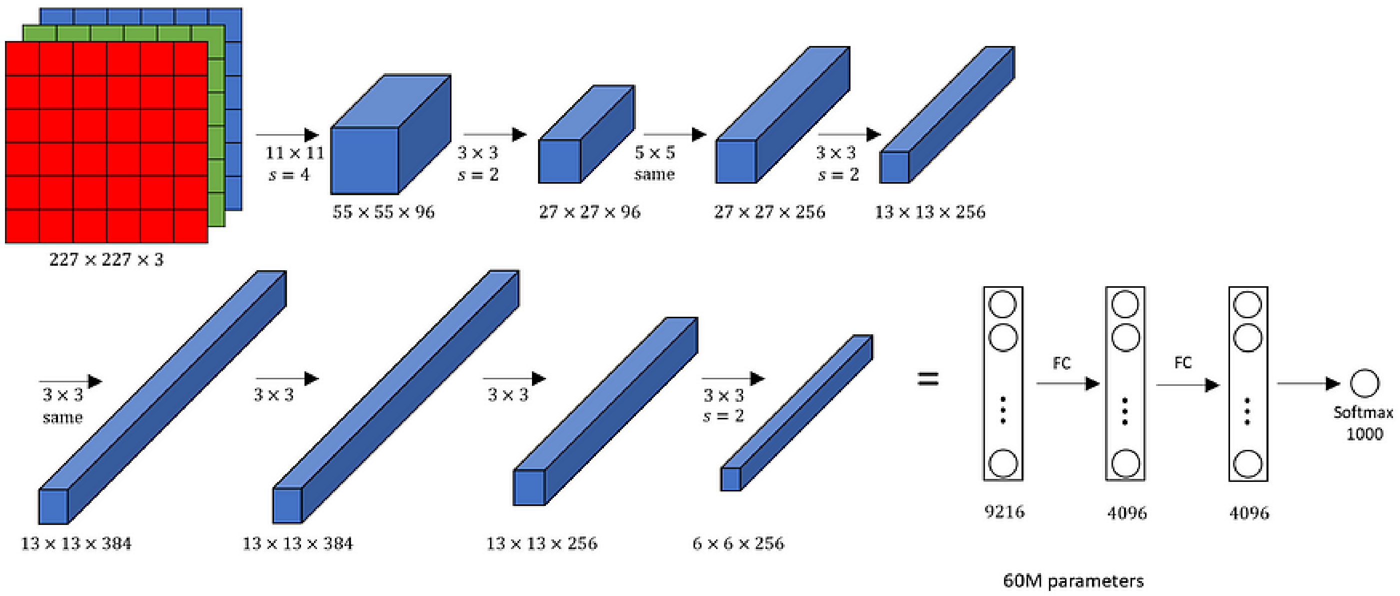
Module 5

AlexNet Architecture



AlexNet Architecture

This was the first architecture that used GPU to boost the training performance. AlexNet consists of 5 convolution layers, 3 max-pooling layers, 2 Normalized layers, 2 fully connected layers and 1 SoftMax layer. Each convolution layer consists of a convolution filter and a non-linear activation function called "ReLU". The pooling layers are used to perform the max-pooling function and the input size is fixed due to the presence of fully connected layers. The input size is mentioned at most of the places as 224x224x3 but due to some padding which happens it works out to be 227x227x3. Above all this AlexNet has over 60 million parameters.

Key Features:

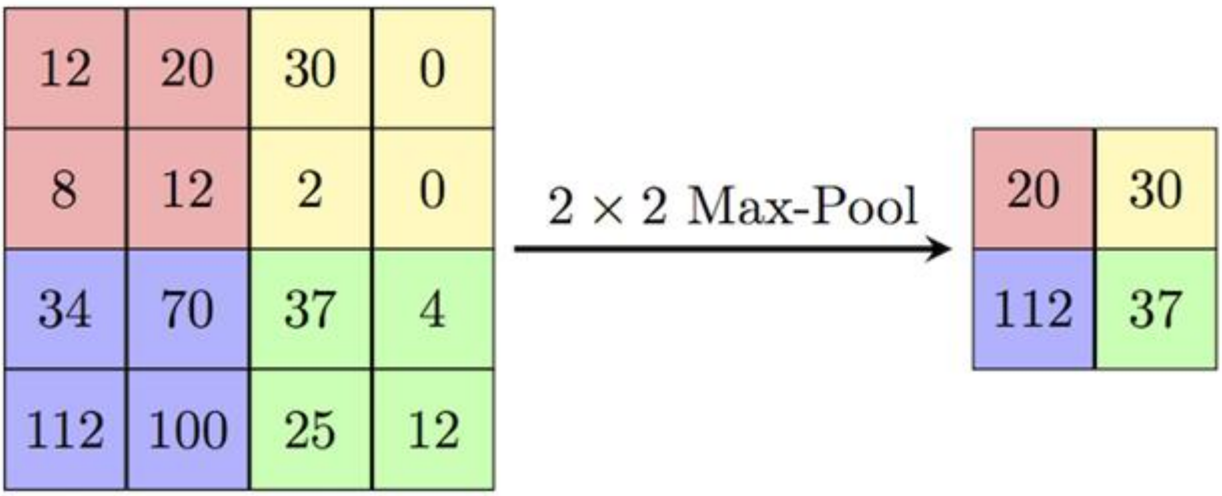
- 'ReLU' is used as an activation function rather than 'tanh'
- Batch size of 128
- SGD Momentum is used as a learning algorithm
- Data Augmentation is been carried out like flipping, jittering, cropping, colour normalization, etc.

Module 5

AlexNet was trained on a GTX 580 GPU with only 3 GB of memory which couldn't fit the entire network. So the network was split across 2 GPUs, with half of the neurons(feature maps) on each GPU.

Max Pooling

Max Pooling is a feature commonly imbibed into Convolutional Neural Network (CNN) architectures. The main idea behind a pooling layer is to "accumulate" features from maps generated by convolving a filter over an image. Formally, its function is to progressively reduce the spatial size of the representation to reduce the number of parameters and computations in the network. The most common form of pooling is max pooling.



Max Pooling

Max pooling is done in part to help over-fitting by providing an abstracted form of the representation. As well, it reduces the computational cost by reducing the number of parameters to learn and provides basic translation invariance to the internal representation. Max pooling is done by applying a max filter to (usually) non-overlapping sub-regions of the initial representation.

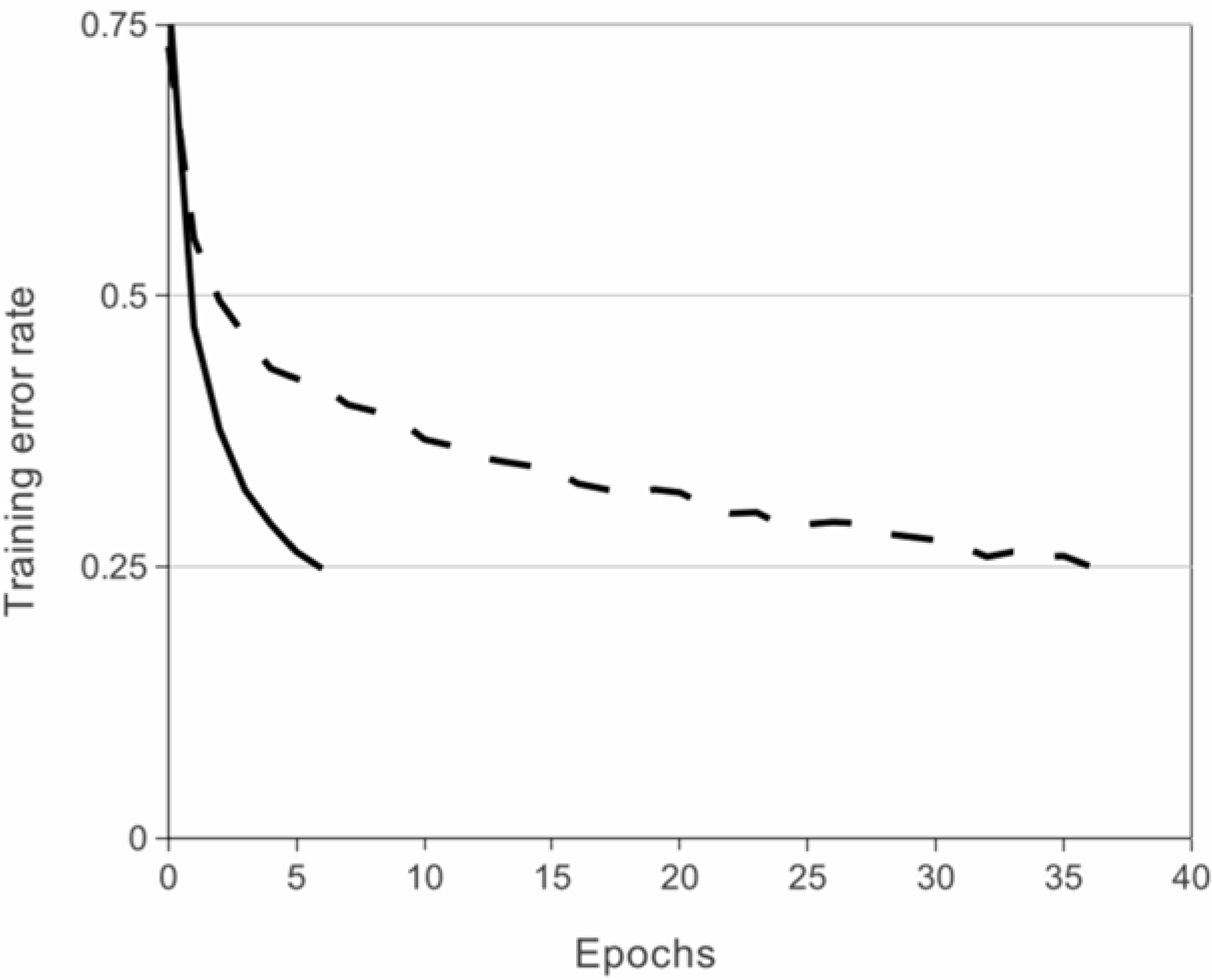
The authors of AlexNet used pooling windows, sized 3×3 with a stride of 2 between the adjacent windows. Due to this overlapping nature of Max Pool, the top-1 error rate was reduced by 0.4% and the top-5 error rate was reduced by 0.3% respectively. If you compare this to using non-overlapping pooling windows of size 2×2 with a stride of 2, that would give the same output dimensions.

ReLU Non-Linearity

AlexNet demonstrates that saturating activation functions like Tanh or Sigmoid can be used to train deep CNNs much more quickly. The image below demonstrates that AlexNet can achieve a training

Module 5

error rate of 25% with the aid of ReLUs (solid curve). Compared to a network using tanh, this is six times faster (dotted curve). On the CIFAR-10 dataset, this was evaluated.



ReLU

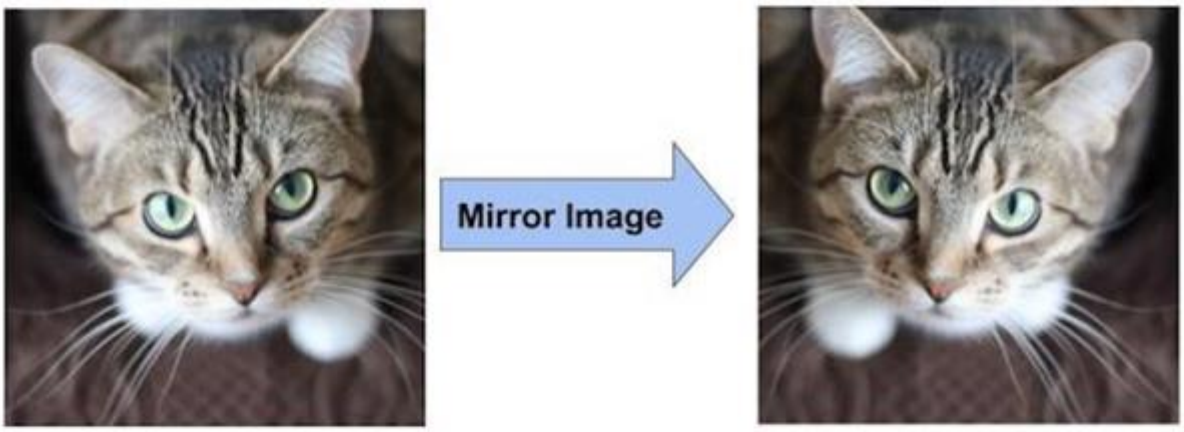
Data Augmentation

Overfitting can be avoided by showing Neural Net various iterations of the same image. Additionally, it assists in producing more data and compels the Neural Net to memorise the main qualities.

- **Augmentation by Mirroring**

Module 5

Consider that our training set contains a picture of a cat. A cat can also be seen as its mirror image. This indicates that by just flipping the image above the vertical axis, we may double the size of the training datasets.



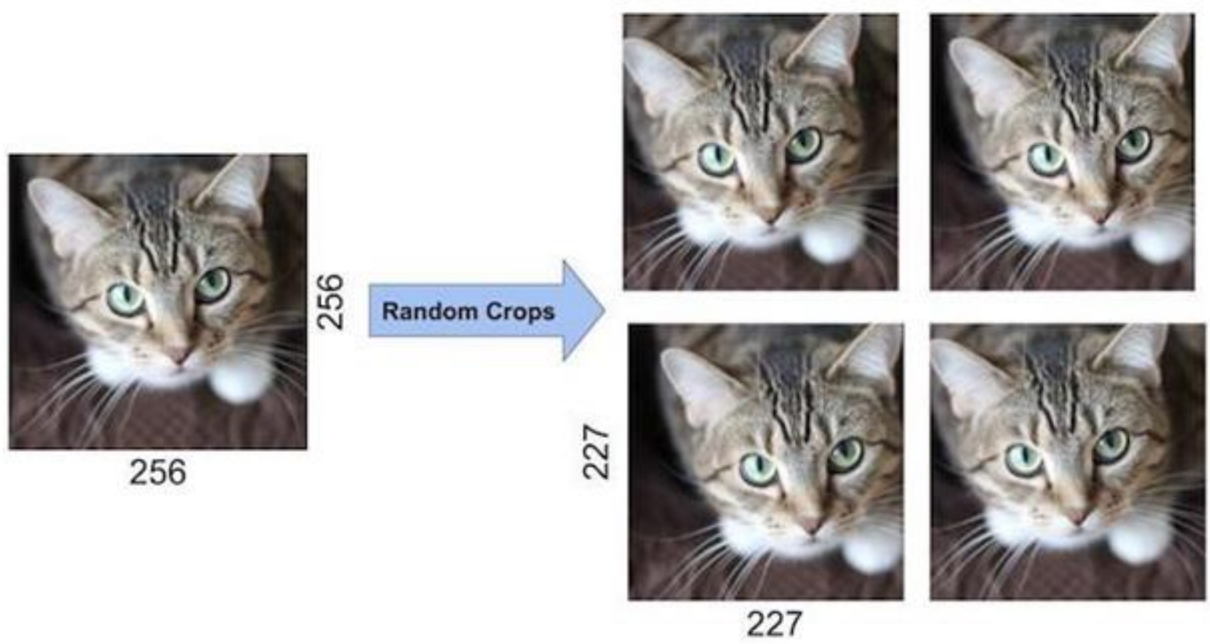
Data Augmentation by Mirroring

- **Augmentation by Random Cropping of Images**

Randomly cropping the original image will also produce additional data that is simply the original data shifted.

For the network's inputs, the creators of AlexNet selected random crops with dimensions of 227 by 227 from within the 256 by 256 image boundary. They multiplied the size of the data by 2048 using this technique.

Module 5

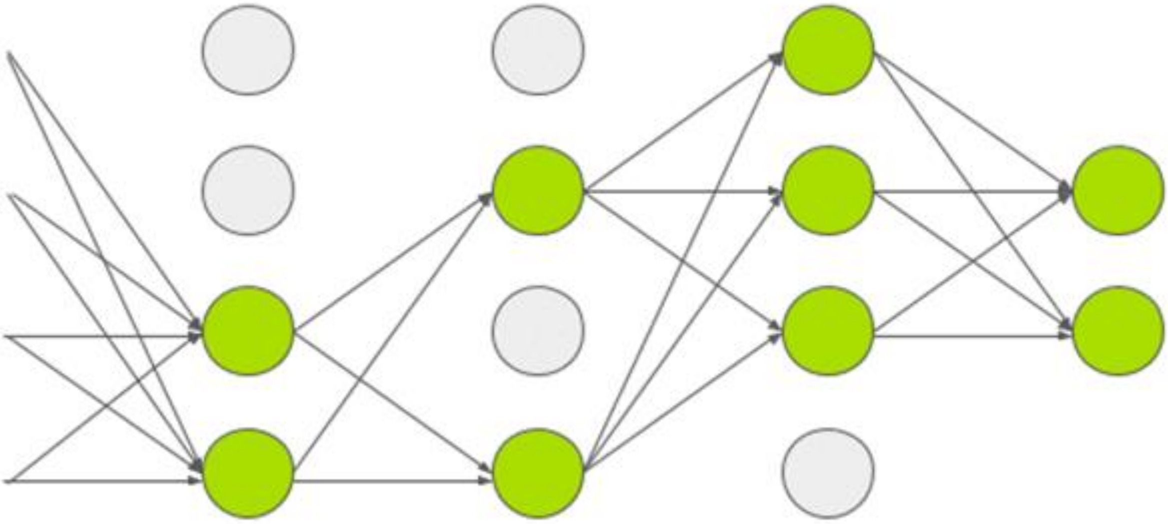


Data Augmentation by Random Cropping

Dropout

A neuron is removed from the neural network during dropout with a probability of 0.5. A neuron that is dropped does not make any contribution to either forward or backward propagation. As seen in the graphic below, each input is processed by a separate Neural Network design. The acquired weight parameters are therefore more reliable and less prone to overfitting.

Module 5



AlexNet Summary

Module 5

Layer		Feature Map	Size	Kernel Size	Stride	Activation
Input	Image	1	227x227x3	-	-	-
1	Convolution	96	55 x 55 x 96	11x11	4	relu
	Max Pooling	96	27 x 27 x 96	3x3	2	relu
2	Convolution	256	27 x 27 x 256	5x5	1	relu
	Max Pooling	256	13 x 13 x 256	3x3	2	relu
3	Convolution	384	13 x 13 x 384	3x3	1	relu
4	Convolution	384	13 x 13 x 384	3x3	1	relu
5	Convolution	256	13 x 13 x 256	3x3	1	relu
	Max Pooling	256	6 x 6 x 256	3x3	2	relu
6	FC	-	9216	-	-	relu
7	FC	-	4096	-	-	relu
8	FC	-	4096	-	-	relu
Output	FC	-	1000	-	-	Softmax