iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VI **Month of publication:** June 2022

DOI: https://doi.org/10.22214/ijraset.2022.44112

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VI June 2022- Available at www.ijraset.com

Object Detection Technique Using a Fast R-CNN Based Technique

Amlan Dutta[1], Abhijit Pal[2], Mriganka Bhadra[3], Md. Akram Khan[4], Rupak Chakraborty[5], Sourish Mitra[6], Rafiqul Islam[7], Nirupam Saha[8]
[1, 2, 3, 4, 5, 6, 7, 8]*Department of Computer Science and Engineering, Guru Nanak Institute of Technology, Kolkata, West Bengal, India*

Abstract: ***Object detection stands as the current interest in locating the objects. Reducing the run time of the algorithms holds a challenge in this field. Deep Convolutional Neural Network (CNN) like SPPnet, Fast R-CNN etc. had been effectively proposed to reduce the execution time. Encouraged by these networks, a Region Program Network (RPN) has been proposed here to portray features of images with less computational cost and time. This network is capable of predicting the bounds of an object and generates scores at each position. In the proposed network, Fast R-CNN is also integrated with RPN so that exceptional region programs can be generated, and entire network can easily understand which area should be focused. The proposed network has been tested on MS COCO, PASCAL VOC 2007, 2012 datasets. Results infer that the proposed RPN along with Fast R-CNN outperforms some popular models like deep VGG-16 [3], ImageNet, YOLO etc. in terms of all the accuracy parameters.***
Keywords: ***Fast R-CNN, Detection of Objects, Identification of Regions***

I. INTRODUCTION

In computer vision, detection of objects is a process to find out objects from the classes of an image. Objects can be localized by interpreting in various ways. This can be achieved by bounding boxes around objects or pointing the image pixels which are identified as objects. Objects can be detected in multiple ways : One step, Two step, and Heatmap-based object detection. Object detection can be implemented by the networking algorithms: Mask R-CNN,Faster R-CNN, and R-CNN,where images are processed, and objects are detected.

Both the networks are used to train and test images after rescaling them such that the size of their shorter side becomes 600px. The total stride for both the ZF and VGC nets on the last convolutional layer becomes 16px, providing good results [3] [17]. Introduced novel Region Proposal Networks (RPNs) which share their convolutional layers with other popular object detection networks so that computation cost for computing proposals becomes small [4] [11] [21]. Some secured multimedia obtained techniques can also be found in the literature [22-24].

The SSD with MobileNetV1 has a high detection speed but the accuracy is low compared to Faster R-CNN with InceptionV2 that has lower speed but good accuracy [5] [12]. Have shown an algorithmic change by communicating the proposals with a deep CNN-lead to an appropriate and proper solution. [3] [14] [18]

Effectiveness of region proposals in images is verified by comparing different object detection techniques. Fast R-CNN shows the highest detection rate for the panel and speech balloon whereas Faster R-CNN shows the highest detection rate for character, face and text objects [6] [15].

Our approach is based on the current revolutionary FASTER R-CNN model, and designed two domain adaptation components with the aim of reducing the domain discrepancy [7] [16] [20]. Proposed a multi-component CNN model, each component of which is steered to focus on a different region of the object thus enforcing diversion of the appearance factors [8] [10] [19].

Here, Section 2 describes Problem Formulation of the paper in the Object detection field. Proposed approach is discussed in Section 3. Base Network is described in Section 4. Section 5 explains results and discussion of proposed approach. Finally, Section 6 draws a conclusion of the paper and its future scope.

II. PROBLEM FORMULATION

Faster R-CNN is the widely used version of R-CNN. Faster R-CNN is a method for detecting objects, which extract features from the pretrained CNN network. The network is subdivided into two networks, which can be trained.

A. Architecture of Region Proposal Network (RPN)

1) At first, the feature map can be obtained by passing the image as an input to the backbone CNN. The size of the input image can be rescaled not exceeding 1000 pixels in the longer side and 600 pixels in the shorter side.

2) The backbone network features are way smaller than the actual image taken as input, having a pace of backbone network.

3) Taking into consideration the output feature map, the network identifies the object from the input image and approximates its position and dimensions in the image. Then an Anchor set is placed in the loaded picture for every position of resultant feature map. This set signifies probable object of different sizes and phase ratios on that location.

4) Output feature map is checked by each and every pixel one by one and finds the corresponding anchors, if they have objects or not. And strains out the frame of bounding boxes to give the region of interest.

5) The process is followed by ROI pooling, upstream classifier and bounding box regressor, likewise Fast R-CNN.

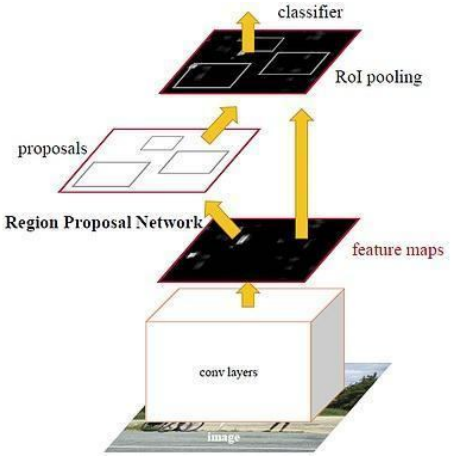


Figure 1: Traditional Region Proposal Network(RPN)

III. THE PROPOSED APPROACH

The overall framework of the proposed approach is demonstrated in figure 2. If the default Faster R-CNN framework is required to metamorphose, red boxes apex the proposed intensification in Figure 2. In the initial step, Mobile Net architecture has been used to assemble the base intricacy layer in spite of default Faster R-CNN framework in which VGG-16 architecture lies. Soft-NMS algorithm has been utilized to work out matters of heavy car repression in the RPN. Context-aware RoI pooling layer retards RoI pooling layer to sustain actual edifice of small cars. To classify proposals, MobileNet architecture has been built at the closing stage within the car background; also rearranges boundary box for every detected car. Below is the explanation of our approach.

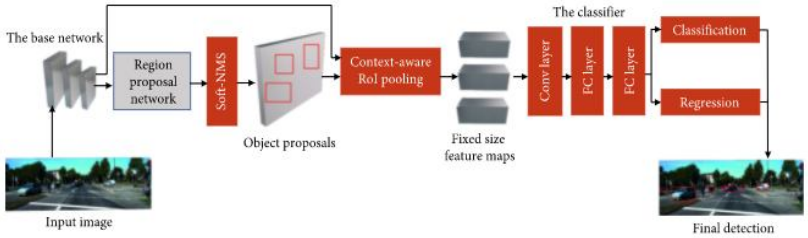


Figure 2**:** The proposed approach.framework

IV. BASE NETWORK

VGG-16 as a base network is used by default-faster R-CNN. It has been proved that 80-85% of the forward time used in base network makes the whole framework much faster. To split the convolution into a 3×3 depthwise convolution, MobileNet architecture is used in 1×1 pointwise convolution, minimizing the no. of criterion and cost of calculation. MobileNet initiates two criterions to fit to tune the manoeuvre/precision trade-off, including resolution coefficient along with width coefficient. Instead of VGG-16, MobileNet is embraced to create the base convolutional layers in Faster R-CNN in the default framework object detection in this paper.

Table 1: Model of MobileNet differentiation with VGG

Used Models	ImageNet precision (%)	Multiply-adds(in millions)	Attributes(in millions)
MobileNet	69.2	571	5.3
VGG-16	72.7	15335	134

The Depth-wise separable convolution has two layers, namely Depth-wise convolutions and Pointwise convolutions. The number of output feature map channels, the kernel size square and the computational cost reduction are commensurate with each other.

A. The Region Proposal Network (RPN)

At first, RPN creates a group of anchor boxes out of a base network generated feature map which is convolutional. Three anchor box having scales of 128, 256 and 512 and ratios of 1 : 1, 1 : 2, and 2 : 1 are utilized for whichever anchor of the research paper as in for swapping between revoke and processing speed by relenting nine anchors at each position of the slider. As of figure 3, there are 1764 anchors for a 14×14 size convolutional feature map.

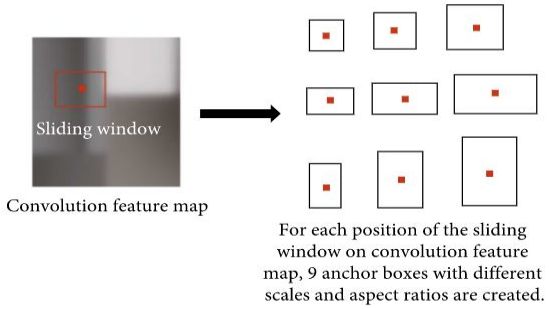


Figure 3: RPN generated anchor boxes

Then for each of the anchors, RPN holds every anchor box and outcomes as two discrete outcomes. The first one is the objectness result, which defines the probability of how many objects are anchored. As shown in Figure 3, The 2nd output, bounding box regression is used to modify the anchors to suit the object better. A valid set of ideas for vehicles is generated by applying the ultimate proposal coordinates and their objectness score. Proposals finish as overlying on the same object as anchors normally. The problem of duplicate proposals can be solved by using Soft non-maximum suppression (NMS) algorithm. In advanced object detection, Faster R-CNN is also utilized to separate similar proposals. Conventional non-maximum suppression separates any other proposal that has an overlay more than a predetermined approach with a conquering proposal. With the help of soft-NMS; adjacent proposals together with conquering proposals neither fully extinguished.

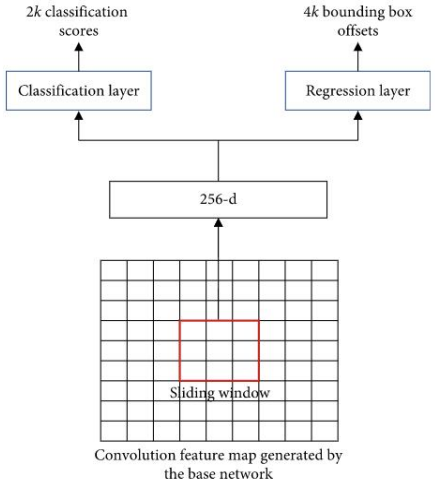


Figure 4: The Network of Region Proposal

(I) (II)

Figure 5: Observation result with (I) Soft-NMS and (II) NMS. Due to dense car obstruction, NMS left 1 vehicle within observation outcomes but soft-NMS determined 2 vehicles individually

A.a. Soft Non-Maximum Suppression Algorithm

Consider Pm = {p1, p2, p3,...,pn} designates a stated proposal set yield, from which proposals are adjured by objective result. The adjacent proposal approach T is fixed for cross-validation to 0.5. Consider Si to indicate the objective result of pi, that being the highest value in the vector of classification score pi. Consider pi denotes the conquering proposal and pj be an adjacent proposal of pi. The modified objective result of pj (indicated as Suj) is calculated by the formula below:

$$S_j^u = S_i\left(1 - O_{p_i,p_i}\right), \qquad (1)$$

Here Opi,pi is the Intersection of union (IoU) in the middle of conquering proposal pi and adjacent proposal pj and is calculated as follows :

$$O_{p_i,p_i} = \frac{\text{area}\left(p_i \cap p_j\right)}{\text{area}\left(p_i \cup p_j\right)}. \qquad (2)$$

Structural outline about soft-NMS algo is shown in figure below.

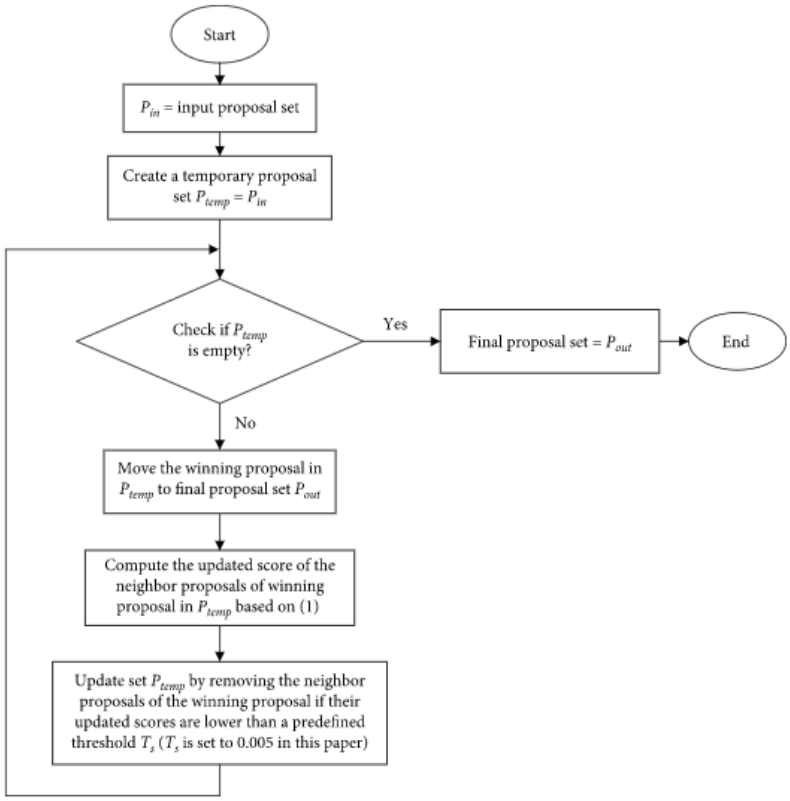


Figure 6: The soft-NMS algorithm Flowchart.

B. *Context-Aware RoI Pooling*

The RoI pooling layer algorithm is utilized for balancing the dimensional proposals to constant sizes. To turn out the features inside any valid region into a small feature map with a fixed dimensional magnitude of H X W, RoI pooling layer uses its highest pooling. RoI max pooling works in such a way that it divides the RoI proposal into an H X W grid of subcells of roughly size (h/H)X(w/W) , for maximum pooling the values in each subcell to the corresponding output grid cells. Any proposed value was less than the H X W; it's perspective can be built up as H X W on appending duplicate values for filling the generated space. Appending duplicate values into tiny proposals are preferable, mainly with tiny cars, because of pulling down actual formation of tiny cars. Then, the efficiency of detecting small cars will be brought down. For proposals less constant size of outcome feature map, we used deconvolution operation and to do that we have used the following formula:

$$y_k = F_k \oplus h_k, \qquad (3)$$

Here,
yk = Output Feature Map
Fk = Input Proposal
hk = Kernel of the Deconvolution Operation

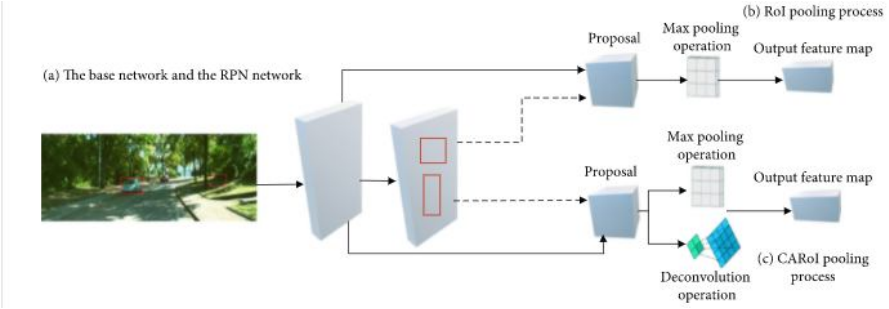


Figure 7: CARoI pooling strategy. (a) Proposals and Feature maps produced by RPN and base network; (b) Conventional RoI pooling operation. (c) Context-aware RoI pooling operation.

C. *Classifier*

The appellation in the dispensed substructure is called Classifier. After taking out attributes for every individual proposal through context-aware RoI pooling, these attributes come into the picture for classification.. The classifier is divided into two Fully Connected layers, the 1st is a box classification layer and the 2nd one is box regression layer.1st Fully Connected layer has been led to the softmax layer to calculate the probabilities of the cars and the background. Another Fully Connected layer along a linear activation function degenerates detected cars' bounding box. Each and every convolutional layer is gone along with a ReLU layer and batch normalization layer.

V. RESULTS AND DISCUSSION

COCO dataset is utilized here to compare the outcomes of the proposed approach with different approaches. Experiment has been implemented on Google Colab Platform with TPU accelerator. Deep CNN frameworks have been implemented using TensorFlow whereas OpenCV has been chosen for the processing of real-time data.

A. *Dataset*

The current interest of the popular dataset named as Common Objects in Context dataset (COCO) has been chosen here for experiment. This object detection dataset is widely used nowadays for benchmark comparison. In addition to that, a new custom object detection dataset can also be created as the COCO dataset follows the go-to format to store annotations of data. Annotations for Segmentation data are also available in this COCO dataset.

B. *Metrics used for Evaluation*

Performance of the presented approach has been evaluated by two parameters. Intersection over Union (IoU) and Average Precision (AP) benchmarks. Three difficulty levels of the COCO dataset are tested here. Different kinds of object detection algorithms have been assessed by these criteria. In this experiment, the IoU has been chosen as 0.7.

C. Model Training

Throughout the paper, the pretrained MobileNet model applied to ImageNet dataset, which has been chosen as a base network and it is later fine-tuned through COCO dataset. Further training process is accelerated, and overfitting has been reduced by freezing the weights of every batch normalization layer at time of training the model. First of all, Region Proposal Networks (RPN) along with the classifier are trained on a small-batch basis and later the attributes of the RPN with the modified base network. After that RPN generated two kinds of proposals (positive and negative) for training and updating the classifier. Then the attributes of the classifier and base convolutional layers have been updated one-by-one. The balancing parameter (λ) has been fixed for loss function. learning rate of the RPN initially is fixed to 0.0001 where the decay rate of learning has been fixed to 0.0005/small-batch. We have trained the final network up to 200 epochs to get the effective outcomes.

D. Performance Results

Table 5: Detection results by the different methods applied on COCO dataset

	Average Precision			
Method	Easy (%)	Average (%)	High(%)	Extracting time(s)
Faster R-CNN [3]	87.72	82.85	72.13	1.9
YOLO [2]	47.69	35.74	29.65	0.03
Proposed approach	88.21	86.85	73.73	0.14

Figure 8: Detection result based on the proposed approach implanted on the image.

E. Loss Value Comparison

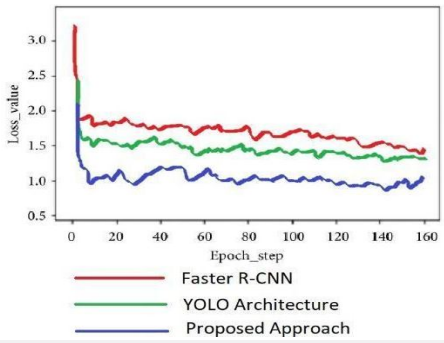


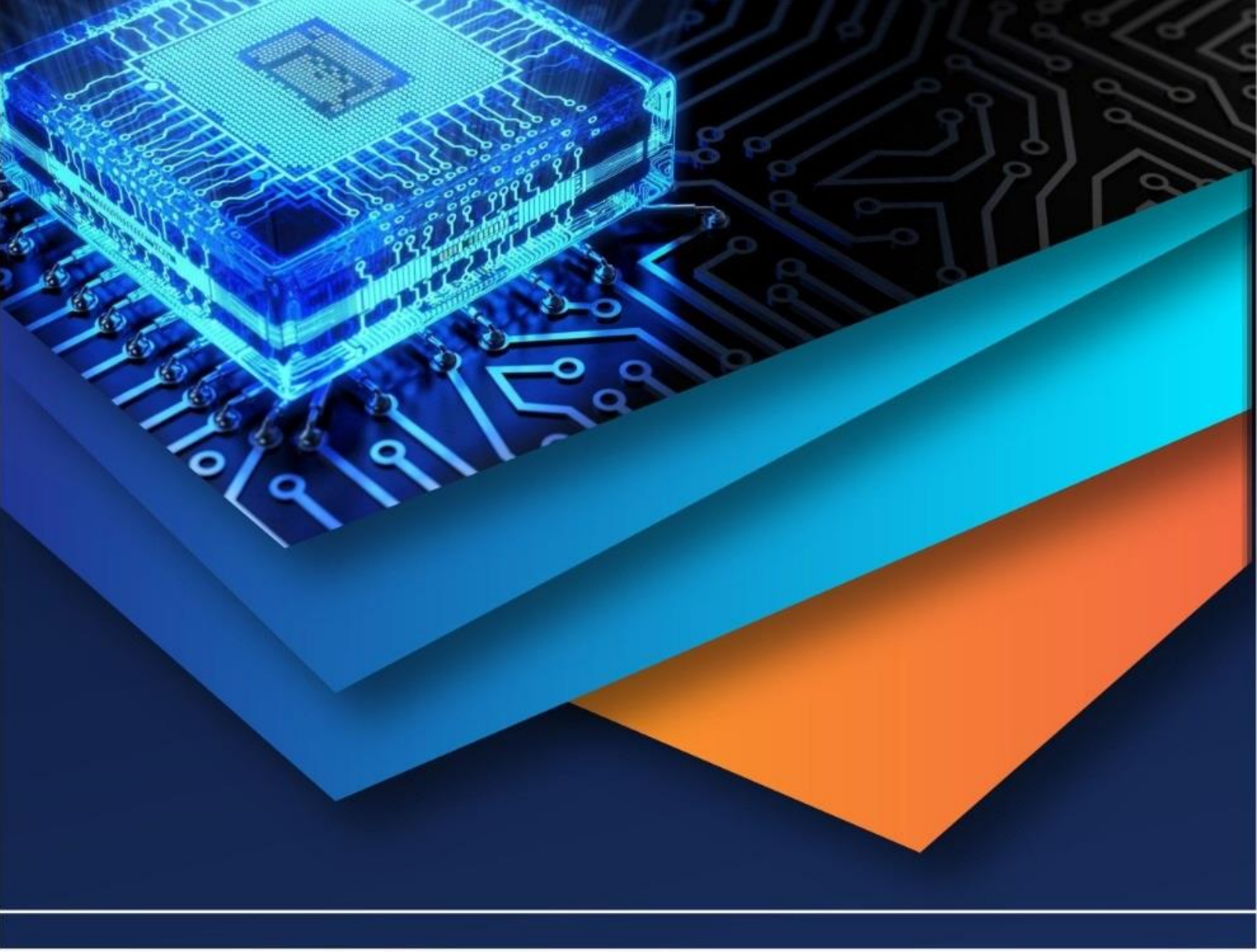
Figure 9: Loss Value Comparison between Faster R-CNN , YOLO Architecture and Proposed Approach

VI. CONCLUSIONS AND FUTURE WORK

So, hereby we can conclude that RPN can be presented more efficiently and accurately. Using proposed approach in Fast R-CNN, we have got more optimised loss value, more Average Precision value than Faster R-CNN and less processing time. In the future we wish to work on reducing the overall loss of the model as much as possible.

REFERENCES

[1] Buric, M., Pobar, M., Ivasis-Kos, Marina.: Ball detection using Yolo and Mask R-CNN. In:International Conference on Computational Science and Computational Intelligence (CSCI),DOI 10.1109/CSCI46756.2018.00068, pp. 1-5 (2018).
[2] Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks: IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 39(6), 5-13 (2017).
[3] Du, Juan.: Understanding of Object Detection Based on CNN Family and YOLO.: IOP Conf. Series: Journal of Physics, 1004(012029), 1-9 (2018).
[4] Ren, S., He, K., Girshick, R., Sun, J.: R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, 1-14 (2016)
[5] Galvez, R., L., Bandala, A., A., Vicerra, R., R., P., Dadios, E., P., Maningo, J., M., Z.: Object Detection Using Convolutional Neural Networks. In: Proceedings of TENCON IEEE Region 10 Conference, pp. 28-31 (2018).
[6] Yanagisawa, H., Yamashita, T., Watanabe, H.: A Study on Object Detection Method from Manga Images using CNN. : International Workshop on Advanced Image Technology (IWAIT), 1-4 (2018).
[7] Chen, Y., Li, W., Sakaridis, C., Dai, D., Gool, L., V.: Domain Adaptive Faster R-CNN for Object Detection in the Wild. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3339-3348 (2018).
[8] Gidaris, S., Komodakis, N.: Object detection via a multi-region & semantic segmentation-aware CNN model. In: IEEE International Conference on Computer Vision (ICCV), pp. 1134-1142 (2015).
[9] Guindel, C., Martin, D., Armingol, J., M.: Joint Object Detection and Viewpoint Estimation using CNN features. In: IEEE International Conference on Vehicular Electronics and Safety (ICVES), pp. 145-150 (2017).
[10] Wu, M., Yue, H., Wang, J., Huang, Y., Liu, M., Jiang, Y., Ke, C., Zeng, C.: Object detection based on RGC mask R-CNN.: IET Image Processing. 14. 10.1049/iet-ipr.0057, (2019).
[11] Liu, Y., LI, H., Yan, J., Wei, F., Wang, X., Tang, X.: Recurrent Scale Approximation for Object Detection in CNN. In: IEEE International Conference on Computer Vision (ICCV), pp 2-7 (2017).
[12] Cai, Z., Vasconcelos, N.: Cascade R-CNN: Delving into High Quality Object Detection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1-9 (2018).
[13] Pang, J., Chen, K., Shi, J., Feng, H., Ouyang, W., Lin, D.: Libra R-CNN: Towards Balanced Learning for Object Detection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1-10 (2019).
[14] Sun, P., Zhang, R., Jiang, Y., Kong, Tao., Xu, C., Zhan, W., Tomizuka, M., Li, L., Yuan, Z., Wang, C., Luo, P.: Sparse R-CNN: End-to-End Object Detection with Learnable Proposals. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1-10(2021).
[15] Zhang, H., Chang, H., Ma, B., Wang, N., Chen, X.: Dynamic R-CNN: Towards High Quality Object Detection via Dynamic Training. In: European Conference on Computer Vision Springer, Cham,DOI 10.1007/978-3-030-58555-6, pp. 260-275 (2020).
[16] Hung, J., Carpenter, A.: Applying Faster R-CNN for Object Detection on Malaria Images. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 56-61 (2017).
[17] Roska, T., Leon, O.: The CNN Universal Machine: An Analogic Array Computer. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing 40,163-173 (1993).
[18] Leon, O., Chua, Roska, T.: The CNN Paradigm. Fundamental Theory and Applications 40, 147-156 (1993).
[19] Wang, P., Liu, Y., Guo, Y., Sun, C., Tong, X.: O-CNN: Octree-based Convolutional Neural Networks for 3D Shape Analysis.: ACM Transactions on Graphics (TOG) 36, 1-11 (2017).
[20] Gupta, H., Jin, K, H., Nguyen, H, Q., McCann, M, T.: CNN-Based Projected Gradient Descent for Consistent CT Image Reconstruction.: IEEE transactions on medical imaging 37, 1-14 (2018).
[21] A. G. Howard, M. Zhu, B. Chen et al.: MobileNets: efficient convolutional neural networks for mobile vision applications, arXiv:1704.04861v1



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY