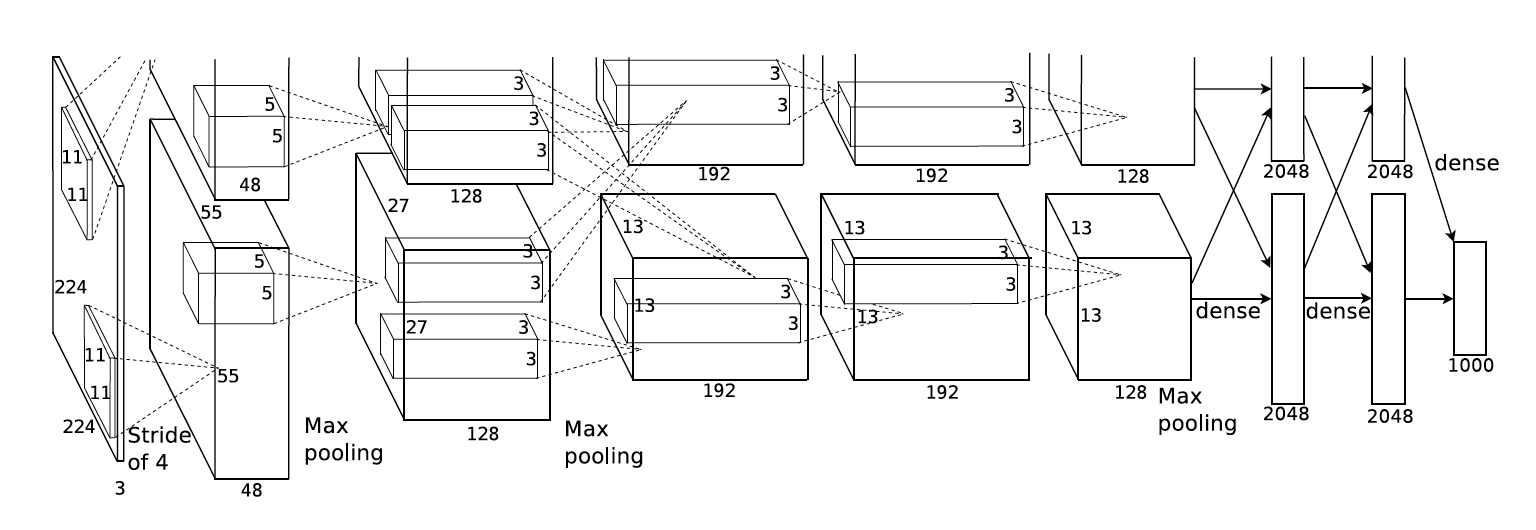
**AlexNet (2012)**



* Use Relu instead of Tanh to add non-linearity. It accelerates the speed by 6 times at the same accuracy.
* Use dropout instead of regularisation to deal with overfitting. However the training time is doubled with the dropout rate of 0.5.
* Overlap pooling to reduce the size of network. It reduces the top-1 and top-5 error rates by 0.4% and 0.3%, repectively.

https://airtable.com/shrArXKRCau4KhAwZ/tbloN5WYjFYpKGUEt?blocks=hide

It contains

* 5 convolutional layers
* 3 fully connected layers.
* Relu is applied after very convolutional and fully connected

layer.

* Dropout is applied before the first and the second fully connected year.
* The image size in the following architecutre chart should be 227 \* 227 instead of 224 \* 224, as it is pointed out by Andrei Karpathy in his famous CS231n Course.
* More insterestingly, the input size is 224 \* 224 with 2 padding in the pytorch torch vision. The output width and height should be (224–11+4)/4 + 1=55.25! The explanation here is pytorch Conv2d apply floor operator to the above result, and therefore the last one padding is ignored.

It consisted

11x11, 5x5,3x3 -convolutions

max pooling

Dropout

data augmentation

ReLU activations

SGD with momentum.

It attached ReLU activations after every convolutional and fully-connected layer.