Facial Emotion Detection

Rupam Kundu

We are going to make a project on facial emotional detection of human using deep neural network mainly convolution neural network. I am going to do a detail discussion of my code, to show how did I complete my project.  
Now a days, facial emotion detection using deep neural network is one of the most important focuses to work on. Emotion detection using images is quite useful for identification like driver's drowsiness detection, student's behaviour detection etc.  
In my code I have import it some packages from the Keras library which uses Tensorflow. It will be useful to create CNN, process images and etc.  
We will be using to image data generator that taste data generator and the train data generator.  
At first, I have used the train data generator and used flow from directory in it. From directory is one of the most interesting things that goes through the datas and collect and pre-process them. Each image will be resized to 48,48 with the batch size 64 color mode grey scale and class mode categorical.  
Again we will use the test or validation generator and we will again use the flow from directory.  
The configurations will be remaining same, that is the images will be resized to 48, 48 and with bath size 64 color mode Grayscale and class-mode categorical!  
Generator is ready to be trained, and all training test data is pre-processed.  
From keras, I am accessing the sequential model and inside adding different layers.  
In this part I have a convolution 2D layer with 32 size and kernel size 3 by 3 and activation function relu, with an input shape, say peace 48 by 48 with colour code Grayscale (only 1 color).  
[For RGB images we need to pass 3 but we don't do that as it will consume more time]  
  
We are again adding a convolution 2D layer with  
64 size kernel size 3/3 and activation function relu.  
We are going to add pulling of 2 by 2. To avoid over fitting we add drop out of 0.25.  
Adding another convolution layer and adding another pooling layer.  
Again you are reading another convolution layer and another pooling layer.  
And again to get avoid over fitting we are adding drop out of 0.25.  
At last we are adding the flatten layer to flatten values and adding a dense layer of dense 1024, dropout of 0.5.  
  
That in we are going to get over 7 classifications or categories. So  we are adding the last dense layer with 7 categories.  
And adding the activation function as soft max!  
And this is our complete CNN.  
To compare this convolution layer we will use a loss function optimizer and matrices from the particular model.  
  
We are adding loss function as categorical cross entropy and adding optimizer as Adam with learning rate 0.0001 and decay of exp(-6) and using matrices as an accuracy.  
Now we have to train our CNN with train data.  
We are using fit generator and all our data is stored in train generator above, and now we are passing that data here.  
Step\_per\_epoch = total no of images/64 and training will happen for 50 epoch. Validation data equal to validation generator.  
We are passing it while training or model it will evaluate how our model will be working. At the stage our model will get trained.  
Once our model is trained, we can store our model.  
We can see we are storing all model structure in emotion\_model.json file, and all learning weights in emotion\_model.h5 file.  
Now we will test our data.  
We have imported some packages.  
We are using open cv to access camera feed or read video content!  
Keras.model is to load the  emotion\_model.h5 file and emotion\_model.json file.  
I have created the emotion\_dect to index the attributes.  
Next we are loading, whatever model we have stored in the loaded\_model.json.  
Then, we will use model\_from\_json to convert the loaded json into the model.  
In the step we are creating the convolutional neural network which we have sotred in json file and in next step we are using h5 model. Using the weights we are applying the learning which we have made.  
Now for input we will use opencv.  
  
We can use our webcam, by the hastagged comment and use video for emotion detection by the next comment, whose path we are providing.  
Before going further we are resizing the frame to 1280/720 for a good fit in our screen.  
Now one more important thing!  
We have to detect faces from the video or camera feed, in order to tell the emotion of the person.  
So we are going to use haarcascade here, which will detect the faces from the videos or any live camera feed.  
Then the face will be passed to our emotion detection model to detect the emotion from face!  
For RGB image, there may be some kind of anomalies. So, we converted it to Greyscale image.  
This next phase we are positioning the detection frame in perfect position in the video or camera feed, we are converting the detected face in 48 by 48 size.  
At finally we are passing the pictures of the faces to emotion detection model to make the prediction of emotion!  
And, we can get our prediction in the left above corner of the marked box in the output!  
  
Thank You!