Real Time Vehicle Detection and Counting Using Deep Neural Network

Shilpa Yadav, Shakshi Verma, Prof. Ashish Baiswar
Department of Information Technology,
Shri Ramswaroop Memorial College of Engineering and Management

Abstract

The number of vehicles on the road grows daily. Real-time vehicle counting can efficiently improve traffic management system Aiming at efficiently collecting traffic information in real time, the author proposes an effective vehicle counting system for detection and. tracking vehicles in complex traffic scenes. This article presents a template for detecting different vehicles and counting them within a given video frame.. Using YOLOv4, we can detect the various types of vehicles in the given video.
Additionally, it uses the DeepSORT algorithm to help count the number of vehicles going through the video efficiently. Performance parameters such as iou and map are calculated in order to measure the performance and validation of the work. After implementing the algorithms, Based on the YOLOv4 test results yielded a detection accuracy rate with mAP of 84.50%. Where YOLOv4's combination with the in-depth sorting algorithm can detect, track and count four vehicle types.

Keywords: YOLOv4, DeepSort, mAP ,Tracking

I. INTRODUCTION

With the surge in smart traffic and connections, Multi-vehicle detection and metering has become an important technique to collect traffic data and plays a vital role in smart traffic management and control of the highway [2]. With the popular installation of traffic cameras, it is possible to get a large traffic video database for analysis purposes. These traffic management systems not only reduce delays and blockages due to traffic, but also play an important role in the resolution of major road issues like - Identification of accidents and vehicles moving on incorrect lanes, Verify that the traffic police are performing their duties and display traffic flow data [7]. Modern technologies using deep learning have great potential to replace these material-based systems in a cost-effective manner with less labour.

Most vehicle metering systems can be classified as hardware and software detection systems [9]. The use of standard sensors such as loop detectors or magnetometers, security cameras requires considerable maintenance and the cost of installing these sensors are also expensive. The necessary things that must be kept in mind when designing these models must be compared with the previous research that is being carried out on these traffic challenges, understand the various methods it accuracy and performance statistics in various meteorological conditions like heavy rains, dusty and dense weather [3]. On the other hand, performance also diminishes with the shadows formed by the big buildings, the dense clouds. With all those challenges in mind, an effective data set and training algorithm should be selected to make a contribution to society.. To solve these difficulties various models are brought forward which can accurately detect and count the number of vehicles under different conditions which helps in solving the real time problems in day to day life.

II. RELATED WORK

Different experiments were performed on different datasets. The results have shown clear improvements achieved with the help of object detection and Tracking . To simplify matters, the researchers suggested various methods of real-time vehicle detection and counting. Yang et al. [12] proposed a vehicle detection that uses background subtraction methodology. A low rank decomposition technique is used in the detection process. Despites its favorable performance on constant scenes, it performance decrease when the background scenes change rapidly. In addition, the vehicle counting process is still difficult, and it is important to deal with partial blocking of objects and variations in brightness and contrast. In the future, the paper should be able to accurately detect the objects.

A new approach to count vehicles using R-CNN and track the vehicles using the KLT (Kanade–Tomsi) tracker has been proposed by Abdelwahab et al.[12] combining these two approaches yields better results on the training dataset. Zhe Dai et al.[2] also present a vehicle counting framework that consists of three stages:object detection using yolo3,object tracking using KCF algo,and trajectory processing using region encoding. According to this result,the accuracy of object detection in high traffic and bad weather conditions was 87.6%.Based on Gram and CD2014 datasets, Adson M. Santos et al. [1] design a system that uses YOLOv4 for object detection and Deep SORT for multiple object tracking algorithm; it is shown to be 99.15% accurate in the global count. In addition to counting vehicles, Zuraim et al.[14],also suggested a model based on TensorFlow and looking only once for real-time detection of vehicles.Combining these two and other required dependencies,the presented paper compares the previous version of yolo and picks yolov4 for implementation. In addition ,the system uses DeepSORT to help count the number of vehicles passing in the video effectively .According to this paper,Yolo4 achieves the best results among all available YOLOs having an AP50 of 82.08%.

III. DATA DESCRIPTION

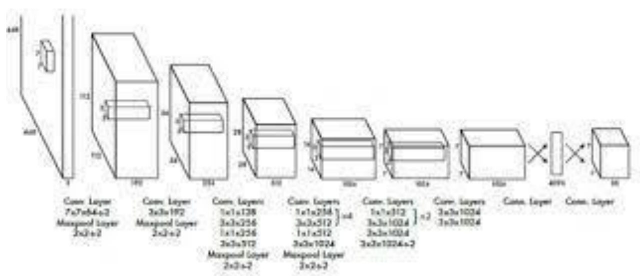
We have accrued images of different classes which include vehicles, buses ,vehicles ,and bikes with the help of Kaggle, robo-flow, and google. After the collection of statistics, we filtered the noisy and blurred pix for higher education of our model. moreover, we additionally adjusted the brightness, hue, and assessment. in the next step with the assist of CVAT (computer imaginative and prescient Annotation device), we've got created bounding boxes and annotation and divided our dataset into two elements:-
(i) 90% for training the model.
(ii) 10% for testing our model

The training of our version on our local system is really time taking and calls for a whole lot of dependencies if we don't have a powerful GPU. With the intention to keep away from this we've got chosen to run our code on Google Colab since it does provide a free GPU and online surroundings. we've got amassed imagesof distinct forms of motors and finished some data augmentation methods on it like resizing the snap shots, brightness adjustment, coloration adjustment, rotation of the pics (clockwise/anti-clockwise), cropping their majestic and created bounding boxes and annotations on them. We've got cut up our information into two scenes: day and night time time and skilled them into 8 instructions (four classes in every scene, which might be bike, truck, bus, and motorcycle).

Images in datasets used for training and evaluation

IV. METHODOLOGY

YOLO model: YOLO is an abbreviation for the term 'You simplest look as soon as'. That is an set of rules that detects and recognizes various objects in apicture (inreal-time).object detection in YOLO is finished as a regression problem and provides the elegance probabilities of the detected photographs.



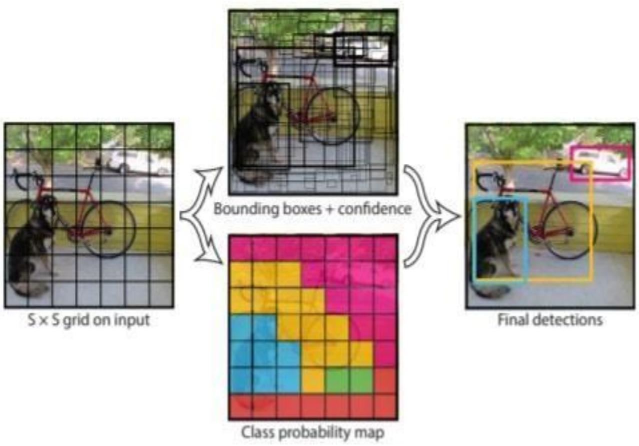
Architecture of YOLO model [5]

This architecture takes a photograph as enter and resizes it to 448*448 by using maintaining the component ratio the same and appearing padding. This photo is then passed on the CNN community. This model has 24 convolution layers, four max-pooling layers followed by means of 2 completely linked layers. For the discount of the wide variety of layers (Channels), we use 1*1 convolutions this is followed by means of 3*3 convolutions. Be aware that the last layer of YOLOv1 predicts a cuboidal output. That is completed via producing (11,470) from the final fully linked layer and reshape in gittosize (7,730). This architecture uses Leaky ReLU as its activation feature in the entire architecture besides the layer where it uses linear activation function. The definition of Leaky ReLU can be determined here. Batch normalization additionally helps regularize the version. With batch normalization, we can put off dropout from the model without over fitting it.

The YOLO set of rules employs convolutional neural networks (CNN) to stumble on items in actual-time.As the name indicates, the algorithm calls for handiest a unmarried ahead propagation through a neural network to locate objects which means prediction inside the whole photograph is accomplished in a unmarried algorithm run. The CNN is used to expect numerous elegance chances and bounding containers concurrently.
The YOLO algorithm consists of various editions. Some of the commonplace ones consist of tiny YOLO and YOLOv3.YOLOv4.

First ,the image is divided into various grids .Each gridhasa measurement of S x S. the following image suggests how an input photo is divided into grids.each grid cell will hit upon items that seem within them. For example, if object middle appears inside a sure grid cell, then this cellular will be liable for detecting it. A bounding container is an outline that highlights an item in an image.Every bounding box in the photo consists of the subsequent attributes: Width (bw), top (bh), elegance (for example, individual, car, visitors mild, and so on.)-
This is represented by the letter ,Bounding box center (bx,via).YOLO makes use of a single bounding field regression to expect the peak, width, center, and sophistication of gadgets.The picture above represents the chance of an item performing inside the bounding container.Intersection over union(IOU) is a phenomenon in item detection that describes how boxes overlap. YOLO makes use of IOU to provide an output box that surrounds the gadgets flawlessly.

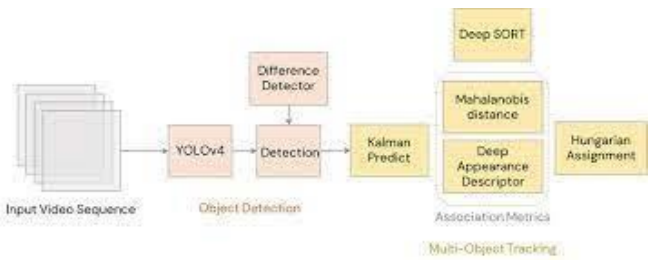
Every grid cellular is accountable for predicting the bounding bins and their self assurance ratings. The IOU is equal to 1if the anticipated bounding field is the same as the real box. This mechanism removes bounding boxes that are not equal to the real box



Object Detection using YOLOv4[6]

***DeepSort**:*

While detecting gadgets in an photo has been getting a number of interest from the scientific network, a lesser recognized and yet an area with full-size packages is tracking objects in a video, something that requires us to merge our understanding of detecting items in static snap shots with reading temporal statistics and using it to quality expect trajectories. Think monitoring sports activities occasions, catching burglars, automating rushing tickets or if our existence is a bit more miserable, alert our self while our 3-12 month old youngster.



Architecture of DeepSort[8]

The Kalman clear out is a critical element in deep type. Our kingdom incorporates eight variables; (u,v,a,h,u',v',a',h') where in (u,v) are facilities of the bounding boxes, a is the element ratio and h, the height of the photo. the opposite variables are the respective velocities of the variables. The variables have simplest absolute position and pace elements, because we're assuming a easy linear pace version. The Kalman and uses prior nation in predicting an awesome suit for bounding boxes. For each detection, we create a "track," that has all of filter helps us element in the noise in detection the essential nation records. It also has a parameter to music and deletes tracks that had their ultimate successful detection long again, as the ones objects would have left the scene. Additionally, to dispose of replica tracks, there may be a minimal quantity of detections threshold for the primary few frames. The Kalman clear out works satisfactory for linear structures with Gaussian processes involved. In our case the tracks hardly ever go away the linear realm and also, most approaches and even noise fall into the Gaussian realm. So, the problem is proper for the usage of Kalman filters [18].

Below diagram clearly explains the working of our proposed model.

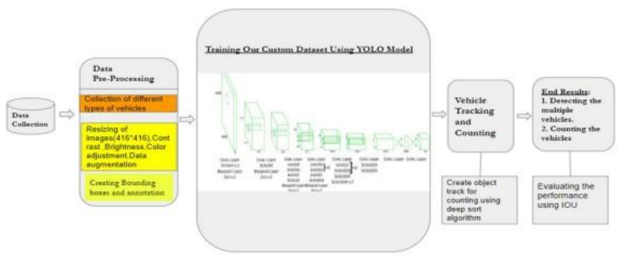


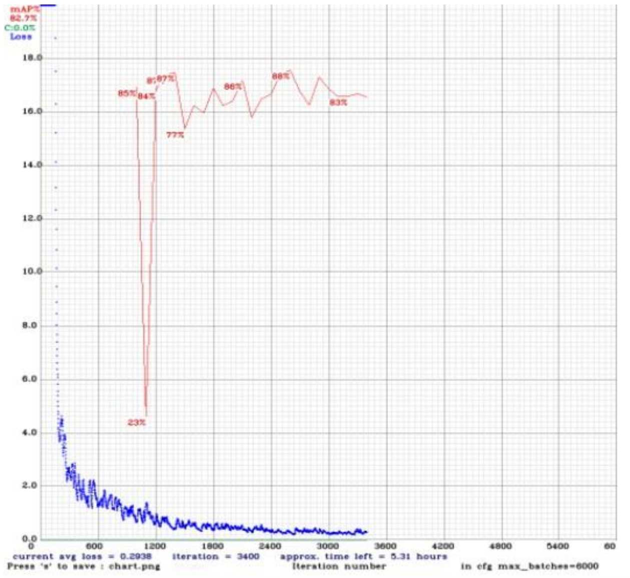
Fig 3.1. Basic Architecture

In our proposed architecture, we have first of all accumulated the snap shots of automobiles using kaggle, roboflow and so forth. Within the next step we've filtered the dataset through adjusting the brightness, hue and comparison. After preprocessing the statistics we've got educated our dataset the use of the YOLOv4 model and are able to reap the weights which is utilized by DeepSORT to matter the automobiles (cars, buses, trucks, bikes) inside the given video efficiently.

IV. RESULTS

In this section we offer the end result obtained through the implementation of the selected technique and assist us to justify the use of the proposed objection detection and tracking approach. The dataset utilized in schooling and checking out the item detection model consists of 25000 pix. Each picture consists of the object belonging from the four training i.e. motorbike, car, bus, truck. And few of the snap shots in the dataset do now not encompass any of the objects belonging from the above-referred to four training, so we've got deleted the ones pix from the dataset. The performance of the object detector and tracker is evaluated on IOU and mAP. The dataset we organized is trained with the help of the YOLOv4 version. we have decided on the v4 version of yolo among all the available alternatives. It uses CNN having twenty four convolution layers, 4 max-pooling layers and two absolutely connected layers.The counting of the objects is applied the usage of Deepsort. This will be achieved with the assist of Kalman filter. In phrases of looks, capabilities similarity,
and motion distance, demonstrated tracks and detections are evaluated. The affiliation findings of proven tracks and detections are then generated the use of the Hungarian approach.With the assist of intersection-over-union (IOU) performance metrics we choose the bounding container on the overlapping inside the video.The Kalman clear out and the motion prediction version are used to replace the a couple of monitoring in the movement nation. Further more we build new tracks for unrelated detections.

Below is the graph plotted between the loss and number of iteration. The graph suggests the two curves one is of blue color and some other is of purple shade. Blue curve shows the loss whilst the red curve shows the mean common precision(mAP) at 50% Intersection-over-Union(IOU) threshold (mAP@0.five). common loss at 3400 iterations is zero.2938 and mAP is eighty four.five.



Graph showing the loss and mAP while training the Yolov4 on custom dataset

The under desk represents the mAP percentage on growing the wide variety of iterations. The better the value of mAP the higher is your detection result.

Iteration	mAP Percentage
1020	85%
1110	23%
1200	84%
1320	86%
1380	87%
1500	77%
2100	86%
2580	88%
3000	83%
3400	84.5%

The under image is output of the monitoring code that is showing the tracking of object in each frame of the video dataset .Each new object in the frame has assigned a few particular identity .This item has been tracked in the subsequent consecutive body similarly if any new item identification is created then general object counter is extended through on.

Snapshot showing the output of the tracking code.

V. CONCLUSION

We are able to see that we will efficiently stumble on and count the cars inside the given video frames containing 4 instructions of motors: motors, buses, vehicles and motorcycles. After training our dataset at the YOLOv4 version we obtain an mAP(suggest common precision) of eighty four.50% and we are also able to locate and depend motors in horrific weather conditions. Our outcomes also help us in understanding diverse deep gaining knowledge of fashions and selecting Yolov4 and deep type for implementation and which enables us in obtaining favored outcome and additionally come ahead with the challenges which want to be advanced in our purposed system

VI. FUTURE WORK

Within the future, we plan to paintings on enhancing the restrictions of our mission that the model is unable to count Indian cars like autos that are used extensively in India. We additionally desire to work on training our model on a dataset containing images of bad climate conditions like heavy rainfall, dusty weather and dense fog and attain higher accuracy and performance.

ACKNOWLEDGEMENT

We appreciate the support of our college Shri Ramswaroop Memorial College of Engineering and Management, Lucknow for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

REFERENCES:

[1] Santos, Adson M., et al. "Counting motors with excessive-precision in Brazilian roads the use of yolov3 and deep type."2020 33rd SIBGRAPI Convention on pictures, styles and photos (SIBGRAPI). IEEE, 2020.

[2] Dai, Zhe, et al. "Video- primarily based vehicle counting framework." IEEE get right of entry to 7 (2019): 64460-64470.

[3] Meng, Qiao, et al. "Video-based vehicle counting for freeway: A novel method primarily based on vehicle detection and correlation-matched tracking the use of photo data from ptz cameras." Mathematical Problems in Engineering 2020 (2020).

[4] Youssef, Yomna, and Mohamed Elshenawy. Computerized Vehicle Counting and Tracking in Aerial Video Feeds Using Cascade R-CNN and Feature Pyramid Networks. No. TRAM-21- 03070. 2021.

[5] Doan, Thanh-Nghi, and Minh-Tuyen Truong. "Real-time automobile detection and counting primarily based on YOLO and DeepSORT." 2020 12th International Conference on Knowledge and Systems Engineering (KSE). IEEE, 2020.

[6] Mahto, Pooja, et al. "Refining yolov4 for car detection." W o r l d w i d e Journal of Advanced Research in Engineering and Technology (IJARET) 11.5 (2020) .

[7] Hu, Hou-Ning, et al. "Joint monocular 3D automobile detection and monitoring ." complaints of the IEEE/CVF global Convention on Computer Vision. 2019.

[8] Lin, Jia-Ping, and Min-Te Sun. "A YOLO-based visitors counting gadget ." 2018 Convention on Technologies and program of synthetic Intelligence (TAAI). IEEE, 2018.

[9] Xiang, Xuezhi, et al. "automobile counting based on vehicle detection and monitoring from aerial films ." Sensors 18.8 (2018): 2560.

[10] Hou, Wei, Dongsheng Xia, and Heo Kyung Jung. "Video avenue automobile detection and tracking primarily based on OpenCV." 2020 International Conference on Information technology know-how and Education (ICISE-IE). IEEE, 2020.

[11] Youssef, Yomna, and Mohamed Elshenawy. "Automatic vehicle counting and tracking in aerial video feeds using cascade vicity -based totally convolutional neural networks and function pyramid networks." Transportation studies Record 2675.8 (2021):
304-317.

[12] Abdelwahab, Mohamed A. "correct vehicle counting approach based on deep neural networks." 2019 global Conference on modern tendencies in pc Engineering (ITCE). IEEE, 2019.

[13] Tang, Yong, et al. "Vehicle detection and recognition for intelligent traffic surveillance systems." Multimedia tools and applications 76.4 (2017): 5817-5832.

[14] Zuraimi, Muhammad Azhad Bin, and Fadhlan Hafizhelmi Kamaru Zaman. "automobile Detection and monitoring using YOLO and DeepSORT." 2021 IEEE 11th IEEE Symposium on pc packages & commercial Electronics (ISCAIE). IEEE, 2021.

[15] Alpatov, Boris A., Pavel V. Babayan, and Maksim D. Ershov. "Car detection and counting machine for real-time traffic surveillance." 2018 7th Mediterranean Conference on Embedded Computing (MECO). IEEE, 2018.

[16] Maity, Madhusri, Sriparna Banerjee, and Sheli Sinha Chaudhuri. "Faster r-cnn and yolo based totally car detection: A survey." 2021 5th International Conference on Computing Methodologies and Communication (ICCMC). IEEE, 2021.

[17] Can, Vuong Xuan, et al. "Vehicle Detection and Counting beneath Mixed Traffic Conditions in Vietnam Using Yolov4." International magazine Advanced Research Engineering a Technology (IJARET) 2 (2021).

[18] Amitha, I. C., and N. K. Narayanan. "Object Detection Using YOLO Framework for smart visitors monitoring." System imaginative and prescient and Augmented Intelligence—idea and Applications.
Springer, Singapore, 2021. 405-412

[19] Hou, Xinyu, Yi Wang, and Lap-Pui Chau. "Vehicle tracking using deep sort with low self assurance track filtering." 2019 16th IEEE worldwide Convention on superior Video and Signal Based Surveillance (AVSS). IEEE, 2019.

[20] Kejriwal, Rahul, H. J. Ritika, and Arpit Arora. "Vehicle Detection and Counting using Deep Learning basedYOLO and Deep SORT Algorithm for Urban Traffic Management System." 2022 First International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT). IEEE, 2022.
.