Journal of Fundamental and Applied Sciences

ISSN 1112-9867

Available online at http://www.jfas.info

Research Article

Special Issue

PERFORMANCE EVALUATION OF FASTER R-CNN ON GPU FOR OBJECT DETECTION

B. Adam[1], F. H. K. Zaman[1,*], I. M. Yassin[1], H. Z. Abidin[1] and Z. I. Rizman[2]

[1]Faculty of Electrical Engineering, UniversitiTeknologi MARA, 40450 Shah Alam, Selangor, Malaysia

[2]Faculty of Electrical Engineering, UniversitiTeknologi MARA, 23000 Dungun, Terengganu, Malaysia

Published online: 10 September 2017

ABSTRACT

This paper presents a performance evaluation of Faster Region-based Convolutional Neural Network method with different parameters to observe the mean average precision. Faster R-CNN replaces the previous proposal method with Region Proposal Network to complete the network. RPN predicts object bounds and its scores at each region making it a fully convolutional network. RPN produces almost cost-free region proposals since the network shares fully image convolutional features with detection network. The use of this technique improve training and testing speed and mean average precision (mAP) compared to SPPnet. It can achieves approximately 10ms per image for object detection and time cost in region proposal. The dataset used to train and test is on VOC 2007. This technique is implemented in MATLAB R2017a using Caffe on NVidia GTX 1060 and GTX 1080.

Keywords: R-CNN; RPN; mAP; object detection.

Author Correspondence, e-mail: fadhlan@salam.uitm.edu.my

doi: http://dx.doi.org/10.4314/jfas.v9i3s.64

1. INTRODUCTION

Convolutional Neural Network is known as a feed-forward artificial neural network that is originally inspired by biological process that is the form of the animal visual cortex more specifically, cat. It is a powerful visual models that has the ability to train end-to-end and pixels-to-pixels even between low or high resolution images [1-2]. Furthermore, object detection is a process to detect instances of semantic objects of a category such as cars, humans or animals. It also identifies how image windows should cover the whole object into a class [3]. This method can combine high-level context with several low-level images features that gives the best performance when combine with R-CNN [4].

Region-based Convolutional Neural Network (R-CNN) method was initially developed by using deep ConvNet to categorize object proposals. After that, spatial pyramid pooling networks (SPPnet) was introduced by sharing computation with R-CNN for speed improvement. SPPnet classifies object proposals by computing convolutional feature map of input image using feature vector that is extracted from the featured map[5]. However, there are few disadvantages of this system such as the training is a multi-stage pipeline, training may cost expensive in memory space and object detection takes 47 seconds per image on a GPU [2].

Therefore, Fast R-CNN was proposed to overcome the drawbacks that is originally the work of Ross Girshick. The whole idea of Fast R-CNN is that the network takes a set of object proposal and its entire image as input, processing the image with few max pooling and convolutional layers to produce a feature map [6]. This network also noticeably takes advantage of using GPU to accelerate proposal computation.

Additionally, Faster R-CNN was developed not long after the previous network with the introduction of Region Proposal Network (RPN) enabling the network to share convolutional layers. Hence, reducing the cost for computing proposals to 10ms per images [7-8].

It has come to our knowledge that handcrafted technique or the previous technique consumes a lot amount of time and various parameters can affect the precision of the results. This paper presents the latest implementation of convolutional neural network that is the Faster R-CNN and performs performance evaluation of input parameters for better precision.

This paper is organized as the following sections: In section I, introduction of this paper is explained. In section II explains the methodology of this paper. Next, section III shows and discuss the results obtained. Lastly, section IV will conclude this paper.

2. METHODOLOGY

2.1. Fast Region-Based Convolutional Neural Network (R-CNN)

R-CNN is a visual object detection system that combines bottom-up region proposals with features previously computed by convolutional neural network. This system will initially computes the region proposal with selective search technique and pass the proposal to the convolutional neural network for classification. Fig. 1 shows the illustration of R-CNN, the original model of which Fast R-CNN is based on.

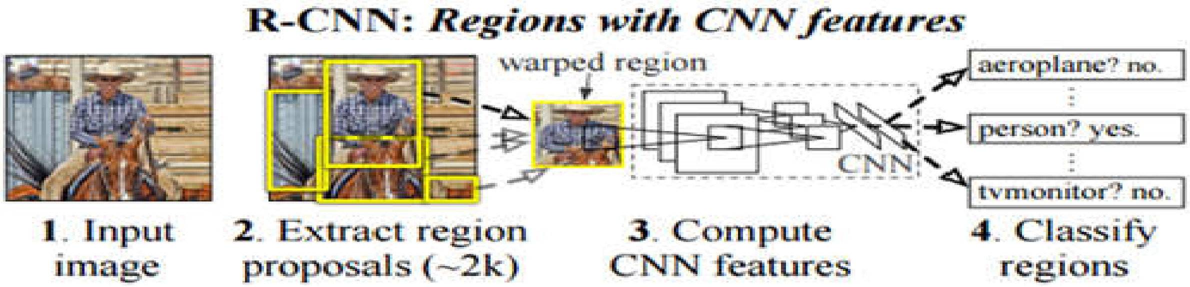


Fig.1. System flow of R-CNN [6]

After that, Fast R-CNN is applied with the ability to shares computation which makes it the most important factor. Fast R-CNN is improved with higher detection quality, a single stage training using multi-task loss, the training can update all layers of network and feature caching does not require storage as shown in Fig. 2 [6].

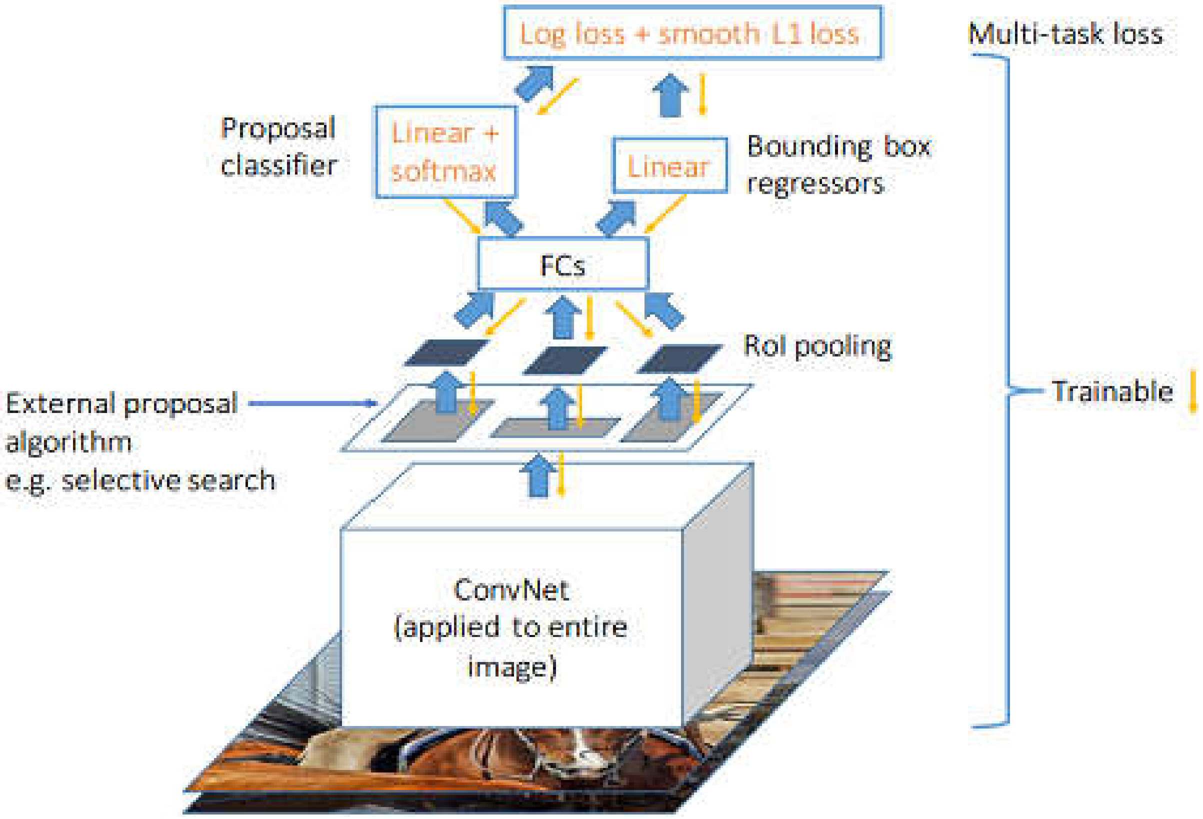

Log loss + smooth L1 loss
Multi-task loss
Proposal classifier
Linear + softmax
Linear
Bounding box regressors
FCs
RoI pooling
External proposal algorithm e.g. selective search
Trainable
ConvNet (applied to entire image)

Fig.2. System flow of fast R-CNN [6]

2.2. Detection Object Proposal Methods

This technique is similar to interest point detection but the time when interest points were proposed, the computation of feature descriptors was very expensive [9]. This is because feature descriptors compute interest points that are useful for detection, classification and retrieval. Despite that, object proposals do not necessarily improve the quality of interest point detection. Moreover, these methods uses low-level features of image to discriminate whether a window is included for detection or absence similar to using a classifier to remove unrequired proposals. These few methods reviewed to generate detection proposals:

- SelectiveSearch capture all objects scales even though some objects have less boundaries using hierarchical algorithm [10]. Therefore, colour, shading, texture or enclosed parts can influence regions to form an object and this algorithm is reasonably fast to compute as depicted in Fig. 3.

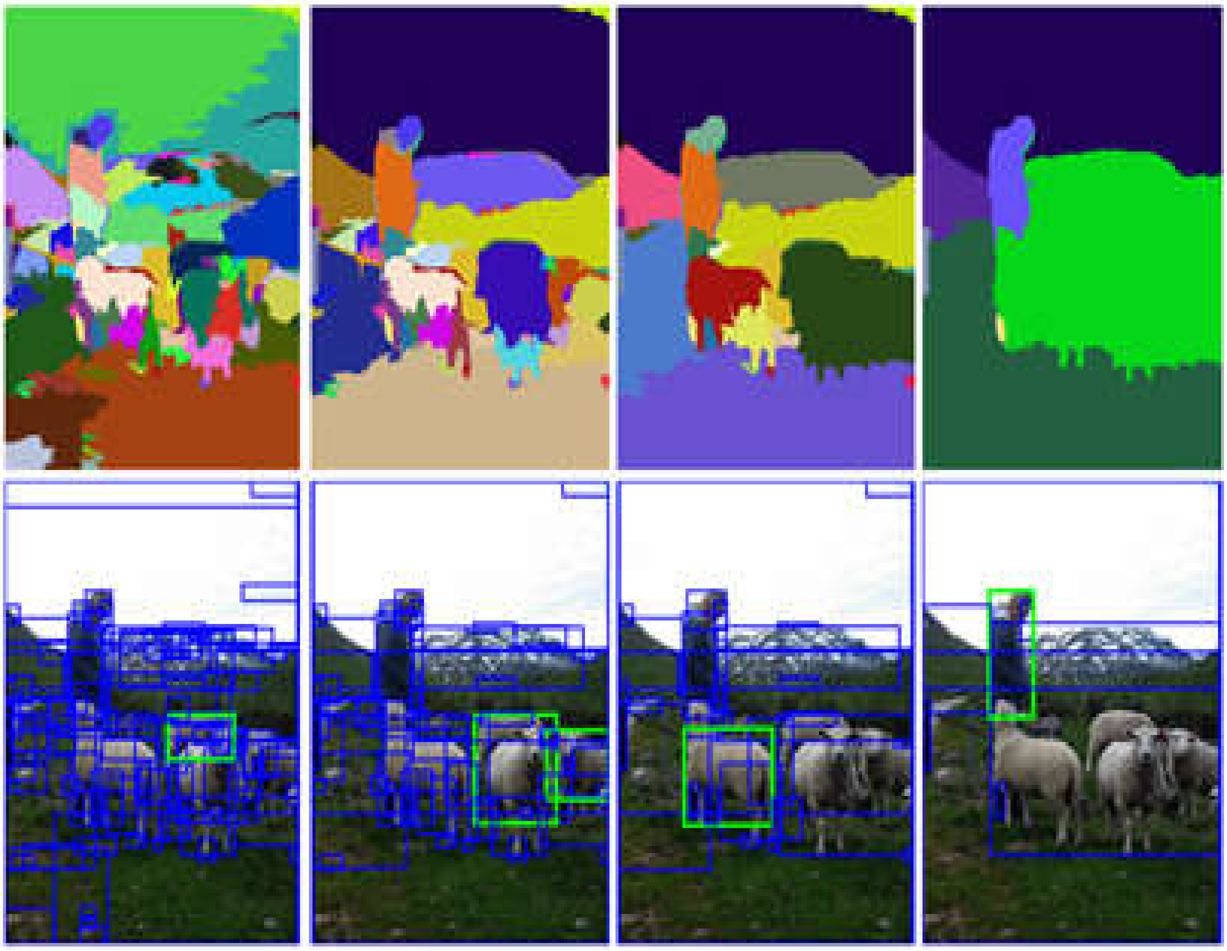


Fig.3. Example showing many different scales of objects [10]

- EdgeBoxes is another method in finding object proposals allowing separation of fully enclosed contours thus measuring the score of the edge-based function [11]. A simple coarse-to-fine search refines the method in order to find top-ranked object proposals across aspect ratio, scale and position evaluation as in Fig. 4. A pair of groups s_i and s_j is computed based on mean positions x_i and x_j with mean orientations θ_i and θ_j where θ_{ij} is the angle between x_i and x_j and $_\gamma$ adjust the affinity's sensitivity to the changes within orientation using:

$$a(s_i, s_j) = \left|\cos(\theta_i - \theta_{ij})\cos(\theta_i - \theta_{ij})\right|^{\gamma} \tag{1}$$

$w_b(s_i)$ is a continuous value that places whether s_i is contained in b with value 1 or 0 and T is an ordered path of edge groups with length t. This is to find the path with the highest affinity between edge group and the group that overlaps the boundary of the box as describes:

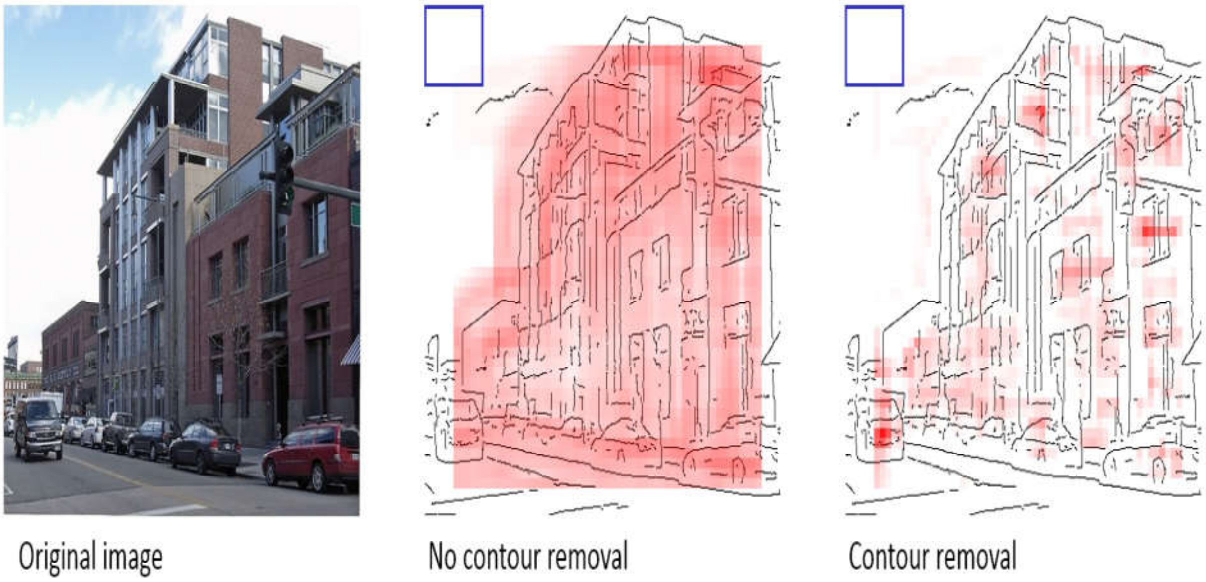
$$w_b(s_i) = 1 - \max_{T} \prod_{j}^{|T|-1} a(t_j, t_{j+1}) \tag{2}$$

The width and height of the box are defined as b_w and b_h, k is used to offset the bias of large windows with the value of 1.5 and sum of all m_i is computed using integral image to speed computation using:

$$h_b = \frac{\sum_i w_b(s_i) m_i}{2(b_w + b_h)^k} \tag{3}$$

Lastly, subtract the edge magnitudes from b^{in} as edges in the center of the box are less likely to be important than the box's edges [7]. Thus, using integral image to compute the sum of edge magnitudes in b^{in} as shown below:

$$h_b^{in} = h_b - \frac{\sum_{p \in b^{in}} m_p}{2(b_w + b_h)^k} \tag{4}$$



Original image

No contour removal

Contour removal

Fig.4. Illustration of removing contours that overlap the boundary of the box [11]

2.3. Region Proposal Network (RPN)

Region Proposal Network takes image feature map and set of rectangular object proposals as outputs with each object score. This network is included in fully convolutional network. Besides that, this network will evaluate the region obtained from different anchors and sliding windows and hence, the bounding box regression. Image feature map is taken by applying 3*3*256 convolutional kernel in the fifth convolution layer. Thus, producing a 256 dimension of intermediate layer. The layer will feed into two branches that are object score and regression. Object score determines if region is a thing or stuff whereas regression determines how bounding box should change to become similar to ground truth box. Fig. 5 shows an overview of Faster R-CNN with RPN.

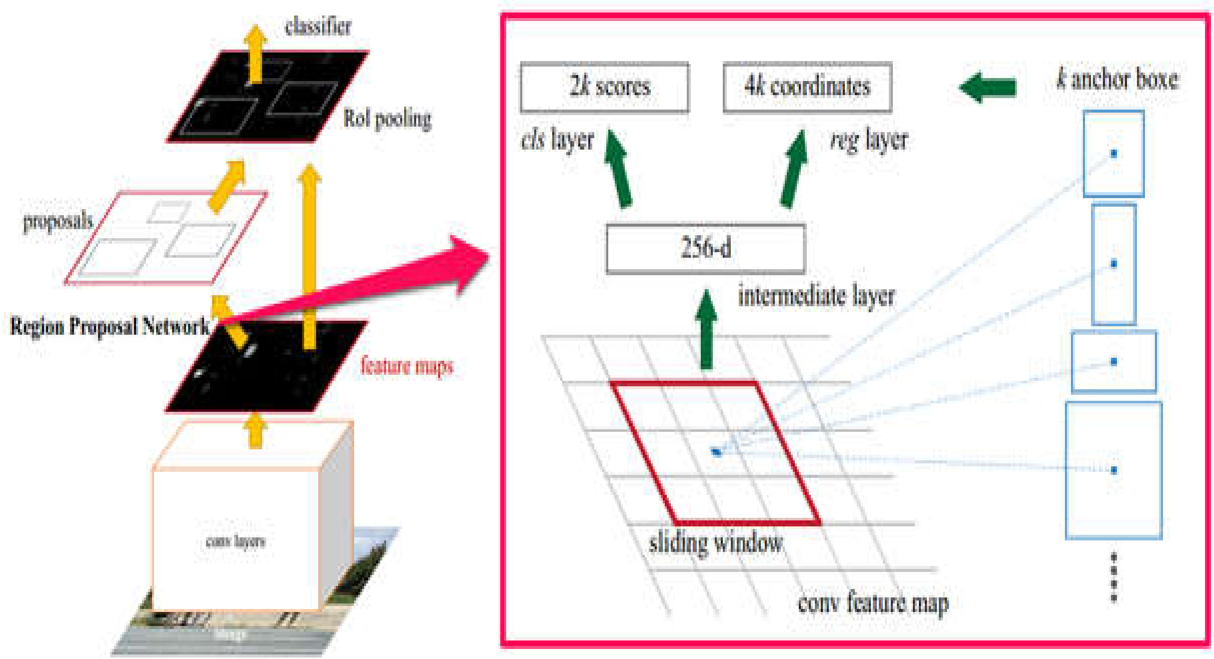


Fig.5. Left: Overview of faster R-CNN network. Right: Region proposal network [7]

3. RESULTS AND DISCUSSION

In this section, we observe the mean average precision of different type of parameters. Average precision (AP) is the common metric used to recognize object for each categories [12]. The parameters are the minimum batch size, maximum pixel size of a scaled input image and the image scales that is also the short edge of the input image. We vary five of each parameters to observe its influence on the mAP. The dataset used for detection benchmark is PASCAL VOC 2007. The dataset consists of approximately 5000 training images and 5000 testing images with 20 object categories that are airplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, dining table, dog, horse, motorbike, person, potted plant, sheep, sofa, train and television. In addition, Zeiler and Fergus (ZF) network is implemented in the experiment that has five convolutional layers and three fully connected layers [13]. We evaluate the training and testing in NVidia graphic card since it is the only GPU that MATLAB supports as of now. Besides that, GPU speeds up training and testing over any central processing unit (CPU) greatly [14].

Table 1 shows that the higher mini-batch size slightly increase for some individual categories on VOC 2007. In the bottle category, a mini-batch size of 129 increase drastically than the previous numbers.

Table 1. Results on various minimum batch size

	Mini-Batch Size				
Category	**64**	**80**	**96**	**112**	**128**
mAP	58.3	59.1	59.5	59.7	59.7
Aero	65.3	63.3	65.8	65.1	65.7
Bike	71.3	71.7	69.5	73.6	70.5
Bird	56.5	55.4	56.9	53.9	56.4
Boat	44.3	43.0	44.3	42.0	43.4
Bottle	28.0	30.0	30.8	31.5	62.3
Bus	66.7	66.8	69.0	66.8	69.2
Car	72.3	72.9	73.7	73.5	73.2
Cat	71.5	71.0	71.6	72.9	71.9
Chair	33.1	33.3	35.6	35.7	34.7
Cow	63.3	65.0	64.7	65.6	64.3
Table	58.7	62.1	62.6	64.4	62.4
Dog	63.6	67.6	67.6	68.4	67.5
Horse	75.0	76.7	76.0	79.0	75.5
Motorbike	68.4	67.5	67.8	66.3	66.9
Person	64.5	64.3	64.7	64.9	65.3

Plant	28.2	30.4	26.6	30.8	28.7
Sheep	56.8	59.3	60.4	58.7	59.9
Sofa	54.0	55.5	56.4	52.6	56.6
Train	69.0	68.8	68.8	69.9	71.2
TV	55.8	57.7	57.4	59.2	59.2

In Table 2, increasing the maximum pixel size of the scale input image gives better results compared to mini-batch size for most individual categories. This is one of the important factors influence the average precision for each categories. Hence, giving a better mean average precision for overall categories.

Table 2. Results on various maximum pixel size of A scaled input image

	Maximum Pixel Size				
Category	**600**	**650**	**700**	**750**	**800**
mAP	56.2	57.1	58.4	59.3	60.2
Aero	59.1	61.9	63.6	62.7	65.2
Bike	69.7	68.9	71.6	71.6	74.3
Bird	52.3	55.3	57.6	57.6	58.2
Boat	40.2	38.7	43.8	44.6	44.8
Bottle	26.9	29.4	27.9	29.4	33.4
Bus	65.3	63.5	65.6	63.1	67.3
Car	72.2	71.9	73.4	74.0	73.3
Cat	65.1	67.7	70.3	69.0	71.5
Chair	32.8	32.9	34.0	35.5	38.4
Cow	62.4	63.8	65.0	66.0	66.1
Table	58.8	56.7	61.6	60.7	62.2
Dog	61.3	65.6	66.2	68.5	68.2
Horse	74.0	75.6	75.8	76.5	77.9
Motorbike	64.6	66.5	69.1	68.6	66.9
Person	63.1	64.8	63.9	64.7	64.9
Plant	29.6	28.3	30.3	29.9	29.2

Sheep	58.4	56.7	58.3	58.7	58.2
Sofa	48.8	50.8	47.4	56.5	55.2
Train	67.8	67.7	69.2	71.3	69.9
TV	51.7	54.8	53.9	58.0	58.4

Lastly, in Table 3, increasing the short edge of the input image or image scales give significance difference in overall mean average precision than previous numbers. Image scales is also one of the major parameters to influence the average precision for each categories. Hence, most categories give better results with the increase of image scales.

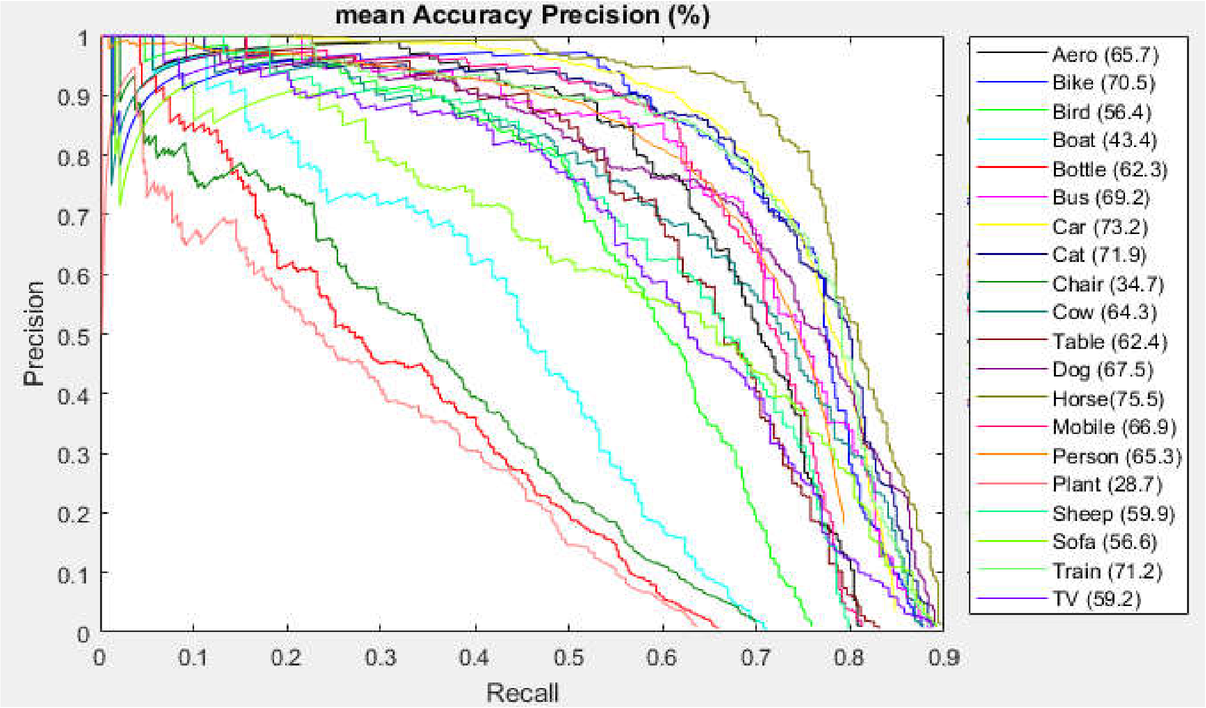


Fig.6. Precision vs. Recall on PASCAL VOC 2007 dataset

Table 3. Results on various image scales

	Image Scales				
Category	**300**	**350**	**400**	**450**	**500**
mAP	45.9	49.8	54.3	56.3	58.1
Aero	50.4	53.6	56.0	61.9	65.3
Bike	53.6	59.3	66.0	69.7	70.6
Bird	38.7	45.6	53.5	55.4	52.6
Boat	31.2	31.3	37.8	38.1	41.3

Bottle	24.3	25.7	28.1	28.3	31.1
Bus	48.9	55.4	58.3	61.8	63.4
Car	62.9	67.7	70.7	72.2	72.7
Cat	48.0	54.1	61.0	67.5	68.4
Chair	27.6	28.2	32.2	33.3	34.0
Cow	53.2	57.6	63.7	60.4	63.3
Table	43.0	54.1	57.7	58.3	63.7
Dog	43.5	44.3	57.6	62.5	64.9
Horse	66.0	66.1	71.6	75.0	75.5
Motorbike	55.3	61.7	63.7	66.5	66.8
Person	56.1	59.6	61.9	63.5	64.1
Plant	22.2	23.9	24.3	27.0	28.2
Sheep	51.4	54.4	56.7	55.6	57.3
Sofa	38.7	42.0	46.6	50.2	53.2
Train	55.1	62.1	65.7	67.3	68.0
TV	47.3	49.5	52.8	52.1	58.1

Another evaluation is analyzed using Simonyan and Zisserman model (VGG-16) to observe the mean average precision. VGG-16 model has 13 convolutional layers and 3 fully connected layers [15]. In Table 4, we set the parameters of 800*800 maximum pixel size, 500*500 image scales and mini-batch size of 128. The mean average precision result in 7.7% better than using ZF model.

Table 4. Results on VOC 2007 using VGG-16 network compared with ZF network

Category	**VGG-16 Network**	**ZF Network**
mAP	67.9	60.2
Aero	68.2	65.2
Bike	78.6	74.3
Bird	64.6	58.2
Boat	53.5	44.8
Bottle	47.8	33.4

Bus	76.0	67.3
Car	79.4	73.3
Cat	80.6	71.5
Chair	48.9	38.4
Cow	77.5	66.1
Table	64.5	62.2
Dog	78.3	68.2
Horse	81.7	77.9
Motorbike	75.3	66.9
Person	76.0	64.9
Plant	34.9	29.2
Sheep	67.7	58.2
Sofa	63.5	55.2
Train	75.7	69.9
TV	65.8	58.4

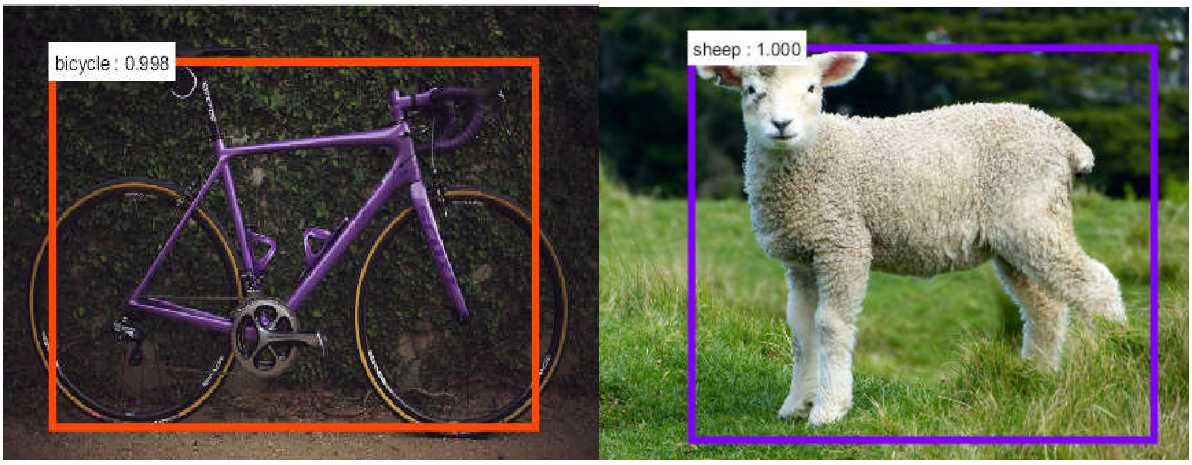
Table 5 shows the best mAP for each categories when the given parameters are set with the best value. It seems that adjusting the image scales does not give the best mAP at any values compared to the other two parameters for all categories. On the other hand, minimum batch size and maximum pixel size give the best mAP performance at higher value for some categories.

Table 5. Results on best mAP based on the given best parameters

Category	Mini-Batch Size	Max Pixel Size	Image Scales	mAP
Aero	96	-	-	65.8
Bike	-	800	-	74.3
Bird	-	800	-	58.2
Boat	-	800	-	44.8
Bottle	128	-	-	62.3
Bus	128	-	-	69.2
Car	-	750	-	74.0

Cat	112	-	-	72.9
Chair	-	800	-	38.4
Cow	-	800	-	66.1
Table	112	-	-	64.4
Dog	-	750	-	68.5
Horse	112	-	-	79.0
Motorbike	-	700	-	69.1
Person	128	-	-	65.3
Plant	112	-	-	30.8
Sheep	96	-	-	60.4
Sofa	128	-	-	56.6
Train	-	750	-	71.3
TV	128	-	-	59.2

In Fig. 6 shows the comparison of average precision between each categories with mini-batch size of 128, 1000*1000 maximum pixel size and 600*600 image scales. The plot shows that the horse category has the most precision while plant has the least precision due to difficulty to extract the features from the category.



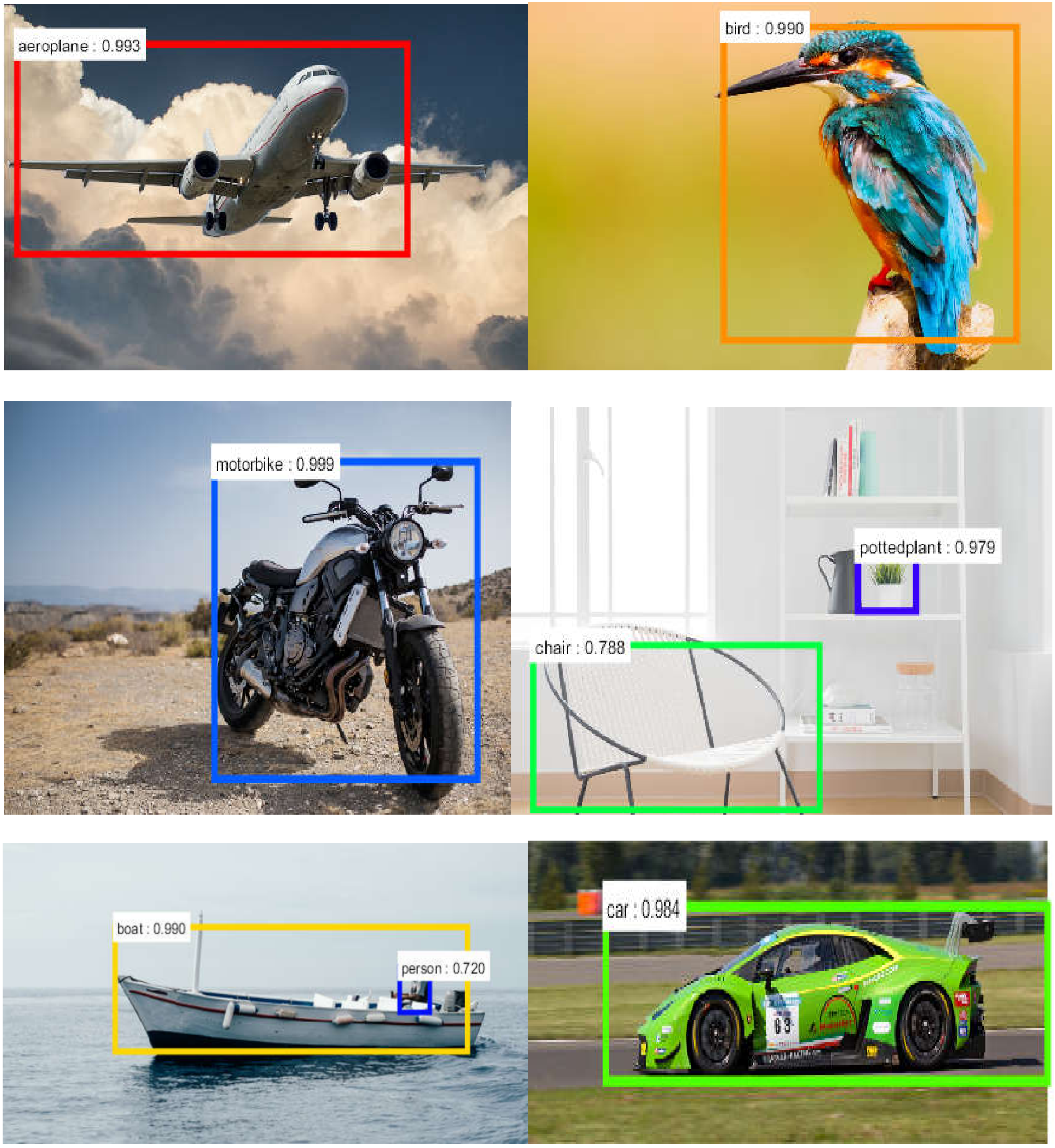


Fig.7. Examples of object detection results using PASCAL VOC 2007 dataset for mini-batch size of 128

Fig. 7 shows four selected examples to detect object on VOC 2007 dataset using ZF model with the best parameters. The mAP for this test is 59.7%. Each output of the color boxes represents each categories label with softmax score within [0, 1]. The score threshold was also left by default in value 0.6 to display the images given.

4. CONCLUSION

In conclusion, this paper has presented the performance evaluation when three different parameters are changed that are maximum pixel size, minimum batch size and image scales with the average precision of each categories on PASCAL VOC 2007 dataset. Although increasing the values of the parameters do not necessarily improve the average precision of each categories, it may slightly improve the overall mean average precision for all categories of the dataset. There are many parameters can be tweaked to observe the changes of mean average precision [16].

5. ACKNOWLEDGEMENTS

I would like to acknowledge PASCAL Visual Object Classes Challenge 2007 for providing the dataset for open-source use. This work is financially supported by Research Acculturation Grant Scheme, RAGS (600-RMI/RAGS 5/3 (16/2015). We also would like to thank Universiti Teknologi MARA for the management support and consideration throughout the project.

6. REFERENCES

[1] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation.In IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431-3440

[2] Dong C, Loy C C, He K, Tang X.Image super-resolution using deep convolutional networks.IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(2):295-307

[3] Alexe B, Deselaers T, Ferrari V. Measuring the objectness of image windows. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(11):2189-2202

[4] Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation.In IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580-587

[5] He K, Zhang X, Ren S, Sun J. Spatial pyramid pooling in deep convolutional networks for visual recognition.In European Conference on Computer Vision, 2014, pp. 346-361

[6] Girshick R. Fast R-CNN.In IEEE International Conference on Computer Vision, 2015, pp. 1440-1448

[7] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks.In Advances in Neural Information Processing Systems, 2015, pp. 91-99

[8] Zhang L, Lin L, Liang X, He K. Is faster R-CNN doing well for pedestrian detection?In European Conference on Computer Vision, 2016, pp. 443-457

[9] Hosang J, Benenson R, Schiele B. How good are detection proposals, really?In British Machine Vision Conference, 2014, pp. 1-25

[10] Uijlings J R, Van De Sande K E, Gevers T, Smeulders A W. Selective search for object recognition. International Journal of Computer Vision, 2013, 104(2):154-171

[11] Zitnick C L, Dollár P.Edge boxes: Locating object proposals from edges.In European Conference on Computer Vision, 2014, pp. 391-405

[12] Everingham M, Van Gool L, Williams C K, Winn J, Zisserman A.The pascal visual object classes (VOC) challenge. International Journal of Computer Vision, 2010, 88(2):303-338

[13] Zeiler M D, Fergus R. Visualizing and understanding convolutional networks.In European Conference on Computer Vision, 2014, pp. 818-833

[14] Vasilache N, Johnson J, Mathieu M, Chintala S, Piantino S, LeCun Y. Fast convolutional nets with fbfft: A GPU performance evaluation.In 3rd International Conference on Learning Representations, pp. 1-17

[15]Simonyan K, Zisserman A.Very deep convolutional networks for large-scale image recognition.In 3rd International Conference on Learning Representations, 2015, pp. 1-14

[16] Fadhlan H K Z, Md. Hazrat A, Amir A S, Zairi I R.Development of mobile face verification based on locally normalized gabor wavelets. International Journal on Advanced Science, Engineering and Information Technology, 2017, 7(3):1026-1031

How to cite this article:
Adam B, Zaman F H K, Yassin I M, Abidin H Z, Rizman Z I. Performance evaluation of faster r-cnn on gpu for object detection. J. Fundam. Appl. Sci., 2017, *9(3S), 909-923*