Project Report

Task 3: Vehicle Detection, Tracking and Speed Estimation

DLBAIPCV01 :- Project computer vision

Professor :- Konstantinos Amplianitis

Name :- Aditya Modi

Matriculation number :- 92115852

Date :- 01-08-2023

Table of content

List of figures

1. Introduction

The world is changing very rapidly so as the traffic management system around the world. But these days road accidents are increasing significantly due to more vehicles on the road and vehicle speed is one the main reason for road accident. To control the rate of accidents on road an effective and convenient traffic management system is needed. An intelligent traffic management system helps to control the vehicle speed and especially the traffic jams which create a chaos mainly during the daytime when everyone is busy in their day jobs. In the past vehicle speed estimation is mainly done by radar systems, which are not very convenient, less accurate and needs high technical support (Khan et al., 2014). Therefore, it is very necessary to build a complete autonomous system which can deal with this problem and to improve safety in road transportation. The task of identifying different objects and extracting meaningful information in an image or video is done by computer vision.

The system tracks the location of the vehicle when it first appeared in video and what direction and the last point before it disappeared. In this way speed is calculated by computing the distance between one point to another and dividing it with time (Khan et al., 2014). However, video calibration also does not give the adject speed of the vehicle but it very accurate and faster than the traditional methods.

The following report will analyse different object detection algorithm for vehicle detection and its performance in different situation based on processing speed and accuracy. Most of the object detection technology based on convolutional neural network (CNN). CNN is a part of deep learning which is most used to analyse visual in an image. Back then the task of analysing different components of an image takes too much time and is very inaccurate as compared to today. Because of the recent improvement in the power of GPUs and the amount of data available to train the artificial intelligence modal completely changed the tech and surveillance industry.

The process of object detection is done by pre-processing the frames of image in a video through different algorithm. We are explaining each algorithm in detail and how each of these algorithms works. The advantage and disadvantages of working with these algorithm and limitations in different situations. There are many different algorithms available on the internet. But for this project we are comparing Faster R-CNN , SSD (single shot multi box detector) and YOLO (you only look once).

2. Faster R-CNN

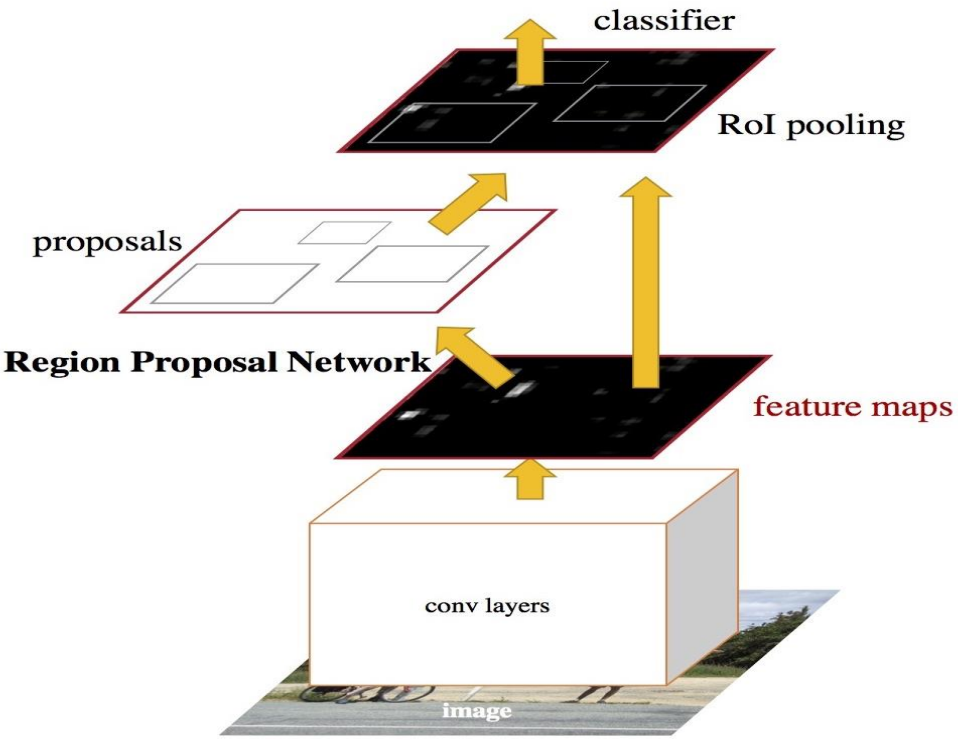
Faster R-CNN modal was developed by a group of researchers at Microsoft. Faster R-CNN is an upgraded version of the R-CNN (Regional based convolutional neural network). Faster R-CNN is a successor of Fast R-CNN. The R-CNN model can do different object detection task gives us the desirable results. The R-CNN modal have some speed issues with training of data and with the accuracy of the results.

To understand the basics of faster R-CNN , we must have a look at R-CNN and Fast R-CNN. The R-CNN modal use search selective algorithm for the extraction of most significant feature of an image. While in Fast R-CNN instead of taking into account all of the sub-segments, it runs the entire picture through the pre-trained Convolutional Neural Network. Faster R-CNN is mainly consisting of two network of reginal proposed network (RPM) and Fast R-CNN (Ren et al., 2017). To provide useful outputs, the region proposal network (RPN) computes pictures at a variety of sizes. An image of any size is fed into a region proposal network (RPN), which generates a series of rectangular 182 object suggestions, each with a score for objectiveness. A selective search strategy is used by fast R-CNN models to compute the area suggestions. This current technique is replaced by the superior region proposal network of the Faster R-CNN approach.

Faster R-CNN composed of two layer one is regional proposed network and second fast R-CNN detector that use proposed region. We need to slide a network of size n*n over the convolutional layer in order to create regional proposals. Once the regional proposals are created, they need to align with feature maps. For each feature maps, there is different anchor boxes which have different varying scale and aspect ratio. Now the region of interest pooling layer features passed through the fully connected layer or the purpose of classification and bounding box regression. The obtained information is then pass-through NMS (non-maximum suppression) to remove any unnecessary duplicate data. Through the elimination of duplicate or overlapping findings, this procedure makes sure that only the most precise forecasts are kept (Ren et al., 2017).

If our compare this to other CNN approaches, the prediction time is quicker. While Fast R-CNN takes around 2 seconds and the Faster R-CNN provides the best result in just about 0.2 seconds, R-CNN typically takes between 40 and 50 seconds to estimate the objects in an image (K, 2022). It shows that faster R-CNN is 200 times faster than the R-CNN modal.

Figure 1: Faster R-CNN modal is a single, unified network for object detection.



2.1 Limitations

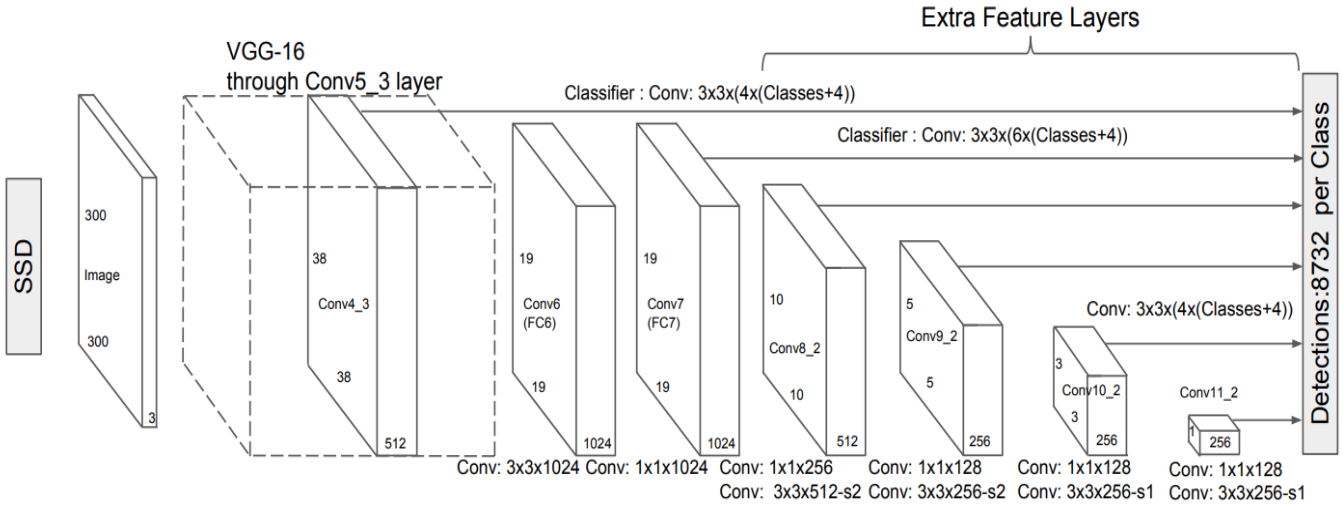
One of the main limitations of Faster R-CNN is time .It is very much faster than other C-NN algorithms, but it takes more time in proposition of different objects when compared to others (K, 2022).

3. Single shot multi box detector (SSD)

Single Shot multi box Detector is one of the fastest ways of detecting object. SSD is the first deep neural network based object detector that does not require features of bounding boxes and accurate as other algorithms are. The SSD method uses a convolutional layer that generate a fixed size bounding boxes and scores for the pre-existing object class instance. The SSD architecture is basically divided into three main parts. In the first step the process of feature extraction is done. Using a set of conventional filters, each additional feature layer may generate a defined set of detection predictions. The fundamental component for predicting the parameters of a possible detection for a feature layer of size m*n with p channels is a 3*3*p tiny kernel that generates either a score for a category or a shape offset in relation to the default box coordinates (Liu et al., 2016). After

applying the kernel at all the m*n locations an output is produced .The second step is training, To train the network properly, we must have to figure out which default boxes correspond to a ground truth detection. For each of the ground truth box we have compare the location and aspect ratio with from the default box and then select the box with maximum overlap.

Figure 2: working of single shot detector.



Most of the default boxes are negatives after the matching stage, especially when there are several alternative default boxes. In the final stage our main task is to reduce the error caused by the default bounding boxes during the overlapping. We do this by choosing the highest confidence loss for each default box. We discovered that doing so results in quicker optimization and more reliable training.

3.1 Limitations

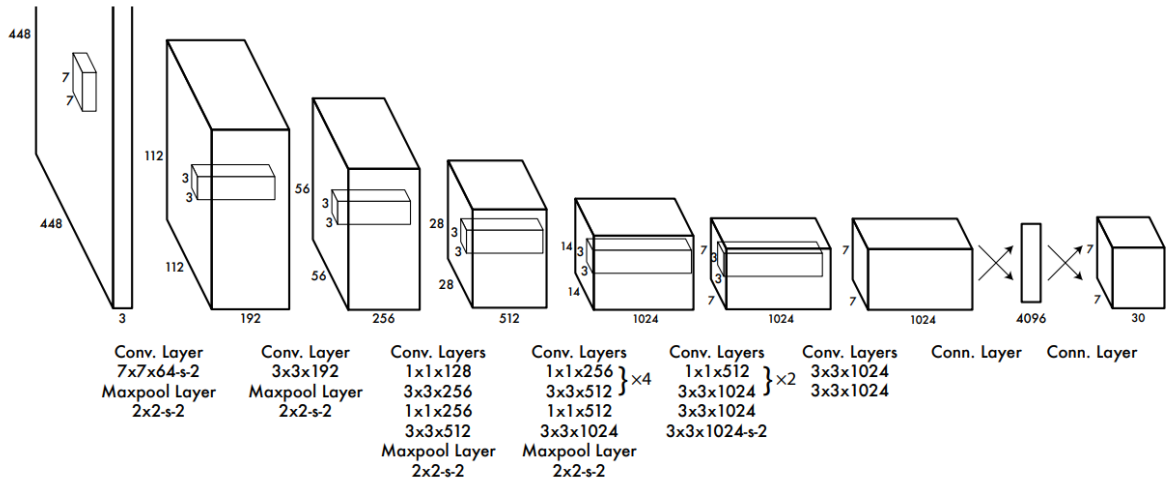
SSD is a great algorithm in boosting the overall performance but significantly decrease the image quality. It can be used for faster prediction and detecting larger object, where the accuracy is not the most important concern for example in case of table, chair and humans it works very well (K, 2022). According to the COCO test-dev dataset result Faster R-CNN is way better than SSD in detecting smaller object. Because Faster R-CNN two improvement steps mainly RPN and Fast R-CNN.

The accuracy of SSD can be improved by data augmentation process for smaller object. Another option to enhance the performance of SSD is to create a better default box tiling that better aligns each location's position and size with the receptive field of the feature map. With these options an increase of 2-3% can be seen in the provided dataset.

4. You only look once (YOLO)

One of the most popular algorithms for object detection is YOLO. It is first created by Joseph Redmon in 2016 (Choudhary, 2022). YOLO is based on Darknet which is a research framework. The algorithm performs object detection using convolution neural network (CNN). YOLO is very popular among its users because of its high accuracy and better performance. YOLO implicitly encodes contextual information about classes in addition to their outward appearance since it views the full image during training and testing. That's why yolo make less than 50% error as compared to Fast R-CNN. YOLO is a very simple algorithm. To forecast detections, we only execute our neural network on a fresh picture at test time. Moreover, it achieves very high mean average precision as compared to other algorithms.

Figure 3: Architecture design of YOLO convolutional layer.



Firstly, the image is divided into grid of boxes. The size of each grid is n*n. As seen in the image. Each of these grids serves as a center point, and a specific estimation is made for each grid in accordance with that. YOLO predicts multiple bounding boxes per grid cell. Each grid cell is responsible for the prediction of bounding boxes. IOU (interaction of union) is used by YOLO to precisely enclose output boxes for objects. If the actual box is similar to bounding box in term of height, width, and class than the IOU score is equals to 1 (Redmon et al., 2016).

4.1 Limitations

Due to the fact that each grid cell may predict just two boxes and only one class, YOLO places substantial spatial limits on bounding box predictions. This geographical restriction

limits how many close items our model can predict. Furthermore, it is very hard for YOLO to predict objects very close to each other due to the limitations to bounding boxes (K, 2022). This limitation can be seen in the initial version of the YOLO but the upgraded version YOLOv5 is way more powerful and improved than the previous versions. It is a PyTorch implementation rather than the initial Darknet framework (Choudhary, 2022).

After its initial launch YOLO have upgraded a lot. It has over 15 different versions ranging from YOLOv1 to YOLOv8. Each of these upgrades is slightly improved then the previous version. The new architecture of YOLO has the ability to learn and develop a clear understanding of different object with time.

5. Comparison of different algorithm

For our project we are comparing three different algorithms to detect vehicles from a CCTV camera footage. While comparing the algorithm on different dataset, it can be clearly seen that the accuracy of Faster R-CNN is relatively high. Because detection is performed in two different steps, but the speed of processing is significant lower as compared to others. So, it causes some problem in detecting faster moving vehicle in each frame (Kim et al., 2020). On the other hand, SSD is relatively faster but very less accurate in detecting objects especially than the smaller ones. The accuracy is also very low that it failed to detect some vehicles on the road. Which is not a good indication as it will not able to detect vehicles that looks small in size. YOLO is best when it comes to accuracy. It detects every vehicle in the video frames. It has an average precision of more than 94% in detecting vehicles and dividing them into categories such as car, jeep and truck.

	YOLO	Faster R-CNN	SSD
Average precision	98.19%	93.4%	90.3%
Frames per second	82.1	36.82	105.14
Accuracy	96%	90%	88%

(Kim et al., 2020)

6. Speed estimation

For speed estimation there is a very simple formula. We are interested in calculating the distance covered by vehicle in two different time frame and then dividing it with the time it

take to go from one frame to another. The speed comes through this method comes in pixel per second and we have to convert it into kilometre per second.

Speed S = d / t , where t = time between two frames

Speed obtained from this formula is in pixel per second. But we need to convert it into kilometres per hour

7. My project

I have used a pre-trained modal available online, which is made using python programming language. The machine learning modal I am using is trained with YOLOv8 and Deepsort libraries. YOLOv8 used for different vehicle detection. In this modal the tracking of different vehicle is done using deep sort. Deep sort is a deep learning algorithm of object tracking. It is an extension to sort (simple online real time tracking). Deep sort is very fast and good in terms of accuracy as compared to alternates like track r-cnn. Here for driving speed the time constant = 15 *3.6, where 15 represents the frame per second while 3.6 is the constant which we can adjust, and as speed = distance/time, but here we are doing speed = distance in meters * time constant. If the speed of the vehicle goes beyond the speed of 60 km/h than our system automatically changes the color of tracking box.

I have used my modal on a video footage to find out the results. The link of that video provided below.

https://drive.google.com/file/d/1-53FSLvL1kdJU-4U1NsCkQtHkuVWRKnd/view

Well, I think our modal stands on our expectations when it comes on to detecting and tracking the object in different time frames. When it comes to speed than I believe there is a little room for improvement because I don't believe that it is 100% accurate but still, it is more accurate than the traditional methods. The speed of the vehicles can be differ in different video footage due to the difference in the camera angle.

8. Conclusion

It is clear that speed estimation through video camera is possible and, it is more efficient and accurate than the traditional methods. After learning about the working and limitations of these algorithms we can now understand that which algorithm can be used for which

situation. Single-shot detectors utilize more intricate designs to be more accurate, and region-based detectors speed up the processes to be quicker. The YOLO, single shot detector, for instance, has qualities that are similar to those of other detectors; the specific difference may not be in the fundamental idea, but rather in the specifics of implementation. It can be clearly seen through different datasets that Faster R-CNN is the fastest modal among all the R-CNN modal. YOLO and SSD have very light weight precision that's why it is very faster than the Faster R-CNN modal. The upgraded version of YOLO improves itself in term of accuracy and speed. Because its structure uses FPN (feature pyramid network), which solve the problem of accuracy in detecting small objects (Kim et al., 2020). Three of these algorithms have different impact on the detection of vehicle and its speed over time. The selection of any of the algorithm depends upon many factors including speed, accuracy or resources availability. Moreover, for higher accuracy more data is needed for training and the performance of our tracking system can be increased with increase in GPU's. YOLO can be a good option in vehicle detection. It is good in terms of both accuracy and speed as compared to SSD and Faster R-CNN which have problem with accuracy and speed. Furthermore, YOLO also many powerful upgrade after its initial launch in 2016. It's most recent version YOLOv8 is launched in 2023 by ultralytics.

Bibliography

Kim, J., Sung, J.-Y., & Park, S. (2020). Comparison of Faster-RCNN, YOLO, and SSD for Real-Time Vehicle Type Recognition. *2020 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia)*, 1–4. https://doi.org/10.1109/ICCE-Asia49877.2020.9277040

K, B. (2022, July 22). *Object Detection Algorithms and Libraries*. Neptune.Ai. https://neptune.ai/blog/object-detection-algorithms-and-libraries

Khan, A., Ansari, I., Sarker, Md. S., & Rayamajhi, S. (2014). Speed Estimation of Vehicle in Intelligent Traffic Surveillance System Using Video Image Processing. *International Journal of Scientific and Engineering Research*, *5*, 1384–1390. https://doi.org/10.14299/ijser.2014.12.003

Choudhary, A. S. (2022, September 22). Object Detection Using YOLO And Mobilenet SSD. *Analytics Vidhya*. https://www.analyticsvidhya.com/blog/2022/09/object-detection-using-yolo-and-mobilenet-ssd/

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single Shot MultiBox Detector. In B. Leibe, J. Matas, N. Sebe, & M. Welling (Eds.), *Computer Vision – ECCV 2016* (pp. 21–37). Springer International Publishing. https://doi.org/10.1007/978-3-319-46448-0_2

Khan, A., Ansari, I., Sarker, Md. S., & Rayamajhi, S. (2014). Speed Estimation of Vehicle in Intelligent Traffic Surveillance System Using Video Image Processing. *International Journal of Scientific and Engineering Research*, *5*, 1384–1390. https://doi.org/10.14299/ijser.2014.12.003

Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-Cnn: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). *You Only Look Once: Unified, Real-Time Object Detection* (arXiv:1506.02640). arXiv. https://doi.org/10.48550/arXiv.1506.02640