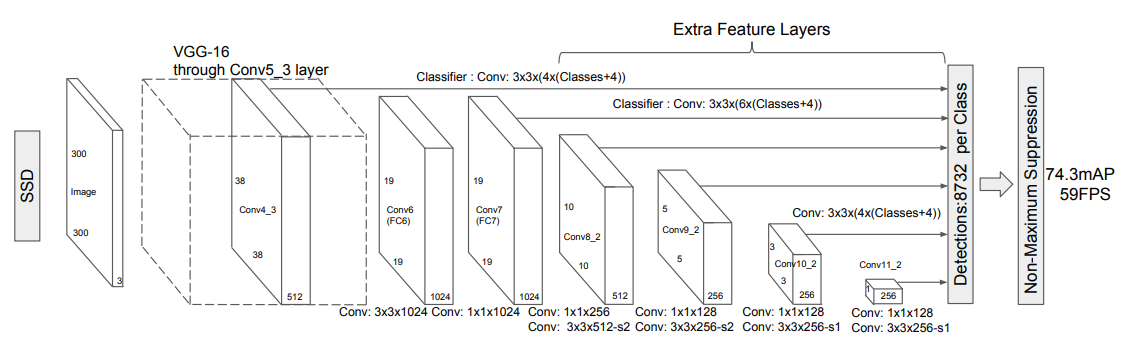
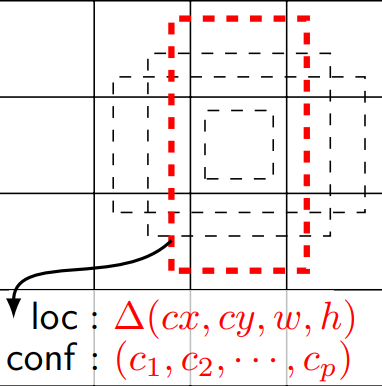
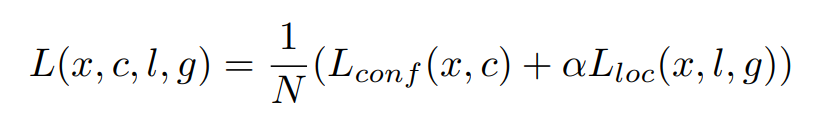
Single Shot Detector:

When we use SSD we only need one **single  
shot to detect** multiple objects within the image.  
  
Our model is SSD with Mobile-Net backbone, which means that it’s an improvement of the SSD architecture or in other words the backbone of the network is for feature extraction that are to be used for prediction in the later stages of the network. So as a result we will explain on the SSD network and then on the Mobile-Net network and will see if it is an improvement over the default.

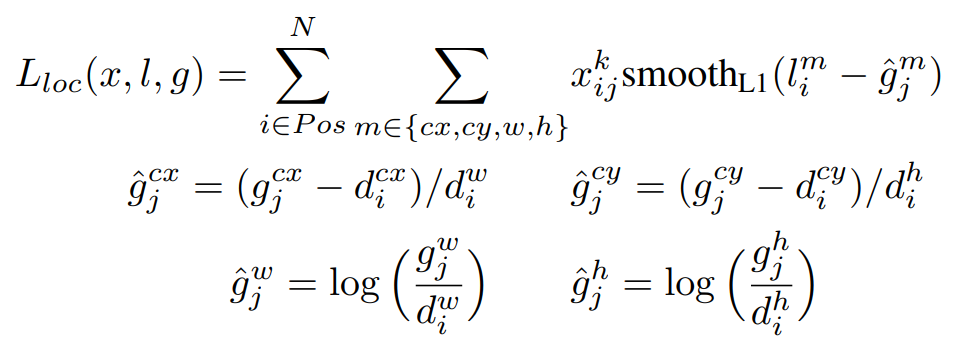


  
SSD architecture:   
the network is constructed by using the base layer (which the default is vgg-16 and ours will be Mobile-Net) to extract features and moving to a large array of smaller and smaller kernels of convoluted layers to detect objects.  
Each layer will construct 4 bounding boxes, so if the layer k is m x n there will be (classes + 4) kmn outputs.  
e.g. let’s take layer conv4\_3, k = 4, m, n = 38 and let’s assume there are 10 classes.  
Output: 38 x 38 x 4 x (10 + 4) = 80864  
the amount of bounding boxes = 38 x 38 x 4 = 5776

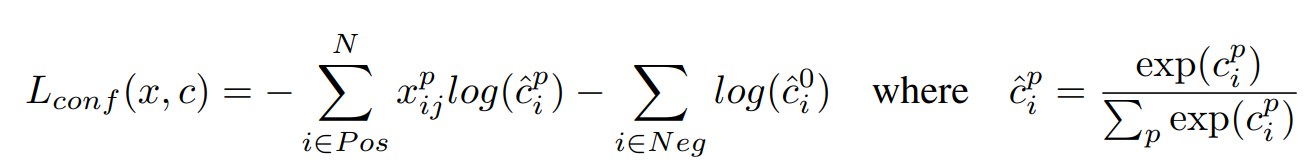
SSD Loss Function:



The localization loss is the mismatch between the ground truth box and the predicted boundary box. SSD only penalizes predictions from positive matches. We want the predictions from the positive matches to get closer to the ground truth. Negative matches can be ignored.



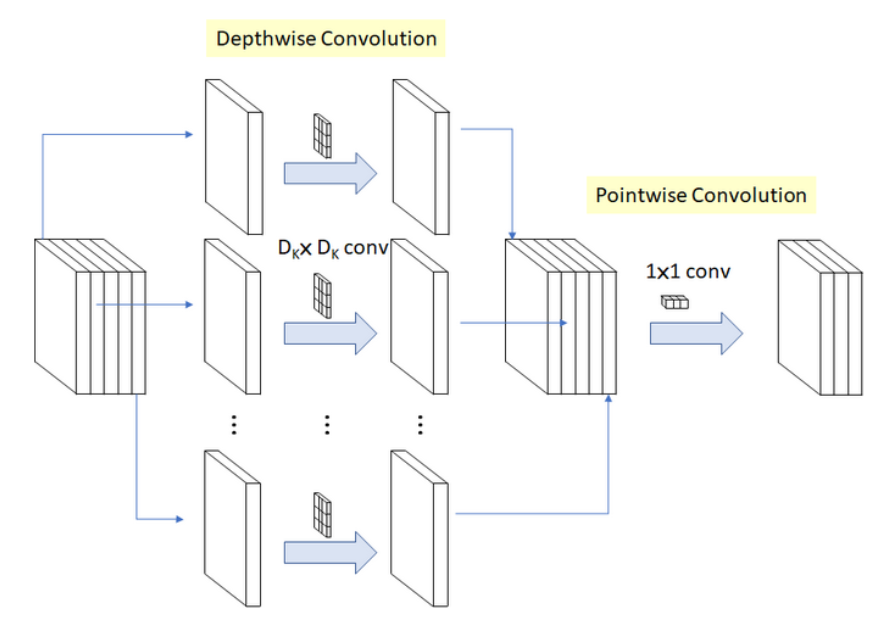
The confidence loss is the loss of making a class prediction. For every positive match prediction, we penalize the loss according to the confidence score of the corresponding class.

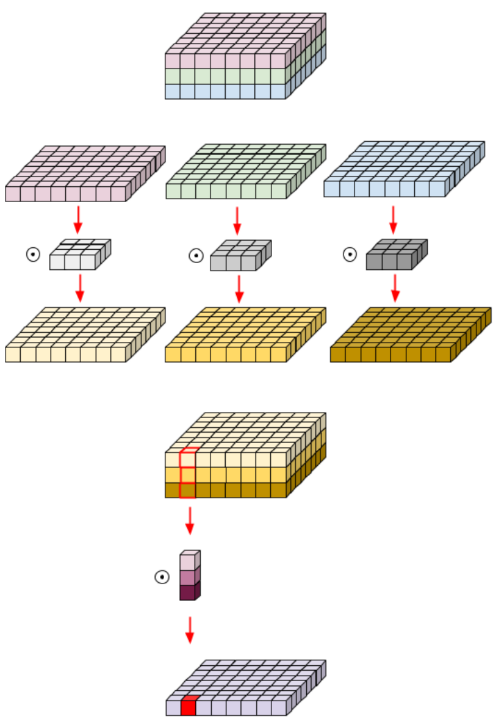


In this network the base layer can be change according to the need of more accurate results or less computationally heavy.

Mobile-Net:

As the name suggests this neural network is meant to be used for mobile devices and by so it is less computationally heavy but accurate none the less.



Mobile-Net architecture:  
the mobile net network uses **Depth-wise separable convolution.  
Depthwise Separable Convolution** splits the computation into two steps: depthwise convolution applies a single convolutional filter per each input channel and pointwise convolution is used to create a linear combination of the output of the depthwise convolution. The comparison of standard convolution and depthwise separable convolution is shown to the right. As we can see we first divide the plain to his layers and conduct a conv layer on each point and then combine each resulting plain and perform a 1 x 1 conv layer on each resulting plain point.

Mobile-Net also uses global hyperparameters to effectively reduce the computational cost further.

* Width Multiplier: Thinner Models:   
  For each layer, the width multiplier α will be multiplied with the input and the output channels (N and M) in order to narrow a network.  
    
  Here α will vary from 0 to 1, with typical values of [1, 0.75, 0.5 and 0.25]
* Resolution Multiplier: Reduced Representation:  
  For a given layer, the resolution multiplier ρ will be multiplied with the input feature map. we can express the computational cost by applying **width multiplier** and **resolution multiplier** as:  
  

Mobile-Net SSD:   
the network will be as follows:  
first the Mobile-Net as the base layer of the network and then the resulting inputs will be given to the later layers in the SSD network.  
  
