YOLOv4 Object Detection on Webcam

MD. Samiul Hauque Chowdhury (18-36072-1), MD. Sabbir Hasan (17-33918-1), Mehedi Hasan Joy (18-38413-2),Shihab Hossain(18-38131-2),

Fahad, Abdullah Al (18-37904-2)

*Department of Computer Sciences, American International University-Bangladesh*

# Abstract

# There are many features designed to improve the accuracy of Convolutional Neural Networks (CNN). The combination of these features needs to be actually tested on a large data set, and the results are theoretically supported. Some functions are only applicable to certain models and only to certain tasks or only to small data sets; while some functions such as batch normalization and residual concatenation are applicable to most models, problems, and data sets. We assume that such general functions include Weighted ResidualConnections (WRC), Cross Stage Partialconnections (CSP), Cross-miniBatch Normalization (CmBN), Self adversarial training (SAT), and Mish activation. We use new features: WRC, CSP, CmBN, SAT, Mish activation, mosaic data expansion. , CmBN, DropBlock regularization, and CIoU loss, we combine some of them to produce cutting-edge results: 43.5% AP (65.7% AP50) MS COCO recorded in real time on Tesla V100 at ~ 65 FPS.

# Introduction

Most CNN-based object detectors are usually only suitable for recommendation systems. For example, city camera is associated with a slow and accurate model to search for free parking spaces, while automatic collision warning is associated with a fast and inaccurate model. The real-time accuracy of the target detector enables not only to be used in the recommendation system to generate prompts, but also to independently control the process and reduce human intervention. The real-time operation of the target detector in the traditional GPU (GPU).)) Massive use of at an affordable price . The most accurate modern neural network does not run in real time and requires a large number of GPUs to train large-size mini data packets. We solved these problems by creating a CNN that runs on a regular GPU in real time​​and only the regular GPU is needed for exercise.[2]

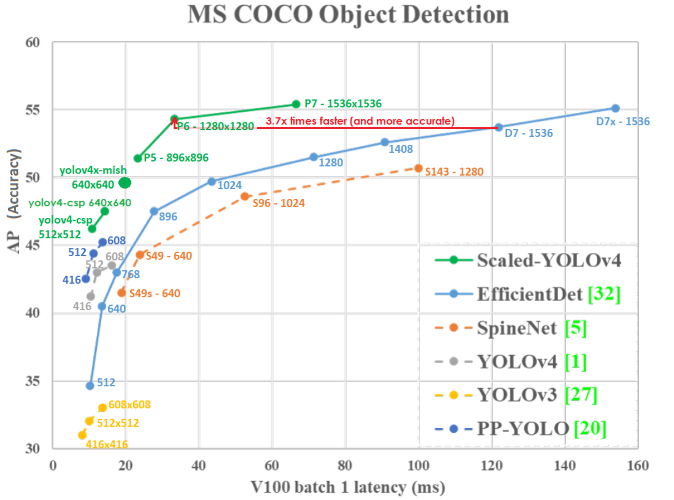


Figure 1: Comparison of the proposed YOLOv4 and other state-of-the-art object detectors. YOLOv4 runs twice faster than EfficientDet with comparable performance. Improves YOLOv3’s AP and FPS by 10% and 12%, respectively

The main goal of this paper is to develop a fast object detector with a speed of in the production system and optimize it for parallel computing, not for the theoretical low computational load (BFLOP) indicator. We hope that the designed facilities are easy to train and use. For example, anyone who uses a traditional GPU for training and testing can obtain compelling, high-quality real-time object detection results, as shown in Figure 1 for the YOLOv4 results. An efficient and powerful object recognition model. In this way, anyone can use 1080 Ti or 2080 Ti GPU to teach ultra-fast and accurate object recognition.

2. We tested the influence of the latest generation of object recognition methods Bagof Freebies and Bag-of-Specials during the training of the detector.

3. We are modifying the most modern methods to make them more efficient and suitable for training on a single GPU, including CBN, PAN, SAM, etc.

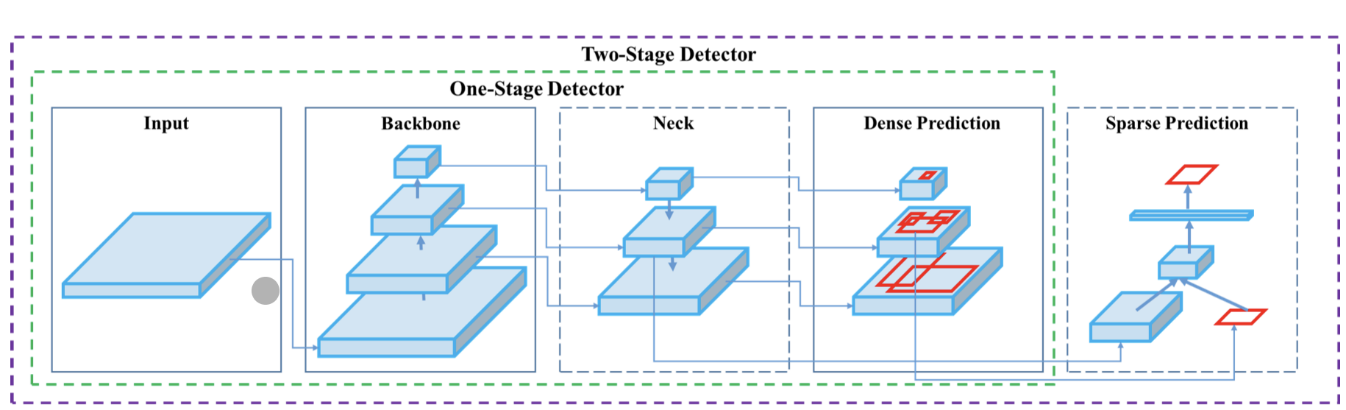


Figure 2: Object detector

**Related work**

# 

# Object detection models

Modern detectors usually consist of two parts: the basis for pre-training in ImageNet and the head used to predict object categories and bounding boxes. For GPU-based detectors, VGG, ResNet, ResNeXt, or DenseNet can be your basis. For a detector running on a CPU platform, your backbone network can be SqueezeNet, MobileNet or ShuffleNet. Regarding the headend, is generally divided into two types: single-stage object detector and two-stage object detector. The most typical two-stage target detector is the RCNN series, which includes Fast RCNN , Fast RCNN, RFCN , and Libra RCNN . It is also possible to turn a two-stage object detector into an object detector without anchor points, such as RepPoint. The most representative models of single-stage object detectors are YOLO , SSD, and RetinaNet . In recent years, anchorless desktop detectors have been developed for CenterNet , CornerNet , FCOS and other objects. Object detectors developed in recent years often insert multiple layers between the spine and the head, and these layers are usually used to collect feature maps at different stages. We can call it the neck of the object detector. Usually the neck consists of several ascending and descending routes. Networks equipped with this mechanism include Functional Pyramid Network (FPN), Path Aggregation Network (PAN), BiFPN and NASFPN.CNN model accomplishes better outcomes in right discoveries as per F1-score measure. Comment that the Faster R-CNN model doesn't utilize any unique ascribes for vehicle discovery though GMM foundation deduction utilized in AlexNet model. Indeed, as the union rates (MR) result shows, GMM foundation deduction actually has issues with fixed vehicles and blocked situations. In Faster R-CNN, the RPN part results could be improved giving some metropolitan setting

In addition to the above models, some researchers put their emphasis on directly building a new backbone (DetNet ,DetNAS) or a new whole model (SpineNet, Hit Detector) for object detection. [1]

**Bag of specials**

Generally, traditional object detectors learn offline. Therefore, researchers are always happy to take advantage of this and develop more efficient training methods that can improve the accuracy of the object detector without increasing inference costs. We will name only these ways to change learning. Strategy or just increase the cost of training is like a bag of rolls. What is often used in object recognition methods and conforms to the definition of a bag of snacks is the increase in the amount of data. The purpose of data expansion is to increase the variability of the input image, so that the developed object recognition model is more reliable than the images obtained from different environments. For examples usually use Photometric Distortion and Geometric Distortion. Data expansion method and definitely improve the object recognition problem. When using luminosity distortion, we will adjust the brightness, contrast, hue, saturation and image noise. For geometric distortion, we added random scaling, cropping, flipping, and rotation. All the above data magnification methods have a pixel setting, and the original pixel information in the adjusted area is all retained. In addition, there are approximately data magnification researchers focusing on the overlap of modeling objects. Lead to image classification and object recognition. For example, using the Random Erase [5] and Cut Out functions, you can randomly select a rectangular area in the image, and then enter a random value of or zero padding. In the case of Hide and Grid randomly or uniformly select several rectangular regions in the image and replace them with zeros. If a similar concept is applied to feature maps, there are DropOut ,DropConnect and DropBlock . In addition, some researchers have proposed multiple image sharing methods to perform data expansion. MixUp uses two images to multiply and overlap with different coefficient ratios, and then use these overlap ratios to adjust the labels. CutMix should cover the cropped image with the rectangular area of ​​other images and adjust the label according to the size of the overlapping area. In addition to the method mentioned above, GAN-style transmission [6] also uses to expand the data. Such use can effectively reduce the texture offset received from CNN. Compared with the various methods suggested above, has developed some other free methods to solve the problem, that is, the semantic distribution in the data set may be skewed. There is a problem of data imbalance between different categories. This problem is usually solved by negative hard sample analysis or online hard sample analysis on a two-stage target detector .However, the exemplary mining process does not apply to the single-stage object detector, because this type of detector is the tight prediction architecture. Therefore, Lin et al. It is recommended to use a focus to solve the

problem of data imbalance between different categories. [4] Another very important point is that is difficult to use a strict one-hot representation to express the degree relationship between different categories. This kind of visualization scheme is often used for annotation. The label smoothing proposed in is used to convert hard labels used for training into soft labels, which can make more reliable. In order to obtain a better "soft" label, Islam et al. introduced the concept of knowledge distillation to develop the tag clarification network. [3]

3.4. YOLOv4

In this section, we shall elaborate the details of YOLOv4.

YOLOv4 consists of:

• Backbone: CSPDarknet53 [81]

• Neck: SPP [25], PAN [49]

• Head: YOLOv3 [63]

**YOLO v4 uses**:

• Bag of Freebies (BoF) for backbone: CutMix and Mosaic data augmentation, DropBlock regularization, Class label smoothing

• Bag of Specials (BoS) for backbone: Mish activation, Cross-stage partial connections (CSP), Multiinput weighted residual connections (MiWRC)

• Bag of Freebies (BoF) for detector: CIoU-loss,

CmBN, DropBlock regularization, Mosaic data augmentation, Self-Adversarial Training, Eliminate grid sensitivity, Using multiple anchors for a single ground truth, Cosine annealing scheduler [52], Optimal hyperparameters, Random training shapes

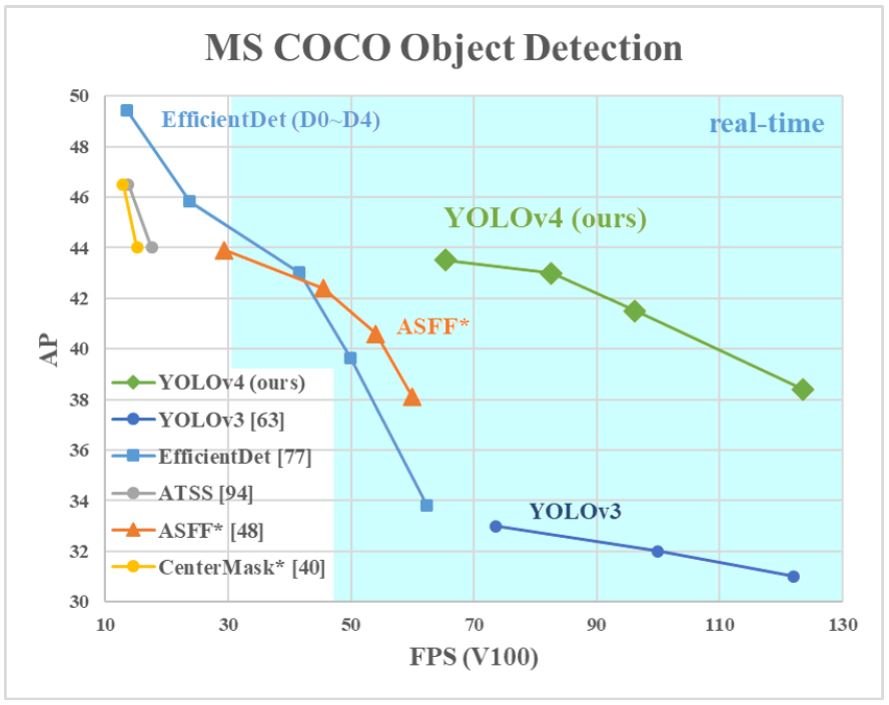
• Bag of Specials (BoS) for detector: Mish activation,SPP-block, SAM-block, PAN path-aggregation block, DIoU-NMS

**Results**

The comparison with the results obtained by other modern object detectors is shown in Fig. 8. Our YOLOv4 is located on the Pareto optimization curve and is better than the fastest and most accurate detector among Speed ​​and Speed ​​Accuracy.Since different methods use GPUs of different architectures to check the output time, we ran YOLOv4

on the widely used Maxwell, Pascal, and Volta 98​​7 GPUs and compared them with other next-generation methods and Maxwell GPU, it may be GTX Titan X (Maxwell) GPU or Tesla M40. Table 9 shows the results of using Pascal GPU to compare the frame rate, which could be Titan X (Pascal), Titan Xp, GTX 1080 Ti, or Tesla P100. Table

10 lists the results of the frame rate comparison. Use Volta 98​​7 GPU, which can be Titan Volta GPU or Tesla V100.



# . Discussion

# In the experiment of detecting MS COCO objects, the default hyperparameters are as follows: the number of learning steps-500 500; the learning rate programming strategy with decreasing steps is adopted, and the initial learning rate is 0.01, 400,000 or 450,000 steps multiplied by a coefficient of 0.1; weight The gain and weight loss are set to 0.9 and 0.0005, respectively. All architectures use a single GPU to perform multi-scale learning, with a burst size of 64, and a minibatch size of 8 or 4, depending on the architecture and GPU memory limitations. For hyperparameter search experiments, all other experiments use the default settings. The genetic algorithm uses YOLOv3SPP for training, loses GIoU and searches, and performs 300 iterations on 5000 Minval sets. We assume that the target learning rate is 0.00261 and the pulse is 0.949, showing that the IoU threshold of Ground Truth is 0.213, and the loss normalizer of the genetic algorithm experiment is 0.07. We tested a large number of BoF, including Mesh Sensitivity Removal, Mosaic Data Augmentation, IoU Threshold, Genetic Algorithm, Class Label Smoothing, Mini-Burst Cross-Normalization, Self-Shading Training, Cosine Annealing Programmer, Mini-Burst Dynamic Size, DropBlock, optimized anchors, different types of IoU loss. We also tested several BoS, including Mish, SPP, SAM, RFB, BiFPN and Gaussian YOLO [8]. For all experiments, we only used GPUs for training, so we didn't use methods like syncBN to optimize multiple GPUs.

# 

# Conclusion

# We provide a modern detector that's faster (FPS) and extra accurate (MS COCO AP50...ninety five and AP50) than all to be had opportunity detectors. The detector described may be educated and used on a traditional GPU with 8-16 GB-VRAM this makes its wide use possible. The original idea of one-degree anchor-primarily based totally detectors has tested its viability. We have established a massive wide variety of capabilities, and decided on to be used such of them for enhancing the accuracy of each the classifier and the detector. These capabilities may be used as best-exercise for destiny research and developments

# References

[1] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-NMS–improving object detection with one line of code. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 5561–5569,2017. 4 [2] Architecture comparison of AlexNet, VGGNet, ResNet, Inception, DenseNet. (2021). Retrieved 1 July 2021, from <https://towardsdatascience.com/architecture-comparison-of-alexnet-vggnet-resnet-inception-densenet> beb8b116866d?fbclid=IwAR04NbVP8UsBS9nhsZ8cTB4TjFhx295BfGEoCW6K8OUUVBvsByso3geP1IQ

[2] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 6154–6162, 2018

[3] Jiale Cao, Yanwei Pang, Jungong Han, and Xuelong Li. Hierarchical shot detector. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 9705–9714, 2019

[4] Ping Chao, Chao-Yang Kao, Yu-Shan Ruan, Chien-Hsiang Huang, and Youn-Long Lin. HarDNet: A low memory traffic network. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2019.

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos,Kevin Murphy, and Alan L Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI),40(4):834–848, 2017

[6] Pengguang Chen. GridMask data augmentation. arXiv preprint arXiv:2001.04086, 2020..

|  |  |  |
| --- | --- | --- |
| Name | ID | PART |
| MD. Sabbir Hasan | 17-33918-1 | Abstract & Conclusions |
| Mehedi Hasan Joy | 18-38413-2 | Introduction |
| MD. Samiul Hauque Chowdhury | 18-36072-1 | Literature Review |
| Shihab Hossain | 18-38131-2 | Results |
| Fahad, Abdullah Al | 18-37904-2 | Discussion |