**Project Report**

**On**

**Cats and Dogs Classification**

(CSE VI Semester Mini Project)



Submitted By: Rohan Singh Bisht

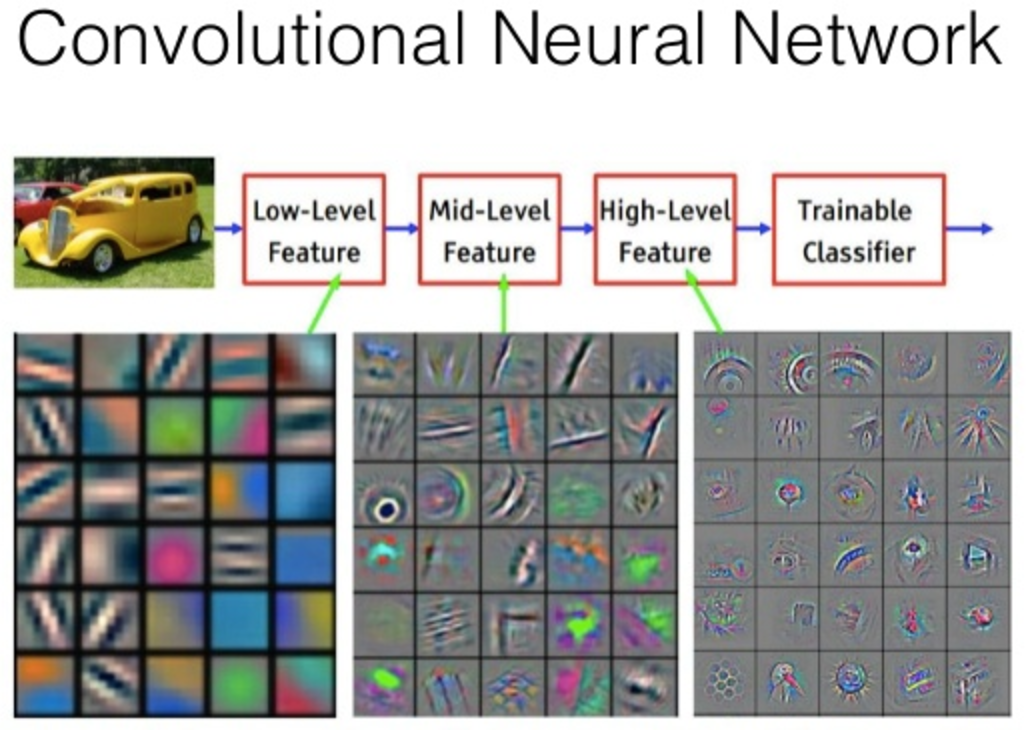
University roll no.: 2015187

Section: DS

Semester: VI

## Cat and dog classification using CNN

Convolutional Neural Network (CNN) is an algorithm taking an image as input then assigning weights and biases to all the aspects of an image and thus differentiates one from the other. Neural networks can be trained by using batches of images, each of them having a label to identify the real nature of the image (cat or dog here). A batch can contain few tenths to hundreds of images. For each and every image, the network prediction is compared with the corresponding existing label, and the distance between network prediction and the truth is evaluated for the whole batch. Then, the network parameters are modified to minimize the distance and thus the prediction capability of the network is increased. The training process continues for every batch similarly.



## Dogs vs. Cats Prediction Problem

**The main goal of this project is to develop a system that can identify images of cats and dogs.** The input image will be analyzed and then the output is predicted. The model that is implemented can be extended to a website or any mobile device as per the need. The Dogs vs Cats dataset can be downloaded from the Kaggle website. The dataset contains a set of images of cats and dogs. **Our main aim here is for the model to learn various distinctive features of cat and dog.** Once the training of the model is done it will be able to differentiate images of cat and dog.

### Installing Required Packages for Python 3.6

1. Numpy -> 1.14.4 [ Image is read and stored in a NumPy array ]
2. TensorFlow -> 1.8.0 [ Tensorflow is the backend for Keras ]
3. Keras -> 2.1.6 [ Keras is used for implementing the CNN ]

## Import Libraries

1. NumPy- For working with arrays, linear algebra.

2. Pandas – For reading/writing data

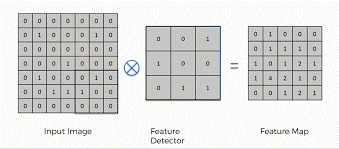
3. Matplotlib – to display images

4. TensorFlow Keras models – Need a model to predict right

5. TensorFlow Keras layers – Every NN needs layers and CNN needs well a couple of layers.

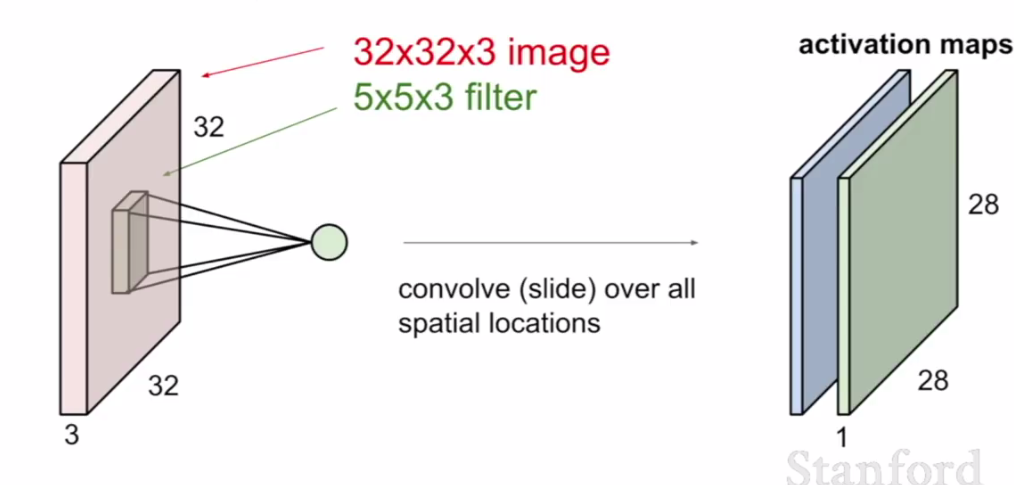
## Convolution

Convolution is a linear operation involving the multiplication of weights with the input. The multiplication is performed between an array of input data and a 2D array of weights known as filter or kernel. The filter is always smaller than input data and the dot product is performed between input and filter array.



## Activation

The activation function is added to help ANN learn complex patterns in the data. The main need for activation function is to add non-linearity into the neural network.



## Pooling

The pooling operation provides spatial variance making the system capable of recognizing an object with some varied appearance. It involves adding a 2Dfilter over each channel of the feature map and thus summarise features lying in that region covered by the filter.

So, pooling basically helps reduce the number of parameters and computations present in the network. It progressively reduces the spatial size of the network and thus controls overfitting. There are two types of operations in this layer; Average pooling and Maximum pooling. Here, we are using max-pooling which according to its name will only take out the maximum from a pool. This is possible with the help of filters sliding through the input and at each stride, the maximum parameter will be taken out and the rest will be dropped.

The pooling layer does not modify the depth of the network unlike in the convolution layer.

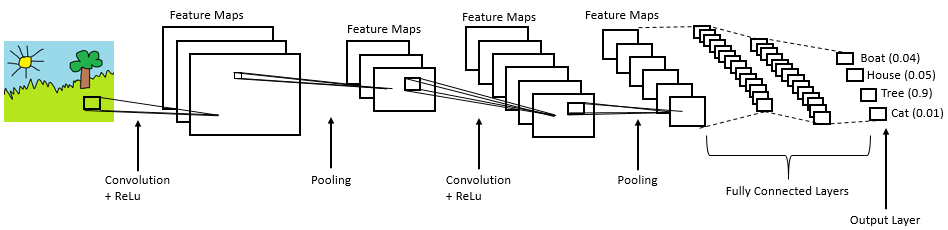


## Fully Connected

The output from the final Pooling layer which is flattened is the input of the fully connected layer.

The Full Connection process practically works as follows:

The neurons present in the fully connected layer detect a certain feature and preserves its value then communicates the value to both the dog and cat classes who then check out the feature and decide if the feature is relevant to them.



## Features Provided:

· We can test our own images and verify the accuracy of the model.

· We can integrate the code directly into our other project and can be extended into a website or mobile application device.

· We can extend the project to different entities by just finding the suitable dataset, change the dataset and train the model accordingly.

## Applications:

We get a general idea of how image classification can be performed. The scope of the project can be further extended to different industries with the scope for automation by just modifying the dataset as needed for the issue.

## Conclusion:

## After this project we get the basic understanding of Convolutional Neural networks and can classify images of cats and dogs.