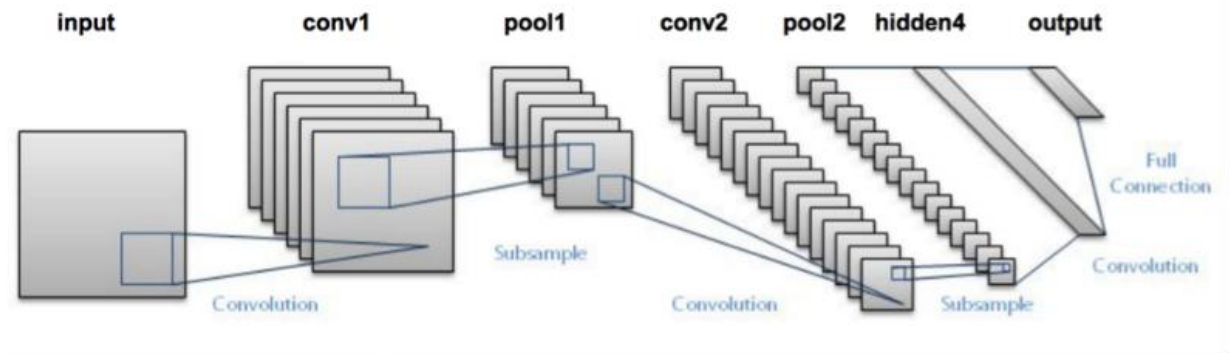
Algorithms Used For Object Detection

Brief working of a CNN (Convolutional Neural Networks) is as follows:



- We pass an image to the network,
- It is then sent through various convolutions and pooling layers,
- Finally, we get the output in the form of the object's class.

Different types of algorithms are as follows:

1. CNN - Convolutional Neural Network:

   a) First, we take an image as input:

b) Then we divide the image into various regions:

c) We will then consider each region as a separate image.

d) Pass all these regions (images) to the CNN and classify them into various classes.

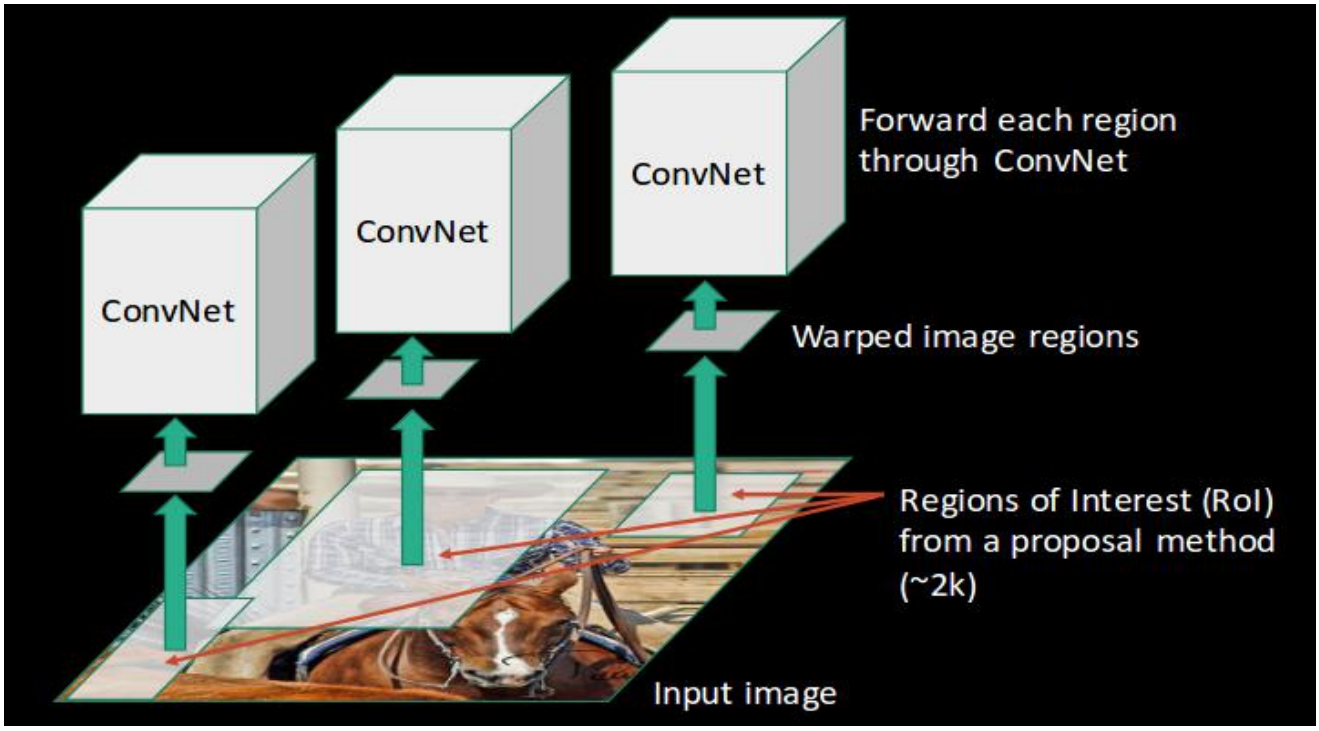
e) Once we have divided each region into its corresponding class, we can combine all these regions to get the original image with the detected objects:

2. RCNN - Region-based Convolutional Neural Network:

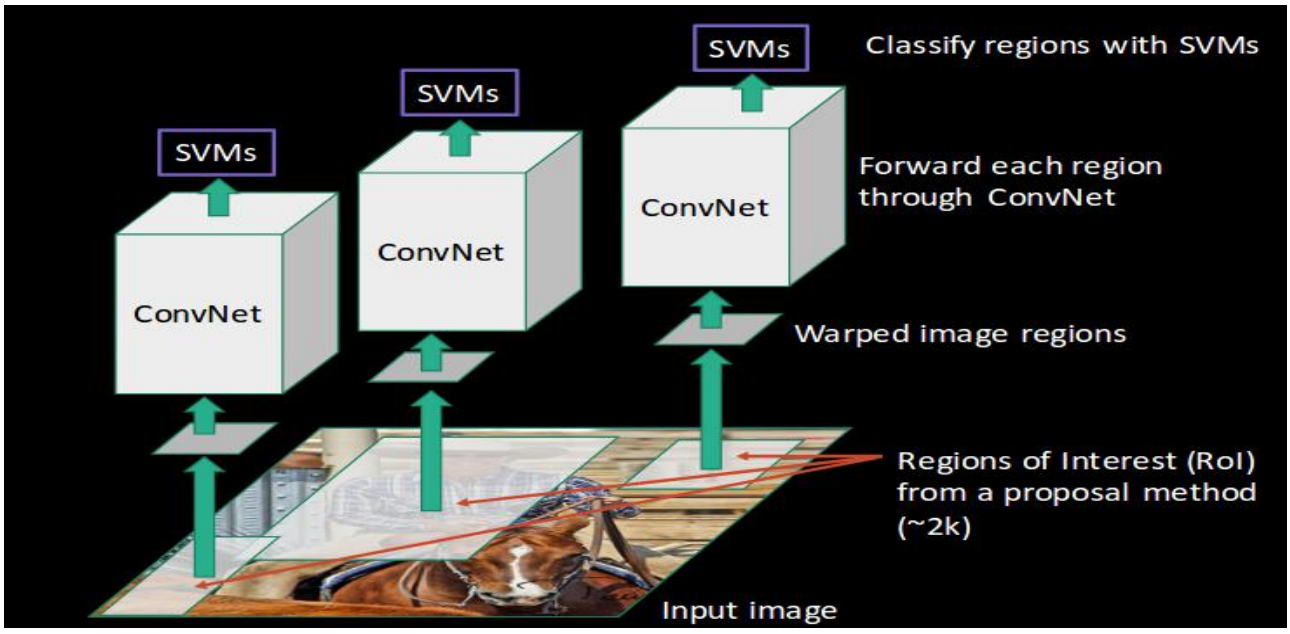
a) First, an image is taken as an input

b) Then, we get the Regions of Interest (ROI) using some proposal method (for example, selective search as seen above)

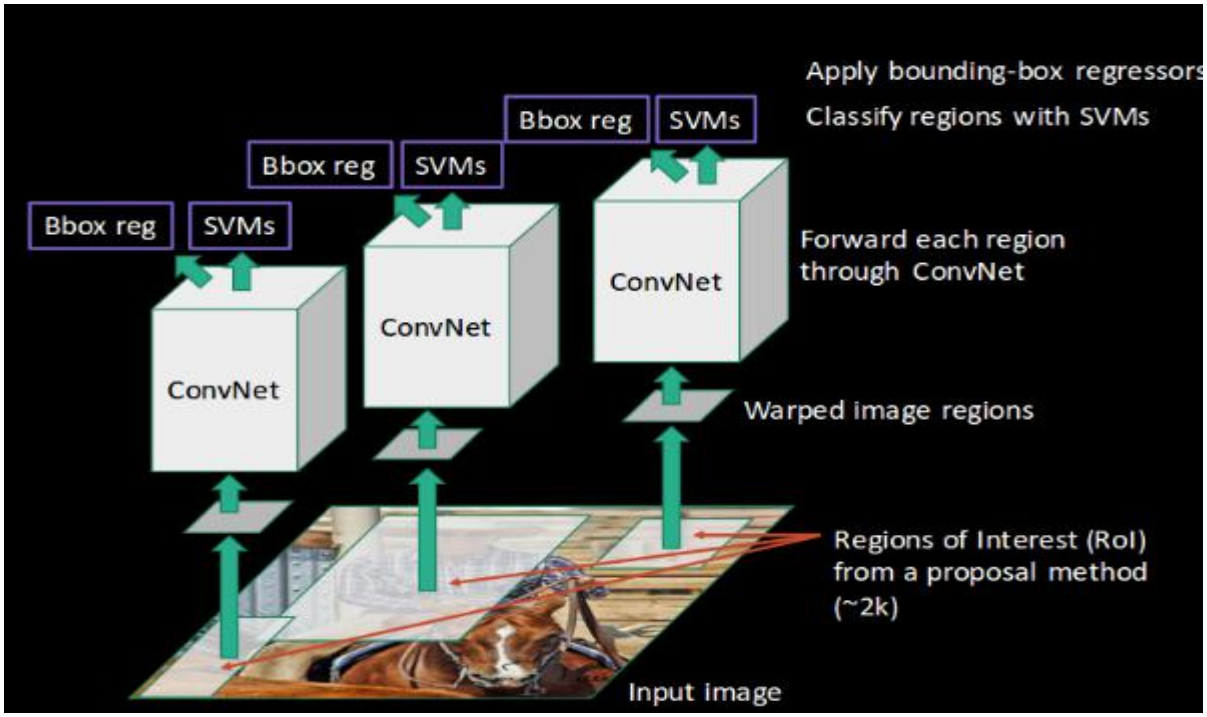
c) All these regions are then reshaped as per the input of the CNN, and each region is passed to the ConvNet:



d) CNN then extracts features for each region and SVMs are used to divide these regions into different classes



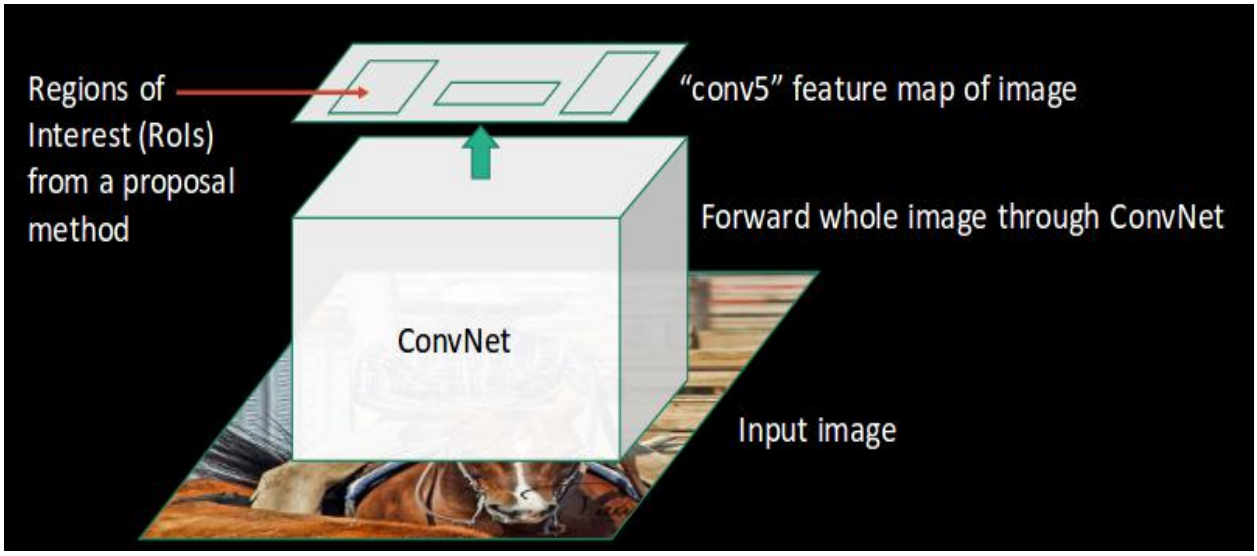
e) Finally, a bounding box regression (Bbox reg) is used to predict the bounding boxes for each identified region



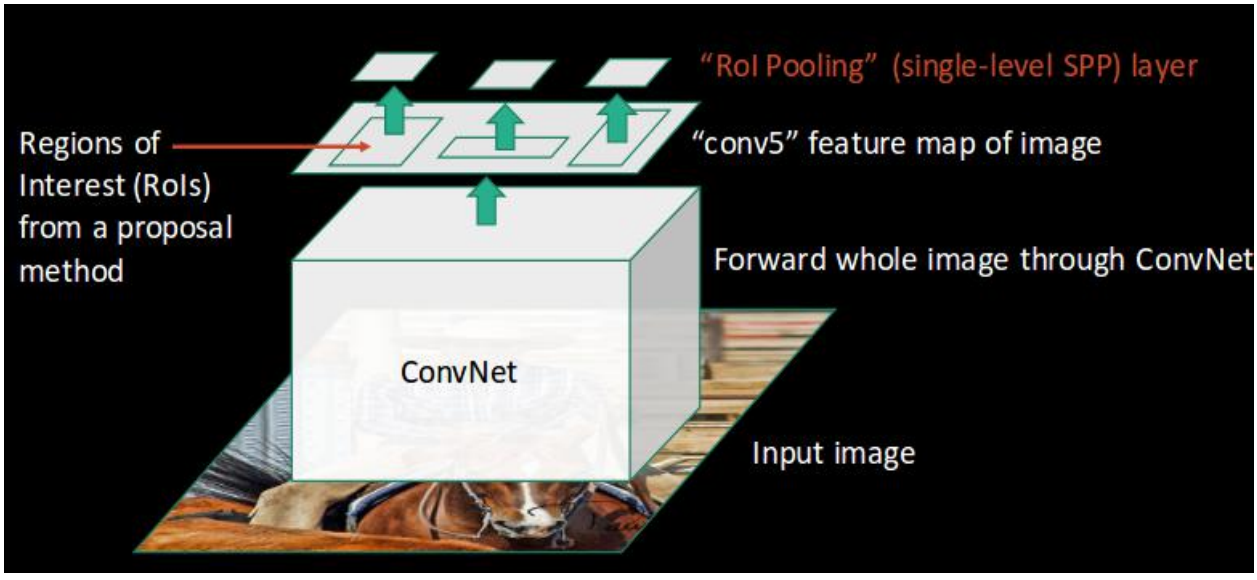
3. Fast RCNN:

a) We follow the now well-known step of taking an image as input

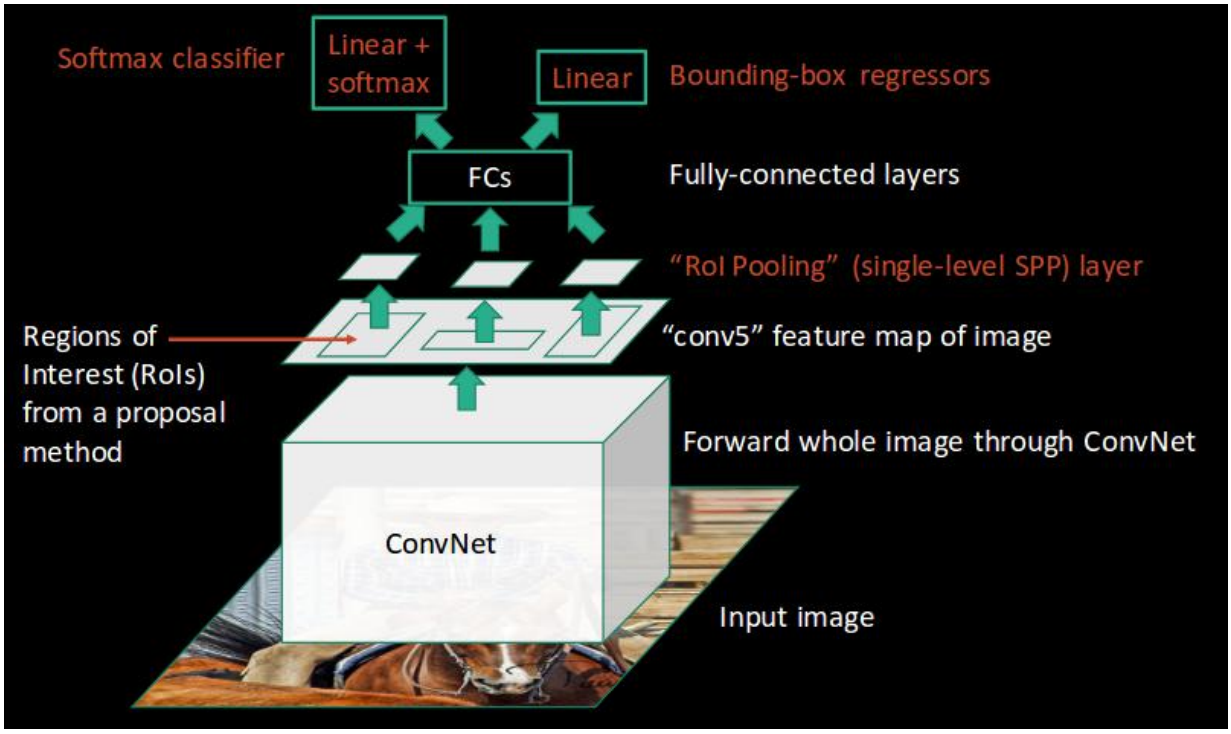
b) This image is passed to a ConvNet which returns the region of interests accordingly



c) Then we apply the RoI pooling layer on the extracted regions of interest to make sure all the regions are of the same size

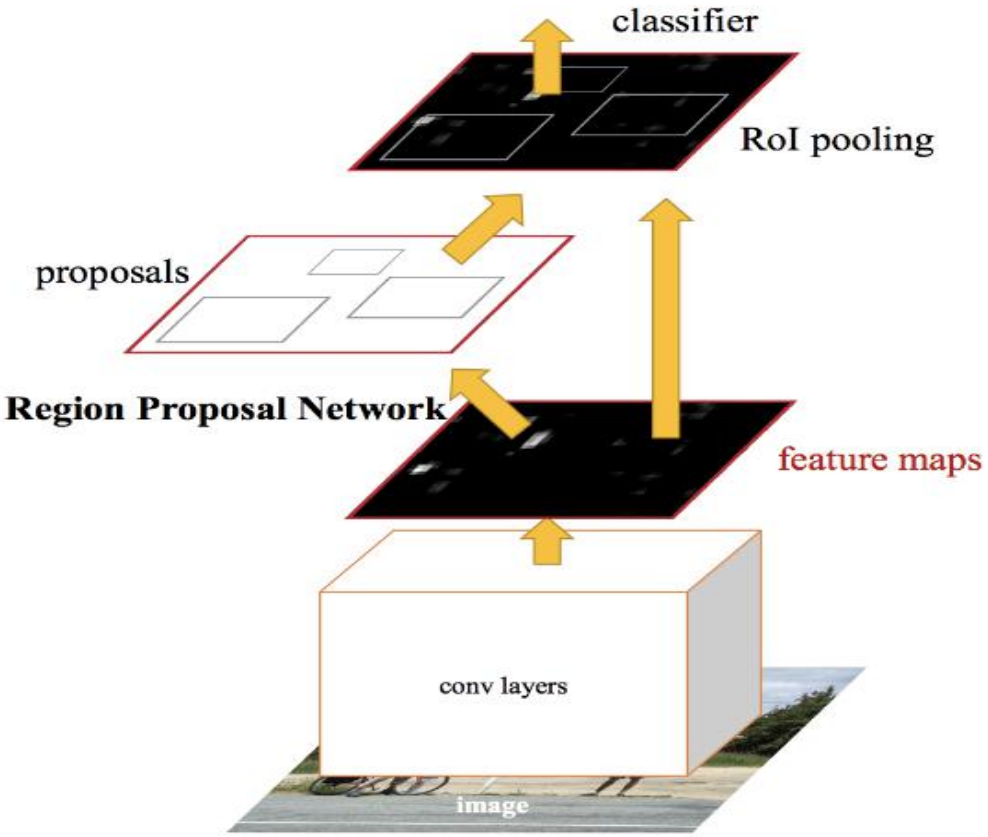


d) Finally, these regions are passed on to a fully connected network which classifies them, as well as returns the bounding boxes using softmax and linear regression layers simultaneously

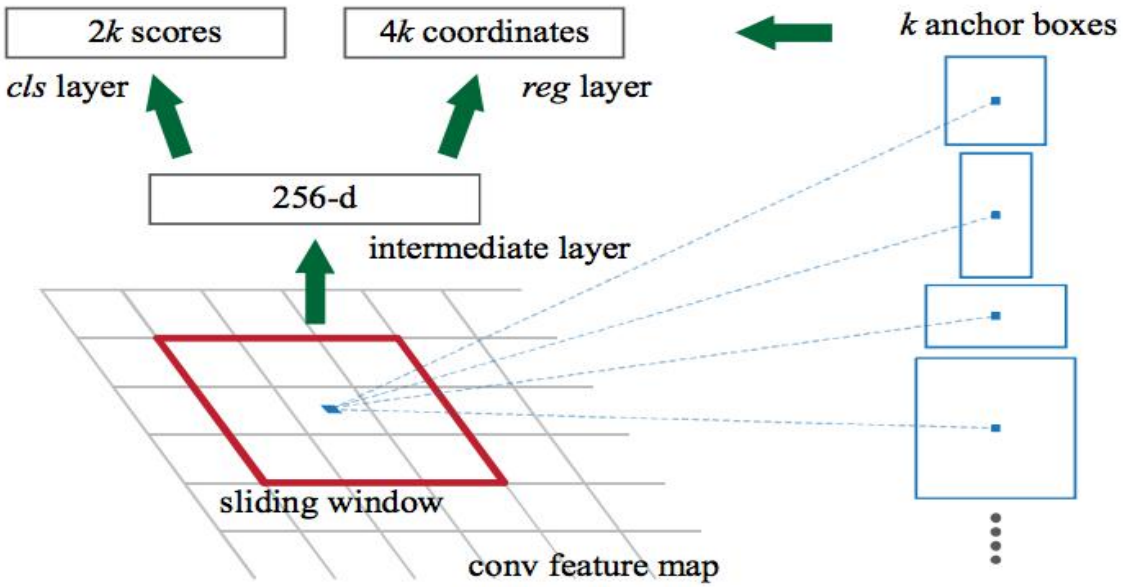


4. Faster RCNN:

a) We take an image as input and pass it to the ConvNet which returns the feature map for that image.
b) Region proposal network is applied on these feature maps. This returns the object proposals along with their objectness score.
c) A RoI pooling layer is applied on these proposals to bring down all the proposals to the same size.
d) Finally, the proposals are passed to a fully connected layer which has a softmax layer and a linear regression layer at its top, to classify and output the bounding boxes for objects.



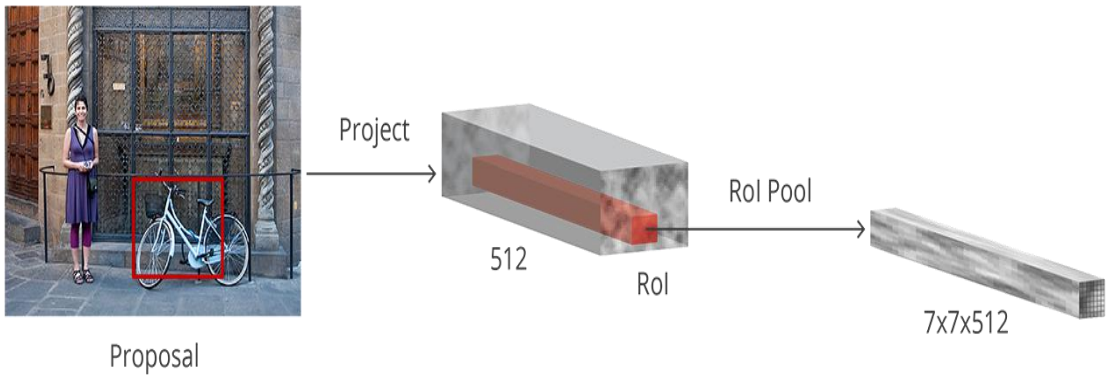
Faster RCNN takes the feature maps from CNN and passes them on to the Region Proposal Network. RPN uses a sliding window over these feature maps, and at each window, it generates k Anchor boxes of different shapes and sizes:



Anchor boxes are fixed sized boundary boxes that are placed throughout the image and have different shapes and sizes. For each anchor, RPN predicts two things:

- The first is the probability that an anchor is an object (it does not consider which class the object belongs to)
- Second is the bounding box regressor for adjusting the anchors to better fit the object

We now have bounding boxes of different shapes and sizes which are passed on to the RoI pooling layer. Now it might be possible that after the RPN step, there are proposals with no classes assigned to them. We can take each proposal and crop it so that each proposal contains an object. This is what the RoI pooling layer does. It extracts fixed sized feature maps for each anchor



Then these feature maps are passed to a fully connected layer which has a softmax and a linear regression layer. It finally classifies the object and predicts the bounding boxes for the identified objects.

Comparison of the above algorithms:

Algorithm	Features	Prediction time / image	Limitations
CNN	Divides the image into multiple regions and then classify each region into various classes.	–	Needs a lot of regions to predict accurately and hence high computation time.
RCNN	Uses selective search to generate regions. Extracts around 2000 regions from each image.	40-50 seconds	High computation time as each region is passed to the CNN separately also it uses three different model for making predictions.
Fast RCNN	Each image is passed only once to the CNN and feature maps are extracted. Selective search is used on these maps to generate predictions. Combines all the three models used in RCNN together.	2 seconds	Selective search is slow and hence computation time is still high.
Faster RCNN	Replaces the selective search method with region proposal network which made the algorithm much faster.	0.2 seconds	Object proposal takes time and as there are different systems working one after the other, the performance of systems depends on how the previous system has performed.