IJARCCE

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 8, August 2021

DOI 10.17148/IJARCCE.2021.10805

A Real Time Object Detection, Classification and Counting of Bengaluru Traffic Surveillance Using AI and ML

G TEJA KRISHNA[1], TEJAS KUMAR Y N[2]

Dept of Information Science and engineering, RV College of engineering, Bengaluru, India[1]

Dept of Information Science and engineering, RV College of engineering, Bengaluru, India[2]

Abstract: Object Detection and tracking of objects is a field that has many applications in this rapidly developing society with more cameras being set up all over the world. Traffic surveillance has become the most pressing issue in increasingly developing cities. Due to poor traffic management in the city of Bangalore, a lot of manpower and hours are being used up. Our project provides a system that detects and monitors vehicles, pedestrians, traffic signals, and signboards and keeps a count of the number of objects per class passing through. This is built using a custom YOLOv4 dataset and functions pertaining to Bengaluru Traffic and implemented using YOLOv4 and Tensorflow. Our model managed to raise the number of objects detected by over 60% to 94% but went down in predicting accuracy from 90% to 65% compared to foreign datasets.

Keywords: YOLOv4, Tensorflow, Non-max Suppression, Darknet.

I. INTRODUCTION

Object detection is a branch of computer vision that is used for identifying and detecting things in images and videos. Object detection is being used to count objects in an image or video, tracking their precise locations, and accurately label them by drawing a rectangular bounding box around them. Classification of the image consists of predicting the class of the object that is being detected. Object detection combines both identification and classification into one to detect and predict the objects. In recent years, Video surveillance of traffic has become a significant problem. Sensing Vehicles and pedestrians during traffic situations is an important aspect in safe driving, accident avoidance, and efficient traffic flow. The fundamental problem here is to identify various classes of objects such as vehicles, pedestrians, road lanes, etc in changing environments and illumination. Especially in rapidly progressing cities where the number of vehicles on the road is immense. Due to the heavy traffic situation in the metropolitan cities of India, proper traffic rules and lane disciplines are not followed which then leads to a wide variety of issues for the detection and classification of objects on the road such as vehicles, pedestrians, sign boards, etc. This in turn leads to multiple vehicles overlapping over each other. Although we are running a Non-max suppression algorithm on the model, there are simply too many vehicles in disorder for the model to recognize. The collection of the custom dataset and labeling and annotating all of them was a particularly challenging task.

In this project, we are using a custom dataset related to Bengaluru traffic collected and sorted through Google and having it run through the model using YOLOv4 and Tensorflow. The applications of this surveillance system can be to keep track of the number of vehicles passing through a certain area at a certain time. This will certainly be helpful for the traffic department to implement this into a more application by adding and modifying this to their needs. The model has been trained using Google Colab and runs using Darknet. This uses a video camera backed with an object detection model which keeps scanning the oncoming traffic which detects objects such as vehicles, pedestrians, and sign boards and sends it over to a control system. Here the video is analyzed using YOLOv4 architecture. The machine learning algorithm utilizes this and helps in detecting objects for the given input. The counting part is done through a custom function using flags.

II. RELATED WORK

Object detection is a programme that uses a specific measure or approach to detect an object in a given scene. Existing to the advent of deep learning technology, object detection was generally performed through the creation of mathematical models based on prior knowledge.

Authors of [1] proposed to use a modified Hungarian algorithm for linear sum assignment problem for vehicle detection and applying linear regression to track the trajectory of the vehicle being detected and predicted. The dataset

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 8, August 2021

DOI 10.17148/IJARCCE.2021.10805

for the proposed project was obtained from publicly available local video surveillance footage and some urban footage that was commonly available. This by no means is ready for universal vehicle detection as the model was trained for local dataset which is very limited.

Another method of utilizing the UV view method for detecting traffic elements using binocular cameras was suggested by authors [2]. By testing the model on various datasets the accuracy obtained was 89%. Peripheral envelope algorithm was used to point out the objects on the road such as vehicles and pedestrians.

Deep learning and AI concepts like MASK R-CNN, CenterNet were used for tracking stationary vehicles and for counting vehicles by [3]. Achieving a 94% accuracy for the vehicles which were stranded but also faced difficulties with detecting distant stationary vehicles.

An overview conducted by [4] on the different techniques for vehicle detection and tracking of different elements of the traffic scene. This went into depth about all the techniques being used and comparing all of them.

According to [5] using daytime labeled images to detect objects in traffic in nighttime images through the means of Faster R-15 CNN produced an accuracy of 82% whereas the other approaches accurately predicted the objects at 86%.

Another deep learning approach to real-time object detection, tracking and distance estimation by [6] using SSD and YOLOv3 resulted in YOLOv3 coming out on top of SSD giving more accuracy than SSD when it came to real- time object detection.

A unique approach by [7] focusing on vehicles and pedestrians detection in real-time, using convolutional neural networks for object detection. The model being capable of real-time detection and that all the results are from real world environments working with different weather conditions too. While maintaining an average fps of 18 the model was able to predict with an accuracy of 62.4%.

According to [8] YOLOv4 can perform almost double the performance of EfficientDet. Improves YOLOv3's performance when compared to its predecessor by almost 10 to 12%. YOLOv4 being the successor to the already wonderful YOLOv3 as it brings in a significant increase in the performance and accuracy for real-time object detection.

As YOLOv4 being the current best object detection algorithm for real-time object detections. The authors of this paper have decided to go ahead with using YOLOv4 paired with Tensorflow for detecting the traffic elements on scene as well as classifying and counting them.

III. DESIGN AND IMPLEMENTATION

A. Architecture

In this paper we investigated the YOLO v4 algorithm as well as tensorflow and its applications on object detection. YOLOv4 is an improvement over the already good YOLOv3 algorithm. Through training and testing, YOLO examines the intact image in order to quietly encode circumstantial information classes and their features. YOLOv4 being a significant upgrade to the previous model of YOLO has increased the mAP and FPS significantly. Now, training the model over a single GPU has become noticeably faster. Object Detection using YOLOv4 and Tensorflow helps in detecting, locating, and tracing an object from an image or a video.

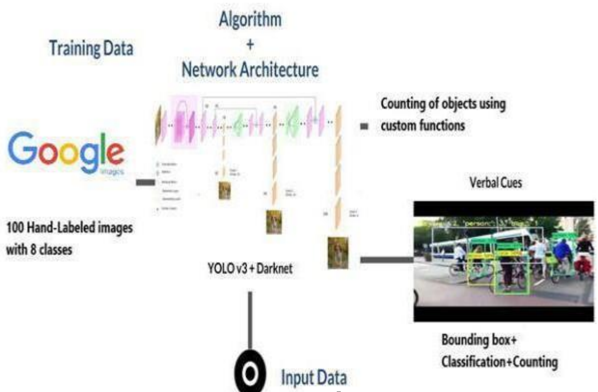


Fig.1: Architecture

B. Methodology

This project uses a video camera backed with an object detection model which keeps scanning the oncoming traffic which detects objects such as vehicles, pedestrians and sign boards and sends the feed over to a control system. Here

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 8, August 2021

DOI 10.17148/IJARCCE.2021.10805

the video is analyzed using YOLOv4 architecture. The YOLOv4 algorithm labels the objects according to its class and then provides a bounding box around it. The model is personally trained with data sets of custom images obtained from google and after sorting it out. The machine learning algorithm utilizes this and helps in detecting objects in the given input. The counting of the detected objects is done using custom functions using flags. A video is a combination of images that forms a moving picture. So we will be detecting the objects in each of these frames and display it one after another in a sequence so that it looks like a video.That is exactly what the Detect program will do. It will take a frame to detect the object in it. Make a box around an object and show the frame. And return the frame with an object bound by a bounding box.

A custom function was written for counting the number of objects detected as a whole and counting the number of objects as per their class. The count will be displayed on the top left corner showing the number of objects detected as per their class. This was done by assigning each class a token and then counting them as they are detected.

C. Dataset Details

The Dataset we have gathered for this project is very limited as it only consists of images gathered from google website related to bengaluru traffic. The dataset has been gathered on following classes:

1. Car
2. Auto
3. Bus
4. Truck
5. Motorcycle
6. Person
7. Traffic Lights
8. Signboards

The data set contains about 100+ images gathered and manually sorted from google about the above mentioned classes. The data set was later labeled and annotated manually for all the images using a labeling tool. Later it was trained using Darknet according to YOLOv4 architecture for around 16,000 iterations for about 8 hours. Although after about nearly 8000 iterations the model showed signs of overfitting after reaching an average loss of 1.67. We compared both the weights at 8000 iterations and 16000 iterations and the accuracy did not vary much.

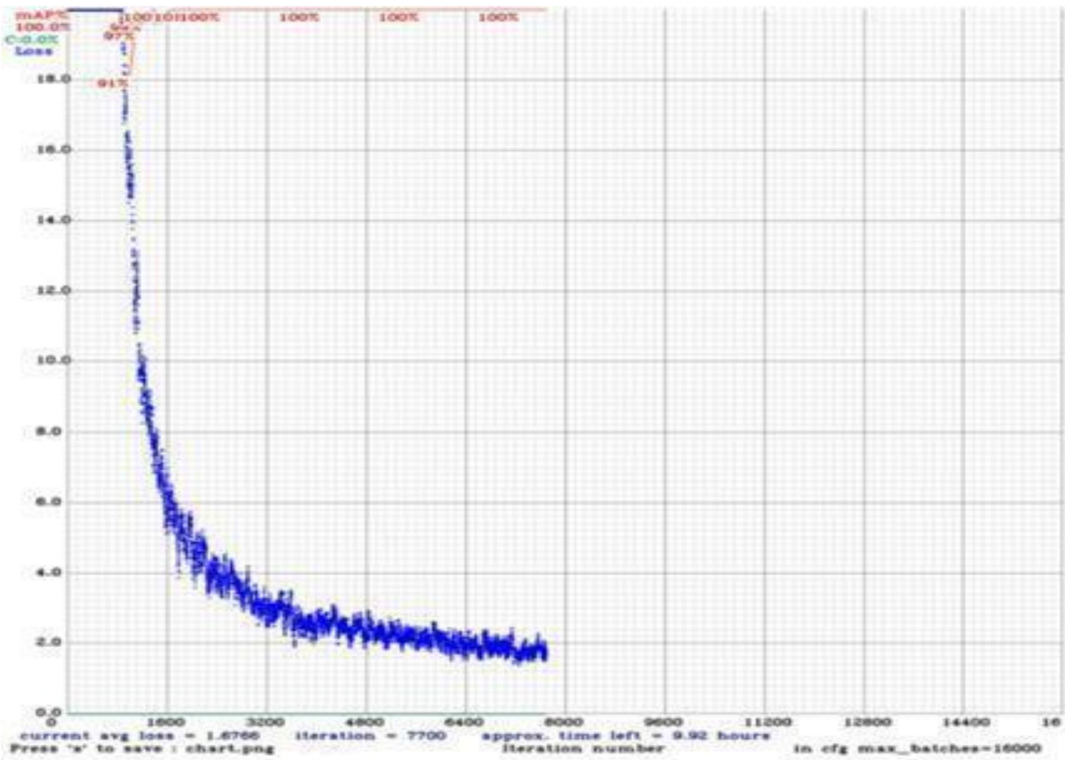


Fig.2: Avg loss vs Iterations at 8000

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 8, August 2021

DOI 10.17148/IJARCCE.2021.10805

D. Implementation details

TensorFlow is a Python library that was used to build the network. Google Collab was used as an environment for training our model. Along with tensorflow some other python libraries like numpy, pandas, keras, scikit learn, opencv, matplotlib, Darknet neural network framework and custom yolov4 weights were also used.

Data preprocessing was done in moderation by resizing all the images and video to a 416x416 scale. The images are also subjected to grayscale conversion. Different variations of the image are fed into the data for accurate predictions.

IV. RESULT ANALYSIS

Our object detection model while running predictions based on limited dataset is able to provide accurate predictions by detecting almost 95% of the objects. Whereas for accurately predicting and classifying the objects, our model is lacking behind the other systems from the literary survey. As our model is using an SSD (Single Shot Detector) that is best for real time object detections, which focuses more on speed of the prediction required for real time rather than the accuracy. Also as the dataset used is limited in nature and the traffic situation of bengaluru is very bad with many vehicles overlapping over each other a lot the predictions may vary a little based on the traffic of bengaluru.

The below figure gives the precision, recall, f1-score, support, avg loss and the avg IoU of the model:

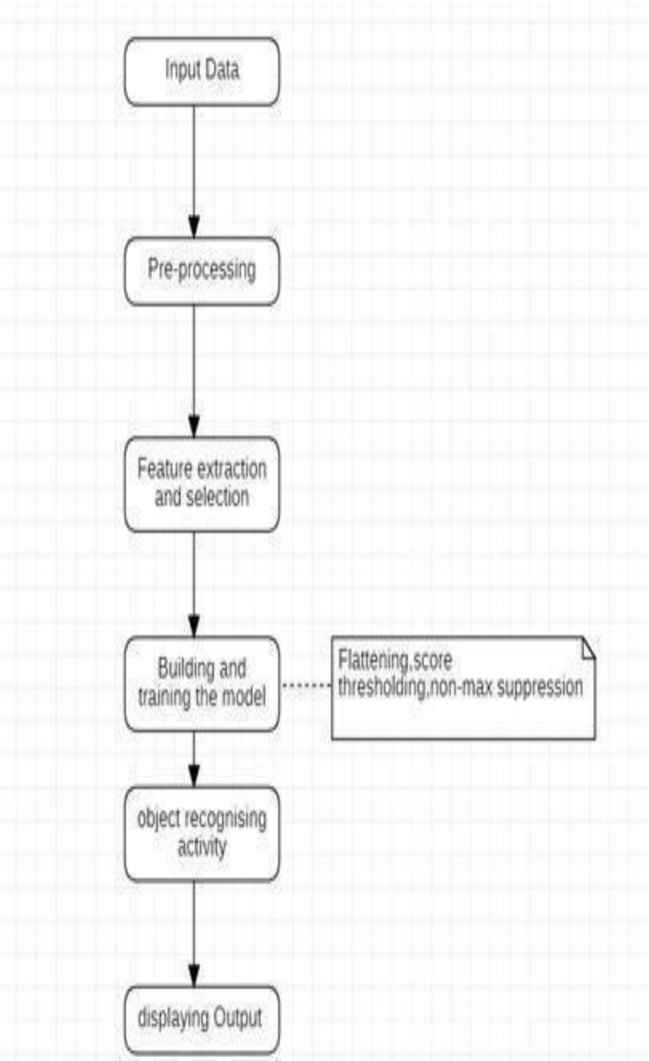
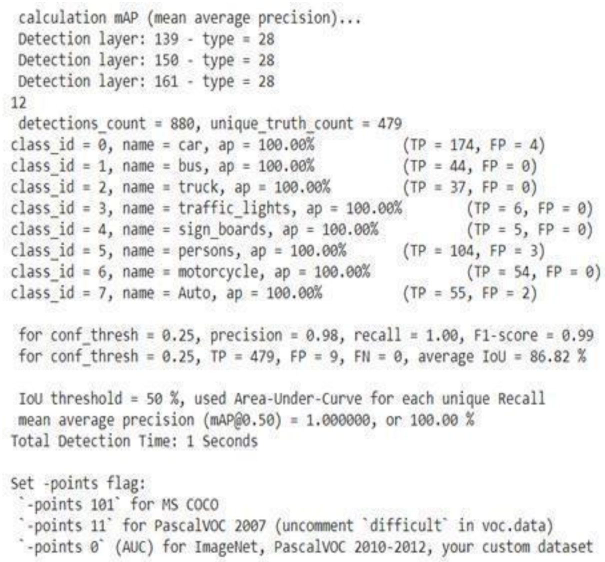


Fig.3: Workflow of the model.



```
calculation mAP (mean average precision)...
Detection layer: 139 - type = 28
Detection layer: 150 - type = 28
Detection layer: 161 - type = 28
12
detections_count = 880, unique_truth_count = 479
class_id = 0, name = car, ap = 100.00%          (TP = 174, FP = 4)
class_id = 1, name = bus, ap = 100.00%          (TP = 44, FP = 0)
class_id = 2, name = truck, ap = 100.00%        (TP = 37, FP = 0)
class_id = 3, name = traffic_lights, ap = 100.00%       (TP = 6, FP = 0)
class_id = 4, name = sign_boards, ap = 100.00%          (TP = 5, FP = 0)
class_id = 5, name = persons, ap = 100.00%      (TP = 104, FP = 3)
class_id = 6, name = motorcycle, ap = 100.00%           (TP = 54, FP = 0)
class_id = 7, name = Auto, ap = 100.00%         (TP = 55, FP = 2)

for conf_thresh = 0.25, precision = 0.98, recall = 1.00, F1-score = 0.99
for conf_thresh = 0.25, TP = 479, FP = 9, FN = 0, average IoU = 86.82 %

IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
mean average precision (mAP@0.50) = 1.000000, or 100.00 %
Total Detection Time: 1 Seconds

Set -points flag:
`-points 101` for MS COCO
`-points 11` for PascalVOC 2007 (uncomment `difficult` in voc.data)
`-points 0` (AUC) for ImageNet, PascalVOC 2010-2012, your custom dataset
```

Fig.4: Result score

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 8, August 2021

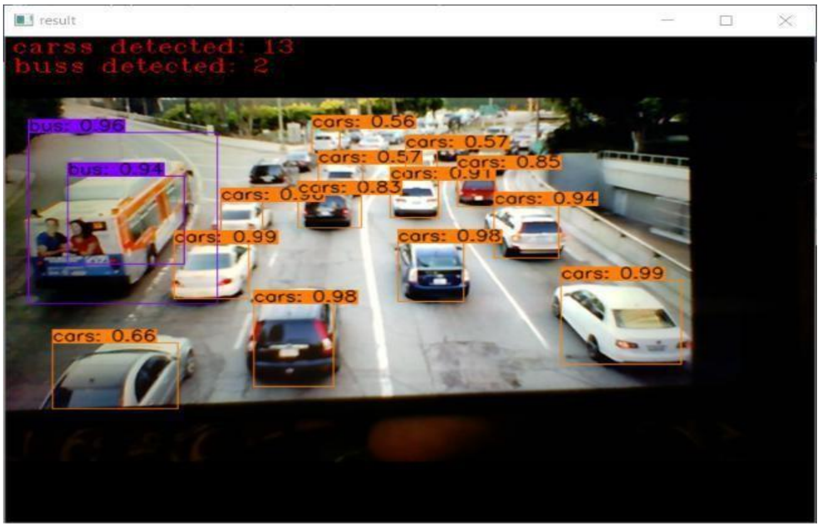
DOI 10.17148/IJARCCE.2021.10805

The below figures show the probability of the object being detected and classified after running the model on a random traffic image of Bengaluru:

Fig.5: Result after running on random image.

Fig.6: Result after running on traffic video.

Real-Time Objection with object counting feature per class results are shown below:



V. CONCLUSION AND FUTURE SCOPE

Our model performs very well according to the limited dataset provided. By detecting objects with an accuracy of more than 90% and prediction accuracy of well over 65%. Training the model in gpu requires more data and time so training for multiple object classes consumes a lot of time. As our training dataset is limited and since we are using SSD for Real-time detections we see a decrease in the number of objects detected in a trade for increasing speed which is needed for Real-Time applications.

To deal with the accuracy problem as well as the overlapping problem a more well defined dataset consisting of more than 1000 to 3000 dataset could be taken and more variations of the images can be provided such as different weather, environment and time of the day images and videos. An alternative algorithm for the Non-max suppression algorithm can also be adopted to solve the overlapping problem.

The traffic monitoring system consisting of video has been growing consistently with the major improvements brought by deep learning and also by the improvements in the field of high end computers. The object detection system can still be improved in its areas of recognition. Yolo has its own drawbacks with its strict requirements of having to use CUDA with GPU to achieve better results in terms of frame rates. This can be avoided by trying to implement yolo using OpenCL.

The performance in the counting of the objects was a success. However, the object counting part of the aspect can be further improved and researched to give a more accurate counting of the objects. This could be useful for many

IJARCCE

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 8, August 2021

DOI 10.17148/IJARCCE.2021.10805

companies which are of the transportation background. With new architectures being discovered and invented, low GPU, simple and more accurate object detection can be achieved in most average hardware. For commercial and industrial application of object detection or any computer vision.

REFERENCES

[1] Vitalijs Komasilov, Aleksejs Zacepins, Ammands Kviesis and Claudio Estevez, "Traffic monitoring using an object detection framework with limited datasets,"5th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2019),DOI: 10.5220/0007586802910296, ISBN: 978-989-758-374-2.

[2] Yongchao Song, Jieru Yao, Yongfeng Ju and Yahong Jiang, "Automatic Detection and Classification of Road, Car, and Pedestrian Using Binocular Cameras in Traffic Scenes with a Common Framework," Research Article, Volume 2020, Article ID 2435793, 17 pages, https://doi.org/10.1155/2020/2435793.

[3] Vishal Mandal, Abdul Rashid Mussah, Peng Jin, "Artificial intelligence-Enabled Traffic Monitoring System," *2 Oct 2020,* arXiv:2010.0127[cs.CV].

[4] Amit Khare, Anshul Vishwakarma, "Vehicle Detection and Tracking for Traffic Surveillance Applications," International Journal of Computer Science and Engineering, Vol.6(7), Jul 2018, E-ISSN: 2347-2693.

[5] Ji, J., Xu, Z., Yu, H., Fu, L., & Zhou, X. (2020). Domain Adaptation For Vehicle Detection In Traffic Surveillance Images From Daytime To Nighttime. Transportation Research Board the 99th Annual Meeting.

[6] Zhihao Chen, Redouane Khemmar, Benoit Decoux, Amphani Atahouet, Jean-Yves Ertaud. Real Time Object Detection, Tracking, and Distance and Motion Estimation based on Deep Learning: Application to Smart Mobility. 2019 Eighth International Conference on Emerging Security Technologies (EST), Jul 2019, Colchester, United Kingdom. 10.1109/EST.2019.8806222. hal-02343350.

[7] Ahmed Abdulkader and Chippen Vlahija, "Real-time vehicle and pedestrian detection, a data-driven recommendation focusing on safety as a perception to autonomous vehicles," Malmö universitet/Teknik och samhälle , 2020. , p. 36.

[8] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," arXiv:2004.10934.