Multiple Object Tracking using Deep Learning with YOLO V5

Abhinu C G
UG Student, Department of Computer Science and Engineering
College of Engineering Kidangoor
Kottayam-686583, India,

Aswin P
UG Student, Department of Computer Science and Engineering
College of Engineering Kidangoor
Kottayam-686583, India,

Kiran Krishnan
UG Student, Department of Computer Science and Engineering
College of Engineering Kidangoor
Kottayam-686583, India,

Bonymol Baby
UG Student, Department of Computer Science and Engineering
College of Engineering Kidangoor
Kottayam-686583, India,

Dr. K S Angel Viji
Associate Professor, Department of Computer Science and Engineering
College of Engineering Kidangoor
Kottayam-686583, India,

***Abstract* - The MOT (Multiple Object Tracking) is an important tool in the modern world. It has various uses like object detection, counting objects, security tools ,etc. The Object tracking is a prominent technology in image processing which has a large future scope. The MOT has made significant growth in a few years due to deep learning, computer vision, machine learning, etc. This paper aims to provide a software solution that keeps track of the objects so that it can handle object list and count. By using YOLO "You Only Look Once" Technology with the help of Pytorch, the system aims in object detection, tracking and counting. Also unlike the general yolo object detection tool which detects all objects at the same time ,this MOT system also detects only objects which are needed to be detected by the user and thus helps in improving the performance of the system.**

Keywords - Multiple Object Tracking (MOT); YoloV5; Deep Learning; Dataset/Model

I. INTRODUCTION

Tracking is one of the necessary technologies needed for the upcoming world. Tracking can broadly be divided into multiple Object Tracking (MOT) and single object tracking. Multiple Object Tracking (MOT) plays an important role in solving many basic problems in computer vision [1].Tracking multiple objects in videos requires detection of objects in individual frames and combining those across multiple frames. Many Computer Vision techniques have been used to build MOT systems, and day to day the technology is growing rapidly providing an area of opportunities called image processing is done by providing a labeled dataset which is trained and is used as a model for the system which can detect objects in different frames comparing to the objects that was provided in the model by mapping the same pattern of model in the frame [2].

YOLO("You Only Look Once"), OPENCV, PYTORCH,COCO dataset, TKINTER with MYSQL(MySQL is optional),GPU are the methodology used to detect, count and track the objects in MOT.The proposed system uses the Latest YoloV5 which is used to detect the objects.YoloV5 uses pytorch classifier for training as well as detection. Yolo begins its journey with darknet technology ,which was later developed to yolov2 ,then yolo v3 and later to yolo v4 [9].And now for easy building of object detection yolo v5 was introduced leading to better performance of object detection [6].Yolo V5 is constructed with Pytorch Classifier in deep learning and after object detection the opencv module is used for inputting real-time or file format video input to the algorithm and also tracks, and counts the objects detected in the output obtained making the system an efficient MOT system. And also the Tkinter makes the MOT easy for user interaction making it user to choose particular model according to the user requirements and only the particular model object is detected.

The proposed system can be used for various object crowded environments for detection of particular classified objects according to the environment. The MOT makes users take

count of the same type of objects and also used to detect the particular object from the bulk crowd helping users to save time in searching for a particular object.

II. RELATED WORK

A. *Object Detection*

Object detection consists of three stages: classification, detection and segmentation [3].Classification is the to classify the image with a unique id in order to identify it while object detection. Detection is the next step where using the trained model the object is detected in frames. This gives a good perception for the system giving the idea about the image. This step detects the objects using the model and provides the location of the object in that frame. The next stage is the segmentation where the detected object is described and segmented for better understanding(commonly used coordinate representation of rectangular detection box as shown in Fig. 1.).

B. *YOLO V5*

The original author of yolo algorithm is Joseph RedMon. When he started the yolo algorithm construction and when it didn't have significant progress in another author Alexey Bochkovskiy published a paper on yolo and then after that a series of yolo arrived which led to yolov2,yolov3 and then upto yolov4. In the heat of yolov4 the Ultralytics LLC team on may 30,2020 issued YOLOV5 [6].The Yolov5 took a move from darknet to pytorch achieving 140 FPS in Tesla P100 where as in yolov4 only 50 FPS.Yolo V5 has the same advantages and has almost similar architecture as yolo v4.Yet Yolov5 makes it convenient to train and detect objects compared to yolov4.

C. *COCO Dataset*

COCO dataset is a scope object recognition, segmentation, and captioning dataset. It begins with object segmentation where image division takes place in order to discover image boundaries and objects in it.

It is used for labelling using bounding boxes in image. After that in recognition in context, a basic correlation architecture is represented between the image and the objects in it. After this we collect or gather the same type of color or grey levels. This is super pixel stuff segmentation where they help in featuring important areas and also they can reduce the input element for calculations.

D. *Open CV*

Opencv is an open source library for computer vision modules like image processing, camera access,etc [4].Now Gpu is also included in the Opencv module which is an essential element for pytorch [5].Opencv technology developed in a fast rate and supports many algorithms to make the algorithm efficient,especially in image processing field. Opencv in python supports libraries like numpy, matplot,utilis,etc. Opencv does few applications like video image stitching, navigation, Medical analysis, etc.

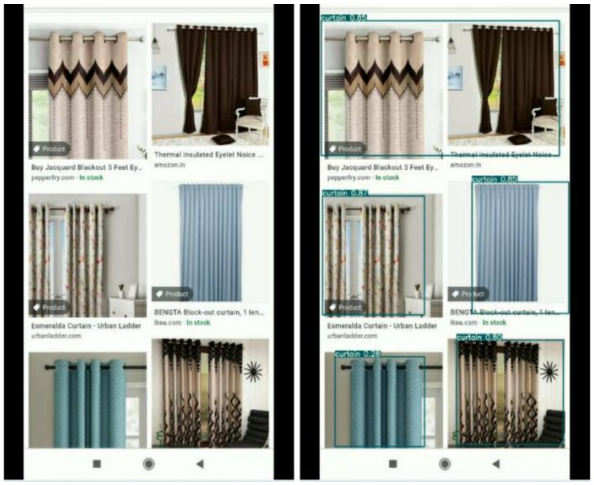

Fig. 1. Object Detection

III. PROPOSED SYSTEM

The system begins with login page and then after optioning, taking video or providing recorded video as input to the system of object detection. Then the inputted data is processed into frames for detection. During detection the The Yolo V5 uses the Model or dataset to detect the object in the input data. By using the model after detecting objects the detected objects are classified and represented by using labels and bounding boxes around the detected objects and then it is processed into output video format (As shown in Fig. 2.).

A. *Input*

Using OpenCV we can access camera module and also we can add video files in different formats. Using OpenCV the the real-time video frame is collected from camera lenses. Before collecting the real-time video frame we assigned a Tkinter window prompt which signifies or collects from the user about which dataset or model should be used for detection. Once the input option is received the user option is sent to neural network module and the camera is enabled using OpenCV and starts collecting the video frames from the camera lens.

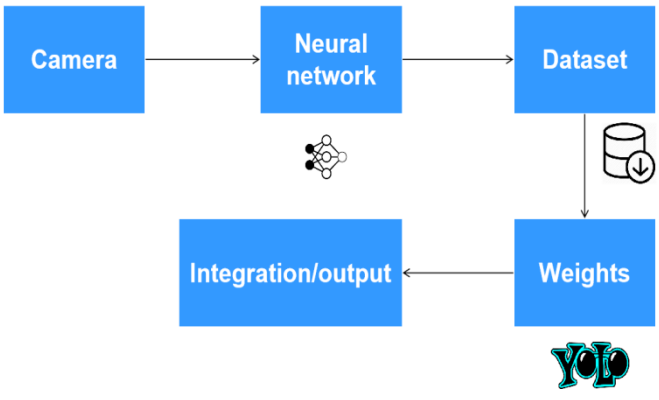

Fig. 2. General working system

B. Neural Network(YOLO V5)

Here using the input received from the camera or video the input data is classified into frames and each frame is sent to yolo detection algorithm with the model which user selected. The model can be a predefined model that is, COCO dataset model or we can create custom models for detection. Once the detection is done, it is bounded with boxes and labels where the object is found and is send to output section where the detected frames are collected and are then compressed into output format. Before merging, the detected frames are used for tracking, counting and sorting using OpenCV and also for better results DeepSORT is also used for sorting and tracking of objects [8].

C. Dataset

This field is used for creating custom dataset from raw images in order for creating a custom model which can be used for detection. For this the first thing is used to collect the raw images from various sources and create a dataset. Then from the dataset images the objects must be annotated and labelled from the images.

For this Python frameworks like "Labeling" is used for annotating and labelling of the objects. Once the annotating and labeling is done then the dataset is split into train and test images in percentage of 70% for train and 30% for test as it is the general ideal percentage used for training. Once this is done it can send to yolo training algorithm where the dataset can be trained and model can be created using COCO dataset model.

D. Weights / Model

Here the labelled dataset obtained from the framework must be configured with ".yaml" (YAML Ain't Markup Language) extension format file which can be used to append the text label to the algorithm. Once the YAML file is configured it is set up in algorithm and using pytorch the given dataset gets trained using GPU according to the epochs given in algorithm for training and with test dataset the testing of trained model after completion also takes place, predicting the objects in test image. Once the objects are predicted the model is compressed into the yolo model format which is configured using the pre-trained model that is using COCO dataset model. Once this is done the model file with the corresponding test result potted in graphs and texts are written in output folder where the evaluation and testing of the model can be done by using it in a detection algorithm.

The activity structure or system architecture of the proposed MOT model is shown in Fig. 3.

E. Integration

Here the above five modules are intersected to make a single system where at front a login page using MySQL can also be set up but it is considered as optional as opencv mostly works under a secure arena . After integrating the modules it is compressed into an executable file which is in ".exe" file format output, which can be used as a MOT application.

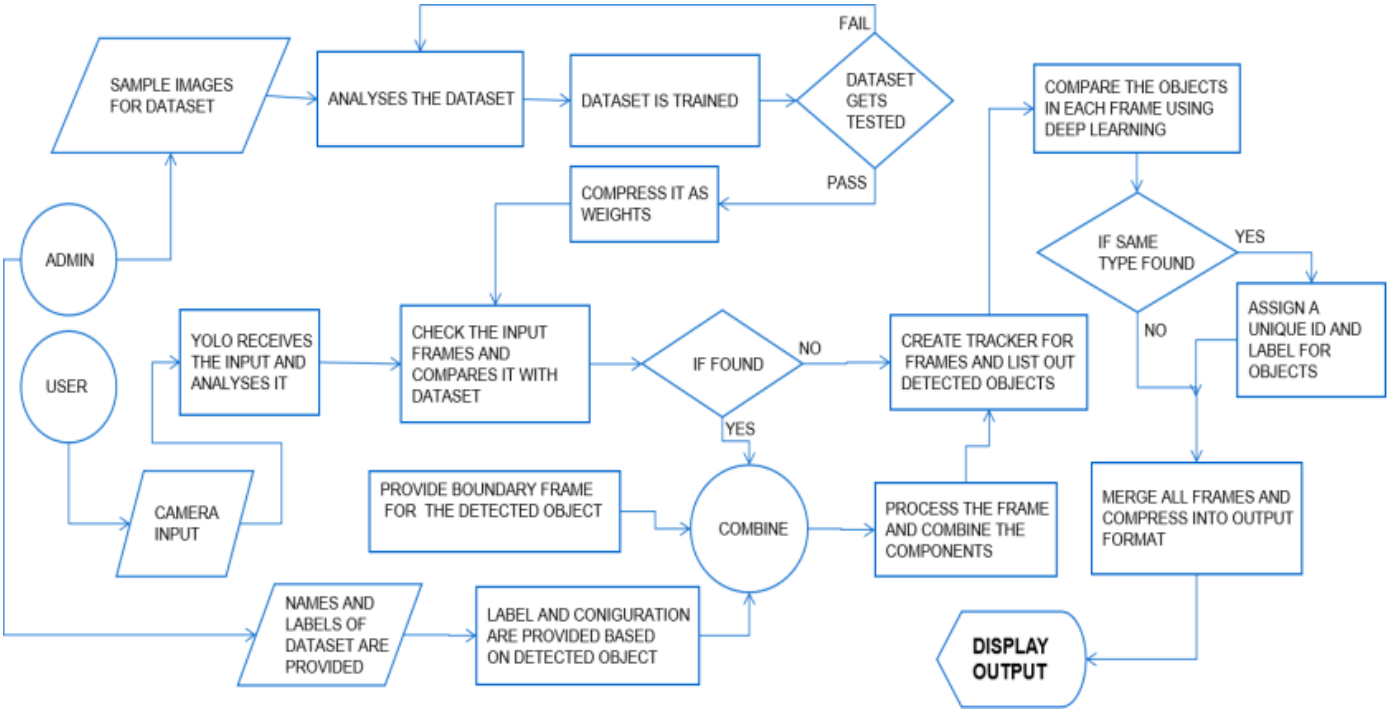

Fig. 3. System Architecture of the proposed system

IV. RESULTS AND PERFORMANCE EVALUATION

The performance of YOLO is measured using mainly three terms: mAP, Precision, Recall [7].

mAP is a measure that combines recall and precision for detecting the accuracy of the object, Where it is calculated with the average precision value for recall value over 0 to 1 with IOU that is intersection over union from 0.5 to 0.95.

$$AP = \frac{\sum_{r=1}^{R} P_r}{R} \quad (1)$$

$$mAP = \frac{1}{N} \Sigma\, AP_k \quad (2)$$

Precision is used to measure how accurately the objects are predicted. Precision can describe how good the model is at predicting the positive class.

$$Precision = \frac{TP}{(TP + FP)} \quad (3)$$

Recall will define how good the objects or classes are detected. Which means, recall or sensitivity calculates the proportion of actual positives that are correctly found.

$$Recall = \frac{TP}{(TP + FN)} \quad (4)$$

IOU stands for Intersection Over Union.

IoU is used to calculate between two boundaries of an image to check whether they have overlapped or not,if so, to know at what rate it is we use IoU. we will be predefining an IoU threshold (say 0.5 in our case) in order to find whether the detection was true positive or false positive.

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (5)$$

The Yolo Accuracy is found using mAP. In COCO mAP, a 101-point interpolated AP definition is used in the calculation. AP corresponds to the average AP for IoU from 0.5 to 0.95 with a step size of 0.05.

Consider the custom dataset model of a single class, say key, with a dataset of 200 images which was trained with the COCO dataset with the "yolov5m6.pt" model which has 51-52 GFLOPS.It was trained under 200 epochs using Pytorch Classifier using Tesla T4 GPU.The results of the main three terms discussed above were plotted in a graph as shown below in Fig. 4.

The results clearly define that at a range of 0-100 images the prediction was growing and mAP was increasing and while reaching the last images the mAP was almost near to one say for the sample dataset key the Accuracy or mAP is approximately 95.39% while training as the Confidence threshold value given was 0.001.Thus we were able to train a good custom yolo model which can be used for detection. The model quality is represented using PR curve shown below in Fig. 5.While using this model for detection, using this data the tracked object in each frame was occurring. The detected object from the crowd had mAP values from 0.2-0.9 according to the clarity of the image in real time.

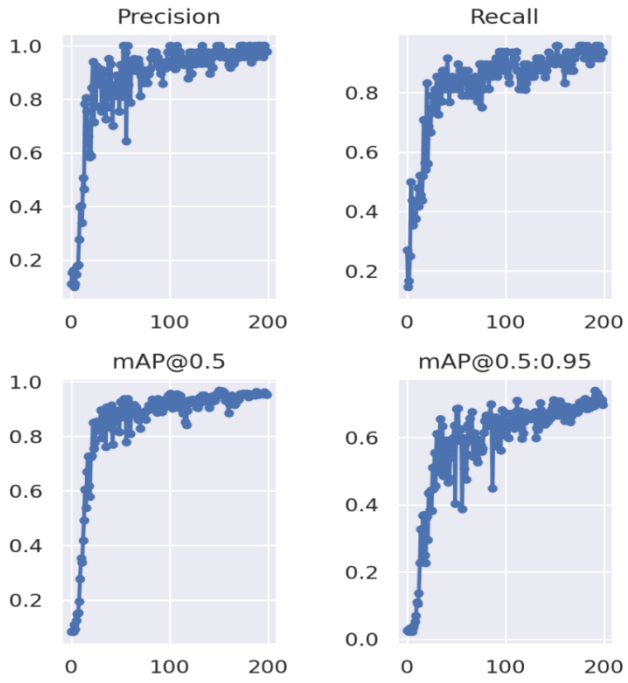

Fig. 4. Precision, Recall, mAP@0.5, mAP@0.5:0.95, of 200 images of class key plotted in graph

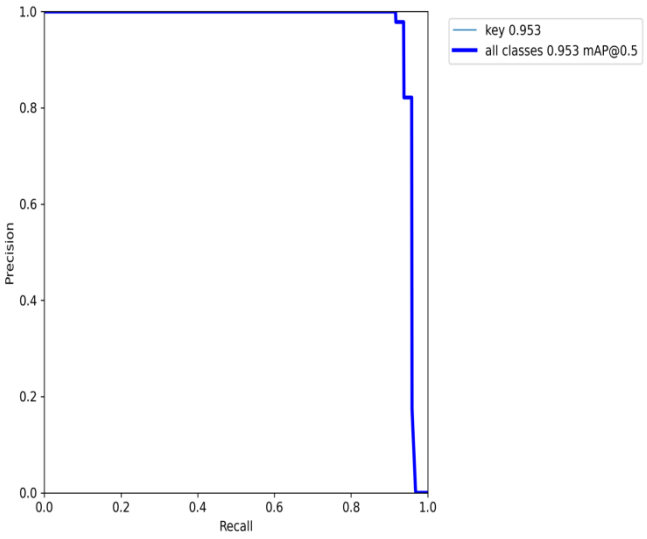

Fig. 5. PR curve of Key dataset model

V. CONCLUSION

In this work, we proposed a Multiple Object Tracking system using yolov5 that can detect objects which were trained and also they can track and take count of objects in each frame. This MOT system has various real-time applications like detecting particular objects from object crowded environments, tracking a particular type of object or detecting a set of object classes or counting a particular object. The MOT is efficient enough to detect objects even on CPU GPU. This MOT requires a local system with GPU which may not be affordable to all systems but with the help of free cloud sources like Google Colab we can use GPU and also make a working

MOT system which can custom train objects with raw-images. And can be used in our local systems. Therefore using the yolov5 algorithms which uses pytorch. The MOT system is capable of tracking various objects which were trained. As a sample the system was trained with a class of object key with 200 images as dataset and after 200 epochs the dataset formed a yolo model with 95.39% mAP which was used for detection .While the custom model was used the object detection and tracking had an accuracy prediction in range of 20-90% according to the clarity of image and appearance of object in image. As this was done in CPU GPU the prediction was better than expected but was not effective as like GPU as GPU system provided a better and accurate prediction of the object this is as the MOT requires to process a model which has 50+GFLOP data. Thus excluding that limitation. The MOT using Deep Learning and yolov5 is executable according to user needs.

ACKNOWLEDGMENT

We would like to convey our special thanks with gratitude to our Tutors and Head of Department as well as our principal who gave us a wonderful chance to do this project, which helped us in involving a lot of Research and we are really thankful to them.

REFERENCES

[1] Li Tan, Xu Dong, Yuxi Ma, Chongchong Yu ,“ A Multiple Object Tracking Algorithm Based on YOLO Detection” ,IEEE, 2018

[2] ShiJie Sun, Naveed Akhtar, HuanSheng Song, Ajmal Mian, Mubarak Shah ,” Deep Affinity Network for Multiple Object Tracking ”, JOURNAL OF LATEX CLASS FILES, VOL. 13, NO. 9, SEPTEMBER 2017.

[3] HASITH KARUNASEKERA , HAN WANG , (Senior Member, IEEE), AND HANDUO ZHANG “ Multiple Object Tracking With Attention to Appearance, Structure, Motion and Size”,IEEE, 104423-104432 ,VOLUME 7, 2019

[4] Shriharsha S. Veni,Ananda S. Hiremath,Mahalakshmi Patil, Mayuri Shinde, Aishwarya Teli “ Video-Based Detection, Counting and Classification of Vehicles using OpenCV ” ICICNIS 2020.

[5] Samira Karimi Mansoub,Rahem Abri,Anıl Hakan Yarıcı,” Concurrent Real-Time Object Detection on Multiple Live Streams Using Optimization CPU and GPU Resources in YOLOv3 ” , IARIA, 2019, ISBN: 978-1-61208-716-0.

[6] Bin Yan , Pan Fan , Xiaoyan Lei , Zhijie Liu and Fuzeng Yang,” A Real-Time Apple Targets Detection Method for Picking Robot Based on Improved YOLOv5 ”, Remote Sens. 2021,1619.

[7] Fadhlan Hafizhelmi Kamaru Zaman,Syahrul Afzal Che Abdullah, Noorfadzli Abdul Razak, Juliana Johari, Idnin Pasya, Khairil Anwar Abu Kassim,“ Visual-Based Motorcycle Detection using You Only Look Once(YOLO) Deep Network” , ICATAS-MJJIC 2020, doi:10.1088/1757- 899X/1051/1/012004.

[8] Hasith Karunasekera, Handuo Zhang and Han Wang, ” Real Time Multiple Object Tracking using Deep Features and Localization Information ”, ICCA,IEEE,July 16-19, 2019.

[9] KangUn Jo, JungHyuk Im, Jingu Kim, Dae-Shik Kim, ” A Real-time Multi-class Multi-object Tracker using YOLOv2 ”, IEEE ICSIPA,Malaysia,
September 12-14, 2017.