International Journal of New Technology and Research (IJNTR)
ISSN:2454-4116, Volume-5, Issue-2, February 2019 Pages 48-52

# Comparison and Analysis of Algorithms used for Detecting Slums in Satellite Images

**Pallavi Saindane, Gayatri Ganapathy, Neha Prabhavalkar, Nilesh Bhatia, Aishwarya Vaidya**

***Abstract*— This paper presents a comparison of some neural network based approaches for analyzing satellite images. The motivation for this study comes from the potential use of such analysis for identification of slums from satellite images. Slums, also formally termed as informal settlements, can be identified from satellite images using image segmentation as well as object detection techniques. Policymakers often spend substantial time and resources to discover certain regions of interest. This paper presents a brief study and comparison of some known algorithms that can be helpful for classifying residential areas as developed and underdeveloped using satellite images as input. The approach can be further improved to identify the slums from satellite images. Such a process is then readily scalable and can reduce the time and resources spent by policymakers for this analysis. This can ultimately help policymakers in developing appropriate policies that lead to economic progress.**

***Index Terms*—Image Segmentation, Slum identification, Satellite images, Convolutional Neural Networks.**

## I. Introduction

Underdeveloped areas in any city are a matter of concern for most policymakers and government officials. Detecting such areas and gaining knowledge of the economic status of such areas often involves work in three different dimensions, that is, legal, social and spatial. Tax/Revenue Department can help in the legal dimension. For the social context, census survey and local groups and communities are approached for better understanding. Even then, there still would exist areas which are not surveyed due to the unawareness of their existence. For this, one needs the spatial dimension, where, using satellite images we identify infrastructural facilities and their linkages. There is a felt need for an automated system which can accurately identify the underdeveloped areas of concern in an ongoing manner as this would have a great influence in planning and policy-making for such cases. It is also understood that the features of a slum in a particular area or region may not be applicable as such to other regions. This paper explores various approaches to detect slums using satellite imagery and the accuracy, efficiency associated with these approaches. [1] [3]

**Mrs Pallavi Saindane**, Assistant Professor, Department of Computer Engineering, VESIT, University of Mumbai, Mumbai, India
**Gayatri Ganapathy**, Final year BE student, Department of Computer Engineering, VESIT, University of Mumbai, Mumbai, India
**Neha Prabhavalkar**, Final year BE student, Department of Computer Engineering, VESIT, University of Mumbai, Mumbai, India
**Nilesh Bhatia**, Final year BE student, Department of Computer Engineering, VESIT, University of Mumbai, Mumbai, India
BE student, Department of Computer Engineering, VESIT, University of Mumbai, Mumbai, India

## II. Methods

### A. CNN

Convolutional Neural Networks (CNN) are types of Neural Networks whose basic constituents are neurons which have weights and bias. A typical CNN has an input layer, an output layer and many hidden layers which are used for the intention of pooling, rectifying, etc.

### B. RCNN

Region Convolutional Neural Network (RCNN) is a variant of CNN in which Selective Search is used to extract region proposals from the input image which are then fed to a CNN whose purpose is to serve as a feature extractor.

### C. Faster RCNN

Faster RCNN is an extended version of CNN. Unlike RCNN which uses Selective Search, a time consuming algorithm, faster RCNN uses a network called Region Proposal Network (RPN) that learns the region proposals, and is considerably fast.

### D. Mask RCNN

Mask RCNN includes two basic modules: a CNN which is used for extraction of features, and a RPN which is used to locate region proposals. Among the four methods, Mask RCNN is the most efficient algorithm.

## III. Study of CNN

We stack the layers given below in CNN such that the output of one layer becomes input for another layer. Layers can be repeated several times. [6] So the image becomes more and more filtered as it goes through the convolution layer and becomes smaller as it goes through the pooling layer. ReLU layer is used to keep the math from breaking out.

### A. Convolution layer

CNN can be roughly classified into CNN for objects and CNN for grasping types. CNN takes the binary image as input with +1 for light, and -1 for the dark pixel. Thus, an image is a two-dimensional array of pixels. The object or pattern inside an image can be straight, translated, scaled, rotated or weighted. While the human can see such differences clearly, it is hard for computer software to differentiate or identify. Hence, the convolutional network analyses the image by breaking it into smaller parts, [6] and using these parts matching is done. For matching the parts of an image with an image, a filtering process is used. Filtering includes the following steps (Fig. 1, [7]):

1. Line up the image and the feature patch.
2. Multiply each image pixel by the corresponding feature pixel.
3. Add them up.
4. Divide by the total number of pixel in the feature.

Using this feature convolution tries to match each feature with the entire image. Hence, after applying a bunch of features, we get a stack of filtered images for each image. This is a convolution layer.

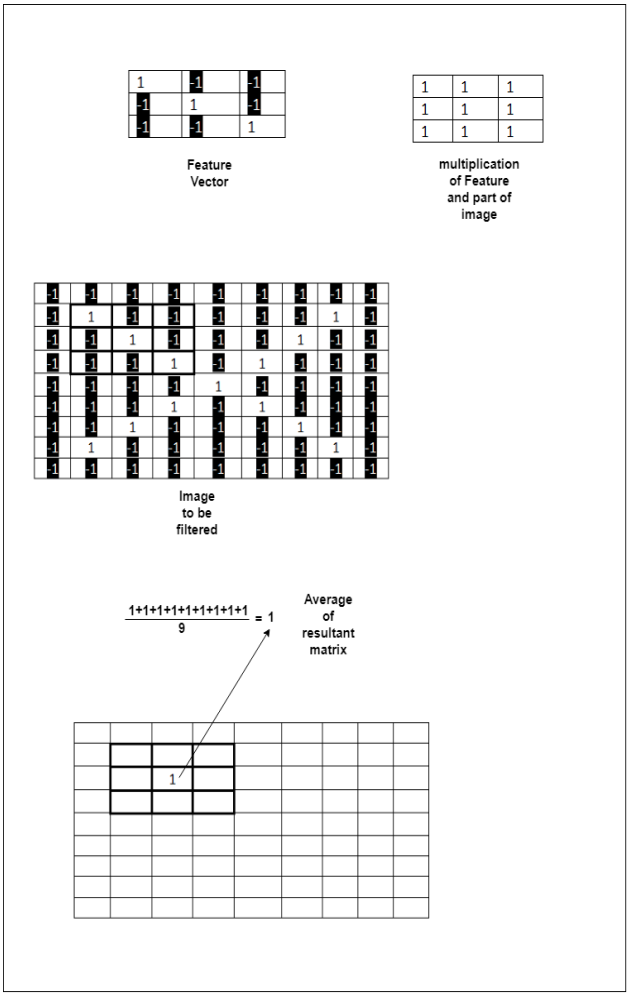


Fig 1. Filtering (Source: [7])

*B. Pooling layer*

In Pooling, we shrink the image stack. Shrinking the size of images makes the calculations much more manageable and easier. The process of max pooling is as follows [7]:

1. Pick a window size (usually 2 or 3).
2. Pick a stride (usually 2).
3. Walk your image across your filtered images.
4. From each window, take the maximum value.

In pooling we don't care about the position of maximum value in that window (Fig. 2). This makes it little less sensitive to position.

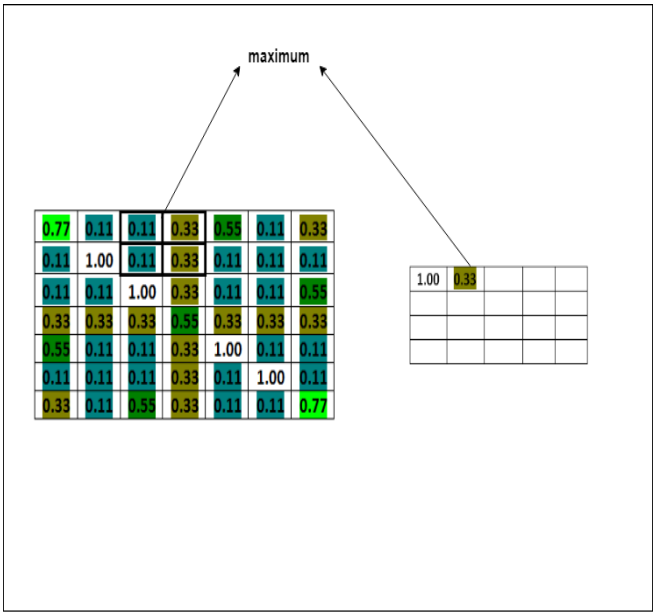


Fig 2. Pooling (Source: [7])

*C. ReLU layer*

Rectified Linear Unit (ReLU) is a computational unit which performs normalization. In this process, all the negative values in the filtered image are replaced by zeros. We perform this process with all other images in the stack, and at last, in the fully connected layer, we take our stack of images and form a single list so that it can become easier for visualization. Certain values in the list have more weightage when we feed certain images (as values represent the presence of features). Feature values in the list act as a list of votes. Initially, we may not get the correct features and correct values (votes) for the features. Hence we have to use back propagation to calculate the error (error = right answer - actual answer), and the error is propagated back to the previous layers. Thus, the feature values will get modified according to the error.

So the designer has to decide about: a number of features and size of features in convolution, window size and window stride in pooling, and a number of neurons in fully connected layers. In architecture, one has to make decisions about the number of each type of layer and the order of the layers. [6]

## IV. Study of RCNN

The process of object detection in RCNN algorithm is classified into the following two sub-processes:-

*A. Selective Search*

In order to locate region proposals, the selective search is used for object detection. The fundamental principle behind the working of selective search is that it performs clustering of similar regions based on their size, texture, color and shape. The initial step in this technique is to over-segment the image considering an important aspect which is the intensity of the pixels. The result expected from this step is the segmented regions. Taking into consideration these segmented regions, the algorithm works as follows:

1. Create bounding boxes corresponding to segmented parts to the list of proposed regions.
2. Group neighboring segments on the basis of similarity.
3. Perform Step 1.

So, the output of each iteration is segments which are smaller in size that adds up into larger segments which get added to the region proposals' list.

### B. SVM and Classification

After obtaining region proposals using selective search algorithm, they are combined into a single unit and fed to the convolutional neural network. Here, CNN is used as a feature extractor and the result generated is a 4096 dimension feature vector. The output is given to an SVM classifier which classifies the presence of objects contained in the proposed regions. An SVM classifier classifies the data points by separating them using a hyper plane. Hence, bounding boxes are created around these areas that can have a potential object. The precision of the bounding boxes drawn can be increased by predicting offset values. These offset values assist in manipulating the bounding boxes of region proposals (Fig. 3).

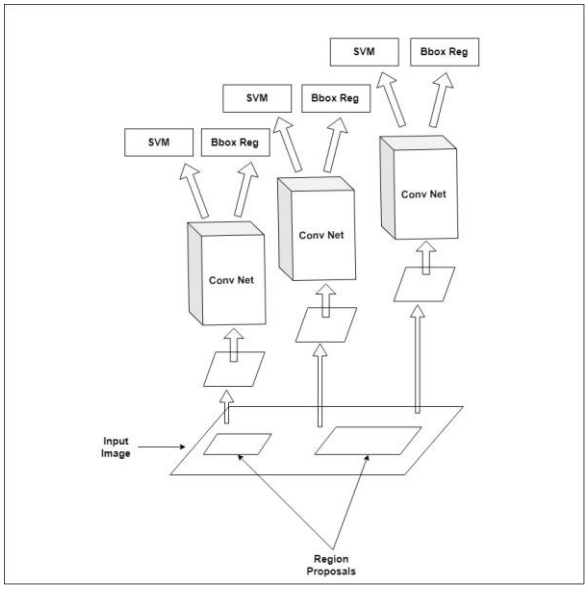


Fig 3. Architecture of RCNN (Source: [8])

## V. Study of Faster RCNN

Faster RCNN uses Fast RCNN and region proposal network as base.

### A. Fast RCNN

In RCNN, we have to send all the region of interest to the CNN which will result in excessive computational time.So, in Fast RCNN we overcome this problem by sending whole image through the CNN and get output as convolutional feature map. This feature maps will be used to generate region of interest. Selective search algorithm will be used for this purpose. Size of the bounding box will be scaled with the size of feature map. Fixed size small regions will be created, using ROI pooling as input to fully connected layers as they accept only fixed size inputs. Class of the proposed regions will be determined using Softmax layer. Softmax classifier gives us the probabilities for each class label. Bounding box regressors will be used to calculate offset values for bounding box. In faster RCNN we use different network for finding regions instead of selective search algorithm.

### B. Region proposal network

After we have found our feature map, instead of using selective search, we will use different network for region proposal. Here, we slide a window over convolutional feature. From each window, we generate k anchor boxes. Anchor boxes can be of different scales and aspect ratio (width of image / height of image). Anchor boxes are the boxes which may contain objects. Here, at this time, we are not concerned about the class of the object, but only the presence of object inside anchor box. From this anchor boxes we try to find the correct coordinate for bounding boxes. Hence, RPN has a classifier and regression layer. Classifier determines the probability (scores) of proposal having the target object and regression regresses the coordinates of proposal. After RPN, we get proposed reasons of different sizes. So to match this with CNN feature map, ROI pooling is done. After that, we feed them to fully connected layers. (Fig. 4).

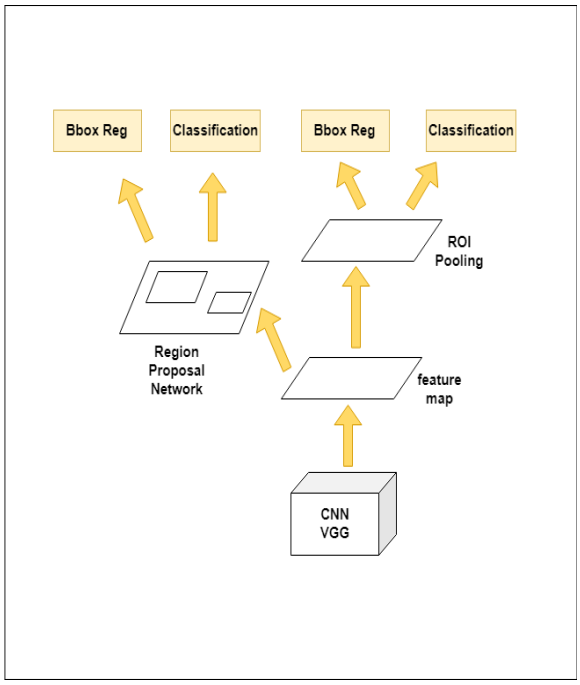


Fig 4. Architecture of Faster RCNN (Source: [4])

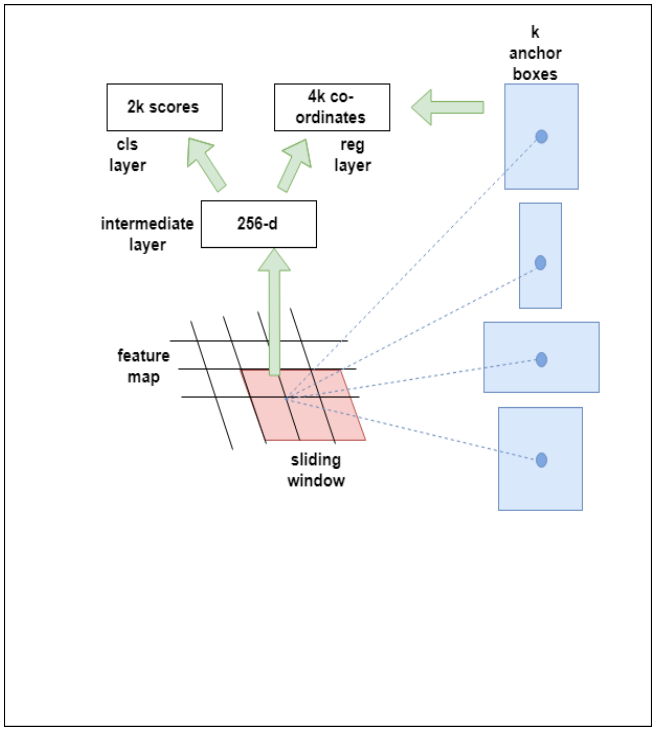


Fig 5. Region Proposal Network (Source: [4])

## VI. Study of Mask RCNN

Mask RCNN is a new type of neural network architecture which is an extension of Faster RCNN which adds a branch of segmentation mask to the latter one. [2] It works towards the problem of instance segmentation which in turn consists of two sub problems viz object detection and semantic segmentation. Mask RCNN consists of two major stages:-

### *A. Region Proposal Network*

Region Proposal Networks are used for the purpose of object detection. The main goal is to draw bounding boxes around the regions which contain an object. Another important task that takes place is ROI Align. ROI Align is the betterment of ROI pooling. Sometimes, data is lost when ROI pooling is used. The accuracy of the model is improved because of ROI Align.

### *B. Fully Convolutional Network*

Fully Convolutional Networks are used for the purpose of semantic segmentation. The network is trained such that it perfectly maps each pixel of the input image to the particular class it belongs to. One thing about FCN is that it does not mark individual instances of a class separately.[2] For example, if an image contains four cats then, the network would mark the region which contains the cats rather than separately specifying the four cats.

Hence, the Mask RCNN model performs two tasks:-

1. Performing object detection in order to obtain instances of objects.
2. Performing semantic segmentation in order to obtain the area in which the object is present.

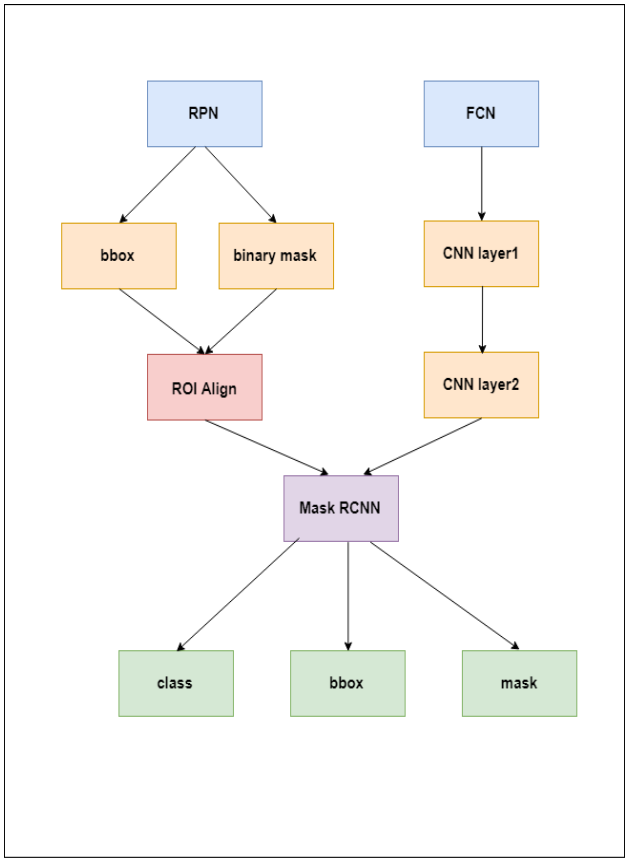


Fig 6. The architecture of Mask RCNN (Source: [9])

## VII. Conclusion

CNN is the base for object detection and image classification. CNN now outperforms humans on ImageNet challenge. RCNN was the first application of CNN for object detection. A typical CNN can tell us about the class of the object in an image. But in RCNN, CNN was forced to focus on a particular region as only a single object will dominate in that region. Thus after detecting regions by selective search algorithm, this resized regions was fed to CNN for further process. Fast RCNN was invented to speed up and simplify RCNN. In CNN, we had three different models for image feature extraction, classification and regression. But in Fast RCNN we used a single network to compute all three. Fast RCNN also used selective search algorithm for region detection which was further improved by Faster RCNN. [4] It used a different network for region proposal. Using the same feature map generated by CNN, it was able to generate region proposal. So for this, only one CNN was needed to be trained. Mask R CNN further extended Faster RCNN for pixel level segmentation. It added a branch to Faster R CNN which gave binary mask (says whether or not a given pixel is a part of an object) as output. [2] In slum identification, to locate and identify slum area in an image, we need to go for pixel level segmentation. Thus Mask R CNN is the best way to identify the slums.

## REFERENCES

[1] Kuffer, M., Pfeffer, K., & Sliuzas, R. (2016). Slums from space-15 years of slum mapping using remote sensing. Remote Sensing, 8(6). http://doi.org/10.3390/rs8060455
[2] He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2017). Mask R-CNN. Proceedings of the IEEE International Conference on Computer Vision, 2017–October, 2980–2988. https://doi.org/10.1109/ICCV.2017.322
[3] Durand-Lasserve, A., & Royston, L. eds. (2002). Holding Their Ground: Secure Land Tenure for the Urban Poor in Developing Countries Risbud, N. , Holding Their Ground: Secure Land Tenure for the Urban Poor in Developing Countries.
[4] Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031
[5] Cheng, B., Wei, Y., Shi, H., Feris, R., Xiong, J., & Huang, T. (2018). Revisiting RCNN: On awakening the classification power of faster RCNN. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11219 LNCS, 473–490. https://doi.org/10.1007/978-3-030-01267-0_28
[6] Xu, Z., Jiang, Z., & Li, Y. (2017). Poverty Prediction by Selected Remote Sensing CNN Features Final Report. Retrieved from www.aaai.org
[7] Brandon Rohrer, How CNNs work, Youtube video
[8] Gandhi, R R-CNN, Fast R-CNN, Faster R-CNN, YOLO – Object Detection Algorithms, Retrieved from https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e
[9] Zhang, X., Simple understanding of Mask RCNN, Retrieved from: https://medium.com/@alittlepain833/simple-understanding-of-mask-rcnn-134b5b330e95