Facial Expression Recognition using Convolutional Neural Networks

Introduction

In this project, we have developed convolutional neural network (CNN) for real time facial expression recognition task. The goal is to classify each facial image into one of seven facial emotion categories. We trained CNN model with different depth using grey - scale images from the **Kaggle** website. To reduce the overfitting of the models, we utilized different techniques including dropout and batch normalisation.

1. Data set and features
   In this project, we use a dataset provided by Kaggle website, which consists of 35887 well structured 48x48 pixel grey-scale images of faces. Each image has to be categorized in to one of the seven classes that express different facial emotions. These facial emotions have been categorized as: 0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral. The given data set is divided in to two different sets which are training and test sets. Training set contain 30000 images and test set contain 5887 images. After reading the raw pixel data we standardized them by subtracting the mean of training images from each image including those in the test sets and further dividing the difference (X-X(mean)) by standard deviation of the training images (standard deviation(X)).
2. Methods
   We developed CNNs to evaluate the performance of two different models for facial expression recognition. We considered the following network architecture:

   [Conv-ReLU-(Dropout) -(Max-pool)]4-Flatten-[Dense-ReLU-(Dropout)]2-[Dense-Sigmoid]

   The first part of the network refers to the 4 convolutional layers that can possess dropout and max-pooling in addition to the convolution layer and ReLU nonlinearity, which always exist in these layers. After four convolution layers, the network is led to 2 fully connected layers that always have ReLU nonlinearity and dropout. The last layer consists of 7 nodes with sigmoid activation function.
3. Analysis
   For the purpose of this project, first we built a network which had two convolutional layers and one fully connected (FC) layer. In the first convolutional layer we had 32 3x3 filters, with the stride of size 1, along with

dropout, but without max-pooling. The second convolutional layer we have 64 3x3 filters, with the stride of size 1, along with dropout and also max-pooling with a filter size of 2x2. In the FC layer, we had a hidden layer with 512 neurons and sigmoid as the loss function. Also, in all the layer, we used Rectified Linear Unit (ReLU) as the activation function. Then, we started training our model from scratch to make the model process faster, we exploited GPU accelerated deep learning facilities on Keras using tensorflow as a backhand. For the training process we used all of the images in the training set with 45 epochs and a batch size of 128. To test the model, we used test set. The accuracy on the test set is 60.42%.
Second, we trained a deeper CNN with four convolutional layers and 2 fully connected layer. The first convolutional layer had 64 3x3 filters, the second one had 128 5x5 filters, the third one had 256 3x3 filters, the fourth one also had 256 3x3 filters in all the convolutional layer, we have a stride of size 1, dropout, max-pooling (2x2, stride (2,2)) and ReLU as activation function. The hidden layer in the first fully connected had 256 neurons and the second had 512 neurons. In both fully connected layers, same as in the convolutional layers, we used dropout and ReLU. Also, we used sigmoid as our loss function. Then, using 70 epochs and batch size of 128 we trained a network with all images in training set. This time we obtained an accuracy of 63.85% on the test set.

Expression	Shallow Model	Deep Model
Angry	56.39%	54.35%
Disgust	62.33%	85.07%
Fear	47%	51.77%
Happy	78.4%	80.64%
Sad	48.25%	54.29%
Surprise	74.81%	77.14%
Neutral	50.29%	54.18%

4. Results

To compare performance of the shallow model with the deep model, we plotted the loss history and the obtained accuracy in these models. It is interesting to see that both models performed well in predicting the happy level, which implies that learning the features of a happy face is easier than other expressions. Additionally, the matric reveal which labels are likely to be confused by the trained networks. For example, we can see the correlation of angry label with the fear and sad labels. These mistakes are consistent with what we see when looking at images in the dataset; even as a human, it can be difficult to recognize whether the angry expression is actually sad or angry. This is due to the fact that people don not all express emotions in the same way. In addition to the confusion matric, we computed the accuracy of each model for every expression. The above table shows these results. As seen in this table, the accuracy of predicting happy expression has the highest value among all emotions in shallow model and second highest value in deep model. Also, for most of the expression using deep network has increased the classification accuracy. For angry expression using deep model not only did not help to get better accuracy but also decrease its prediction accuracy it means that for some expression going deeper does not necessarily provide better features.

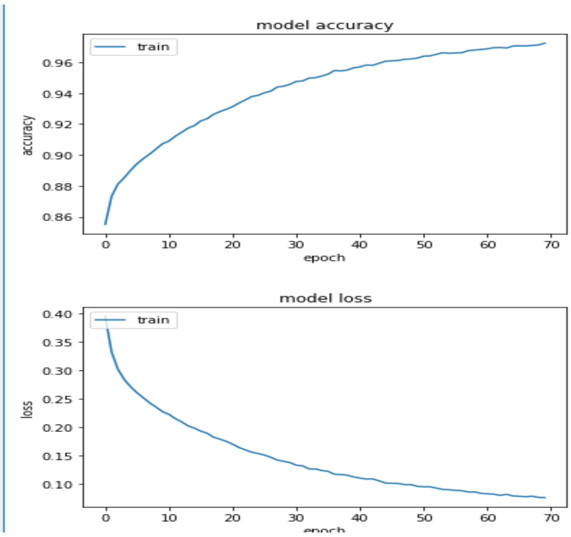
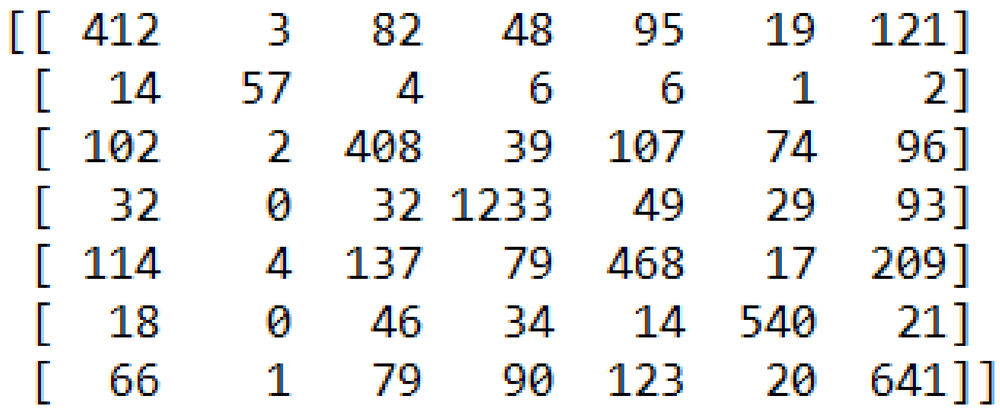


Figure 1: Accuracy and loss history graph for deep model



	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	412	3	82	48	95	19	121
Disgust	14	57	4	6	6	1	2
Fear	102	2	408	39	107	74	96
Happy	32	0	32	1233	49	29	93
Sad	114	4	137	79	468	17	209
Surprise	18	0	46	34	14	540	21
Neutral	66	1	79	90	123	20	641

Figure 2: The confusion matrix for the deep model

5. Conclusion

   We developed various CNNs for real time facial expression recognition problem and evaluated their performances using different post-processing and visualization techniques. The result demonstrated that deep CNNs are capable of learning facial characteristics and improving facia emotion detection.