VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION

The VGG 16 paper starts with an intimation that highly accurate prediction results can be achieved by increasing the depth of layers in CNNs to 16-19 layers, which is further explained in this paper by showing the results in the ILSVRC 2014 Challenge.

First of all the steadily increasing depth was feasible because of the use of small sized filters i.e. 3×3 in all Conv Layers.

Architecture

Firstly, in the input image of 224×224, the mean RGB value was subtracted from each pixel. And as mentioned earlier, filters of small receptive field are used i.e. 3×3. In one of the configurations, they also used 1×1 filters which can actually be seen as a linear transformation of input channels. The Conv Stride is fixed to 1 and the spatial padding is set in such a way that the size of the input doesn't change. Five max pooling layers are used after Conv Layers, although it is also written that all the Conv Layers aren't followed by Max pooling layers. Max pooling is performed over a 2×2 pixel window, with stride 2.

The basic structure is composed of a stack of Conv Layers with different depths which is followed by 3 FC Layers. The first 2 FC layers contain 4096 channels, and the last layer contains 1k channels for 1k classes, as in

ILSVRC. Then follows the softmax layer, which provides the probaility scores.

All the hidden layers are followed by ReLu non linearity. They also concluded that the use of LRN (Local Response Normalization) hasn't showed much betterment in prediction, in the contrary, it decreased the accuract by 0.02%.

Configurations

There were 5 configurations of the same architecture which differ in depths (from 11 to 19 weight layers). The number of channels increases by a factor of 2 after every Conv Layer. It also says that the number of weights to be trained are not that much high in number as compared to shallower networks with larger Conv Layer widths and receptive fields.

Unlike Krizhevsky and Zeiler & Fergus,this configuration focuses more on depth rather than larger receptive fields of 11×11 or 7×7. The reasoning for using a 3×3 filter is given by saying that 2 3×3 Conv Layers stacked up provide and effective receptive field of one 5×5 filter. Another reason is that by using more small 3×3 filters they brought more ReLU activations in use, which made the model more discriminative. One more benefit was that it gave us lesser number of weights to train with relatively higher discriminative ability.

Then it is explained that they made use of 1×1 convolution essentially as a linear projection of the input in the same space (as they didn't change the number of channels), to increase the non linearity of the decision function by adding a ReLU unit after the convolution.

Training

It is said that the training procedure was same as Krizhevsky's,i.e., Multinomial Logistic Regression using Mini batch gradient descent with Momentum. The batch size was set to 256 and momentum coefficient to 0.9. L2 and Dropout Regularizations were used on the first 2 FC layers with penalty multiplier set to be $5/10^4$ and dropout ratio to be 0.5. The LR was set to be 0.01 which was further decreased by a factor of 10 when the val accuracy stopped improving. In total the LR was decreased 3 times and learning was stopped after 370K iterations i.e. 74 epochs.

A very good fact was given that for highly deep networks (16 or 19 layers), the convergence could be reached faster if the weights of the first Conv layers and the FC layers are pre initialised. The LR for pre initialised weights weren't decresed allowing them to change during learning. The biases were initialised to be 0. The randomly initialized weights were sampled from a normal distribution with 0 mean and 0.01 variance.

Some augmentations were done on the training images like cropping them to make them of the size 224×224. The cropped images further underwent horizontal flipping, and random RGB colour shift.

To tackle the varying input image size, two methods were used to rescale the input images viz. Single scale approach and Multi scale approach. In the Single Scale Approach, the smallest side of the image S, was set to be constant and the input image was first rescaled into an image with smallest side S and then a crop was taken from the image. In the Multi Scale Approach, two values Smin and Smax are set, and the input image is rescaled into an image with smallest side lying between the bound [Smin,Smax], and then a crop was taken from it. They trained multi scaled models by fine tuning all layers of a single scale model with same configuration, pre trained with S=384.

Testing

During testing, instead of using the FC layers, they used a different interpretation of the same, i.e. they converted the FC layers into 7×7 Conv Layers and 1×1 Conv layers (used as classifiers). The main reason to use this method was to make the model able to work on different sized inputs. If the input will be kf difderent sizes then also after the 7×7 conv, one can use GAP(Global Average Pooling) to get the 1×1×4096 output.

Implementation

Multiple GPUs were used to exploit the parallelism provided by GPUs and use them to parallelize the operations done on different batches of data. After completion the results are averaged to get the gradient of the full batch. They used Synchronous Gradient Computation.

Classification Experiments

After all this, it is shown that the model was evaluated on different basis, like for single scale, multi scale, multi crop, Conv Fusion. It is clearly shown that using deeper networks with small receptive fields give better results than shallow networks with larger receptive fields.

At last, the VGG was a really good approach for image recognition and Computer Vision task.