International Journal of Research in Advent Technology, Vol.7, No.4, April 2019
E-ISSN: 2321-9637
Available online at www.ijrat.org

Deep Learning Based Object Detection Using You Only Look Once

Shraddha Kulkarni[1], Gururaj S.P[2]
M.Tech Scholar, Department of CSE[1]
Assistant Professor, Department of CSE[2]
Siddhaganga Institute of Technology – Tumkur[1,2]
Email: shraddhakulkarni@gmail.com, gururajsp@sit.ac.in

Abstract- This paper spotlight on the real time detection and recognition model called as YOLO. It uses single convolutional neural network in order to detect and recognize the objects of the images. The model is first trained on COCO dataset and car dataset of achieving a mAP of 91.28% and 70% respectively. YOLO takes 57 FPS to processes the image to detect the objects in Image. Since YOLO takes whole detection pipeline in a single unified network and it helps to increase and optimize the real time object detection with variety of objects.
Index Terms-YOLO, CNN, COCO dataset, Bounding Box.

1. INTRODUCTION

In the modern years, object detection is one of important areas of computer vision. The marked goal of object detection is to detect and classify the objects in the real time environment. The task of allocating a label and a bounding box to every objects in the image is called object detection. So this paper presents the solution for the object detection that is YOLO-(You Only Look Once)[9] A unified approach for predicting all bounding boxes in the images and videos simultaneously as shown in below Fig.1.

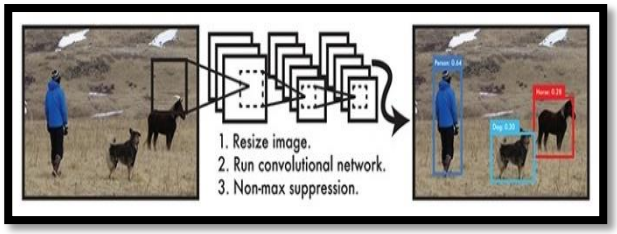

Fig. 1. The YOLO Detection System

Images can be processed using the following three steps. This can be done by

- The first step is to change the size of the input image from original size to 448*448 pixels.
- The second step is to run an individual convolutional network on the image, and
- The next step is too verged with resulting detection by the confidence of model.

The Basic Idea of this Yolo approach is to desperate units of components of the respective object detection to form a single network. The whole network uses the features from the entire image to predict each bounding boxes[7]. This also means that the single neural network detect all the objects in the images. Therefore Yolo design implements end to end training and provide high average precision.

The neural network system will divide the input image into S*S grid cells and if the centre of object falls into that grid cell, then that grid cell is responsible for detecting the object.
Each grid cell predicts B bounding boxes and scores of confidence for those bounding boxes. So these scores of confidence tells us that how confidence and accurate the model will predict the bounding boxes.
Therefore we define probability of finding the scores of confidence is given by $Pr(object)* IOU^{truth}_{pred.}$ If ^{there} are no scores exists in the prediction then that confidence scores should be zero. Each of the bounding boxes predicts 5 predictions: a, b, c, d and score of confidence[2].
For getting the conditional class probability and individual box confidence predictions we multiply them as shown below equ (1).

$$Pr(Class_i|Object)*Pr(object)*IOU^{truth}_{pred} = Pr(Class_i) * IOU^{truth}_{pred} \quad (1)$$

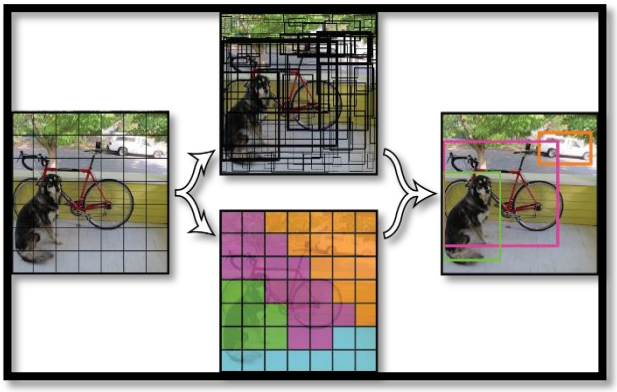

Fig. 2. The Object detection Model

The model of the system detection is also called as regression problem. It predicts bounding boxes by finding the probability using the equation. The next section is the Literature survey of how different systems and algorithms are used for object detection.

2. RELATED WORK

In the recent years Deep Networks is used for locating class-specific or class agnostic bounding boxes. Therefore to increase the accuracy and efficiency of the prediction of bounding boxes they have used different models.

The Object detection was proposed by Ross Girshick, It was proposed on Fast Region-based Convolutional Network method (Fast R-CNN)[5]. It is used to classify object proposals using deep learning networks. This model trains on deep learning model VGG-16 network 9* faster than R-CNN, is 213*faster at test -time, and achieves about 66% mAP on PASCAL VOC dataset. The drawback of this model is to sparse object proposals appear to improve detector quality.

Further Hao Yang et al.[3] Proposed Convolutional neural networks (CNNs) for the general feature representations for the object detection and recognition applications. In this paper the apart from basic feature representation extraction the ground truth bounding box annotations is added for another level of information by using nearest-neighbour relationships for local regions to obtain multi-view pipeline. The drawbacks of this paper is to increase the scalability and possibility of establishing a new proposal selection criteria to filter noisy objects in the Images.

The Next literature is by Rachel Huang et al.[1] Proposed A Real-Time object Detection Algorithm Optimized for Non-GPU computers uses the trained model and trained dataset of PASCAL VOC and COCO dataset and achieving a mAP of 33.81% and 12.26% respectively. The YOLO-LITE takes 21 Fps second to processes the Image frames. The drawback of YOLO-LITE architecture has decrease its accuracy down to 12.26%.

Further A New model is proposed by Joseph Redmon et al.[2] that is YOLO model for real time object detection which processes 45 frames per second and 155 frames per second for the faster version of YOLO. YOLO learns always more generalization representations and runs in single neural network predicting bounding boxes and class probabilities directly from full images in one evaluation.

The drawback of the YOLO method is it makes more localization errors and less with false prediction if nothing is existing in the state of art detection.

Wei Liu et al.[4] has Proposed SSD-Single Shot Multishot Detector is used for detecting objects in the images using the single neural network.

The SSD model does the following task while prediction time that is it generate scores for the presence of each object in every box and makes the adjustments with the other multiple boxes generated for better match in the objects in image. SSD model is tested and trained on the PASCAL VOC, COCO, and ILSVRC dataset to achieve about 74.3% mAP on VOC 2007 test at 59 Frames per second. The Drawback of this model is it unnecessarily creates the multiple bounding boxes for every objects and it is showing very less prediction of objects in the moving video and tracking the objects simultaneously. The next section explains the design Architecture of the system.

3. DESIGN

The Operation of this model is done using the convolutional neural network and classify it on the cars dataset and COCO dataset. The basic convolutional network of layers are used to extract features from the images then the next few layers of the network are used to predict the bounding boxes using the probabilities and coordinates as shown in Fig. 3.

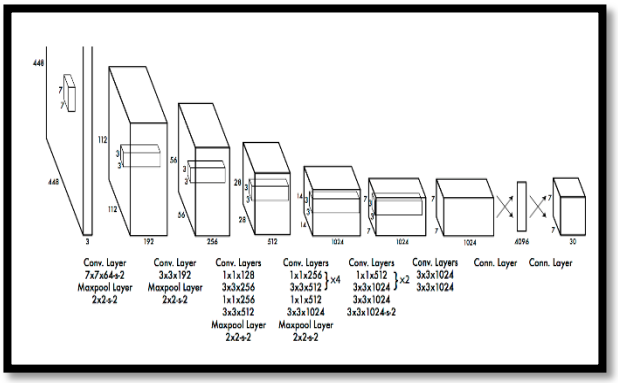

Fig. 3. The process of feature extraction and prediction of objects.

This Neural network has 24 convolutional layers pursued by 2 fully connected layers. When we put our configuration file on to this network it will lessen the features space from previous layers[6].

YOLO network work with only fewer convolutional layers that is it uses 9 layers instead of 24 layers. The first 7 layers are used for feature extraction and 2 fully connected layers are used for prediction of bounding boxes in the Images.

Therefore as per architecture the final output of the model is 7*7*30 tensor of predictions. The architecture is motivated by the GoogLeNet for classification of the images.

The GoogLeNet[2] always try to reduce the layers by 1*1 reduction and convolutional layer by 3*3 as we can see in the diagram. The next section tells the procedure to reach the bounding boxes in the images.

The below given figure is the flowchart of how the images are processed to predict the bounding boxes. Firstly the raw data of images are taken for training and convolutional neural network is applied on the dataset to extract the features of the objects and predict the bounding boxes as shown in Fig. 4.

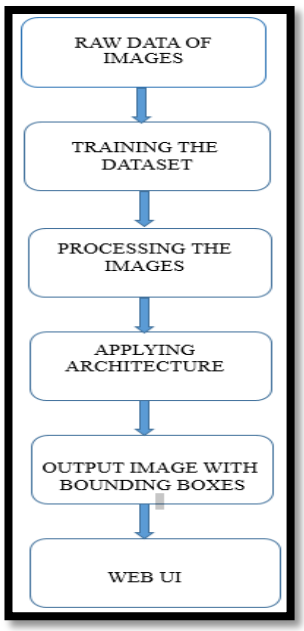

Fig. 4. Flow chart of the process

The first step of our model is training the dataset using 24 convolutional layers as shown in fig.3 as pooling layer and fully connected layers. The network is trained around 4-5 hours in GPU machine to achieve 91.28% of mAP and this is achieved using individual network.

After detection of the object by inserting bounding boxes the network increase the input resolution from 224*224 to 448*448[2].At the final layer we are using linear activation function to optimize the final bounding boxes and form initial layers to the final layers rectified linear activation layer is used to optimize the initial layers output.

So this network gives the output to predict the boxes and probability of classes in form of individual evaluation [11]. Then the predicted images are saved in the output file to processes the image further for the autonomous purpose. The next part of this paper tells the experiment results and their specific quantitative outputs.

4. EXPERIMENTS AND RESULT

The YOLO model is applied on the Car dataset that is firstly labelling the Image in the format of XML file and then generating test and train set. Then the next step is to convert the XML file to the CSV file. Then applying the model configuration file to predict the bounding boxes. This method of YOLO takes 57 Frames per second to processes the Image and predicts approximately 91.28% for detecting all the objects in the Image and for video it takes around 1.04 second for each frame of Image for tracking the moving objects[8]. The comparisons of the real system and the YOLO system is made in the next section. Results are shown in Table. 1.

Table 1. Experimental Results

Dataset	Classes	mAP	FPS
Car Dataset	40	91.28%	57
COCO Dataset	80	70%	40

4.1 Comparison To Other Systems

Many Algorithm are implemented for the object detection which focuses on increasing the accuracy and efficiency of the models. YOLO-LITE on Non-GPU[1] machine takes around 12.26% mAP for COCO dataset and for PASCAL-VOC dataset takes around 33.81% mAP. SSD Mobilenet V1 on COCO dataset produces around 21% mAP[4] and 5.8 frames per second it processes the image.

Therefore our system is implemented on the yolo model using the convolutional neural network producing 91.28% mAP for the car dataset and 70% mAP for the general dataset with 57 and 40 frames per second respectively. This is how our model improves in increasing the accuracy in terms of mean average precision and frames per second as shown in below figures.

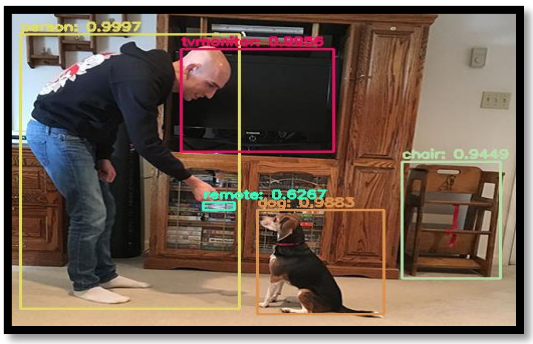

Fig. 5. Object detection

Fig. 6. Object detection

Fig. 7. Object detection

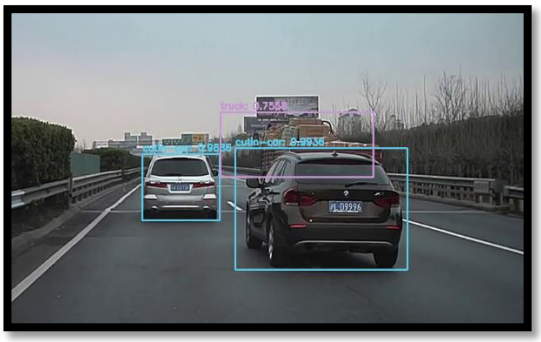

Fig. 8. Object Detection

Fig. 9. Quantitative Result

5. CONCLUSION AND FUTURE WORK

YOLO is achieved the marked goal of bringing object detection for automated industry to achieve the high end automated cars. YOLO is extraordinarily fast Real time object detection and recognition system and it is uncomplicated to design. It can be trained precisely on entire Images and YOLO takes 57 FPS to detect the objects. This is better enhancement of increasing the result in terms of FPS. As compare to other model our model takes less predict false detection and this makes the advantage to predict objects in better way. The purpose of object detection is for Autonomous Industry for detecting wide collection of object in order to safeguard the driving instances of the person. This is how computer vision has the advantage of object detection. In this paper we used the model to increase the performance based on their accuracy and FPS. The future work of this paper is to increase the efficiency and accuracy in terms of detection and recognition of the objects in better way.

REFERENCES

[1] Rachel Huang, Jonathan Pedoeem. YOLO-LITE: A Real-Time Object Detection Algorithm Optimized for Non-GPU Computers. arXiv: 1811.05588v1 [cs.CV] 14 Nov 2018.

[2] Joseph Redmon , Santosh Divvala, Ross Girshick, Ali Farhadi University of Washington, Allen Institute for AI , Facebook AI Research. You Only Look Once: Unified, Real-Time Object Detection. arXiv: 1506.02640v5 [cs.CV] 9 May 2016.

[3] Hao Yang, Joey Tianyi Zhou, Yu Zhang, Bin-Bin Gao, Jianxin Wu, and Jianfei Cai. Exploit Bounding Box Annotations for Multi-label Object Recognition. arXiv: 1504.05843v2 [cs.CV] 3 Jun 2016.

[4] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu1, Alexander C. Berg. SSD: Single Shot MultiBox Detector. arXiv: 1512.02325v5 [cs.CV] 29 Dec 2016.

[5] Ross Girshik. Fast R-CNN. In Microsoft Research computer vision foundation. arXiv: 1504.080843v2 [cs.CV] 27 September 2015.

[6] J.Schmidhuber, "Deep learning in neural networks: An overview," Neural networks, vol. 61, pp. 85–117, 2015.

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.

[8] L. Fridman, D. E. Brown, M. Glazer, W. Angell, S. Dodd, B. Jenik, J. Terwilliger, J. Kindelsberger, L. Ding, S. Seaman et al., "Mit autonomous vehicle technology study: Large-scale deep learning based analysis of driver behavior and interaction with automation," arXiv preprint arXiv:1711.06976, 2017.

[9] O. Akgul, H. I. Penekli, and Y. Genc, "Applying deep learning in augmented reality tracking," in Signal-Image Technology & Internet- Based Systems (SITIS), 2016 12th International Conference on. IEEE, 2016, pp. 47–54.

[10] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning." in AAAI, vol. 4, 2017, p. 12.

[11] Wei Fang, Soufience Djahel "A Novel YOLO-Based Real-Time People Counting Approach" in Conference Paper · September 2017 DOI: 10.1109/ISC2.2017.8090864.