Object recognition using Convolutional Neural Networks (CNN)

Annapurna Bala
HOD, Dept. Of Computer Science,
Ch.S.D.St Theresas Autonomous College for Women
Eluru, Andhra Pradesh, India.

Abstract- An Object classification problem is predicting the label of an image among the predefined labels. It assumes that there is single object of interest in the image and it covers a significant portion of image. Detection is about not only finding the class of object but also localizing the extent of an object in the image. The object can be lying anywhere in the image and can be of any scale. So object classification is no more helpful when there are Multiple objects in image, Objects are small and Exact location and size of object in image is desired. Traditional methods of detection involved using a block-wise orientation histogram(SIFT or HOG) feature which could not achieve high accuracy in standard datasets such as PASCAL VOC. These methods encode a very low level characteristics of the objects and therefore are not able to distinguish well among the different labels. Deep learning i.e. Convolutional networks based methods have become the state of the art in object detection in image. They construct a representation in a hierarchical manner with increasing order of abstraction from lower to higher levels of neural network. One could perform detection by carrying out a classification on different sub-windows or patches or regions extracted from the image. The patch with high probability will 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. Any Deep Neural Network will consist of three types of layers namely The Input Layer, The Hidden Layer and The Output Layer. We can say Deep Learning is the newest term in the field of Machine Learning. It's a way to implement Machine Learning.

Deep Learning is discovered and proves to have the best techniques with state-of-the-art performances. Thus, Deep Learning is surprising us and will continue to do so in the near future. Recently, researchers are continuous in exploring Machine Learning and Deep Learning.

Keywords- Convolutional Neural Network Input Matrix Convolutional Layer Convolution Layer Processing Neural Network.

I. INTRODUCTION

When the data is small, Deep Learning algorithms don't perform well. This is the only reason DL algorithms need a large amount of data to understand it perfectly. Deep Learning depends on high-end machines while traditional learning depends on low-end machines. Thus, Deep Learning requirement includes GPUs. That is an integral part of it's working. They also do a large amount of matrix multiplication operations. Feature Engineering is a general process. here, domain knowledge is put into the creation of feature extractors to reduce the complexity of the data and make patterns more visible to learn the algorithm working. Although, it's very difficult to process. Hence, it's time consuming and expertise. DL algorithms needs to break a problem into different parts to solve them individually. And to get a result, combine them all.

II. Convolutional Neural Networks (CNN) Approach

With IoT converting more and more offline objects into online assets, homes, transports, manufacturing units, agriculture fields, aquaculture bodies have become more vulnerable to e-Threats. Today, not just our smartphone and desktop but everything needs protection, and the businesses that are on the cusp of adopting and deploying technologies need more strategic protection. The technology advisory and consulting firms like Gartner, are quite optimistic about the prospects of using blockchain with the Internet of Things. The nodes described in the blockchain are analogous to the objects that are connected in the IoT ecosystem. Blockchain-based IoT will make our connected home, connected cars and everything based on the similar concept more reliable and secure. Leveraging AI, businesses can make the IoT more extensive in its approach to data analysis and dig out insights. BI Intelligence is helpful in reading the market trend and understanding the consumer's requirements better.

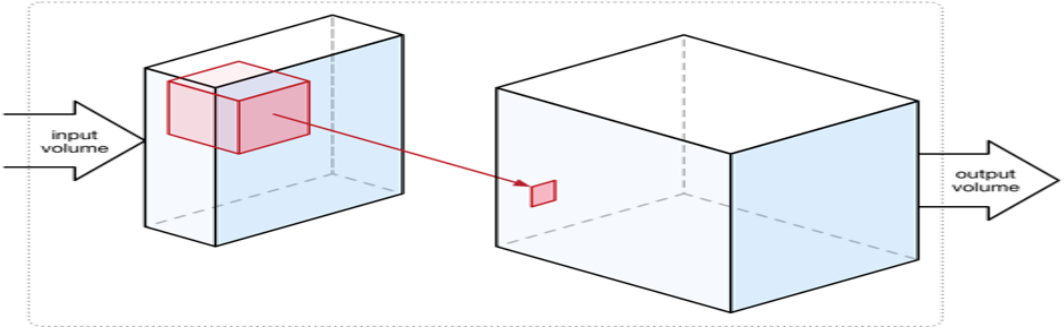
Let us suppose we have a task of multiple object detection. In this task, we have to identify what the object is and where is it present in the image. Deep Learning takes more time to train as compared to Machine Learning. The main reason is that there are so many parameters in a Deep Learning algorithm.

This will be a tedious process from computational time point of view as each sub-window would require passing it through CNN and calculating the feature for that region. R-CNN therefore uses a object proposal algorithm like selective search in its pipeline which gives out a number(~2000) of TENTATIVE object locations and extents on the basis of local cues like color rgb, hsv etc. This does not use any fancy supervised algorithm and therefore is class-agnostic which can be used independent of the domain. These object regions are warped to a fixed sized(227X227) pixels and are fed to a classification convolutional network which gives the individual probability of the region belonging to background and classes. The tricky part is feeding the appropriate regions labelled as background during the training of convolutional

network. If random regions that do not have anything to do with the object classes are fed as background the network wont be able to distinguish between the object regions and regions which are partially containing the objects. Therefore regions which have an IOU greater than 0.5 with the objects are marked with class of that object and those with overlap < 0.3 are marked as background. As in the classification training, SGD is used to train the network end to end. The second stage of RCNN involves improving the localization(coordinates of the extent of object) accuracy by minimizing the error of predicted coordinates against the ground truth coordinates. This is required because SS need not necessarily produces regions which can encompass the objects perfectly. For this a linear regression layer is optimized on top of Conv5 layer after fine-tuning the network for classification. Since this training is done independent of classification training, conv5 layer and layers preceding it cannot be fine-tuned once their weights have been optimized for classification(because we will be using the same network for both to reduce computation time). In the test time an additional technique Non Maximal Suppression is used to merge highly overlapping regions which are predicted to be of same class.

III. CONVOLUTIONAL LAYERS

The real power of these deep learning networks, especially for image recognition, comes from convolutional layers.

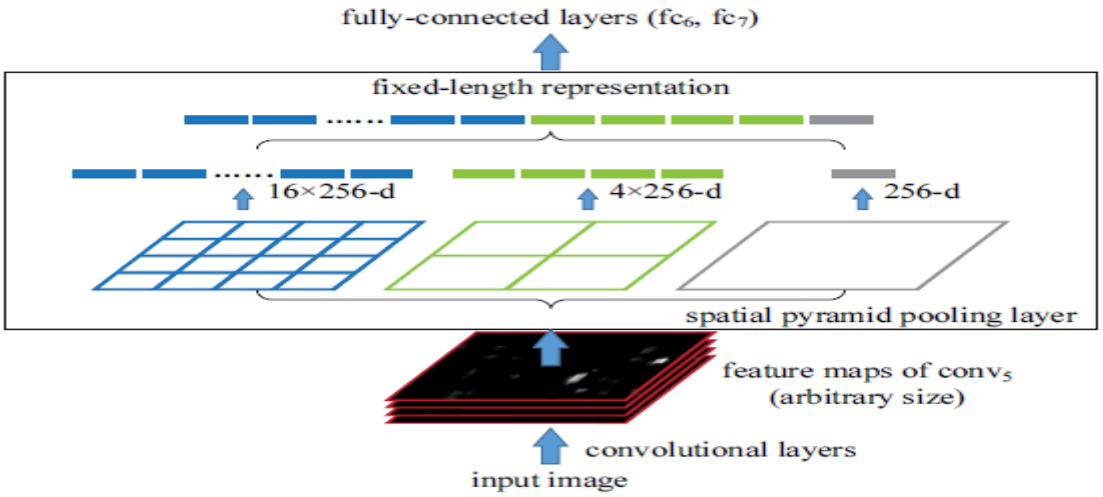


IV. EXPERIMENTAL ANALYSIS AND RESULTS

With a fully-connected layer, the input and output are one-dimensional vectors of numbers. However, a convolutional layer works on three-dimensional volumes of data (also called tensors). The output usually has the same width and height as the input volume but a larger depth. You can think of an image as a three dimensional "cube". The width and height are as usual, but each of the RGB components gets its own plane. Convolutional layers are great for image recognition because they scan the input much like a human eye does. Just as with the fully-connected layer, what we learn is weights — but here these weights are inside the convolution kernels.

A. Region based convolution neural networks (R-CNNS) approach:

Spatial Pyramid Pooling is one of the turning points for making a highly accurate RCNN pipeline feasible in run-time. RCNN had to pass all the ~2000 regions from SS independently through CNN and is therefore a very slow algorithm. Spatial Pyramid Pooling allows the whole image to be passed through the convolutional layer only once. This saves a lot of time because same patch may belong to multiple regions and convolutions on them are not calculated multiple time as done in RCNN thereby enabling a shared computation of convolution layers among the regions. Since major chunk of time(~90%) is spent on the convolutional layers it reduces the computation time drastically. After passing the image through convolution layers, independent feature need to be calculated for each of the regions generated by SS. This can be done by performing a pooling type of operation on JUST that section of the feature maps of last convolution layer that corresponds to the region. The rectangular section of convolution layer corresponding to a region can be calculated by projecting the region on convolution layer by taking into account the down sampling happening in the intermediate layers. Normally a max pooling layer follows the final convolution layer of CNN, but the feature vector produced by max pool layer depends upon the size of region and therefore vector obtained cannot be fed into the following FC layers which require a fixed sized vector as input. Spatial Pyramid Pooling solves this problem by replacing max pooling layer with spatial pooling layer. Spatial Pyramid Pooling layer divides a region of any arbitrary size into constant number of bins and max pool is performed on each of the bins. Since number of bins remain the same, a constant size vector is produced. The training windows for positive and negative samples are chosen in the same manner as RCNN. The only difference in SPP net is that since windows of arbitrary size are used for pooling operation, back-propagation through Spatial Pyramid Pooling layer and fine tuning the network end-to-end is not trivial and therefore SPP net fine tunes just FC part of the network and gradients are not propagated across the Spatial Pyramid Pooling layer. The second part follows on the same lines as RCNN wherein it fits a linear regression layer for localization on top of conv5 layers.

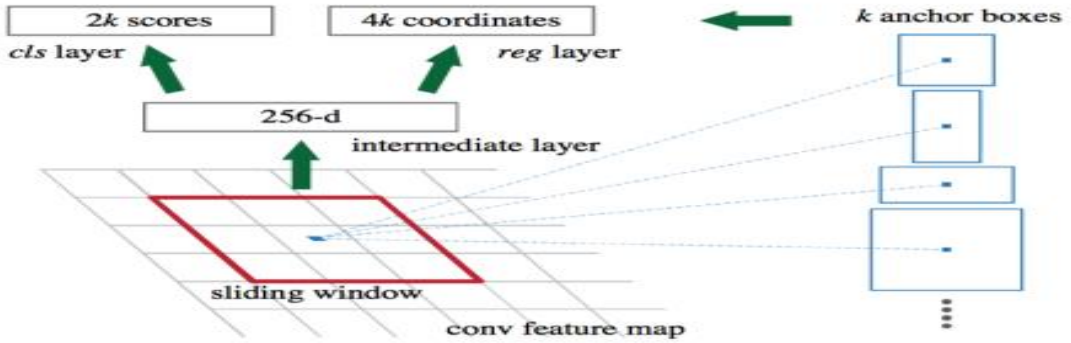


B. Limitations of Spatial Pyramid Pooling

Spatial Pyramid Pooling has certain limitations that it does not use the full potential of CNN because training is not end-to-end. Fast RCNN tackles the downsides by installing the net with the capacity to back-propagate the gradients from FC layer to convolution. layers. It is a simple back-propagation calculation and is very similar to max-pooling gradient calculation with the exception that pooling regions overlap and therefore a cell can have gradients pumping in from multiple regions. Second major change is a single network with two loss branches pertaining to soft-max classification and bounding box regression. This multitask objective is a salient feature of Fast-rcnn as it no longer requires training of the network independently for classification and localization. These two changes reduces the overall training time and increases the accuracy in comparison to Spatial Pyramid Pooling net because of end to end learning of CNN.

In the pipeline of fast-rcnn, the slowest part is generating regions from SS(~2s) or edge boxes(~0.2s). Faster-RCNN replaces SS with CNN itself for generating the region proposals(called RPN-region proposal network) which gives out tentative regions at almost negligible amount of time. This is done by using the convolutional layers from detection network and introducing two convolutional layers on top of this to generate regions at various spatial location. Since the convolution layers are shared it does not add to the computation time and the only additional time involved is the two additional convolution layers which have relatively small number of filters. So for RPN a small network with kernel size of 3X3 is run through the final convolution feature map and a smaller 256 dimension feature is obtained at every spatial location. This is then fed to the two sibling layers just as in previous detection network for the two tasks of classification and localization. Since the task of RPN is to generate potential object like regions classification layer has only two outputs for background and foreground. Its like sliding the window across the feature map. This small layer takes in fixed number(3X3) of pixels for making the prediction about the coordinates of objects which can be of any size or aspect ratio. And therefore different sets of parameters are used for different sizes and aspect ratio. In the following figure three sets of sizes and three sets of aspect ratio(square, vertically elongated and horizontally elongated) are used. These are called anchor boxes and are cantered at the sliding window location. So for these 9 anchor boxes(k), a total of 36(9k) is the number of outputs for regression layer encoding 4 coordinates for each anchor box and 18(2k) outputs for classification layer that gives an estimate of the probabilities of object or not object for each box. The anchor box having the highest overlap with the ground truth box is labelled as an object. Also the boxes having a high overlap percentage (>0.7) are also marked as foreground. All the anchor boxes with overlap percentage < 0.3 are labelled as background.

Training bit is a little tricky and different from previous methods because RPN and Detection network when trained independently would set different weights for convolution layers which would defeat the purpose of shared convolution layers for reducing computation time. Therefore authors proposed an alternating approach as follows. First RPN is trained and regions obtained are used to train detection network. Then RPN is retrained but this time with the convolution part initialized with weights from convolution part of detection network. In this run ,only the newly added convolution layers(not being used in detection network) are fine tuned for RPN and this makes the convolution layers to have the same weights as detection network. Finally the new RPN is used to generate the regions and fed to the training for detection network in which only the FC layers are fine tuned. This is a like alternating optimization used in optimization problem involving two variables when fixing one variable converts the problem into an easy to optimize function dependent on second variable.



V. CONCLUSION

All these methods concentrate on increasing the run-time efficiency of object detection without compromising on the accuracy. Faster-Rcnn has become a state-of-the-art technique which is being used in pipelines of many other computer vision tasks like captioning, video object detection, fine grained categorization etc.

VI. REFERENCES

[1]. Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik: Rich feature hierarchies for accurate object detection and semantic segmentation
[2]. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun: Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition
[3]. Ross Girshick: Fast R-CNN
[4]. Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun: Faster R-CNN: Towards Real- Time Object Detection with Region Proposal Networks http://www.cs.utoronto.ca/~fidler/slides/CSC420/lecture17.pdf
[5]. Matthijs Hollemans. Convolutional neural networks on the iPhone with VGGNet. http://matthijshollemans.com/2016/08/30/vggnet-convolutional-neural-network- iphone/, 2016.