TEST Engineering & Management

*May-June 2020*
*ISSN: 0193-4120 Page No. 4191-4194*

# Object Detection Using Deep Learning from Very High Resolution Imagery

**[1]Vijayalakshmi. S, [2]Magesh Kumar**

[1]Assistant Professor, [2]Associate Professor
[1,2] Dept of CSE, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Thandalam, Chennai, Tamilnadu, India- 602 105
[1]vijilak.sse@gmail.com, [2]mageshkumars.sse@saveetha.com

***Article Info***
***Volume 83***
***Page Number: 4191-4194***
***Publication Issue:***
***May - June 2020***

***Article History***
***Article Received:****19 November 2019*
***Revised:*** *27 January 2020*
***Accepted:*** *24 February 2020*
***Publication:*** *12 May 2020*

## Abstract

In recent technology development of enumerous vehicle surveillance, video surveillance, facial recognition, crowd monitoring application and some more hence object detection algorithm are in demand. These kind of algorithm involves detecting and classifying every object in an image, but also positioning every object by marking the appropriate boundary over the region. Also makes object detection accurately in more complex images than existing image classification algorithms. In this paper gives you a comprehensive survey on newer algorithm introduced for object detection using deep learning.

***Keywords*****:** *object detection, image classification, computer vision, deep learning*

## 1. Introduction

**Object detection** methodologies are enhancement of image classification algorithm. Recently Google has released a newer object detection API using Tensorflow which has a pre built architecture for newer models:

- Region – Based Fully Convolution Networks
- Faster RCNN
- Fast RCNN
- Single Shot Multibox Detector

This paper covers the comprehensive survey on the above models and gives you the better model which can be used for object detection.

## 2. R-CNN

R-CNN or Region based convolution neural network consist of 3 simple steps,

i. The input image is analysed for objects regions using the algorithm named selective search which can generate approximately 2000 region.
ii. Then CNN is applied on over these proposed regions which required from previous step
iii. The results of each CNN are feed as input to SVM to classify the object region and a linear regression to obtain the boundary region of every object.

The above stated three points are illustrates in the image Fig.1

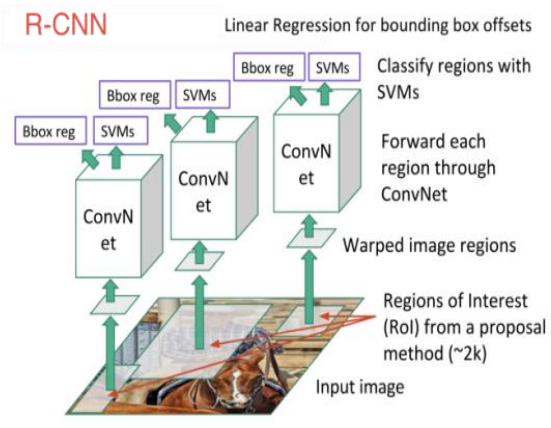


Figure 1: R-CNN

On optional way, at first the suggested regions are extracted and then classification of those object regions based on their features is done, R-CNN was very effective, but very slow when high resolution imagery is used.

May-June 2020
ISSN: 0193-4120 Page No. 4191-4194

## 3. Fast-RCNN

Enhanced version of R-CNN is Fast R-CNN with improvement on its detection accuracy with two main key idea.

i. Feature extraction are done over the image before the object region are proposed, hence only one CNN is implemented over 2000 regions
ii. Using of softmax layer rather than SVM extends the neural network to predict instead of creating a newer region.

The new Fast R-CNN looks like Fig. 2.

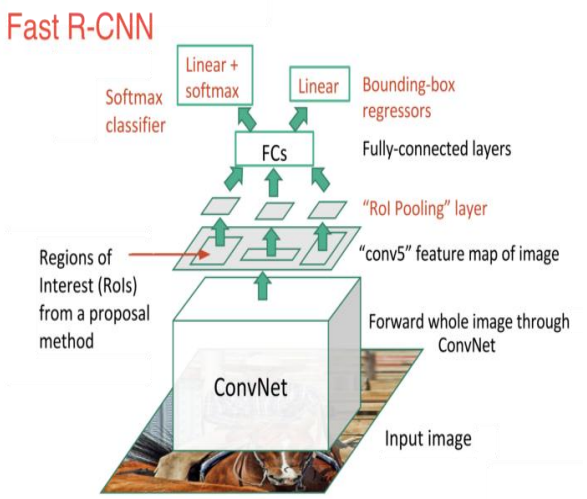


Figure 2: Fast R-CNN

As inference from the above picture we see object region are proposed based on feature extracted and not from original source image. In addition it does not have softmax layer to outputs the different probable class.

Fast R-CNN performs much in speed. But one issue remaining, it's the use of selective search algorithm for proposing specific regions.

## 4. Faster-RCNN

The main aspect of Faster-RCNN is fast neural network are applied instead of selective search algorithm. Thus it introduced a newer Region Proposed Network (RPN). Following are the working of RPN

i. Sliding window which traverse across the feature extraction layer and maps to the lower dimension. It is located at the last layer of an first CNN.
ii. For each sliding window position, it produces maximum possible object region based on k fixed ration boxes.
iii. Each region proposed has an object score value for the region and 4 points which represents the boundary region.

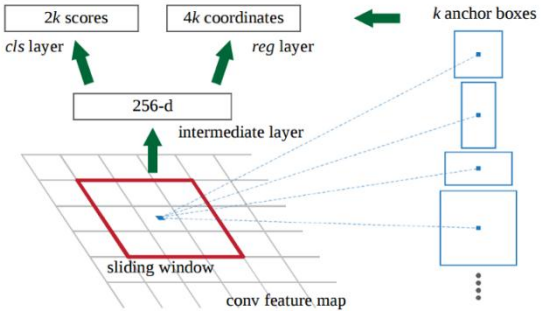


Figure 3: Sliding Window

Fig.3 depicts the working of sliding window. At each location of feature region map consider k different rectangular regions centred over it, a huge box, a wider box etc. For every rectangular regions result is predicted based on presence of the object.

RPN output the coordinates of boundary region; it does not classify any potential objects. An anchor region which contains the object score above the threshold value. It is forwarded as proposed region. Once the region has been extracted it is feed as input into Fast R-CNN. Thus a layer called pooling, fully-CNN and lastly softmax layer classification and bounding box region are projected. Hence Faster R-CNN= RPN+Fast R-CNN. Fig.4 shows how Faster R-CNN works.

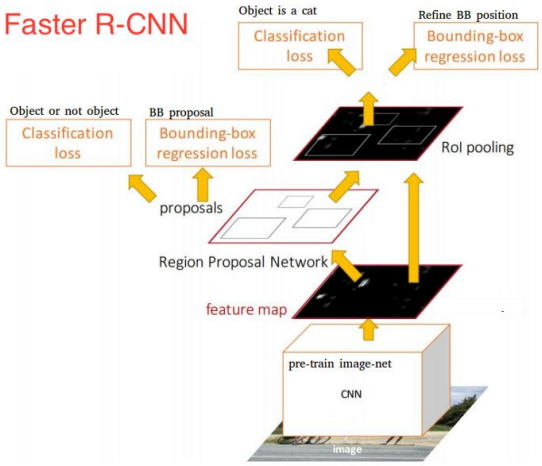


Figure 4: Faster R-CNN

Thus the algorithm achieves greater speed and accuracy. It also gives higher performance than other API models. The best real time example is Tensorflow's Faster R-CNN with Inception ResNet-slowest but most accurate model.

## 5. R-FCN

This works similar to Fast R-CNN which improves the object detection speed by spreading CNN across all proposed region. The same is followed in R-FCN –

*May-June 2020*
*ISSN: 0193-4120 Page No. 4191-4194*

Region- based Fully Convolutional Network, increasing the speed by maximizing shared result.

During the process of classifying an object, additional information on location variance of a model need to be known. For example, in regard of where the dog appears in the given image, at first we need to classify it as a dog. On other hand, during object detection, information of location is needed. if the dog appears to be in the top right-hand corner, we want to draw a covered region in the top right-hand corner. For compromise between location invariance and location variance, R-FCN gives a solution as positive sensitive score maps. Each position-sensitive score map represents *one relative position* of *one object class.* Following are the working of

i. At earlier stages runs a CNN over the given source image

ii. Fully convolution layer is added to produce a image score map– position sensitive score map, it must be $k^2(C+1)$ scores, using $k^2$ it represents the number of similar positions to divide an object and C+1 represents the number of classes along with the background.

iii. A fully RPN is applied to generate regions of proposed interest

iv. Now for every region RoI are divided into smaller $k^2$-bin or sub regions based on the score ma region

v. Every bin region must check the score to check if the bin reqion gets exactly matched to respective position of object

vi. After mapping eccery k2 bin to its respective object the value of every class are averaged to get the score value of per class.

vii. Finally now classification of each RoI can be done with softmax over left out C+1 vector dimension

A working R-FCN is depicted in Fig.5

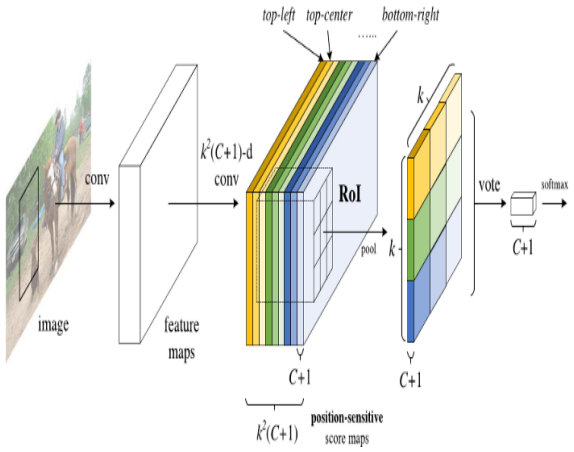


Figure 5: R-FCN

However R-FCN simultaneously detect the location information by object proposal refer to same as score map region. In accordingly all the score map must classify the object as object regardless where it appears. Finally R-FCN is minimize times accurate than Faster R-CNN, and also attains comparable accuracy.

## 6. SSD

**S**ingle-**S**hot **D**etector most Likely to R-FCN, which provides greater speed than Faster R-CNN.As discussed above in the models object region suggestion and region extraction are performed by two methods. At First region proposal network are used to calculate the region of interest, second fully connected layers or CNN are used to classify the regions. SSD accomplish the above two task in a single stage. Prediction of boundary region and its respective classes are processed simultaneously. Fig.6 shows the architecture diagram of SSD.

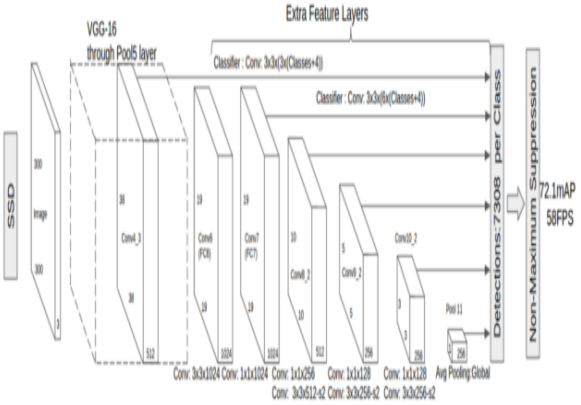


Figure 6: Architecture of SSD

Following points states the working of SSD

i. Initially the input image is passed through a convolution layers, which gives various sets of feature maps.

ii. Equivalent to boundary box region in Faster R-CNN a set of default boundary box region are generated by applying 3x3 convolution filter on each feature maps generated initially.

iii. For each box, prediction on boundary box offset and probable class is done simultaneously

iv. Over the process of training a perfect match is found based on the predicted boundary region. Which will be named as positive with match threshold>0.5

Hence SSD identify and extract the boundary boxes from all position in image with multiple shapes at different scale. As a result it generates larger number of boundary box than other models.

SSD simply avoids the preliminary stages on region proposal instead it consider every single boundary boxes and location to classify. All the process are done in a single shot, hence it is the fastest of all the other models.

*May-June 2020*
*ISSN: 0193-4120 Page No. 4191-4194*

Table 1: Shows the performance comparison with the models

| Model | Model Name | Overall Performance |
|---|---|---|
| R-CNN | Region based convolution neural network | very slow when high resolution imagery is used |
| Fast R-CNN | Fast Region based convolution neural network | performs much better in speed |
| Faster R-CNN | Faster Region based convolution neural network | slowest but most accurate model |
| R-FCN | Region- based Fully Convolution Network | achieves comparable accuracy |
| SSD | **S**ingle-**S**hot **D**etector | performs quite comparably |

## 7. Conclusion

This paper finally give you on various model of object detection using deep learning, and suggest which of these models performs well with one another. Faster R-CNN, R-FCN, SSD are the three models which are widely used currently. Other relative models which tends to be likely to these models which relays on deep neural network for classifying or object detection.

## References

[1] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification with deep convolutional neural networks. In NIPS, 2012.

[2] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional neural networks. In ECCV.

[3] K. Simonyan and A. Zisserman. Very deep onvolutional networks for large-scale image recognition. In ICLR, 2015.

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. CVPR, 2016.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in CVPR, 2014.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. FeiFei. ImageNet: A large-scale hierarchical image database. In CVPR, 2009.

[7] J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, and L. FeiFei. ImageNet Large Scale Visual Recognition Competition 2012 (ILSVRC2012).

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, 2007.

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. IJCV, 2010.

[10] Lin, Tsung Yi, et al. Microsoft COCO: Common Objects in Context. Computer Vision – ECCV 2014. pringer International Publishing, 2014:740-755.

[11] S. Vijayalakshmi, A Newest Data Set Analysis for Remote Sensing Applications – Jour of Adv Research in Dynamical & Control Systems, Vol. 11, 04-Special Issue, 2019.