Facial Expression Recognition from Static Images

Project Report

6 May, 2017

Chintan Saraviya, Dhaval Thummar, Maitrey Mehta, Parth Shah

***Abstract*—Facial expressions are reflection to a persons internal emotional state, intentions, or social communications. Facial expression depicted by humans can be classified into seven main categories namely: fear, anger, joy, surprise, disgust, sad and neutral. Facial expressions of any person can be determined on a computational device viz. computers. A deep learning approach to this problem can be developed using concepts of sequential models and Convolutional Neural Network(CNN) or convnets. We have proposed a deep learning sequential model comprised of CNN, Maximum pooling, Rectified linear Units and fully connected layers. Predictions for any given test case is done by using softmax function. Using softmax we will predict the probability of all emotions for any given image. We also developed a Support Vector Machine classifier and compared the results with the results of CNN. Applications of discussed topic extends from simply facilitating retailers to analyse customer satisfaction to more complex encounters like guessing patients mental state during treatment or estimating audiences reaction to a play or movie.**

Key words: Convolutional Neural Networks(CNN), Pooling, softmax, confusion matrix.

I. Introduction

2016 is the year when machines learn to grasp human emotions Andrew Moore, the dean of computer science at Carnegie Mellon. Facial expression of humans depicts human emotions, thus it becomes necssary for us to train our computers to predict these expressions accurately. Geometric based feature detection or appearance based feature detection are some of the exisiting methods, which extracts the features from a subjects face and make decision based on the matching action units(AU). A self sufficient literature on the above discussed methods can be found in [1]. With changing trends of fashion(variation in beard or mustache) or due to age([1] mentions an issue where expression of an adult is d etermined accurately but same model is not valid for infants) the features of any subject is likely to change and in such cases robustness of the above proposed methods is likely to suffer. [1] also discusses the limitations of the model where sufficient features of a face are not provided(side view of a face).Even authors of [1] acknowledges that neural networks is a better approach to facial expression recognition. We have proposed a deep learning sequential model that facilitates machines to guess facial expressions correctly. Our model highly depends on the work proposed by [2]. With the help of Convolutional Neural networks(CNN) or convnets, fully connected layers and some concepts of Digital signal processing like maximum pooling, 2D convolution, [2] has proposed a sequential approach to recognize human expression. Another approach defines the use of one vs all SVM classifiers. The training datasets are arranged in the feature plane. A binary SVM classifier is used considering the dataset of one class of expression and another dataset of the combination of rest classes of expression. Six different models are hence created and a test data is classfied based on the proximity to each model.

II. Datasets

Selection of datasets is one of the essential part in our deep learning model.This is necessary as we need to train our CNN and this has to be done using datasets which would provide a comprehensive set of subjects which makes our neural network learn the most. Authors of [1] has recommended to use CohnKanade[3] for training neural networks. A similar, yet more comprehensive dataset having varied subjects is available as a practice set on Kaggle[4]. This data set contains appprox 40, 000 test images of size 48x48 pixels which are divided into training and test cases. Using the database of [4] we have trained our CNN and then predictions were made on test cases. The private test dataset i.e. images not identical to trained model are also provided in [4].

III. Convolutional Neural Networks

This section discusses Convolutional Neural Networks, developed for expression prediction, all the layer viz. CNN, Maximum pooling, and fully connected layers along with softmax function are explained here.

A. Convolutional Neural Network(CNN)

1) Features: are rightly coined as mini-images in [5].CNN compares images piece by piece.These pieces are called features.CNN learns when the feature matches in two different images at the same place.

   When presented a new image CNN doesn't know where these features are located, so CNN tries to match it in entire image.For calculating this match we consider our feature as a filter. The math we do for this is convolution, hence the name.

B. Pooling

Another prominent tool CNN uses is pooling [5].Pooling is essentially shrinking down the image size by just keping the most essential part of the image.It takes a nxn square matrix of either 2x2 or 3x3 dimension and takes only the most related

part of the image. Thus image is shrinked down by $1/4$ when we use a 2x2 matrix.

C. Dense layer or fully connected layers

These dense layers are different from sparse or hidden layers, in the sense every input unit is connected to every output unit [5].These dense layers would highly help our softmax function further for predicting probalility of each expression in each image. Below (Fig-1) is a sample of the resized image which is an input to our dense layer.

D. Softmax function

Like sigmoid function, softmax is a decision function. But unlike a traditional sigmoid function giving only a single predicted output, softmax function predicts the probability of all the possible cases. It is more intuitive in the sense, sigmoid function as it gives only single output, is considered errorneous, if it predicts false. In case of softmax function we would predict the probability and hence a sense of correct prediction always pertains.

IV. CNN Sequential Model

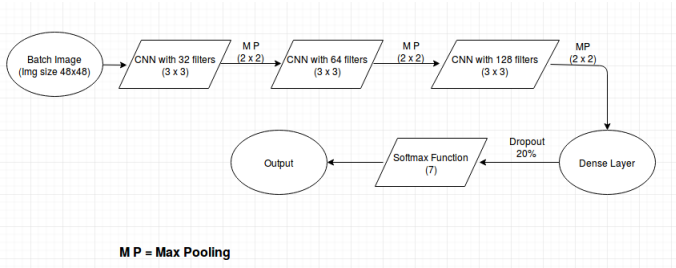

Fig. 1: CNN approach with no dropout (97.037%)

The flow of our sequential model in given in the (Fig-1). A batch of images is given as an input. We have considered a batch size of 128 images. The images are considered as already preprocessed and converted to gray scale. The batch is fed into a Convolutional Neural Network with a specified 32 filters each of size 3x3. These features are unique and have randomly generated weights. The image is convoluted with each filters to create feature maps, one corresponding to each filter. The feature maps are then max pooled using a 2x2 window. The resulting maps are then fed into another CNN with 64 filters. The output is further maxpooled with a 2x2 window. Finally, the output from the maxpooling is fed into a CNN of 128 filters. The output is then maxpooled and passed through a fully connected dense layer. We introduce a 20% drop-out to avoid overfitting. The outputs are passed through the softmax function which decides the appropriate expressions. A confusion matrix is created to assess the accuracy of the model.

V. SVM Model

SVMs are essentially binary classifiers,however, they can be adopted to handle the multiple classification tasks. In our case, SVM classifier uses One-Against-All approach. The One-Against-All approach represents the earliest and most common SVM multiclass approach and involves the division of an N class dataset into N two-class cases. We use the Gaussian kernel for increasing dimensions and dice similarity for better classification. At the end, a confusion matrix is created to assess the accuracy.

VI. Result

The results are generated by simulating our sequential model, the accuracy of the model on public test data set with no drop out is 97.37%. The accuracy of the model on public test data set with 20% drop out is 96.63%. The confusion matrix for the same are given in the figures.The accuracy of the model on private test data set with no drop out is 42.26%. The accuracy of the model on private test data set with 20% drop out is 44.71%. The confusion matrix for the same are given in the figures.

The accuracy of the SVM classifier on private test dataset is merely 31.13%. But the same SVM predicts with 99.87% accuracy on public data set.
The confusion matrix for all the above discussed results are given below.

Figures of public dataset (Fig 3-6), truly depicts the nature of a public dataset, SVM outperforms our CNN model, as SVM works best with already trained dataset.Moreover for the trained dataset we should get a better result with no drop out which is also depicted in the output (Fig 3-4).

The scenario changes when private dataset comes (Fig 7-9), here with an appropriate drop out CNN performs better than CNN model trained without dropout. Also we see that CNN outperforms SVM.

The following figure (Fig 2), depicts that *Happy* emotion is most correctly predicted, and excatly opposite to it *Sad* emotion is predicted least correctly, this analysis is done for a Neural Network approach with 20% dropout.

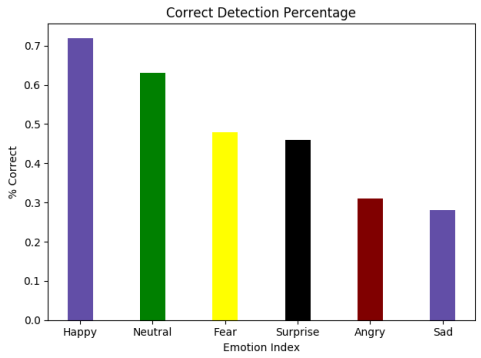

Fig. 2: CNN approach with no dropout (97.037%)

A. Public Dataset Results

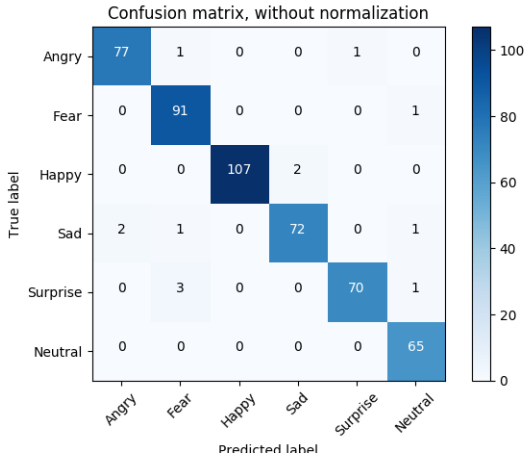

Fig. 3: CNN approach with no dropout (97.037%)

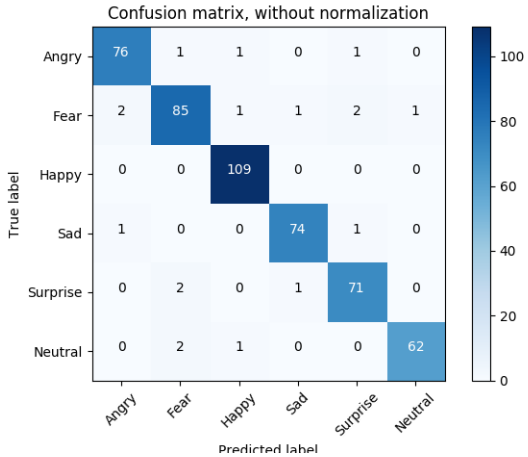

Fig. 4: CNN approach with 20% dropout (96.63)

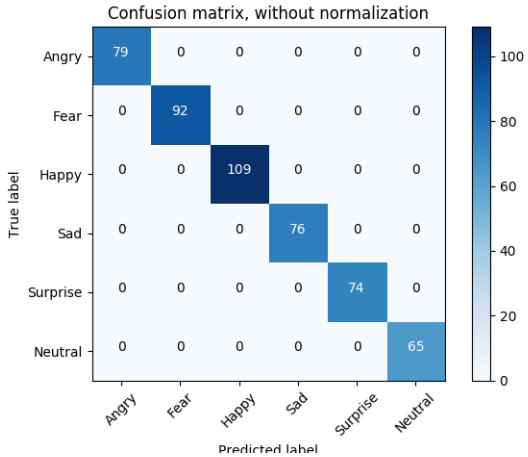

Fig. 5: SVM approach (99.87%)

B. Private Dataset Results

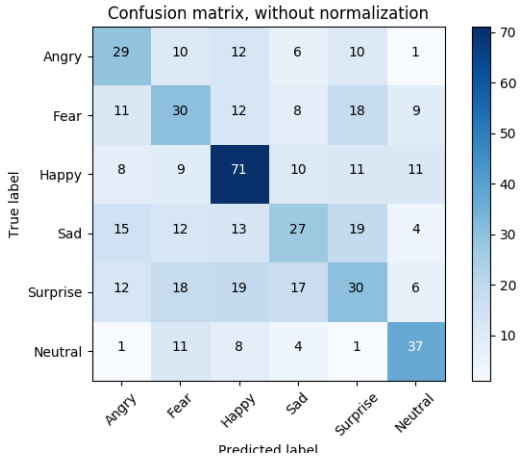

Fig. 6: CNN approach with no dropout (42.26%)

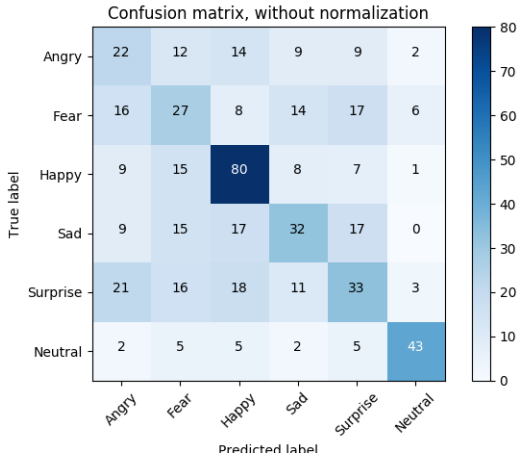

Fig. 7: CNN approach with 20% dropout (44.71%)

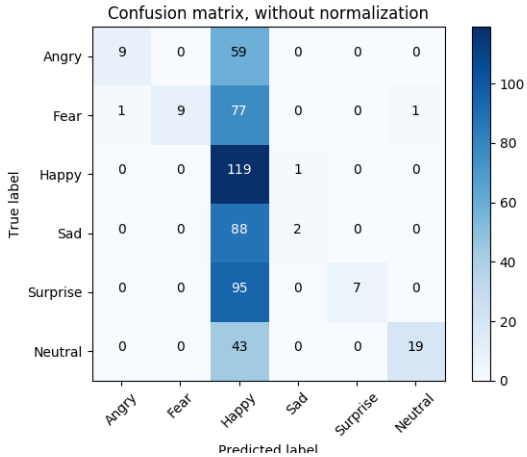

Fig. 8: SVM approach (31.13%)

C. Computation Time

Model	Time Taken
CNN	104 min
SVM	60 min

Table-1:Computational time calculation

Table-1 informs us that though we are getting moe efficiency for sequential CNN model, these model takes a huge time. The computational time is calculated for 128 epoch and 1280 batch size. A similar type of analysis was done for CNN with various batch size and epochs but the takenepoch and batch size proved to have most optimal efficieny.

VII. Future Implementation

A more holistic way of approaching facial detection using SVM, is through landmarks, these improves the accuracy to many folds, this approach can also be used for live feeds instead of static image.Moreover we can also use our Neural network with some more self sufficient heuristics which predicts the expression with more accuracy. The same model can also be trained on a GPU for a better and quicker implementation. Models for detecting expressions can be further extended for animated cartoons.

References

[1] Yingli Tian, Takeo Kanade, and Jeffrey F. Cohn, *Facial Expression Recognition*

[2] Yingli Tian, Takeo Kanade, and Jeffrey F. Cohn, *Comprehensive database for facial expression analysis*

[3] *https://github.com/JostineHo/mememojis*

[4] *https://www.kaggle.com/c/challenges-in-representation-learning-facialexpression-recognition-challenge*