REAL TIME OBJECT DETECTION USING MOBILENET-SSD WITH OPENCV

G. Hema Sudha Rani[1], D. Suvarna[2], P. Teja Ammani[3], E. Jeetesh[4], K. Sai Chandu[5]

[1]Assistant Professor, Department of CSE-Artificial Intelligence and Machine Learning, S.R.K Institute of Technology, NTR, Andhra Pradesh, India, suvarnareddy3725@gmail.com
[2,3,4,5]Student, Department of CSE-Artificial Intelligence and Machine Learning, S.R.K Institute of Technology, NTR, Andhra Pradesh, India

Abstract- The object detection is a computer technology which is related to OpenCV. The task of the object detection is identifying and localizing the objects with in an image or video. It involves the drawing of the bounding boxes around the detected objects and assigning the detected objects with their corresponding class labels. In this paper, we are implementing the object detection in real time. Firstly, we need to load the pre trained model MobileNet and develop the system using SSD (Single Shot Multi-box Detector) framework. And we built the required system using convolutional neural network using deep learning approaches. This real time object detections is used in many purposes like surveillance camera, video stream, self-driving cars and augmented reality to detect and to track the objects as they appear in real time. To implement this we are combining the MobileNet and SSD framework to improve the performance and efficiency for object detection.

Keywords- Object detection, real-time, CNN, Non-Maximum suppression, SSD (Single Shot Multi-box Detector), MobileNet.

I.INTRODUCTION

The real time object detection is a fundamental task in computer vision, which the system is capable of identifying and locating the objects in a video stream in real time. It is an image classification method in which this method aims to detect one or more images in the given input frame and also uses the bounding boxes to display their presence. The foundation of this project is preprocessing step which involves resizing, normalization and other transformations. The main purpose of this project is to enable the efficient and accurate detection of objects in real world environment directly on embedded systems.

The traditional object detection involves the handcrafted features and techniques to detect the objects. This method detects only a few objects and applicable at certain applications only. In this method the features are extracted from the preprocessing image by using the techniques like Histogram of Oriented Gradients (HOG) and Scale-Invariant Feature Transform (SIFT). By using decision tree algorithm it can classify the labels of the objects in the image.

MobileNet with SSD framework is designed to be the lightweighted and optimized for mobile and the embedded devices. This architecture allows for the fast inference of the speed and making it suitable for the real time applications where quick detection of the objects is essential for the augmented reality and object tracking. It is designed for a small storage space and having the lower computational requirements compared to the larger CNN structure. This simplifies the development process by offering a unified framework for the image processing, object detection and class label classification.

The SSD (Single Shot Multi-box Detector) process the detection of objects in a single pass through the network. And this architecture design also having faster inference compared to two stage detector. The SSD utilizes the multi-box technique to allocate bounding boxes around the detected objects through regional proposal. The SSD helps in the improvement of the efficiency in the object detection by avoiding the separate regional proposal step. This is used to recognize

the objects in a video stream of the real time and for the webcam board casting. We can also use the object detection module to determine the objects which are in the video stream. To complete this module, we combining the MobileNet with the SSD framework to create a fast and the efficient deep learning-based item detection technique.

The experiment results reveal that the algorithms average precision for recognising of the various classes such as car, human, bus, and chair with 99.4 percent, 99.3 percent, 95.2 percent and 99.1 percent respectively. This enhances the accuracy of the behaviour recognition of the objects at the handling speed required for the real time location of the object in the detection process. One of the focal points of our work is the incorporation of MobileNet into the SSD system. However the MobileNet with the efficient SSD structure has been a best combination for the investigation topic in the recent years, owing to the practical limitations of the running robust neural network on low end devices such as mobile phones, laptops to broaden the range of the possible outcomes in the real time applications.

The regions are generated automatically, without taking into consideration of the image features during the extraction process. An important trade-off that is made with the region proposals generation on the number of the regions and the computational complexity. We find the objects based on the regions that we used to generate during the process. On the contrary, if we have to exhaustively generate all possible proposals, it won't be the possible to run the object detector in teal time applications, for a instance. Sometimes, it is possible to use a problem specific information to reduce the number of ROI's.

II.LITERATURE SURVEY

In 2014, Ross Girshik, Jeff Donahue, Tervor Darrell, Jitendra malik developed the "Rich Features Hierarchies for Accurate Object Detection and Semantic Segmentation", this is implemented by regional convolutional neural network. In this the performance of the object detection is measured by the PASCAL VOC dataset, and best performing methods are typically combine low level images features to high level context. The best result for the VOC-2012 is having mAP of 53.3%. In this we are using high-capacity convolutional neural network. There are three basic steps, the first step allows to allocate the independent regional proposals. The second step is the high-capacity convolutional neural network extracts the vectors from the regional proposals. The third step is a set of class linear technique and in this we present our decision to the model, after classifying if allocates the bounding objects. In this approach we combine all the techniques for a two key insights in it one can apply the high-capacity of the convolutional neural network to a bottom-up approach of the region proposals in order to localize and the segment of the objects in which labelled training data is scarce and it is supervised pretraining for an auxiliary task, and it is followed by the domain specifications of fine-tuning, yields a significant performance boost. Since we combine our proposals with the convolutional neural network we call this method as the R-CNN (Region with CNN features).

In 2018, Joseph Redmon, Ali Farhadi developed the "YOLOv3: An Incremental Improvement". In this model we predict the objectness score for each displayed bounding boxes using logistic regression. In this process we use the threshold value 0.5 to implement the detection of the object. If the bounding boxes prior is not assigned to the ground truth object it occurs no loss for co-ordinate or class predictions, only classes. To have a good performance we do not use any softmax to detect the objects and also we simply use the independent logistics classifiers. Having the softmax it states the only one class in the prediction during the complete process and it makes difficult to predict the multiple class prediction. YOLOv3 is a comparatively lightweight capacity in which the users prioritize model compactness and easy of the deployment. For instance YOLO (You Only Look Once) is a reliable detector in detection of the objects during the process of object detection based on the deep convolutional neural network and it remains commonly for the real time object detection. Recently, different

algorithms based on the YOLO provide much faster and more accurate when compared with the previous algorithms. The aim of this is to compare the different performances in the process and the versions on image classification.

In 2020 V.N. Tran, T.T. Nguyen developed "Efficient Object Detection using OpenCV with faster R-CNN" implemented by three algorithms that are mask R-CNN, faster R-CNN, R-CNN. The network is designed to train and test the images during the process. The performance is based on the regional proposal network which is built by convolutional layers and also using the MobileNetv1 to improve the efficiency in the object detection process. The fast R-CNN shows the high detection rate for the panel and speech whereas the Faster R-CNN shows high detection rate based on the character, face, features of the object in the given input frame. This method is based on present revolutionary of Faster R-CNN model, and it was designed by adapting the two domain components with a aim of the reducing of domain discrepancy. By proposing a multi component of the different region based on the focus steered by enforcing diversion of the factor appearance in the detection of the objects in the given input frames. This model architecture has a high detection speed but the accuracy is low when compared with the faster R-CNN with Inception V2 that has lower speed but good accuracy and this shows the algorithmic change by communicating the proposals with a deep convolutional neural network.

III.EXISTING SYSTEM

In existing system we use Fast R-CNN which is a spatially-localized and oriented method to classify and localize objects in images. It works at an inexperienced speed with accuracy, and is powered by tensorflow. As a input we give the image to detect the objects and regional proposals are allocated based on the initial candidate region to detect the object. Now we use a pre-trained convolutional neural network to extract the features from the images. These features contain the high level representations of the image, essential for detecting the objects. For each proposal apply RoI pooling to extract the fixed size feature maps. RoI pooling efficiently crops and resizes the features that are extracted by the CNN backbone to the uniform size, regardless of size or aspect of the ratio of region proposals. Now we need to pass this RoI layers to fully connected layers and these layers transform the region-specific features into a format suitable for classification and bounding box regression.

Predict the class probabilities and the bounding boxes offset for each region proposals. And the class probabilities indicate the likelihood of each proposal that containing a particular object class. The bounding boxes offsets the refine of the co-ordinates of the proposed bounding boxes to better fit the objects and drawing the bounding boxes around the detected objects. And also label each bounding boxes with the corresponding object class and confidence score.

Disadvantages: -

- Difficult in real time implementation
- Training time
- Complexity
- Inference speed
- Limited to pretrained classes

IV.PROPOSED SYSTEM

The proposed system uses the MobileNet SSD architecture to quickly and efficiently identify objects in real time. A python script is written using OpenCV that uses a deep neural network to discover objects. Input will be given through the real time video stream or webcam capturing, based on streamlined MobileNet Architecture which used depth-wise separable convolutions to build light weight deep neural networks.

First we need to import the necessary libraries and load the pretrained MobileNet SSD model. The MobileNet SSD model is a deep neural network trained to detect objects. Start

capturing the video frames from the webcam as the input to the system and read each frame from the video stream. Extract the each feature from the image and apply the resizing to the images based on the pixel intensity under the dark area. An image may contain the irrelevant features and few relevant features that can be used to detect the object. The task of the MobileNet SSD layers is to change the pixels from the input image to describe the contents of the image. Preprocess each frame and convert it into a blob and load this blob to the network. Run forward pass to obtain the detections. The model returns a set of bounding boxes along with confidence score for the detected objects in the frame. As the last step is to display the output.

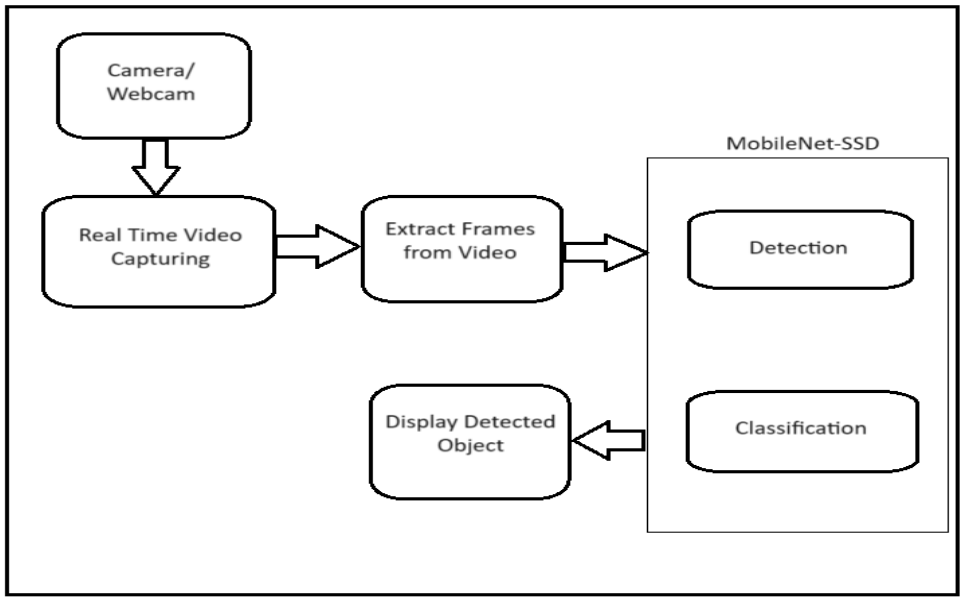

Fig:1 Architecture of Proposed System

Advantages: -

- Real Time Management
- Efficiency
- Flexibility and customization
- Ease of implementation
- Accuracy and scalability
- Low latency

V.RESULT

The evaluation of the performance of the real time object detection model is conducted through many experiments. And the model is demonstrated as very efficient and accurate in detecting the objects in real world applications. The accuracy is achieved overall 87 percent by evaluations of the back ground in the frame of the given video stream or the webcam. The real time processing speed of the system was the consistently measured at the 25 frames per second on a standard desktop computer and meeting in the real time applications requirement.

Furthermore, the system was showcased its adaptability to the resource-constrained devices by achieving an average FPS (Frames Per Second) of 15 on embedded systems and smartphones. A comparative analysis was occurred against state of the art object detection methods, including the CNN (Convolutional Neural Network), YOLO (You Only Look Once) and SSD (Single Shot Multi-box Detector), highlighted the systems superior performance in terms of the accuracy and the speed. The real time object detection system demonstrates the reliable performance in the various of real world scenarios, such as traffic surveillance, pedestrian detection, and object tracking. Its ability to accurately identify and the localization

of the objects of interests in real time application. Although it is a realistic data that would show how the system will perform and improves the performance and the efficiency by the help of the trained model and also try to improve its accuracy by using the testing data from the COCO dataset. After the completion of the task it displays the output which consists of the detected object surrounded by a bounding boxes with its detected class labels and with its respective confidence score.

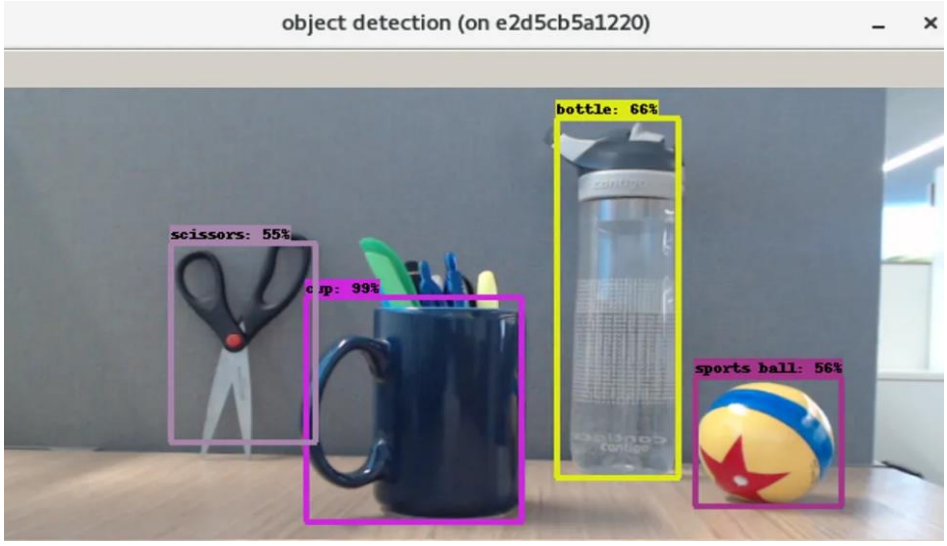

Fig:2 Object detected using webcam

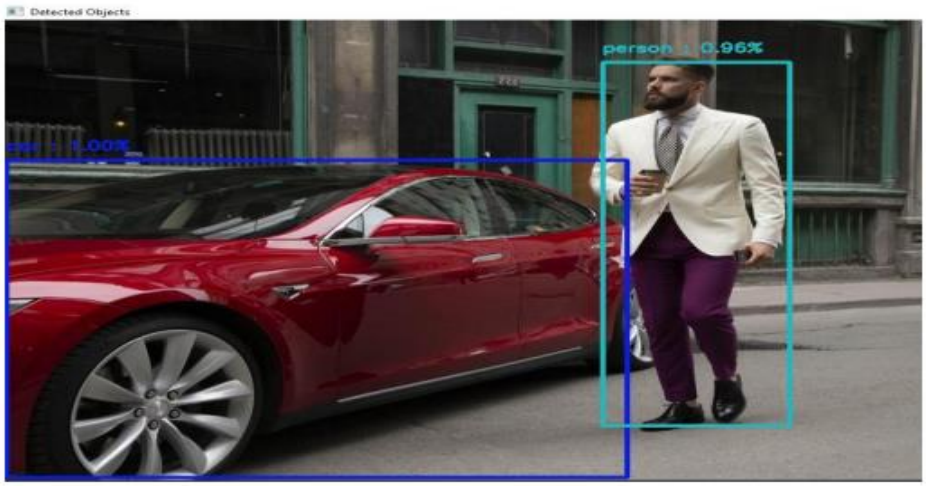

Fig:3 Objects (Car, Person) detected in an image

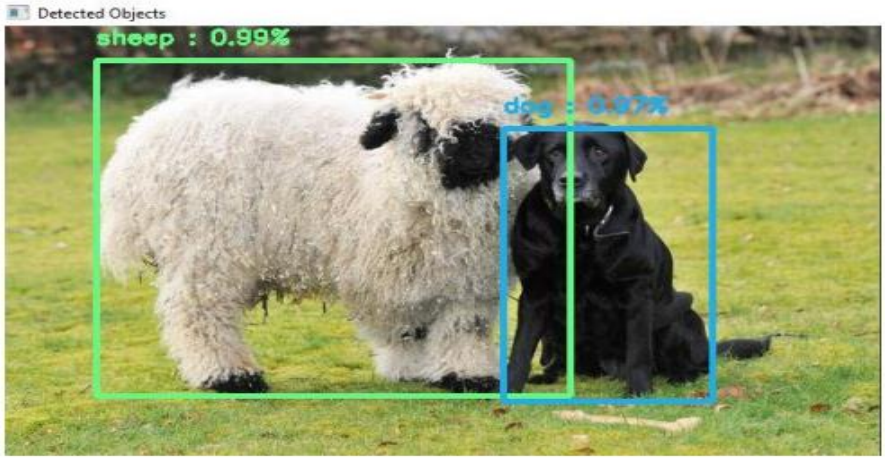

Fig: 4 Objects (Sheep, Dog) detected in an image

VI.CONCLUSION

A high accuracy object detection technique has been achieved by using the combination of the MobileNet and the SSD framework for the object detection. The proposed system is tested with many objects and it can detect and identify the objects quite accurately. This system can detect the items within its dataset such as person, chair, bottle and tv monitor etc. Achieving a mean average precision of 82.3%. Finally, we can perform the real time object detection using the SSD framework and MobileNet which is the faster and more efficient than traditional methods of the object detection. In this each detection provides the coordinates of the bounding boxes around the detected objects. Overlay bounding boxes on the frame corresponding to the detected objects. And also displays the confidence score as labels.

VII.FUTURE SCOPE

We can see continued improvements in accuracy and efficiency. Integration with hardware acceleration and multi-object tracking will enhance the performance for diverse applications. Domain specific optimizations and robustness enhancements will broaden its utility across industries. Segmentation integration will deepen scene understanding capabilities for richer applications.

REFERENCE

[1] S. Parvathi and S. T. Selvi, 2021. Detection of maturity stages of coconuts in complex back ground using Faster R-CNN model, biosystems engineering, pp.119-132.
[2] M. Pawelczyk and M. Wojtyra, 2020. Real world object detection dataset for quadcopter unmanned aerial vehicle detection. IEEE Access.
[3] K. Ayoosh, What's New in YOLO v3?, 2018. [Online] Available: www.towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b.
[4] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement", arXiv preprint arXiv:1804.02767, 2018.
[5] G. Alpaydin, "An adaptive deep neural network for detection, recognition of objects with long range auto surveillance", in ICSC, 2018, pp. 316-317.
[6] B. A. Kitchenham, D. Budgen and P. Brereton, Evidence-based software engineering and systematic reviews. CRC press, 2015,vol. 4.
[7] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," arXiv preprint arXiv:1806.03852, 2018.
[8] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy and P. Tang, "On large-batch training for deep learning: generalization gap and sharp minima", arXiv preprint arXiv: 1609.04836, 2016.