Machine Learning Assignment 2

**Emotion Recognition using CNN in TensorFlow**

# **Aim:**

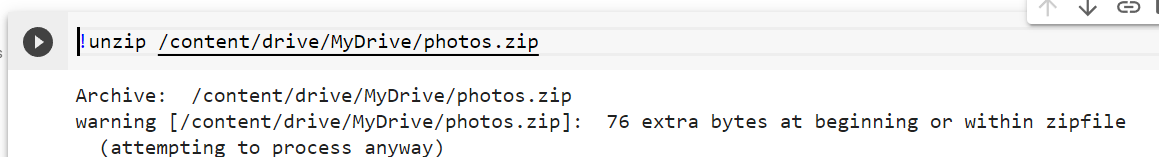
The goal is to create a CNN (convolutional neural network) such that it detects the patterns in the given images, and classifies between the various classes.

# **Import dataset:**

I have taken a dataset from Kaggle link:

<https://www.kaggle.com/datasets/jonathanoheix/face-expression-recognition-dataset>

This dataset has divided the training and testing data and organised various emotions as separate folders for the model to navigate within folders with ease. Next I have imported it into colab and unzipped the images:

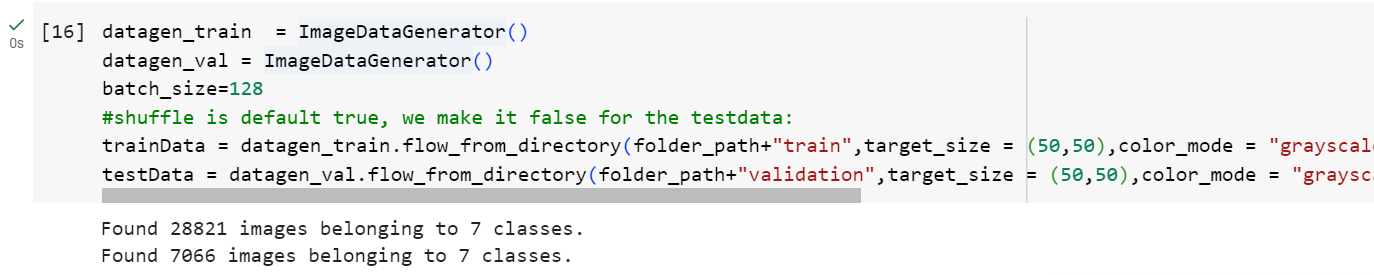


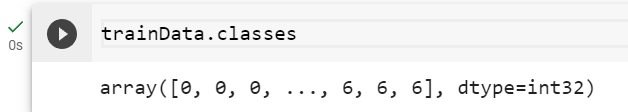
Import necessary modules:



# **Split the training and testing data:**

By using the ImageDataGenerator, that converts the images to array, we have used it on training and testing data separately. Clearly 7 classes or emotions have been identified by it.





# **Create the CNN model:**

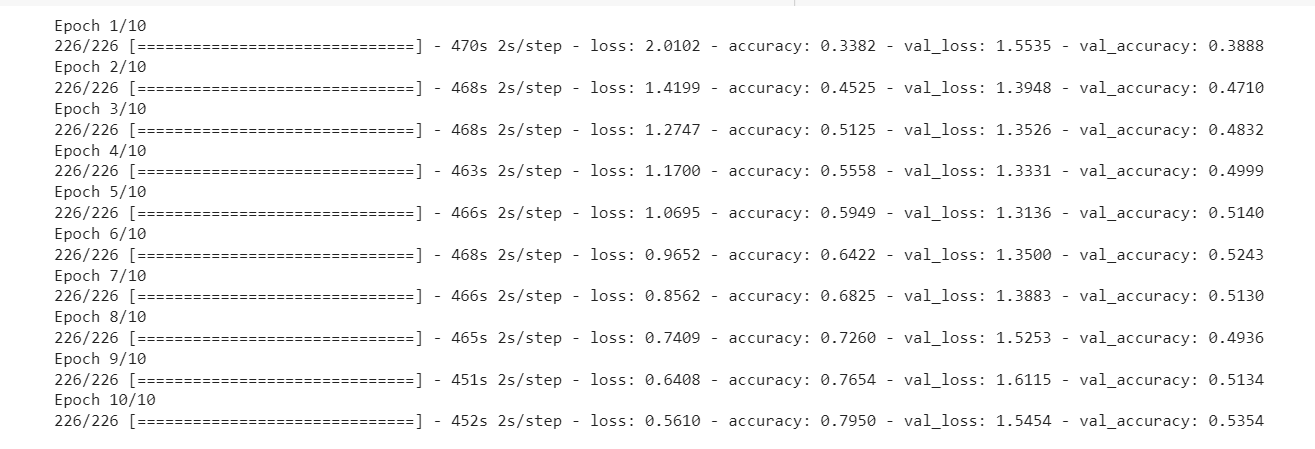
The model is created by using various layers such as convolutional layer, pooling layers, dense layers. Along with it, I have used the batch normalization and dropout, in order to reduce the overfitting cases. At last, we have used the softmax activation function in the output layer.

In compiling, we have used categorical crossentropy, as this is a case of multi class classification. Optimizer used is adam.

Next, we trained our model for 10 epochs, with validation data as the testData.



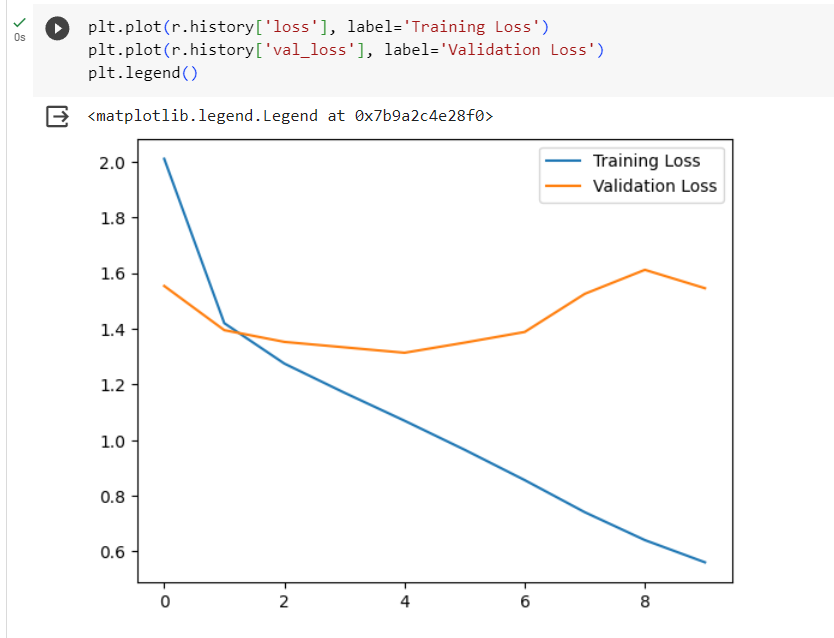
# **Loss and Accuracy Metrics:**

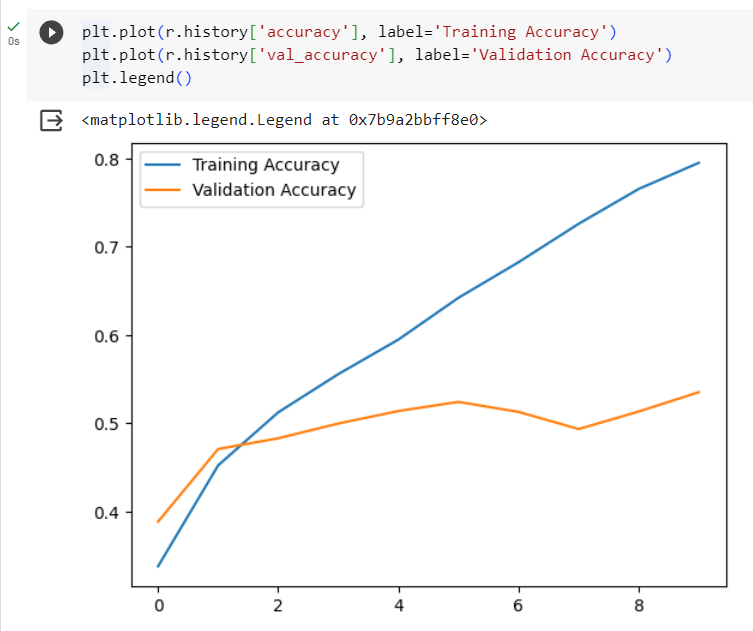


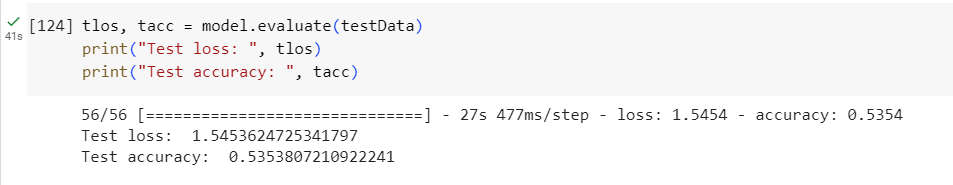
**Training Accuracy: 0.795**

**Training Loss: 0.56**

Loss and accuracy charts for the training set are as follows:





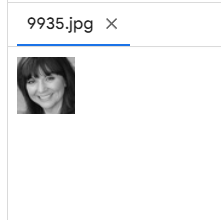


Clearly the **test accuracy is 53.54%.** it is a bit less due to low number of epochs being run.

# **Prediction for a new image:**

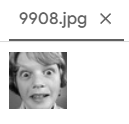
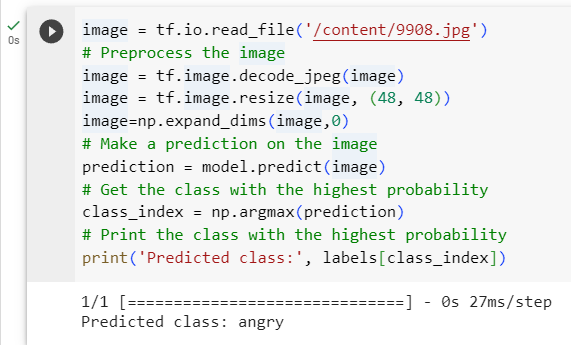


Actual image was:

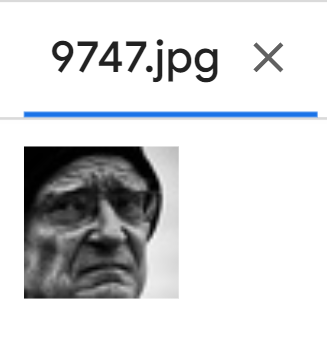
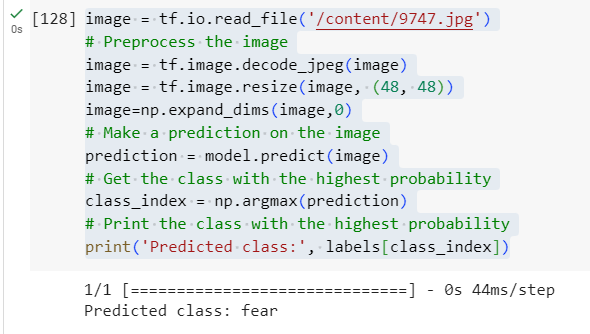
 So yes, our model has learned the prediction nicely.

# **Misclassifications:**

But of course, there are some misclassifications with this model as follows:



**Actual class= SURPRISED**

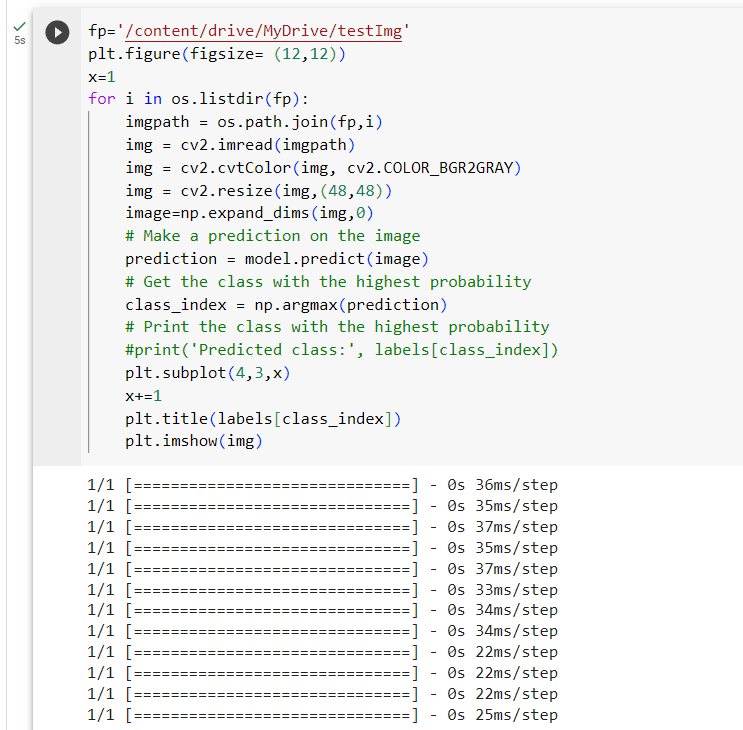


**Actual class=SAD**

But the misclassifications are justified, as the images may show similar patterns. For example, sad and fear may be observed in the above photo.

# **12 Sample images classified from google:**

We have taken 12 images from google of various expressions. I have applied my model to them in order to predict their expression.





It has predicted 6/12 images correctly.

# **Text Analysis in Multimodal Analysis:**

I have tried to analyse the text if some is provided with the photo, to classify them as either positive or negative. Hence, I tried to make a text classification model, but it seems that it is not possible on google colab. I have tried using word2vec module in order to convert the words into vectors so that I can pass them into the model layers, but it shows the error that string\_to\_number cast is not possible. I have several other ways to convert the strings into numbers, but it was not successful.

If the text classification model was built, then we could have generated results based on combining both the results from text and image.