K. J. SOMAIYA COLLEGE OF ENGINEERING

2USI-603

APPLIED MACHINE LEARNING USING TENSORFLOW

Submitted To:
Prof. Nirmala Shinde
Professor
Department of Computer Engineering
Prof. Kavita Kelkar
Professor
Department of Computer Engineering
Prof. Grishma Sharma
Professor
Department of Computer Engineering

Submitted By :
Ebrahim Bhinderwala
1812071
Meet Doshi
1812078
Arnav Gadre
1812079
Rhyme Risi
18120103
Amanshu Tiwari
1922023
Semester VI

Contents

LITERATURE SURVEY: OBJECT DETECTION

Ebrahim Bhinderwala[1], Meet Doshi[2], Arnav Gadre[3], Rhyme Risi[4], Amanshu Tiwari[5]
Department of Electronics Engineering
K.J. Somaiya College of Engineering, Vidyavihar, 400077, Maharashtra, India

I. ABSTRACT

Object detection is a core part of an intelligent surveillance system and a fundamental algorithm in the field of identity identification, which is of great practical importance. Deep neural networks (DNNs) have shown prominent performance in the field of object detection. However, DNNs usually run on powerful devices with high computational ability and sufficient memory, which have greatly limited their deployment for constrained environments such as embedded devices. YOLO is one of the state-of-the-art DNN-based object detection approaches with good performance both on speed and accuracy and Tiny-YOLO-V3 is its latest variant with a small model that can run on embedded devices. Since the YOLO series algorithms have good results in terms of accuracy and speed, YOLO and each subsequent version have been surpassing. In this literature review different object detection models are compared with YOLO-V3 . YOLO-V3 also posses comparable results in mAP and faster runtime speed with smaller model size and BFLOP/s value compared with other lightweight models like SqueezeNet SSD and MobileNet SSD.

II. INTRODUCTION

Object detection is an important task in many popular fields such as medical diagnosis, robot navigation, automatic driving, augmented reality and so on. In these complex scenarios, object detection methods based on deep learning approach, such as Region-based Convolutional Neural Networks (R-CNN) [1], Spatial Pyramid Pooling Networks (SPPNet) [2], fast R-CNN [3], faster R-CNN [4], Region-based Fully Convolutional Networks (R-FCN) [5], Feature Pyramid Networks (FPN) [6], and You Only Look Once (YOLO) [7] show greater advantages than traditional methods. YOLO is one of the fastest object detection methods with good real-time performance and high accuracy, and it has been improved since it was proposed, including YOLO-V1, YOLO-V2, YOLO-V3. YOLO-V1 has two full-connected layers and twenty-four convolutional layers. The model size of YOLO-V1 has reached 1 GB, which occupies very large storage space and requires the running platform with high performance. YOLO-V2 [8] removes the fully-connected layers and introduces anchor boxes to predict bounding boxes, making the YOLO detector faster and more robust than YOLO-V1. YOLO-V3 [9] uses the residual structure to further deepen the network layer and achieves a breakthrough in accuracy. By running on the powerful GPU platform, YOLO and its improvements have reached high accuracy and fast speed. However, the model sizes of these object detection algorithms are too large for constrained environments with limited storage memory devices and they cannot work in constrained environments with real-time performance. Tiny-YOLO-V3 [9] is the latest improvement of YOLO with a relatively small model size for constrained environments. However, its detection accuracy is not high and the real-time performance is still not satisfactory on low computing power devices.

III. LITERATURE REVIEW I

A. *TITLE*

An Optimized Approach to Clinical Object Identification using YOLO v3 in the Cloud Environment.

B. *AUTHORS*

N. Suresh Kumar
Research Scholar and Assitant Professor
School of Computing Science and Engineering
Galgotias University
Greater Noida, India

Amit Kumar Goel
Professor, School of Computing Science and Engineering
Galgotias University
Greater Noida, India

C. *INTRODUCTION*

In this paper *Kumar's* [10] research work includes the identification and description of clinical items. The model they have mentioned in the paper is trainedin combination with photographs and images using YOLO v3 object detection technique to identify medical objects. The model is programmed in the cloud system to do well in a short time of preparation. The work summarizes the model that has been developed with the potential to recognize and detail unknown objects, whether static or moving in a real life context. Their program not only shows the knowledge in the form of text but also spells out text in an artificial voice to help you easily understand the object. The model has an accuracy of 98.62% to detect the objects in clinical practices.

D. METHODOLOGY

The point of object detection is to recognize all classes of items from a known or identified class. This idea is the basis of *Kumar's* [10] model. An area and scale or the degree of an item characterized using a bounding box. The postural data are increasingly dotted and include conditions for clear or straight adjustment in a variety of different circumstances. The localization is marked using the bounding box which works based on the equation(1):

$$BoundingBox = \frac{AreaofOverlap}{AreaofUnion} \tag{1}$$

When two stage detectors are used in the method of object detection improves the accuracy with a greater height. Backbone networks are used to perform the classifications task which involves the mapping of the classes as a network. Stethoscope, Lab coat, sharps containers, ECG Unit return the spatial location and percentage of accuracy about each instance of an object to an image is analysed if present. The clinical or medical objects which are taken as the region of interest (ROI) will be highlighted with the help of the bounding box. The detectors will rescale the original image and try to match with the region of interest. Since the model is trained with one specific size of the object. We need to rescale the objects in the image as shown in Fig 1.

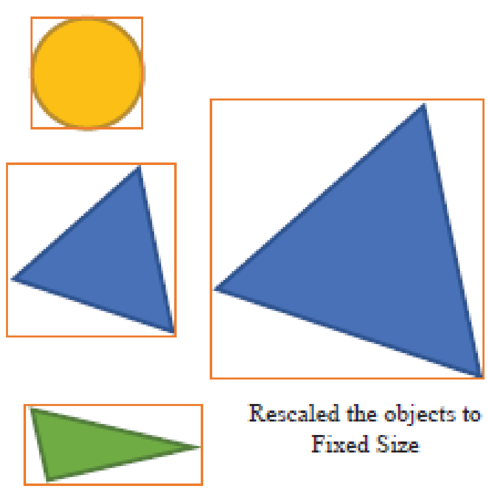

Fig 1: Scale Adaptive Training.

Initially the dataset is prepared with a set of 25 classes containing all the medical related objects. The dataset preparation included data augmentation techniques, such as flip, brightness, and shear. This step increased the size of the dataset and improved performance of the accuracy. The dataset is loaded in Python program with training folder containing 70% of the dataset which undergoes the training and features are extracted. The model was trained using GPU in the cloud environment. The dataset consisted of 25 classes of labels. Figure (1) shows the architecture of the proposed framework. The objects are defined by a qualified model and the image mark is positioned at the top of the objects.

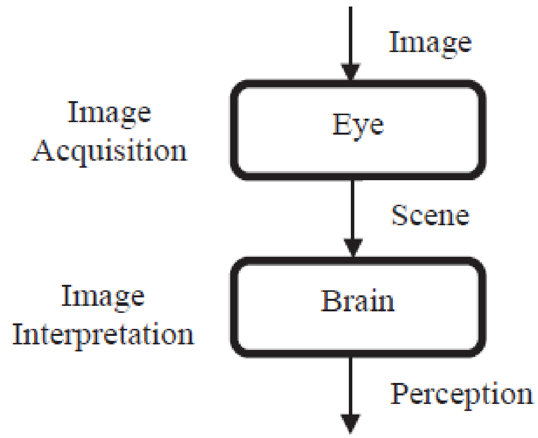

Fig 2: Proposed Framework by Kumar [1].

The model is trained with 100 epochs in the GPU for 3 hours and achieved the best accuracy results. First the model was trained with 10 classes using CPU but takes more than 18 hours to obtain accuracy of 98.46%.

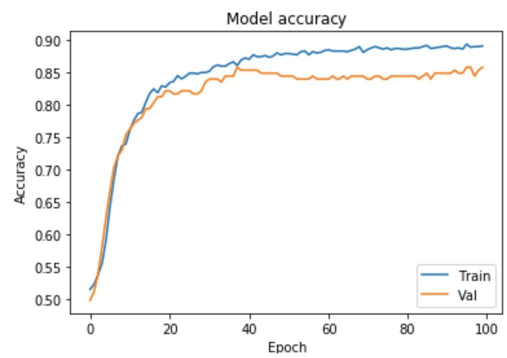

Fig 3: Model: Training and Validation Accuracy.

E. CONCLUSION

This research work discusses experimental studies on different approaches to the recognition and classification of objects and compares the output of each process. The purpose of this work is to establish a method for the recognition of objects in 2D images. The technique of object scale adaptive rescaling given in this paper is used widely in many algorithms and provides a great boost to the accuracy.

F. FUTURE SCOPE

Object recognition technologies can be used for surveillance devices, facial identification, error detection, character recognition, etc. This requires a good and fast model. Although the a network like Tiny YOLO v3 is compact we need better and smaller networks, progress in Tiny YOLO v3 in coming years looks highly promising.

IV. LITERATURE REVIEW II

A. *TITLE*

Tinier-YOLO: A Real-Time Object Detection method for Constrained Environments

B. *AUTHORS*

Wei Fang
School of IoT Engineering, Jiangnan University, Wuxi, China
Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence,
Wuxi, China

Lin Wang
School of IoT Engineering, Jiangnan University
Wuxi, China

Peiming Ren
School of IoT Engineering, Jiangnan University
Wuxi, China

C. *INTRODUCTION*

YOLO is one of the state-of-the-art DNN-based object detection approaches with good performance both on speed and accuracy and Tiny-YOLO-V3 is its latest variant with a small model that can run on embedded devices.

In this paper, Tinier-YOLO, which is originated from Tiny-YOLO-V3, is proposed to further shrink the model size while achieving improved detection accuracy and real-time performance. For improving the proposed Tinier-YOLO in terms of detection accuracy and real-time performance, the connectivity style between fire modules in Tinier-YOLO differs from SqueezeNet in that dense connection is introduced and fine designed to strengthen the feature propagation and ensure the maximum information flow in the network. The object detection performance is enhanced in Tinier-YOLO by using the passthrough layer that merges feature maps from the front layers to get fine-grained features, which can counter the negative effect of reducing the model size..

In order to get a more efficient object detection model for constrained environments, originated from Tiny- YOLO-V3, Tinier-YOLO is proposed in this paper to reduce the model size while achieving improved detection accuracy and real-time performance. Tinier-YOLO draws inspiration from the fire module in SqueezeNet to reduce the number of model parameters that helps to shrink the model size. One of the key challenges to introduce fire module in Tiny- YOLO-V3 is to investigate the number of fire modules as well as their positions in the model. Another key challenge is to determine the connectivity style between fire modules in order to further get the improved detection accuracy and real-time performance. Since there is a reduction in model size, it will inevitably affect the detection accuracy. Therefore the passthrough layer is utilized in Tinier-YOLO to address this issue that can merge feature maps from the front layers to get fine-grained features.

D. *METHODOLOGY*

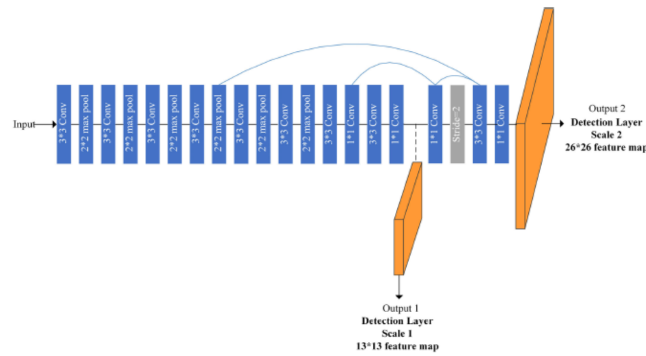

Fig 4: Original Tiny YOLO v3 Network Structure.

Fig. 4 shows the network structure of Tiny-YOLO-V3, which is composed of seven convolutional layers and six maxpool layers for extracting image features and two scales of detection layers. Tiny-YOLO-V3 uses many convolutional layers with 512 and 1024 convolution filters, which results in a large number of parameters, large storage usage and slow detection speed on constrained environments. Another problem of Tiny-YOLO-V3 is that the detection accuracy is not high and the unreasonable compression methods in the network may further reduce the detection accuracy.

The problems in Tinier YOLO v3 are proposed in this paper and it focuses on performance of the model size, the detection speed and accuracy. The purpose of Tinier-YOLO is to get a smaller, faster and better model that can run on constrained environments. First of all, the network structure is optimized by reducing the number of parameters reasonably instead of just deleting the convolutional layers blindly. The fire module of SqueezeNet uses the bottleneck layer network to compress the model and the network module is widen without losing detection accuracy heavily. Therefore, the fire module is introduced in Tinier-YOLO in order to get a smaller and faster network structure. And then, Tinier-YOLO seeks the way to get higher detection accuracy while reducing the number of parameters.

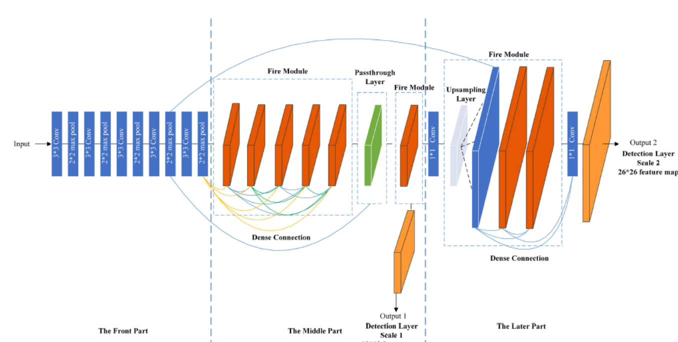

Fig 5: The Network Structure of Tinier YOLO proposed by Fang[11].

The front part of Tinier-YOLO retains the front five convolutional layers of Tiny-YOLO-V3. In the middle part of Tinier-YOLO, the five fire modules are introduced to implement network parameters compression and the dense connection between the fire modules is employed. It also merges previous feature maps with the passthrough layer before the first detection layer and predicts bounding boxes with the first scale. In the latter part of Tinier-YOLO, it adds two fire modules to process the combined feature maps and predicts bounding boxes with the second scale to get the fine-grained features. The fire module is introduced to reduce the number of model parameters and increase the depth and width of the whole network to ensure the detection accuracy. Dense connection is deployed between the fire modules to improve the accuracy by strengthening feature extraction ability and ensuring the maximum information flow in the network as shown in Fig 6.

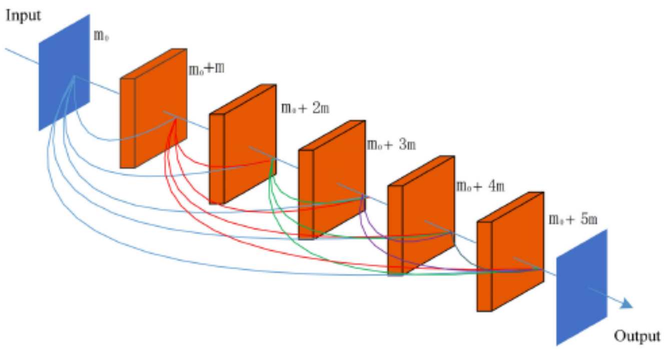

Fig 6: Dense Connection between the fire modules.

Fig. 4 shows the dense connection between five fire modules in Tinier-YOLO. The feature maps of the first l - 1 fire modules are concatenated and utilized as the input of the lth fire module. m0 is the feature map of the previous convolutional layer and m is the feature map of the fire modules. Therefore the lth fire module outputs m0 + m(l-1) feature maps. The feature maps are relatively smaller that with fewer parameters and calculations. If the dense connection is deployed in larger feature maps or connects the feature maps with different size, it will result in a huge amount of calculation, which greatly affects the real-time performance. Therefore, Tinier-YOLO keeps the front convolutional layers and only replaces the five convolutional layers in the middle part and the latter part of Tiny- YOLO-V3 with eight fire modules.

E. CONCLUSION

In this paper, they have proposed Tinier-YOLO which is a real-time object detection method for constrained environments. Tinier-YOLO is designed by introducing the fire module of SqueezeNet into Tiny-YOLO-V3 at first in order to reduce the model size. And then the number of fire models and their positions in the network architecture have been studied. The connectivity method between fire modules in Tinier-YOLO is realized by using dense connections of DenseNet. The dense connections in Tinier-YOLO helped to improve the detection accuracy and real-time performance since the feature propagation is strengthened and the maximum information flow in the network is ensured. By incorporating the passthrough layer in Tinier-YOLO, the detection accuracy has been further improved by merging feature maps from the front layers to get fine-grained features. The computational cost of the network has been reduced by removing the batch normalization from the fire modules of Tinier-YOLO. The experiments on PASCAL VOC and COCO demonstrate Tinier-YOLO is more efficient compared with Tiny-YOLO-V3.

F. FUTURE SCOPE

Comparing with lightweight models like SqueezeNet SSD and MobileNet SSD, Tinier-YOLO also shows its competitive performance in terms of model size. In future, Fang [11] looks forward to further optimizing the performance of Tinier-YOLO, especially for the COCO dataset.

REFERENCES

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580-587.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2014, pp. 346-361.

[3] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV),* Dec. 2015, pp. 1440-1448.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91-99.

[5] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379-387.

[6] T.-Y. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," in *Proc. CVPR*, 2017, vol. 1, no. 2, pp. 2117-2125.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: United, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779-788.

[8] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," 2017, *arXiv:1612.08242*. [Online]. Available: https://arxiv.org/abs/1612.08242

[9] J. Redmon and A. Farhadi, "YOLOV3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: https://arxiv.org/abs/1804.02767

[10] N. S. Kumar and A. Kumar Goel, "An Optimized Approach to Clinical Object Identification using YOLO v3 in the Cloud Environment," *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 2021, pp. 856-859, doi: 10.1109/ICACITE51222.2021.9404592.

[11] W. Fang, L. Wang and P. Ren, "Tinier-YOLO: A Real-Time Object Detection Method for Constrained Environments," in *IEEE Access*, vol. 8, pp. 1935-1944, 2020, doi: 10.1109/ACCESS.2019.2961959.