Proceedings of IOE Graduate Conference, 2017
Volume: 5 ISSN: 2350-8914 (Online), 2350-8906 (Print)

Object Detection in Images using Region Based CNN

Deepesh Lekhak [a], Basanta Joshi [b], Sushma Shrestha [c]

[a, b, c] *Department of Electronics and Computer Engineering, Pulchowk Campus, IOE, TU, Nepal*
Corresponding Email: [a] deepeshlekhak@gmail.com, [b] basanta@ioe.edu.np, [c] sanushr7@gmail.com

Abstract
Object detection is the task of recognizing and localizing objects in an image. Object detection in images have many applications including object counting, Visual Search Engine, security, surveillance etc. Deep Learning based techniques for object detection are divided into two categories as region based approch and single shot approach. In this paper, region based approach technique Faster R-CNN was implemented using ResNET architecture of Convolutional Neural Network(CNN). The architecture of ResNET is modified to incorporate region proposal network to propose probable region of interest in an image and classification and regression network to detect and classify objects and their boundary in an image. The results were compared with Faster R-CNN based on VGG-16 on PASCAL VOC dataset . It was found that Faster R-CNN based on ResNET provides mean average precision of 0.78 which is better performance on PASCAL VOC dataset than VGG-16 architecture with mean average precision of 0.699.

Keywords
Object Detection – Convolutional Neural Network – Region Proposal – Residual Network

1. Introduction

The problem of recognizing objects in images has been studied extensively over the decades but still remains a challenging task. In recent years, the study of deep learning has been a growing interest due to its superior performance in several recognition tasks, for instance, activity recognition, object detection, and scene classification. Deep Convolutional networks based methods have become the state of the art in object detection in image. Object detection can be carried out with classification on different sub-windows or patches or regions extracted from the image. The patch with high probability had not only the class of that region but also implicitly gives its location too in the image. Most of the approaches vary on the type of methodology used for choosing the windows. Convolutional neural networks(CNN) were used as feature extractor for the classification. Region based convolutional neural networks were popular method used to detect objects in images.

Object detection in images was started with classical approaches such as classifier based on features such as Haar-like Features, HOG features [1] and use classifiers such as SVM [2], Bayesian Classifier etc. Deep Learning based approach deep learning models have outperformed other classical models on the task of image classification, deep learning models are now state of the art in object detection as well. Deep learning based approaches are evolved using Convolutional Neural Networks which mimics the visual cortex of the animals. Methods such as Overfeat [3], Region Based CNN [4], Single Shot Multiplex Detector [5] etc are some examples of deep learning approach.

2. Related Work

In Computer Vision Object detection is one of the fundamental problems and has been studied for years to make it more efficient and faster. Most of the classical object recognition methods involve edge (or contour) [6, 7] and patch [8, 9] based feature extraction. In object detection, some of the efficient techniques exploit sliding window [10] and boosting [11]. Convolutional neural network (CNN) has become dominating model due to its outstanding performance in object detection [12, 13]. Girshick et al.[4] demonstrated outstanding performance in terms of detection accuracy for object detection in images using Region based Convolutional

Neural Network (R-CNN). However, this approach has large computational complexity in order to classify a large number proposed regions. Selective search [14] is used commonly to generate object proposals. However, due to exhaustive search and large number of region proposals from an image, it is computationally expensive. R. Girshick [15] drastically reduced computational cost of R-CNN with the sharing convolutions across proposals by Fast R-CNN. Fast R-CNN achieves near real-time rates using very deep networks, when ignoring the time spent on region proposals. Proposals were the computational bottleneck in state-of-the-art detection systems using Fast R-CNN. Shaoqing Ren et al. [16] developed Faster R-CNN method based on VGG-16 using Region Proposal Networks (RPNs) that share convolutional layers with state-of-the-art object detection that leads to an elegant and effective object detection solution for images. Faster Region based CNN (F-RCNN) [16] using VGGNet provided 0.69 mean average precision for object detection using PASCAL VOC dataset [17]. This architecture yields a large model for CNN and provided slow processing rate(frame per second)for object detection. In this research work, performance of the Faster R-CNN is enhanced with implementation of the method using Residual Network.

3. Research Methodology

Deep Residual Network

Deep Residual Network (ResNET) [18] is state-of-the art architecture of CNN based on residual component. Residual Network is composed of various residual Learning Block. In each component batch normalization is performed to normalize the training samples in each batch, which enhance the training process, which is followed by ReLU non linearity activation function.

The final computation of a residual block is:

$$\sigma(F(x)+x) = \sigma([W_2\sigma(W_1x+b_1)+b_2]+x) \quad (1)$$

where σ is ReLU non-linearity activation function. A ResNET architecture is built using residual components, which includes convolution, pooling and activation layers. In this research work, a ResNET network with 50 layer architecture is implemented.

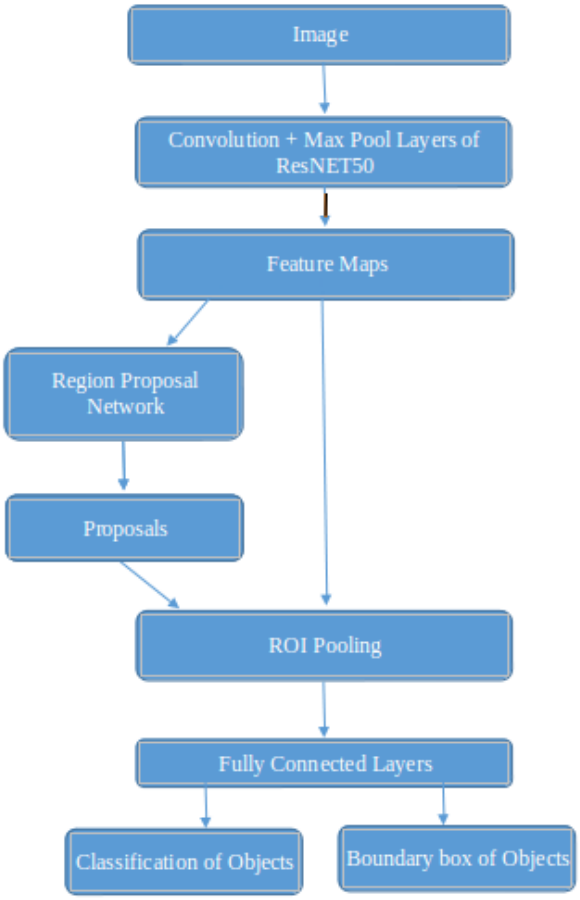

Figure 1: Architecture of object detection using Faster R-CNN based on ResNET50

The above architecture is modified to implement Faster R-CNN method by removing layers after average pool layer at the end and by adding Region Proposal Network using fully convolutional neural network which is followed by Region of Interest (RoI) pooling fully connected layers for classifier and boundary box regressor.

Faster R-CNN

Faster R-CNN implement region proposal mechanism using the CNN and thereby makes region proposal a part of the CNN training and prediction steps. Region Proposal Network is proposed to predict proposal regions in an image. RPN slides over the last shared Convolution feature map of ResNET50 model to determine whether the region is an object or not. The convolution feature maps are shared for RPN and Object recognition. For each image, it is fed forward to

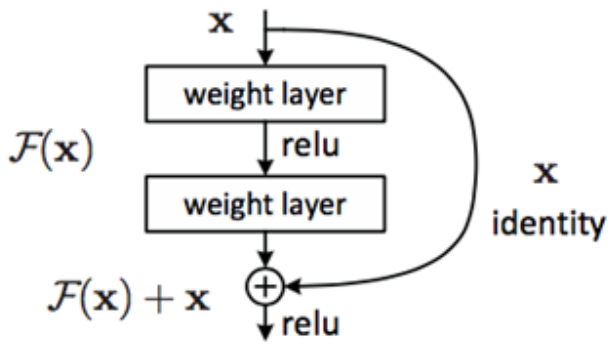

Figure 2: A Residual Component [18]

get a Convolution feature map from the last Convolution layer. Then Region Proposal Network (RPN) is used to determine if any object present or not in the image.

Region Proposal Network

A Region Proposal Network (RPN) takes an image as input and outputs a set of rectangular object proposals, each with an object-ness score. RPN is modelled with a fully convolutional network. Region proposals are generated by sliding a small network over the convolutional feature map output by the last shared convolutional layer. The input of this network is a spatial window of input convolutional feature map, which is then fed into two different fully-connected layers — a box-regression layer and a box-classification. After getting proposals, each proposed region on Convolution feature map is passed into a Region of Interest (RoI) pooling layer, the purpose is to get a fixed feature vector output to later Fully Connected layers. The layer size is varying according to the size of input feature map. Since the layer size is changing, it always outputs a fixed feature vector. The RoI feature vector is fully connected to a Fully Connected layer and performs classification of objects in RoI region.

Training

The input images were re-sized such that the lowest side is equal to 600 pixel while aspect ratio was kept same as original image. The image sets were divided into batch of 8 images and fed to ResNet network, the output from last convolution and pooling layer is forwarded to Region Proposal Network (RPN). RPN consists of fully connected layers which took base convolutional feature maps and number of anchor and their size and aspect ratios as input. For anchors 3 scales with box areas of 128 , 256 , and 512 pixels, and 3 aspect ratios of 1:1, 1:2, and 2:1 were used.

The RPN utilizes a sliding window approach in First, an n by n filter is convolved with last layer convolution feature map. Then the result is projected to a lower dimensional space by convolving with a 1 by 1 filter (which just linearly combines the channels for each position independently), resulting in a fixed-size vector for each position. The vector is separately passed into a box-regression layer and a box-classification layer. In the box-regression layer, k bounding boxes are generated relative to the current position in the conv feature map (the current anchor point).

The box-classification layer generates 2k outputs, where each pair of 2 outputs is the probability that the corresponding bounding box has an object in it or is just background. That is, the sum of each pair of outputs is 1, and is the probability distribution over whether the bounding box contains an object or not. To reduce the number of bounding box proposals, non-maxima suppression is used on proposals that have intersection-over-union (IoU) higher than 0.7 and only top 300 boundary boxes were taken for the training. The boxes are ranked based on the object probability score.

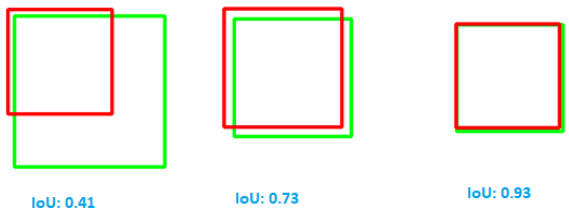

Figure 3: Intersection over Union Calculation

For training RPNs a binary class label of being an object or not to each anchor had been assigned. A positive label had been assigned to two kinds of anchors as the anchor/anchors with the highest Intersection-over-Union (IoU) overlap with a ground-truth box and an anchor that has an IoU overlap higher than 0.7 with any ground-truth box. A single ground-truth box may assign positive labels to multiple anchors. A negative label had been assigned to a non-positive anchor if its IoU ratio is lower than 0.3 for all ground-truth boxes. Anchors that were neither positive nor negative do not contribute to the training

objective.

The objective function had been minimized following the multi-task loss in Fast R-CNN [15] defined as:

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}\,(p_i,\,p_i^*) + \lambda\frac{1}{N_{reg}}\sum_i p_i^*\,L_{reg}\,(t_i,\,t_i^*) \quad (2)$$

Here, i is the index of an anchor in a mini-batch and pi is the predicted probability of anchor i being an object. The ground-truth label p_i* is 1 if the anchor is positive, and is 0 if the anchor is negative. t_i is a vector representing the 4 parameterized coordinates of the predicted bounding box, and t_i* is that of the ground-truth box associated with a positive anchor. The classification loss L_{cls} is log loss over two classes (object vs. not object). For the regression loss L_{reg} (t_i, t_i*) = R(t_i - t_i*) where R is the robust loss function (smooth L1) defined in [18]. The term pi* L_{reg} means the regression loss is activated only for positive anchors (p_i* = 1) and is disabled otherwise (p_i*= 0).
The RPN has been trained end-to-end by backpropagation and stochastic gradient descent (SGD). Unlike gradient descent where gradients were calculated after running on entire dataset, SGD calculates over few examples at a time.
Then, for each object proposal a region of interest (RoI) pooling layer had been extracted as fixed-length feature vector from the feature map. Each feature vector had been fed into a sequence of fully connected layers that finally branch into two sibling output layers: one that produces softmax probability estimates over K object classes plus a catch-all "background" class and another layer that outputs four real-valued numbers for each of the K object classes. Each set of 4 values encodes refined bounding-box positions for one of the K classes. Then RPN and CNN networks has been merged into one network during training. In each SGD iteration, the forward pass had generates region proposals which were treated just like fixed, precomputed proposals when training a CNN detector. The backward propagation had taken place as usual, where for the shared layers the backward propagated signals from both the RPN loss and the CNN loss had been combined.
The output from the network has two component: classifier that provides class of the detected objects and regressor that provides four co-ordinates of the detected objects boundary boxes.

4. Results

The system for detecting objects in images is trained and evaluated in PASCAL VOC dataset [17]. This dataset contains fully annotated images with 5000 images tagged as train/validation set and 5000 images tagged as testing set. 5-Fold cross-validation was used in this research work. The dataset is fully annotated with object category and co-ordinates of boundary box of object in the image.The annotations of the images of all twenty classes are called as ground truth, which contains class name and bounding box i.e. an axis-aligned rectangle specifying the extent of the object visible in the image. Based on these annotations Region Proposal Network was trained to predict bounding boxes in an image.

Training

Faster R-CNN based on ResNet with 50 layers is trained with this dataset for 50 epochs. The images were resized as the lowest side is 600 pixels keeping aspect ratios. The training and validation results after 50 epochs of training, the results are shown in table 1

Table 1: Training results on PASCAL VOC image Dataset

Metrics	Training	Validation
Classifier accuracy bounding box	0.82	0.77
Loss RPN classifier	0.30	0.45
Loss RPN regression	0.15	0.40
Loss Detector classifier	1.20	2.68
Loss Detector regression	0.38	0.33

The trained model based on ResNET50 was found to be smaller (115 MB) than the trained model based on VGG-16 (530 MB)for the implementation of Faster R-CNN.

Figure 4: Output of the system on Sample Images

Table 2: Performance of Faster R-CNN with ResNet in PASCAL VOC Dataset

ID	Class	mAP VGG-16	mAP ResNET
1	Aeroplane	0.7	0.81
2	Bicycle	0.806	0.9
3	Bird	0.701	0.85
4	Boat	0.573	0.84
5	Bottle	0.499	0.86
6	Bus	0.782	0.77
7	Car	0.804	0.86
8	Cat	0.82	0.82
9	Chair	0.522	0.86
10	Cow	0.753	0.84
11	DiningTable	0.672	0.85
12	Dog	0.803	0.68
13	Horse	0.798	0.51
14	Motorbike	0.75	0.91
15	Person	0.763	0.51
16	PottedPlant	0.391	0.72
17	Sheep	0.683	0.58
18	Sofa	0.673	0.79
19	Train	0.811	0.71
20	Tv/monitor	0.676	0.87
	Total mAP	0.699	0.78

Evaluation

The evaluation was done using mean average precision (mAP), which is widely used in object detection/classification. The mean average precision for all twenty categories and total mAP with ResNET is compared with that of VGG-16 from []is compared in table 2

The mean average precision for PASCAL VOC dataset using Faster R-CNN with ResNet with 50 layers was found 0.78 which is better than Faster R-CNN with VGG-16 which is 69.9 as given in [16]. The comparison of the performance of Faster R-CNN with ResNET50 architecture in this research work with Faster R-CNN with VGG-16 architecture is depicted in Figure 5

The timing of the system using ResNET50 on K520 instance is compared with that of VGG-16 in table **??**

Table 3: Comparison of Timing of the system with VGG-16 and ResNET50

Model	System	Rate
VGG-16	RPN + Fast R-CNN	5 fps
ResNET50	RPN + Fast R-CNN	12 fps

The system was then evaluated using test samples of PASCAL VOC dataset. The output from the system are

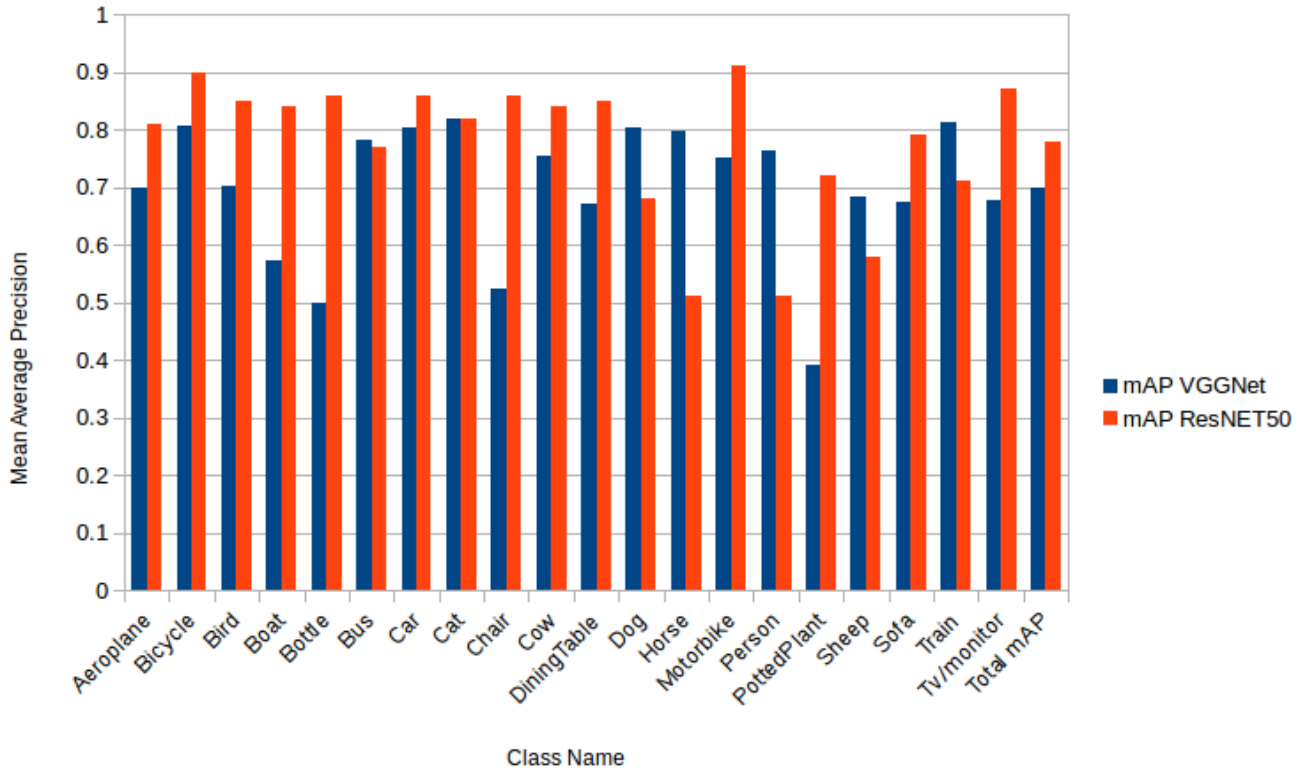

Figure 5: Performance of VGG-16 architecture vs ResNET50 architecture

shown in Figure 4

5. Conclusion

In this research work Faster R-CNN method is implemented based on ResNET with 50 layer architecture with mean average precision of 0.78 on PASCAL VOC Dataset. It provided better performance than Faster R-CNN based on VGG-16 with mean avearage precision of 0.699[16]. Although ResNET is a deeper architecture than VGG-16 , ResNET model is lighter in size, and provides higher accuracy than VGG-16.

Acknowledgments

The authors are thankful to Department of Electronics and Computer Engineering, Pulchowk Campus, IOE, TU for all the support and guidance in this research work.

References

[1] Kosuke Mizuno, Yosuke Terachi, Kenta Takagi, Shintaro Izumi, Hiroshi Kawaguchi, and Masahiko Yoshimoto. Architectural study of hog feature extraction processor for real-time object detection. In *Signal Processing Systems (SiPS), 2012 IEEE Workshop on*, pages 197–202. IEEE, 2012.

[2] Hao Zhang, Alexander C Berg, Michael Maire, and Jitendra Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2126–2136. IEEE, 2006.

[3] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[4] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[5] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European*

conference on computer vision, pages 21–37. Springer, 2016.

[6] Jamie Shotton, Andrew Blake, and Roberto Cipolla. Multiscale categorical object recognition using contour fragments. *IEEE transactions on pattern analysis and machine intelligence*, 30(7):1270–1281, 2008.

[7] Konrad Schindler and David Suter. Object detection by global contour shape. *Pattern Recognition*, 41(12):3736–3748, 2008.

[8] Bastian Leibe, Aleš Leonardis, and Bernt Schiele. Robust object detection with interleaved categorization and segmentation. *International journal of computer vision*, 77(1-3):259–289, 2008.

[9] Bjorn Ommer and Joachim Buhmann. Learning the compositional nature of visual object categories for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(3):501–516, 2010.

[10] Andrea Vedaldi, Varun Gulshan, Manik Varma, and Andrew Zisserman. Multiple kernels for object detection. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 606–613. IEEE, 2009.

[11] Zhiquan Qi, Yitian Xu, Laisheng Wang, and Ye Song. Online multiple instance boosting for object detection. *Neurocomputing*, 74(10):1769–1775, 2011.

[12] Christian Szegedy, Alexander Toshev, and Dumitru Erhan. Deep neural networks for object detection. In *Advances in neural information processing systems*, pages 2553–2561, 2013.

[13] Dumitru Erhan, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. Scalable object detection using deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2147–2154, 2014.

[14] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

[15] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[16] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[17] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.