International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 09 Issue 12 ǁ 2021 ǁ PP. 01-04

An Object Detection Using State-Of-Art Deep Learning Yolo Network

Mr. KIRAN D N [1], Mrs. ANURADHA K N [2], Leena Rani A [3]
[1]Assistant Professor, Department of Computer Science, Seshadripuram College, Tumkur
[2]Assistant Professor, Department of Computer Science, Seshadripuram College, Tumkur
[3]Student, Department of MCA, Siddaganga Institute of Technology, Tumkur

Abstract

Deep learning-based object detection has shown to be quite effective. In the actual world, however, there are numerous issues with photographs, such as noise, blurring, and spinning jitter, to name a few. Object detection suffers as a result of these issues. To train a robust model, the authors utilized the YOLO-V4network. In the field of object detection, deep learning has demonstrated outstanding results. YOLO-V4 is a state-of-the-art deep learning-based object identification method that performs well in terms of both speed and accuracy. To conduct YOLOv4 detections, the model was first trained using the well-known AlexeyAB's darknet dataset. The YOLOv4 object identification neural network based on the CSP technique scales up and down and applies to small and large networks while maintaining optimal speed and accuracy, as demonstrated in this study. We suggest a network scaling method that alters not just the network's depth, width, and resolution, but also its structure.

Keywords: *YOLO network, object detection, deep learning.*

Date of Submission: 08-12-2021 Date of acceptance: 23-12-2021

I. INTRODUCTION

The object detection technology based on deep learning has numerous applications in our daily lives. Object detection is used in a variety of applications, including medical image analysis, self-driving vehicles, business analytics, and facial recognition.

In this work, walk through all the steps for performing YOLOv4 object detections on webcam. Authors will use scaled-YOLOv4 (yolov4-CSP) for this work, the fastest and most accurate object detector there currently is. Cloud computing resources, general GPUs, IoT clusters, or single embedded devices may be required for the purposes listed above. The model scaling technique is critical for designing an effective object detector because it allows object detectors to achieve high accuracy and real-time inference on a variety of devices.

One of the most important study areas in computer vision is object detection. Traditional methods and deep learning methods are the two basic types of object detection algorithms. Furthermore, the latter can be classified into two groups. Some of them are based on region proposal object detection techniques like RCNN, SPP-net, Fast-RCNN, and Faster-RCNN, which are algorithms that construct region proposal networks and then classify them. SSD and YOLO [1, 2], for example, are based on the regression object detection method. These algorithms simultaneously construct a region proposal network and classify these region proposals. All algorithms have good performances in object detection. In [2], Cai et al. attempt to develop strategies that can be used to a variety of device network architectures with only a single training session. They decouple and train many sub-nets using techniques including decoupling training and search, as well as knowledge distillation, so that the full network and sub-nets are capable of processing specified tasks.

The rest of this paper is organized in an accompanying way. Section II: Related Work, Section III: Proposed Work, Section IV: Simulation Results, Section V: Conclusion, Section.

II. RELATED WORK

Authors referred some of the papers to understand the concepts and process of object detection based on the deep learning techniques. Some of them are listed below:-

In [1], discovered that CSPDarknet53, which is the backbone of YOLOv4 matches practically all optimal design features, derived using the network architecture search technique after analyzing state-of-the-art object detectors [2, 3, 6].The traditional model measurement method is to change the depth of the model by adding additional layers. For example, VGGNet [5], designed by Simonyan et al. accumulates additional convolutional layers in different categories, and uses this concept to design the structures of VGG-11, VGG-13, VGG-16, and VGG-19. The following methods usually follow the same model measurement method. In ResNet

[11], proposed by Mingxing Tan et al., Depth measurement can create very deep networks, such as ResNet-50, ResNet-101, and ResNet-152.

Later, Zagoruyko et al. [4], consider the scope of the network, and adjust the kernel number of the dynamic layer to see the scale. So they designed a comprehensive ResNet (WRN), while maintaining the same accuracy. Although WRN has a higher number of parameters than ResNet, the processing speed is much faster. Subsequent DenseNet [12], and ResNeXt [7], also designed a compact scale version that puts depth and breadth into consideration.

As for the description of the image pyramid, it is a common way to make augmentation during a run. It takes an embedded image and creates a different resolution scale, and then inserts these different pyramid combinations into a trained CNN. Eventually, the network will integrate multiple sets of results as its end result. Redmon et al. [8], Use the concept above to create an input image measurement. They use high-resolution image editing to perform well on a trained Darknet53, and the purpose of performing this step is to achieve high accuracy.

In recent years, research related to network architecture (NAS) has intensified, and NAS FPN has sought to integrate the pyramid schemes. We can think of NAS-FPN as a model measurement that is done mainly at the stage level. As for Efficient Net [10], it uses integrated search scales based on depth, width, and input size. The main design concept of Efficient Net [9] is to disassemble modules with different object acquisition functions, and then measure image size, width, #BiFPN layers, and #box / class layer.

III. PROPOSED WORK

In this work, the famous AlexeyAB's darknet repository is using to perform YOLOv4 detections. In order to utilize YOLOv4 with Python code we will use some of the pre-built functions found within darknet.py by importing the functions into our workstation. Feel free to check out the darknet.py file to see the function definitions in detail. Running YOLOv4 on images taken from webcam is fairly straight-forward. Authors will utilize code within Google Colab's Code Snippets that has a variety of useful code functions to perform various tasks.

In this work, authors will be using the code snippet for Camera Capture which runs JavaScript code to utilize your computer's webcam. The code snippet will take a webcam photo, which will then pass into YOLOv4 model for object detection. The function is coded to take the webcam picture using JavaScript and then run YOLOv4 on it. Running YOLOv4 on webcam video is a little more complex than images. Authors need to start a video stream using webcam as input. Then run each frame through YOLOv4 model and create an overlay image that contains bounding box of detection(s). Then overlay the bounding box image back onto the next frame of video stream. YOLOv4 is so fast that it can run the detections in real-time.

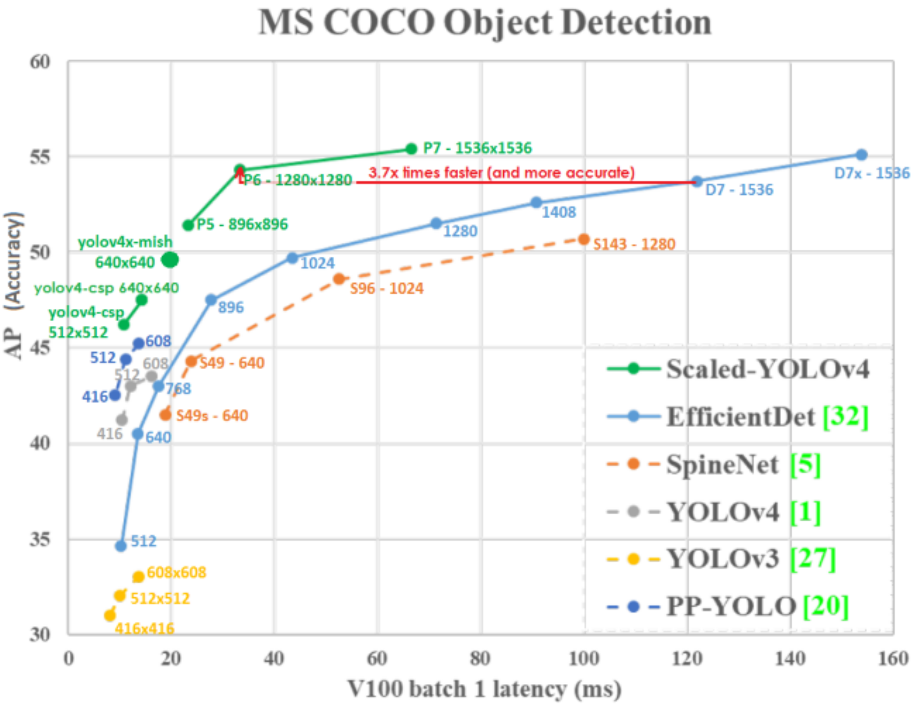

Fig 1: Comparison of the proposed scaled-YOLOv4 and other state-of-art object detectors. The dashed line means only latency of model inference, while the solid line includes model inference and post-processing.

YOLO

The YOLO neural network includes candidate box extraction, feature extraction, and neural networking techniques. The YOLO neural network directly extends candidate boxes to images and objects found in all aspects of the image.

In the YOLO network, images are separated by S × S grids. Candidate boxes are evenly distributed on X-axis and Y-axis. Candidate boxes have object detection and predict confidence of object presence in each candidate's box. Self-confidence indicates whether the images cover the object or not, as well as the accuracy of the object's location. YOLOv4 is built for real-time acquisition in a standard GPU.

IV. SIMULATION RESULTS

YOLOv4 Example on Test Image Fig2 show the result of object detection by given image as fixed input. Make detections properly on a test image.

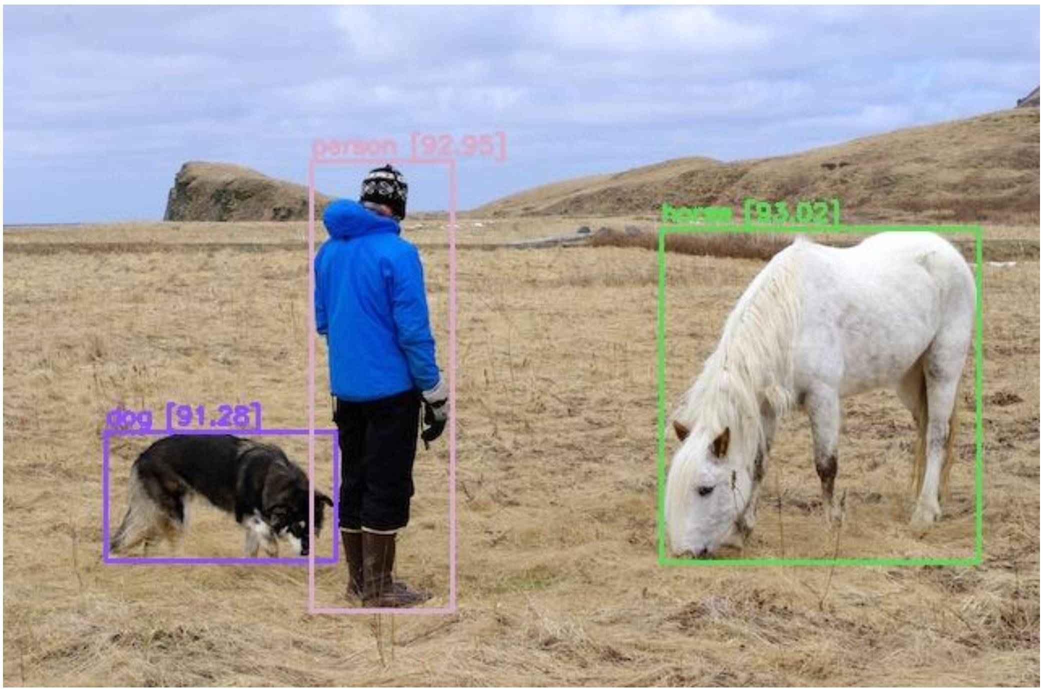

Fig.2 Output diagram of object detection using test image as input

YOLOv4 Example on Webcam Images Fig3 show the result of object detection by given image from Webcam as input. Make detections properly on a Webcam image.

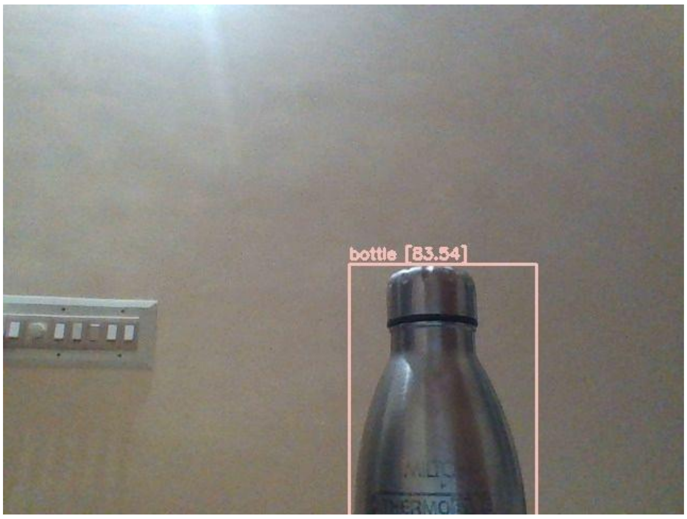

Fig.3 Output diagram of object detection using webcam image

V. CONCLUSION

The main objective of the proposed work was to object detection. The study was done with the YOLOv4 object identification neural network, which is based on the CSP technique and can scale up and down to fit small and big networks. So we achieve the highest accuracy 56.0 percent AP on test-dev COCO dataset for the model YOLOv4-large, extremely high speed 1774 FPS for the small model YOLOv4 .optimal speed and accuracy for other YOLOv4 models.

REFERENCES

[1] Alexey Bochkovskiy, Chien-Yao Wang, and HongYuan Mark Liao. YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934, 2020.

[2] Han Cai, Chuang Gan, Tianzhe Wang, Zhekai Zhang, and Song Han. Once-for-all: Train one network and specialize it for efficient deployment. arXiv preprint arXiv:1908.09791, 2019.

[3] Jiale Cao, Hisham Cholakkal, Rao Muhammad Anwer, Fahad Shahbaz Khan, Yanwei Pang, and Ling Shao. D2Det: Towards high quality object detection and instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 11485– 11494, 2020.

[4] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. arXiv preprint arXiv:1605.07146, 2016.

[5] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014..

[6] Xianzhi Du, Tsung-Yi Lin, Pengchong Jin, Golnaz Ghiasi, Mingxing Tan, Yin Cui, Quoc V Le, and Xiaodan Song. SpineNet: Learning scale-permuted backbone for recognition and localization. arXiv preprint arXiv:1912.05027, 2019.

[7] Saining Xie, Ross Girshick, Piotr Dollar, Zhuowen Tu, and ´Kaiming He. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE Conferenceon Computer Vision and Pattern Recognition (CVPR), pages 1492–1500, 2017..

[8] Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.

[9] Mingxing Tan, Ruoming Pang, and Quoc V Le. EfficientDet: Scalable and efficient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[10] Mingxing Tan and Quoc V Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Proceedings of International Conference on Machine Learning (ICML), 2019.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770–778, 2016.

[12] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 4700–4708, 2017.