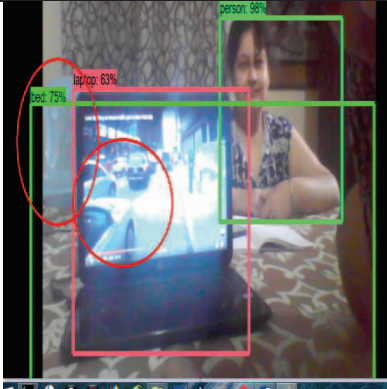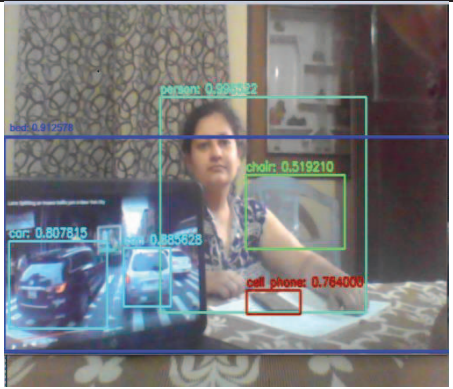TABLE II. Performance of Algorithms with Single Object

Input Image	YOLO	YOLO_V3
a		
Precision %	86.44	96.50
Recall %	84.90	94.98
Inference Time	1.80 Sec	0.345 Sec

B. Multiple Object Output

Results of yolo and yolo_v3 when tested for multiple objects is shown in Table 3. We have tested multiple object test for many objects to correctly analyze the performance. From the below results it is clear that yolo sometimes ignores some when many are present in one frame on the other hand yolo_v3 is able to detect all objects present in the frame. If we compare the time of algorithms for single object and multiple objects we get more time in detecting all multiple objects and sometimes accuracy also decrease.

TABLE III Performance of Algorithms with Multiple Object

Input Image	YOLO	YOLO_V3
a		
Precision %	83.22	93.44
Recall %	82.12	91.45
Inference Time	6.25 Sec	2.59 Sec

C. Distant Object Output

When yolo and yolo_v3 are tested for distance objects yolo is not able to recognize any small object on the other side yolo v3 gives excellent performance with far away objects as shown in Table 4.

TABLE IV. Performance of Algorithms with Distant Objects

Input Image	YOLO	YOLO_V3
a		
Precision %	71.44	89.45
Recall %	70.78	87.98
Inference Time	5.76 sec	3.37 Sec

D. Video Object Output

All the above analysis are based on static type i.e. we are setting the frame and test the performance of the algorithm but in this we have also test the algorithms on real time running video where we find Yolo_v3 is showing better and more accurate performance than Yolo algorithm as shown in Table 5.

TABLE V. Performance of Algorithms with Video Objects

Input Image	YOLO	YOLO_V3
a		
Precision %	78.99	92.89
Recall %	75.78	90.45
Time	7.48 Sec	5.14 Sec

Performance of both the algorithms under every situation on different size of dataset is also calculated and plotted in the form of precision and recall curve as shown in "*Fig. 3*,", "*Fig. 4*,", "*Fig. 5*," and "*Fig. 6*," .

Fig. 3. Precision Curve of Yolo in different size of dataset

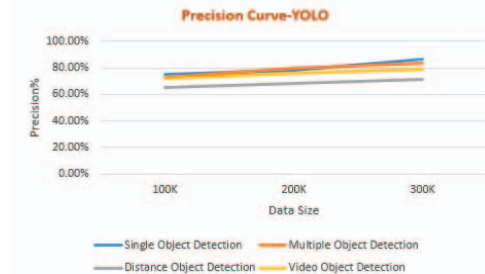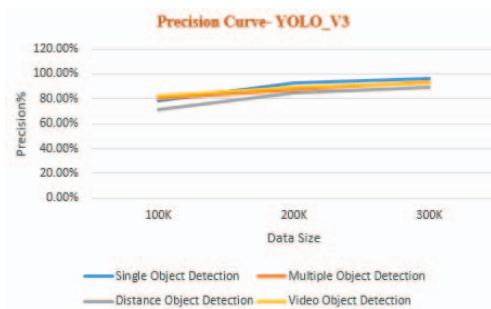*2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence 2021)*

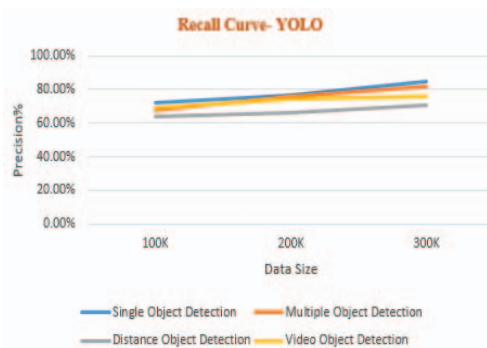Fig. 4. Precision Curve of Yolo_v3 in different size of dataset

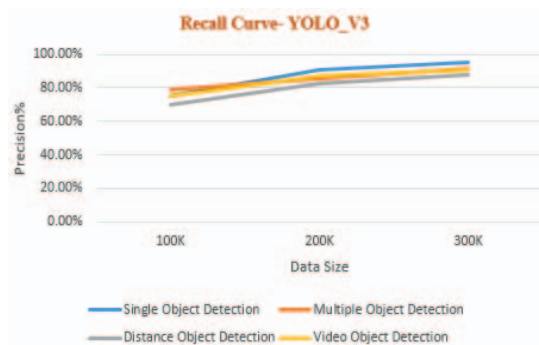Fig. 5. Recall Curve of Yolo in different size of dataset

Fig 6 Recall Curve of Yolo_v3 in different size of dataset

From the graph it is clear that when the size of the dataset is small the model generally under fit and it start stabilizing when the size of dataset start increasing. According to the precision curve of Yolo the precision value of Yolo in all situation is between 65-85 % and to that of Yolo_v3 is 65-98 %. Yolo_v3 is also showing better performance in detection of small objects and distant object in real time. Yolo is showing good accuracy in single object detection and multiple objects detection but not if objects are far and small in size as shown in curve. After above analysis it is concluded that Yolo_v3 is better than Yolo in various situations and can be combine with any IOT technology for the development of the business.

VII. CONCLUSION AND FUTURE SCOPE

In this paper, introduced a comparison between Yolo and Yolo_v3 for detecting and classifying every object present in front of webcam with good accuracy and in less time.

After testing both the algorithms for various situations we find that Yolo_v3 is much more powerful than Yolo in detecting small objects and distant objects. Yolo is fast in detecting all nearby objects but when we are testing it for some complex image it ignore all the small objects and far away objects. There are many deep learning algorithms there working for this application so it is a big challenge to attain good accuracy among all. In this paper we have done small scale evaluation of algorithms in terms of precision, recall and inference time taken by the algorithm. In future we will try to expand our evaluation to other algorithms with more parameters and with more images. This time we have worked on ready mate dataset but in future we will try to implement this analysis on our own small dataset which would be more objective and effective.

REFERENCES

[1] S. Cherian, & C. Singh, "Real Time Implementation of Object Tracking Through webcam," *Internation Journal of Research in Engineering and Technology*, 128-132, (2014)

[2] Z. Zhao, Q. Zheng, P.Xu, S. T, & X. Wu, "Object detection with deep learning: A review," *IEEE transactions on neural networks and learning systems*, *30*(11), 3212-3232, (2019).

[3] N. Dalal, & B. Triggs, "Histograms of oriented gradients for human detection," In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)* (Vol. 1, pp. 886-893). IEEE, (2005, June).

[4] R. Girshick., J. Donahue, T. Darrell, & J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE transactions on pattern analysis and machine intelligence*, *38*(1), 142-158, (2015).

[5] X. Wang, A. Shrivastava, & A. Gupta, "A-fast-rcnn: Hard positive generation via adversary for object detection," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2606-2615), (2017).

[6] S. Ren, K. H, R. Girshick, & J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," In *Advances in neural information processing systems* (pp. 91-99), (2015).

[7] J. Redmon, S. Divvala, R. Girshick, & A. Farhadi, "You only look once: Unified, real-time object detection," In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788), (2016).

[8] J. Redmon, & A. Farhadi, "YOLO9000: better, faster, stronger," In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7263-7271) (2017).

[9]J. Redmon & A. Farhadi, "Yolov3: An incremental improvement," ArXiv preprint arXiv: 1804.02767, (2018).

[10] R. Bharti, K. Bhadane, P. Bhadane, & A. Gadhe, "Object Detection and Recognition for Blind Assistance," International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 06, (2019).

[11]T. Lin, Y. Maire, M. Belongie, S. Hays, J. Perona, P. Ramanan, D., & C.L. Zitnick, "Microsoft coco: Common objects in context," In *European conference on computer vision* (pp. 740-755). Springer, Cham, (2014, September).

[12] http://cocodataset.org/#home

[13] J. Du, "Understanding of Object Detection Based on CNN Family and YOLO," In Journal of Physics: Conference Series (Vol. 1004, No. 1, p. 012029). IOP Publishin, g, (2018, April).

[14] S. Geethapriya, N. Duraimurugan, & S.P. Chokkalingam, "Real-Time Object Detection with Yolo," International Journal of Engineering and Advanced Technology (IJEAT), 8(3S), (2019).

[15] A. Arora, A. Grover, R. Chugh, & S.S. Reka, " Real time multi object detection for blind using single shot multibox detector,". Wireless Personal Communications, *107*(1), 651-661, (2019).

[16]S. Kurleka, "Reading Device for Blind People using Python, OCR," International Journal of Science and Engineering Applications Volume 9–Issue 04,49-52, ISSN:-2319–7560, (2020).

[17]J. Redmon, & A. Farhadi, "Yolov3: An incremental improvement," ArXiv preprint arXiv: 1804.02767, (2018).

[18] J. Li, J. Gu., Z. Huang, & J. Wen, "Application Research of Improved YOLO V3 Algorithm in PCB Electronic Component Detection," Applied Sciences, 9(18), 3750, (2019).

[19] G. Peng, "Performance and Accuracy Analysis in Object Detection," (2019).

[20] A. Jana, P., & A. Biswas, "YOLO based Detection and Classification of Objects in video records," In 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT) (pp. 2448-2452). IEEE, (2018, May).