International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2454-9150 Vol-09, Issue-01, Apr 2023

A Video Streaming Vehicle Detection Algorithm Based On Yolov4

[1]Prof. Vishal R. Shinde, [2]Mr. Aditya J Shirsekar, [3]Mr. Vihang R Waravdekar, [4]Mr. Rushikesh T Mokal, [5]Mr. Pratik P Lingayat
[1]Asst.Prof, [2,3,4,5] UG Student, [1,2,3,4]Computer Engg.Dept. Shivajirao S. Jondhle College of Engineering & Technology, Asangaon, Maharashtra, India.
[1]mailme.vishalshinde@gmail.com , [2]shirsekaraditya@gmail.com, [3]vihangrw@gmail.com , [4]rushi.mokal@gmail.com , [5]pratiklingayat@gmail.com

Abstract - Safe driving depends on being able to sense information about the environment around the car, and one of the most important technologies to solve this problem is computer vision. The Series YOLO and SSD, Retina Net method are two one-stage target identification algorithms that excel in both accuracy and speed. Vehicle detection is still some distance from real-time, but the newest algorithm in the YOLO series, YOLOv4, has improved the rapidity and precision of vehicle target recognition compared to prior versions. This research suggests a faster vehicle target recognition technique based on YOLOv4 to address the issue with the slow detection speed. The YOLOv4 technique is theoretically introduced in this paper, followed by a suggestion for an algorithmic method to quicken the detection speed, and finally, actual road tests. According to testing results, the method presented in this research can increase detection speed without compromising accuracy, which can act as a foundation for decision-making for secure vehicle operation.[1]

Keywords-YOLOv4, Video Streaming, Vehicle Detection, Retina Net, algorithm fusion.

I. INTRODUCTION

With the rapid development of Country's economy and productivity, Country's per capital car ownership is on the rise. While cars bring us the convenience of daily life, they also bring us potential safety hazards. To reduce such safety hazards, it is necessary for vehicles to sense their surroundings and thus make corresponding responses to different environments. In the subject of intelligent driving, computer vision technology is a critical component that enables vehicles to detect other obstacles in front of them, such as other vehicles, people, etc. LIDAR and other data can be combined with other sensors like millimeter wave radar to give the car a better feel of its surroundings and increase driving safety. The two most widely used detection approaches in the field of automobile target recognition right now are deep learning-based techniques and conventional image processing-based techniques. Traditional feature-based detection techniques rely on human feature design and feature extraction. Convolutional neural networks, which have strong generalization, resilience, and detection effects, are often used for learning and feature extraction in deep learning-based approaches, enabling end-to-end instruction and detection. This method is simple to understand and has a quick computational speed, but the algorithm's performance in terms of robustness and generalization is poor. Deep learning-based methods can achieve end-to-end instruction and detection because they frequently use convolutional neural networks for learning and feature extraction, which have good generalization, robustness, and detection effects. This method is simple to understand and has a quick computational speed, but the algorithm's performance in terms of robustness and generalization is poor. The target Convolutional neural network-based detections separated into one-stage target detection algorithm and two-stage target detection algorithm depending on whether the candidate frame of the possible target has to be retrieved in advance. The two-stage representative algorithms R- CNN, fast R- CNN, faster R CNN, R- FCN, and others are examples. SSD, Retina Net, the YOLO series, etc., are examples of one-stage representative algorithms. In this study, vehicle targets are detected using the YOLOv4 algorithm while real-time and accuracy requirements are taken into account.[3]

II. AIMS AND OBJECTIVE

a) Aim: This project's primary goal is to locate and identify one or more useful targets using still image or video data. It thoroughly incorporates a range of crucial methods, including machine learning, artificial intelligence, pattern recognition, and image processing.

b) Objective: The authors claim that YOLOv4 was designed to be a rapid object detector for use in manufacturing that is also optimized for parallel calculations. It had to produce compelling object detection findings quickly and precisely. YOLOv4 was developed as a rapid object detector for usage in production systems that is also optimized for parallel

 DOI : 10.5281/zenodo.7856500

calculations, claim the authors. It had to produce compelling object detection findings quickly and precisely.

III. LITERATURE SURVEY

Paper 1 (YOLOv4): Optimal Object Detection Speed and Accuracy

Many different features claim to increase accuracy of a convolutional neural network (CNN). These feature combinations must be tested in practise on substantial datasets, and the results must be theoretically justified. While some aspects, like as batch-normalization and residual-connections, are relevant to the majority of models, tasks, and datasets, others only function on specific models, for specific issues, or only for small-scale datasets. Weighted-Residual-Connections, Cross-Stage-Partial-Connections, Normalization of cross-mini-Batch, Self-adversarial-Training, and Mish-activation are a few examples of such universal properties taken into consideration in this work. Examples of such universal qualities taken into account in this work include Weighted-Residual-Connections, Cross-Stage-Partial-Connections cross-minor-Batch normalization, Self-adversarial-Training, and Mish-activation. New features, such as wrc, csp, cmbn, SAT, Mish activation, and Mosaic data augmentation, were used in this work. Combining CMBN, Drop Block regularization, and CIoU loss yields cutting-edge results . For the Tesla V100, real-time performance regarding an Ms COCO dataset was 43.5 percent AP (65.7 percent AP50).[7]

Paper 2 (Retina Mask): Free state-of-the-art single-shot detection improved by learning how to recognize masks

In the accuracy vs. speed trade-off, two-stage detectors recently outperformed single-shot detectors. However, single-shot detectors are extremely common in embedded vision applications. This article elevates single-shot sensors to the level of contemporary two-stage techniques.For the first time, instance mask prediction is integrated into training for the cutting-edge individual-shot detector Retina Net. There are more complex scenarios covered, and the error function is more adaptable and stable. The resultant enhanced network is known as Retina Mask. Retina Mask's detection component utilizes the very same amount of work as Retina Net's original version, but it is more precise. Even if the runtime during assessment is the same, the COCO test-dev findings for RetinaMask-101 a maximum of 41.4mAP vs. 39.1 mAP for RetinaNet-101. RetinaMask-101's functionality is increased by Group Normalization to 41.7 mAP. Retina Mask's detecting feature is more accurate. Even if the runtime during assessment is the same, the COCO test-dev findings for RetinaMask-101 a maximum of 41.4mAP vs. 39.1 mAP for RetinaNet-101. RetinaMask-101's functionality is increased by Group Normalization to 41.7 mAP.[5]

Paper 3 : Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition: Present-day neural nets (CNNs) demand a fixed-size (for instance, 224x224) input imageThis "artificial" criterion may decrease computer detection performance for pictures or anti - anti of any size or scale. The networks in this research study all around aforementioned restriction by employing a different pooling technique called "spatial pyramid pooling."No matter how big the imageor scale, the cutting-edge network structure known as SPP-net may offer a fixed-length representation. Small object deformations can be accommodated via pyramid pooling. These benefits suggest that SPP-net will enhance all CNN-based image categorization techniques generally. This study demonstrates that SPP-net enhances accuracy Regardless of the fact that various CNN configurations have different designs, they all performed well on ImageNet 2012 dataset. SPP-net delivers state-of-the-art classification scores just on Cartesian Vox 2007 and Caltech101 data with such a fundamentally sufficient representation and without tweaking.SPP-net delivers state-of-the-art classification scores just on Cartesian Vox 2007 and Caltech101 data with such a fundamentally sufficient representation and without tweaking. Object detection is greatly enhanced by SPP-strength. Only once is it necessary to compute the feature mappings for the entire image using SPP-net, after which the features are gathered into a pool. Fixed-length representations can be provided by using arbitrary sections (sub-images) to train the detectors. The convolutional features are not repeatedly produced when using this method. While maintaining more or equal quality on MATLAB VOC 2007, the new approach is 24-102x quicker than the R-CNN technique while processing test photographs.This techniques place third in picture classification and second in object recognition among the 38 teams competing in the 2014 ILSVRC. The modifications made for this competition are also introduced in this document.[9]

IV. EXISTING SYSTEM

Reducing traffic accidents and accelerating car regulation have received a lot of attention recently. The most used methods for measuring the speed of moving vehicles are technologies used in aircraft or video frames, radar, photo detection and ranging, drones' radar, LIDAR tachometer clocks, speed limit computers, etc. Most of these methods are very expensive and don't always offer enough precision. The accountable authorities have implemented traffic lights and speed of the vehicle detection systems depending on techniques like RADAR or LIDAR. The detection function in Retina Mask uses the same amount of effort as Retina Net's previous incarnation, but it is more accurate. Even though the assessment runtime is the same, the overall COCO testing results for RetinaMask-101 a maximum of 41.4plot compared to 39.1 plot for RetinaNet-101.Group Normalization brings RetinaMask-101's capability up to 41.7 map. undertaken several attempts to curb speeding (Laser Infrared Detection and Ranging). Although RADAR and LIDAR-based techniques perform well, they are also quite expensive. Due to their increased cost, these technologies cannot be widely implemented on roadways.

 DOI : 10.5281/zenodo.7856500

V. COMPARATIVE STUDY

Sr No.	Author	Project Title	Publication	Technology	Purpose
1	Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao	YOLOv4: Optimal Object Detection Speed and Accuracy	IEEE 2021	YOLO v4	Many different features claim to increase Convolutional Neural Network accuracy. It is necessary to evaluate these feature combinations in practise on huge datasets and to theoretically support the findings..
2	Cheng-Yang Fu, Mykhailo Shvets, Alexander C. Berg	RetinaMask: Freely enhances a cutting-edge single-shot detection by learning to anticipate masks	IEEE 2021	RetinaNet	By doing this study, single-shot detectors are brought up to par with current two-stage techniques. We achieve this by improving Retina Net, a state-of-the-art single-shot detector, in three ways: first, by integrating instance mask prediction; second, by making the loss function adaptable and more stable; and third, by including more challenging cases.
3	Zhou Pu, Kai Wang, Kai Yan	Deep Convolutional Networks with Spatial Pyramid Pooling for Recognition	IEEE 2021	Faster R-CNN	This "artificial" criterion might make it harder to correctly identify images or semi of any scale. We employ a distinct pooling strategy known as "spatial pyramid clumping up" on the network in just this study to ensure compliance with the aforementioned requirement.

Table.5.1 comparative study

[1]YOLOv3: YOLOv4 is an improvement over YOLOv3, with higher accuracy and faster speed.

[2] RetinaNet: YOLOv4 is faster and more accurate than RetinaNet on most benchmarks. However, RetinaNet performs better on small object detection tasks.

[3] Faster R-CNN: YOLOv4 is generally faster than Faster R-CNN, while achieving similar accuracy. However, Faster R-CNN has higher memory requirements and is more complex to implement.

Overall, YOLOv4 is a very competitive object detection model, with high accuracy and fast speed. Its design improvements have made it one of the top-performing object detection models available. However, the choice of model ultimately depends on the specific needs of the application, such as accuracy requirements, speed, and available computing resources.

VI. PROBLEM STATEMENT

Finding a moving vehicle with a camera is called vehicle tracking. It is challenging for the application to capture the car in the security camera's video sequence in order to increase tracking performance. Thanks to this technology, there are an increasing number of applications for controls over and monitoring of traffic, traffic flow, security, etc. Utilizing this technology is expected to be relatively inexpensive. In numerous cities and metropolitan areas, traffic surveillance, analysis, and monitoring have all been done using video and image processing. Building an autonomous system that can precisely localize and monitor the speed of any vehicles that show in overhead video frames is the goal.[2]

VII. PROPOSED SYSTEM

Deep learning and traditional image processing-based detection techniques are the foundation of the bulk of vehicle target recognition systems currently in use. Traditional feature-based detection methods rely on manually designed and extracted features. Even though this method is easy to understand and has a quick computing speed, its resilience is less than that of deep learning-based methods. Deep learning-based methods can achieve end-to-end training and detection because they frequently use convolutional neural networks for learning and feature extraction. They offer effective detection, resilience, and generalization properties. There are 2 types of target identification algorithms that use convolutional neural networks: one-stage target detection

 DOI : 10.5281/zenodo.7856500

algorithms and two-stage target detection algorithms. R-CNN, fast R- CNN, faster R CNN, R- FCN, and other two-stage representative algorithms are available depending on whether the candidate frame of the potential target needs to be extracted beforehand. A sample algorithm for one stage. Series, SSD, Retina Net, etc. is called YOLO. In this paper, the YOLOv4 algorithm is utilized to achieve the detection of vehicle targets while taking into account the needs of real-time and accuracy.[10]

VIII. ALGORITHM

Step 1: Begin

Step 2: Loading a sample image

ap = argparse.ArgumentParser()

ap.add_argument("-i","--image",required=True,help="path to input image")

ap.add_argument("-y","--yolo",required=True,help="base path to YOLO directory")

ap.add_argument("-c","--confidence", type=float, default=0.5,help="minimum probability to filter weak detections")

ap.add_argument("-t", "--threshold", type=float, default=0.3,help="threshold when applyong non-maxima suppression")

args = vars(ap.parse_args())

Step 3: Object Detector

print("[INFO] loading Darknet from disk...")

netcv2.dnn.readNetFromDarknet(configPath, weightsPath)

Step 4: Load our input image

image = cv2.imread(args["image"])

(H, W) = image.shape[:2])

Step 5: cycle through each output layer box and populate our lists of observed bounding boxes, confidences, and class IDs.

Assurances = []

in layer, classIDs = [] for output

Outputs:

cycle through each detection for output detection:

Take the current object detection scores and extract the class ID and confident (i.e., likelihood) [5:]

np.argmax classID (scores)

score = confidence[classID]

Step 6: Ensuring the detection

If len(idxs) > 0:

loop over the indexes we are keeping

for i in idxs.flatten():

extract the bounding box coordinates

(x, y) = (boxes[i][0], boxes[i][1])

(w, h) = (boxes[i][2], boxes[i][3])

draw a bounding box rectangle and label on the image

color=[int(c) for c in COLORS[classIDs[i]]]

cv2.rectangle(image, (x, y), (x + w, y + h), color, 2)

text="{}:{:.4f}".format(LABELS[classIDs[i]], confidences[i])

cv2.putText(image, text, (x, y - 5), cv2.FONT_HERSHEY_SIMPLEX,

0.5, color, 2)

Step 7: Output

cv2.imshow("Image", image)

cv2.waitKey(0)

IX. MATHEMATICAL MODEL

The image is divided into grid cells of size $S \times S$ by YOLOv4 before entering the neural network in which grid the center of the object is in the image, and the corresponding mesh is only responsible for predicting the object. Predict B bounding boxes per grid and give the confidence of that box; each box contains five variables, as defined in below Equation.

$$T=[x,y,w,h,confidence]$$

where the target prediction frame's centroid positions are x and y and the prediction frame's breadth and height, accordingly; confidence is the level of certainty associated with the forecast category, and the following equation shows how it is calculated.

$$\text{Confidence} = \mathrm{P}_r(\text{Object}) \times IOU_{pred}^{truth}, \mathrm{P}_r(\text{Object}) \in \{0,1\}$$

Additionally, there is Class information, which indicates the class of own dataset and is shown in the equation below.

$$\text{Class}= [\text{Class1}, \text{Class2}, \ldots, \text{Class}_C]$$

This yields the final tensor output, where Class is the lot of options available in the dataset and B is often considered to be 2, is determined as stated in the following equation.In YOLOv4, Loss is distributed into three parts: one is the error brought by the x, y, w and h, which is the loss brought by the bounding box location; another is the error brought by the category; the third is the error brought by the confidence level. Unlike YOLOv3, YOLOv4 uses CIOU. Based on IOU, CIOU takes into account scale information on border

 DOI : 10.5281/zenodo.7856500

overlaps, enter length, and aspect ratio; the loss function's formula is given by

$$Loss = L_{CIOU} + L_{conf} + L_{cls}$$

$$L_{conf} = \sum_{i=0}^{S^2}\sum_{j=0}^{B} 1_{ij}^{obj}\left[\hat{O}_i \log(O_i) + (1-\hat{O}_i)\log(1-O_i)\right] -$$

$$\lambda_{noobj}\sum_{i=0}^{S^2}\sum_{j=0}^{B} 1_{ij}^{noobj}\left[\hat{O}_i \log(O_i) + (1-\hat{O}_i)\log(1-O_i)\right]$$

$$L_{ds} = -\sum_{i=0}^{S^2} 1_{ij}^{ojj} \sum_{o\in dass}\left[\hat{P}_i(o)\log(p_i(o)) + (1-\hat{P}_i(o))\log(1-P_i(o))\right]$$

$$L_{IOU} = \sum_{i=0}^{S^2}\sum_{j=0}^{B} 1_{ij}^{obj}(2 - w^{gt} \times h^{gt})[1 - CIOU(X,Y)]$$

$$CIOU(X,Y) = IOU(X,Y) - \frac{\rho^2(X_{dr}Y_{ctr})}{m^2} - uv$$

Fig.9.1 loss function

X. SYSTEM ARCHITECTURE

The backbone, neck, and head are the three components that make up the you just look once version 4 (YOLO v4) object recognition network, which would be a one-stage object detection network.

[1]The backbone can be a pretrained convolutional neural network such as VGG16 or CSPDarkNet53 trained on COCO or ImageNet data sets. The feature extraction network, which creates image features from the input photos, serves as the foundation of the YOLO v4 network.

[2]The neck connects the backbone and the head. It is composed of a spatial pyramid pooling (SPP) module and a path aggregation network (PAN). The neck concatenates the feature maps from different layers of the backbone network and sends them as inputs to the head.

[3]The head processes the aggregated features and predicts the bounding boxes, objectness scores, and classification scores. One-stage object detectors, like YOLO v3, are used as detection heads in the YOLO v4 network.

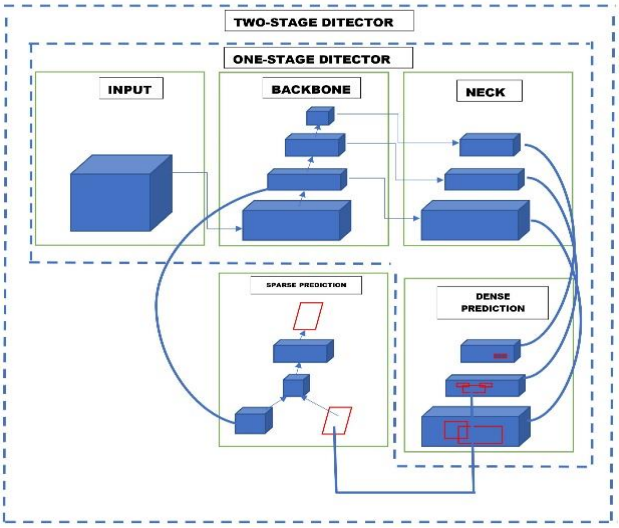

Fig.10.1 system architecture

XI. ADVANTAGES

The YOLOv4 algorithm first suggests an algorithmic method to increase the detection speed before doing actual road tests. The approach presented in this paper can increase detection speed without sacrificing accuracy.

When comparing performance, YOLOv4 is twice as quick as EfficientDet (a competitive recognition model). Additionally, AP (Overall Precision) & FPS improved to 10% and 12% in comparison with old , respectively.Improved network architecture: YOLOv4 incorporates several architectural improvements, including residual connections, spatial pyramid pooling, and a novel CSPDarknet53 backbone architecture, which enhance its performance. Open-source code: YOLOv4 is open-source, which means that it can be freely downloaded, modified, and used by researchers and developers for a wide range of applications.

XII. DESIGN DETAILS

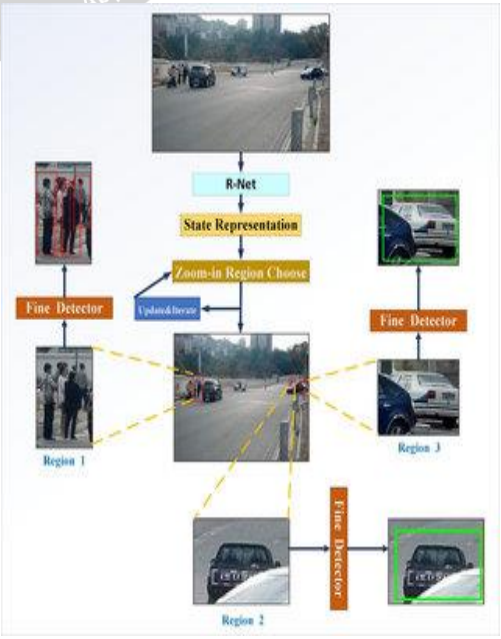

Fig12.1.design details

 DOI : 10.5281/zenodo.7856500

Here are some key design details of YOLOv4:

[1] Backbone network: YOLOv4 uses the CSPDarknet53 network as its backbone. It is a modified version of the Darknet-53 architecture, which consists of several convolutional layers and residual connections.

[2]Spatial pyramid pooling (SPP): YOLOv4 uses SPP to improve its ability to detect objects at different scales. SPP extracts features at multiple scales and concatenates them to produce a final feature map.

[3]Path aggregation network (PAN): YOLOv4 uses PAN to aggregate features from different scales. It consists of two branches, one for fine-grained details and the other for coarse features. The two branches are merged to produce a final feature map.

[4]Mish activation function: YOLOv4 uses the Mish activation function instead of the commonly used ReLU function. Mish is a smooth function that has been shown to improve model performance.

[5] Spatial attention: YOLOv4 uses spatial attention to highlight important regions of the input image. This helps the model focus on relevant information and ignore irrelevant background.

[6] Self-adversarial training: YOLOv4 uses self-adversarial training to improve its robustness against adversarial attacks. In self-adversarial training, the model is trained on adversarial examples generated from the same model.

[7]Bag of freebies (BoF) and bag of specials (BoS): YOLOv4 uses BoF and BoS to further improve its performance. BoF refers to a set of training techniques such as label smoothing and mixup, while BoS refers to model design choices such as anchor-free detection and multi-input weighted residual connections.

XIII. CONCLUSION

Thus we have tried to implement “A Video Streaming Vehicle Detection Algorithm Based on YOLO V4”,Hu, X., Wei, Z., & Zhou, W. IEEE, 2021 and the conclusion as follow:

The results of thereal-world road testing demonstrate that the enhanced YOLOv4 detection algorithm is capable of accurately detecting approaching vehicles and pedestrians. When tracking with the Camshift algorithm, the fused vehicle detection method successfully increases the overall detection speed from around 10 FPS to about 16 FPS while essentially maintaining the algorithm's detection effect. As a result, the technique developed in this research has increased the speed at which video streams can be detected.

REFERENCE

Hu, X., Wei, Z., & Zhou, W. (2021). A video streaming vehicle detection algorithm based on YOLOv4. 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC).

[1] Prof.Vishal R.Shinde,"An application of deep learning algorithm for automatic detection of unexpected accidents under bad CCTV”in IJREAM, ISSN : 2454-9150, Volume 08, Issue 01, APR 2022 Special Issue.

Yang, X., Dong, F., Liang, F., & Zhang, G. (2021). Chip defect detection based on deep learning method. 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA).

[2] YOLOv4: Optimal speed and accuracy of object detection. arXiv 2020 A Bochkovskiy, CY Wang, HYM Liao - arXiv preprint arXiv:2004.10934, 2020 2086

[3] Cheng-Yang Fu, Mykhailo Shvets, and Alexander C Berg. RetinaMask: Learning to predict masks improves stateof-the-art single-shot detection for +2019

[4] Gayatri Naik,”An Efficient Multi User Searchable Encryption Scheme without QueryTransformation over Outsourced Encrypted Data”,International Journal for Research in Engineering Application & Management (IJREAM) ISSN : 2454-9150,ISSN:2454-9150,Special Issue -iCreate - 2019.

[5] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-FCN:Object detection via region-based fully convolutional networks. In Advances in Neural Information Processing Systems (NIPS), pages 379–387, 2016.

[6] Xiu C. and Ba F. 2016 2016 Chinese Control and Decision Conference (CCDC) (Yinchuan) Target tracking based on the improved Camshift method 3600-3604

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatialpyramid pooling in deep convolutional networks for visual recognition.IEEE Transactions on Pattern AnalysisandMachine Intelligence(TPAMI), 37(9):1904–1916, 2015.

[8] Girshick, R. (2015). Fast R-CNN. 2015 IEEE International Conference on Computer Vision (ICCV).

 DOI : 10.5281/zenodo.7856500