



Shri Vile Parle Kelvani Mandal's

**DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**

Approved by AICTE and Affiliated to the University of Mumbai



**Department of Electronics & Telecommunication Engineering**

## **Mini Project On**

**Title: Cat and Dog Recognition**

**SUBMITTED BY:**

<b>Name of Student</b>	<b>SAP ID</b>
Sanjeet Krishna	60002160050
Viren Baria	60002160005

**Teacher's Name: Prof. Vishakha Kelkar**



Shri Vile Parle Kelvani Mandal's

**DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**

Approved by AICTE and Affiliated to the University of Mumbai



**Department of Electronics & Telecommunication Engineering**

## **CERTIFICATE**

This is to certify that Mr. Viren Baria,  
SAP ID 60002160005 of TE EXTC 1: has submitted their  
Mini Project for Subject Name for the Academic Year 2018-2019.

Guide

Examiner

Head of Department

EXTC Department



Shri Vile Parle Kelvani Mandal's

**DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**

Approved by AICTE and Affiliated to the University of Mumbai



**Department of Electronics & Telecommunication Engineering**

## Index

Sr No.	Topic	Page No.
1	Introduction	4
2	Software used	4
3	Flowchart	4
4	Theory	5
5	Results	6
6	Conclusion	7
7	References	8



## Department of Electronics & Telecommunication Engineering

### Introduction:

Image classification is a fundamental problem in computer vision. Deep learning provides successful results for machine learning problems. Many algorithms like minimum distance algorithm, K-Nearest neighbour algorithm, Nearest Clustering algorithm, Fuzzy C - Means algorithm, Maximum likelihood algorithm are used for the purpose of image classification. Here, image classification is performed using convolutional neural network which is became standard after since AlexNet (Alex Krizhevsky, Geoff Hinton and Ilya Sutskevar) won Image Net in 2012. Generally convolutional neural network uses GPU technology because of huge number of computations but, in proposed method we are building a very small network which can work on CPU as well. The network is trained using a subset of Kaggle Dog-Cat dataset. This trained classifier can classify the given image into either cat or dog. The same network can trained with any other dataset and classify the images into one of the two predefined class.

**Software Used:** Python

### Flowchart:

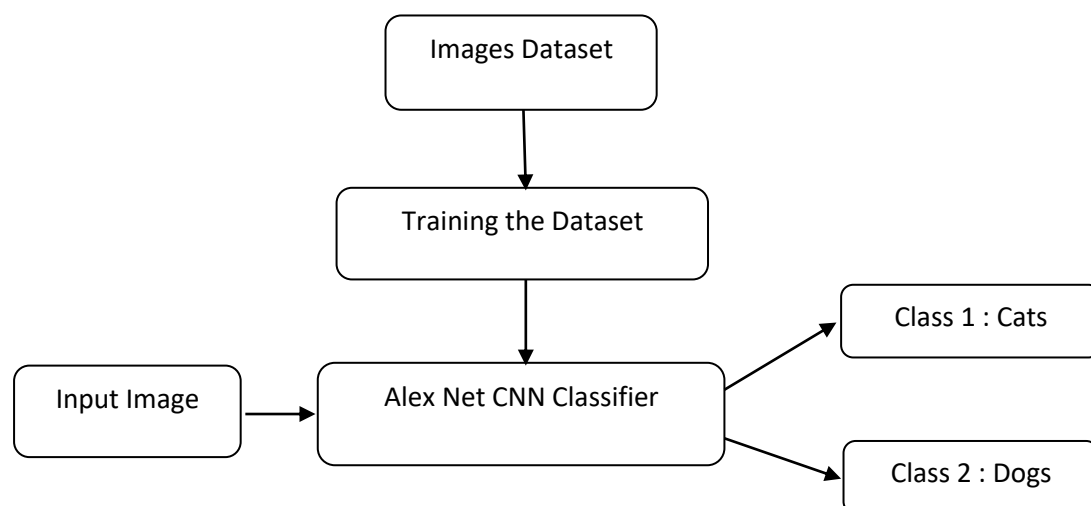


Fig.1. AlexNet CNN Model



## Department of Electronics & Telecommunication Engineering

### Theory:

AlexNet solves the problem of image classification where the input is an image of one of 1000 different classes (e.g. cats, dogs etc.) and the output is a vector of 1000 numbers. The  $i^{\text{th}}$  element of the output vector is interpreted as the probability that the input image belongs to the  $i^{\text{th}}$  class. Therefore, the sum of all elements of the output vector is 1.

The input to AlexNet is an RGB image of size  $256 \times 256$ . This means all images in the training set and all test images need to be of size  $256 \times 256$ .

If the input image is not  $256 \times 256$ , it needs to be converted to  $256 \times 256$  before using it for training the network. To achieve this, the smaller dimension is resized to 256 and then the resulting image is cropped to obtain a  $256 \times 256$  image.

If the input image is grayscale, it is converted to an RGB image by replicating the single channel to obtain a 3-channel RGB image. Random crops of size  $227 \times 227$  were generated from inside the  $256 \times 256$  images to feed the first layer of AlexNet. Note that the paper mentions the network inputs to be  $224 \times 224$ , but that is a mistake and the numbers make sense with  $227 \times 227$  instead.

### AlexNet Architecture

AlexNet was much larger than previous CNNs used for computer vision tasks (e.g. Yann LeCun's LeNet paper in 1998). It has 60 million parameters and 650,000 neurons and took five to six days to train on two GTX 580 3GB GPUs. Today there are much more complex CNNs that can run on faster GPUs very efficiently even on very large datasets.

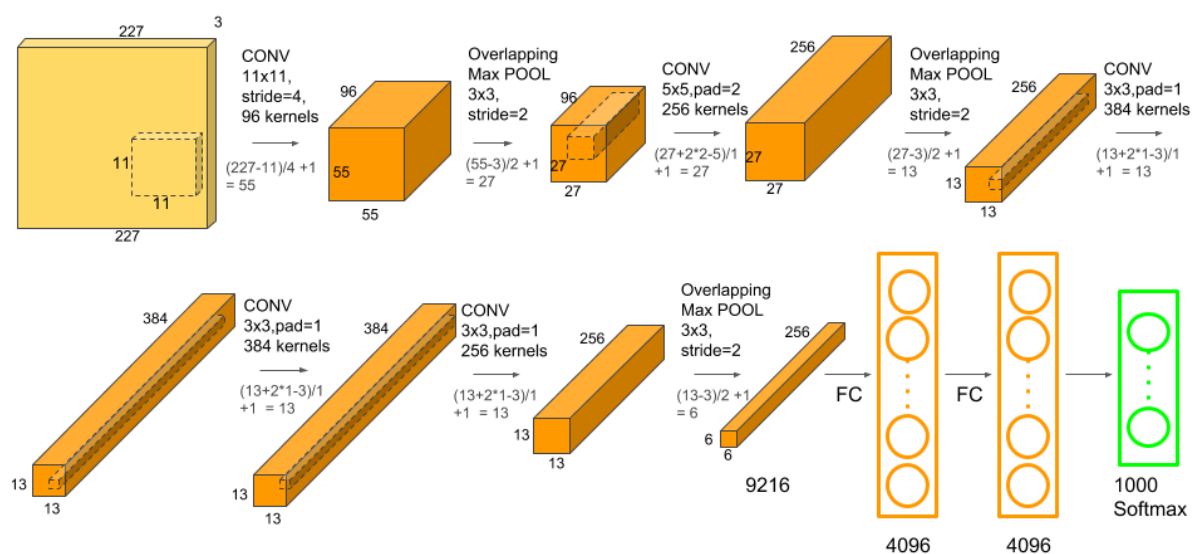


Fig.2.AlexNet CNN Architecture



## Department of Electronics & Telecommunication Engineering

AlexNet consists of **5 Convolutional Layers** and **3 Fully Connected Layers**.

Multiple Convolutional Kernels (a.k.a filters) extract interesting features in an image. In a single convolutional layer, there are usually many kernels of the same size. For example, the first Convolution Layer of AlexNet contains 96 kernels of size 11x11x3. Note the width and height of the kernel are usually the same and the depth is the same as the number of channels.

The first two Convolutional layers are followed by the Overlapping Max Pooling layers that we describe next. The third, fourth and fifth convolutional layers are connected directly. The fifth convolutional layer is followed by an Overlapping Max Pooling layer, the output of which goes into a series of two fully connected layers. The second fully connected layer feeds into a softmax classifier with 1000 class labels.

ReLU (Rectified Linear Unit) nonlinearity is applied after all the convolution and fully connected layers. The ReLU nonlinearity of the first and second convolution layers are followed by a local normalization step before doing pooling.

## Results:

```
Cat image classifier > Train_net.py
Run Train_net
2560/6400 [=====>.....] - ETA: 13:47 - loss: 0.1926 - acc: 0.9207
2688/6400 [=====>.....] - ETA: 13:19 - loss: 0.1950 - acc: 0.9193
2816/6400 [=====>.....] - ETA: 12:51 - loss: 0.1948 - acc: 0.9183
2944/6400 [=====>.....] - ETA: 12:24 - loss: 0.1918 - acc: 0.9202
3072/6400 [=====>.....] - ETA: 11:57 - loss: 0.1895 - acc: 0.9212
3200/6400 [=====>.....] - ETA: 11:29 - loss: 0.1936 - acc: 0.9200
3328/6400 [=====>.....] - ETA: 11:02 - loss: 0.1913 - acc: 0.9213
3456/6400 [=====>.....] - ETA: 10:34 - loss: 0.1905 - acc: 0.9216
3584/6400 [=====>.....] - ETA: 10:07 - loss: 0.1885 - acc: 0.9222
3712/6400 [=====>.....] - ETA: 9:39 - loss: 0.1878 - acc: 0.9227
3840/6400 [=====>.....] - ETA: 9:12 - loss: 0.1875 - acc: 0.9227
3968/6400 [=====>.....] - ETA: 8:44 - loss: 0.1857 - acc: 0.9236
4096/6400 [=====>.....] - ETA: 8:17 - loss: 0.1834 - acc: 0.9250
4224/6400 [=====>.....] - ETA: 7:49 - loss: 0.1819 - acc: 0.9259
4352/6400 [=====>.....] - ETA: 7:22 - loss: 0.1826 - acc: 0.9256
4480/6400 [=====>.....] - ETA: 6:54 - loss: 0.1819 - acc: 0.9257
4608/6400 [=====>.....] - ETA: 6:27 - loss: 0.1817 - acc: 0.9256
4736/6400 [=====>.....] - ETA: 6:00 - loss: 0.1818 - acc: 0.9261
4864/6400 [=====>.....] - ETA: 5:32 - loss: 0.1812 - acc: 0.9262
4992/6400 [=====>.....] - ETA: 5:05 - loss: 0.1794 - acc: 0.9269
5120/6400 [=====>.....] - ETA: 4:37 - loss: 0.1788 - acc: 0.9273
5248/6400 [=====>.....] - ETA: 4:09 - loss: 0.1772 - acc: 0.9282
5376/6400 [=====>.....] - ETA: 3:42 - loss: 0.1765 - acc: 0.9286
5504/6400 [=====>.....] - ETA: 3:14 - loss: 0.1753 - acc: 0.9290
5632/6400 [=====>.....] - ETA: 2:46 - loss: 0.1762 - acc: 0.9286
5760/6400 [=====>.....] - ETA: 2:18 - loss: 0.1758 - acc: 0.9288
5888/6400 [=====>.....] - ETA: 1:51 - loss: 0.1762 - acc: 0.9283
6016/6400 [=====>.....] - ETA: 1:23 - loss: 0.1750 - acc: 0.9292
6144/6400 [=====>.....] - ETA: 55s - loss: 0.1768 - acc: 0.9284
6272/6400 [=====>.....] - ETA: 27s - loss: 0.1765 - acc: 0.9287
6400/6400 [=====] - 1396s 218ms/step - loss: 0.1758 - acc: 0.9291

acc: 93.47%
Saved model to disk

Process finished with exit code 0
```



## Department of Electronics & Telecommunication Engineering

Fig.3. Training the Dataset

```
Run: Train_net Test_net
Instructions for updating:
Please use 'rate' instead of 'keep_prob'. Rate should be set to 'rate = 1 - keep_prob'.
Loaded model from disk
[[593 407]
 [145 855]]
acc: 72.40%
Process finished with exit code 0
```



```
Loaded model from disk
dog.jpg
[[0.869275]]
dog
Process finished with exit code 0
```



```
Loaded model from disk
cat1.jpg
[[0.21600792]]
cat
Process finished with exit code 0
```

Fig.4. Testing the Dataset

## Conclusion:

We have successfully performed a classification process using Alex Net convolutional neural networks. Although there are many algorithms that perform image classification, convolutional neural network is considered to be a standard image classification technique. Convolutional neural network uses GPU technology because of large number of layers which increases the number of computers. Therefore, we presented a very small convolutional neural network which can work on CPU as well. This network classifies the images into one the two predefined classes say Cat and Dog.



Shri Vile Parle Kelvani Mandal's

**DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**

Approved by AICTE and Affiliated to the University of Mumbai



## **Department of Electronics & Telecommunication Engineering**

### **References:**

1. Shanmukhi, M., et al. "CONVOLUTIONAL NEURAL NETWORK FOR SUPERVISED IMAGE CLASSIFICATION." *International Journal of Pure and Applied Mathematics* 119.14 (2018): 77-83.
2. You, Yang, et al. "100-epoch imagenet training with alexnet in 24 minutes." *ArXiv e-prints* (2017).
3. Gonzales and Woods, "Digital Image Processing", Pearson Education, India, Third Edition.