1. Document containing following information

a)      Which Neural Network and why?

I’m using Convolution neural network (CNN).Because CNN  automatically detects the important features without any human supervision. CNNs are used for image classification and recognition because of its high accuracy. The CNN follows a hierarchical model which works on building a network, like a funnel, and finally gives out a fully-connected layer where all the neurons are connected to each other and the output is processed.

b)      Which optimizer and why?

I used Adam() optimizer for my project , because Adam optimization is a stochastic gradient descent method that is based on adaptive estimation of first-order and second-order moments. The method computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients*.*

c)      Which accuracy metric and why

I’m using accuracy class as accuracy metric which will compile mu model according to the accuracy. The metrics argument should be a list — you model can have any number of metrics.It is the list of metrics to be evaluated by the model during training and testing.

d)      Which loss function and why

I used categorical\_crossentropy which will Computes the categorical crossentropy loss. The categorical crossentropy is well suited to classification tasks

e)      Brief information on how cleaning was done (if any)

For this dataset I did not perform any cleaning.

f)       How data was got into the right shape (if any)

The pixcels are in the dataset are compared with live recognized facial pixcel .

g)      What functions/features of OpenCV were used

CascadeClassifier() - which load the classifiers from the file.

VideoCapture() - Its argument can be either the device index or the name of a video file. Device index is just the number to specify which camera. Normally one camera will be connected . So I simply pass 0 . You can select the second camera by passing 1 and so on.

cvtColor() – This is used for BGR colored image.

detectMultiScale() - Detects objects of different sizes in the input image. The detected objects are returned as a list of rectangles

putText() – This method is used to draw a text string on any image

imshow() - **imshow uses** the default display range for the image data type and optimizes figure, axes, and image object properties for image display.

waitKey() - waitKey() is a **keyboard** binding function. Its argument is the time in milliseconds.

h)      Which dataset have you used

I used fer2013.csv dataset which is taken from Kaggle

2)      Your code “ipynb” notebook or python program file

My code file is of ipynb notebook.