Facial Emotion Recognition Using Deep Convolutional Neural Network

Pranav E.
School of Engineering
Cochin University of Science and Technology
Kochi, India
pranaveswr96@cusat.ac.in

Suraj Kamal
Department of Electronics
Cochin University of Science and Technology
Kochi, India
surajkamal@cusat.ac.in

Satheesh Chandran C.
Department of Electronics
Cochin University of Science and Technology
Kochi, India
satheeshchandran@cusat.ac.in

Supriya M.H.
Department of Electronics
Cochin University of Science and Technology
Kochi, India
supriya@cusat.ac.in

***Abstract*—The rapid growth of artificial intelligence has contributed a lot to the technology world. As the traditional algorithms failed to meet the human needs in real time, Machine learning and deep learning algorithms have gained great success in different applications such as classification systems, recommendation systems, pattern recognition etc. Emotion plays a vital role in determining the thoughts, behaviour and feeling of a human. An emotion recognition system can be built by utilizing the benefits of deep learning and different applications such as feedback analysis, face unlocking etc. can be implemented with good accuracy. The main focus of this work is to create a Deep Convolutional Neural Network (DCNN) model that classifies 5 different human facial emotions. The model is trained, tested and validated using the manually collected image dataset.**

Keywords— Facial Emotion Recognition, Deep Convolutional Neural Network, Classification, Adam.

I. Introduction

The development and usage of computer systems, software and networks are growing tremendously. These systems have an important role in our everyday life and they make human life much easier. Facial emotion recognition system assumes a lot of importance in this era since it can capture the human behaviour, feelings, intentions etc. The conventional methods have limited speed and have less accuracy while facial emotion recognition system using deep learning has proved to be the better one. This system aims to build a deep convolutional neural network model that recognizes 5 different human facial emotions and this can be used for applications such as customer feedback analysis, face unlocking etc.

In the field of computer science, machine learning is one of the emerging technologies that is considered to have an impact of 90% in the next 4 years. Deep learning, a subset of machine learning uses artificial neural network, which is an algorithm inspired from the human brain. Convolutional Neural Network (CNN) is a class of deep neural network that uses convolution as the mathematical operation. As the dataset consists of images, the system uses a 2D CNN for the recognition task. The proposed deep convolutional neural network is trained not only to classify 5 different human facial emotions, but also to yield a good accuracy. The model is trained using the dataset which is collected manually using a mobile phone camera.

II. Related Works

G. Cao *et al* [1] used a Convolutional Neural Network model to recognize human emotion from the ECG dataset, which can in turn be used to classify brain signals as well. The system gives an accuracy around 83% on testing. G. Yang *et al* [2] proposed a DNN model which uses vectorized facial features as input. The model can predict different emotions with an accuracy of 84.33%. Liu *et al* [3] uses the fer2013 dataset and two layer CNN to classify different facial emotions. They have also compared it with 4 different existing models and the proposed model yields a test accuracy of 49.8%. S Suresh *et al* [4] proposed a sign language recognition system that classifies 6 different sign languages using a Deep Neural Network (DNN). Two models with different optimizers (Adam and SGD) are compared and the model with Adam optimizer is found to have more accuracy. K. Bouaziz *et al* [5] has demonstrated an analytics workflow that integrates the tools and techniques of image recognition. The proposed model classifies different handwriting with CNN architecture. F. Zhou *et al* [6] used a deep convolutional neural network model to detect ship in motion for PolSAR images. The model uses a faster region based CNN (Faster-RCNN) method to recognize ships with different sizes. The model is validated using NASA/JPL AIRSAR dataset.

III. Proposed Facial Emotion Recognition Model

This section describes CNN and the architecture of the DCNN model of the proposed system.

A. Convolutional Neural Network

Neural network is a set of algorithms that mimic the human brain and it finds a relationship between the data to get solutions using these algorithms. CNN is a type of Neural Network where the mathematical operation used to find the relationship between the data is Convolution [7]-[9]. Traditional neural network fails when coming to complex problems such as image classification, video classification, pattern recognition, etc. but CNN has achieved great success in these applications, yielding good accuracy.

CNN consists of mainly 4 Layers, viz. convolutional, pooling, dropout and fully connected layers. These layers together extract the features from the input data. The algorithm learns from the feature, where the features of interest are represented by each convolution filter. The convolutional layer

978-1-7281-5197-7/20/$31.00 ©2020 IEEE

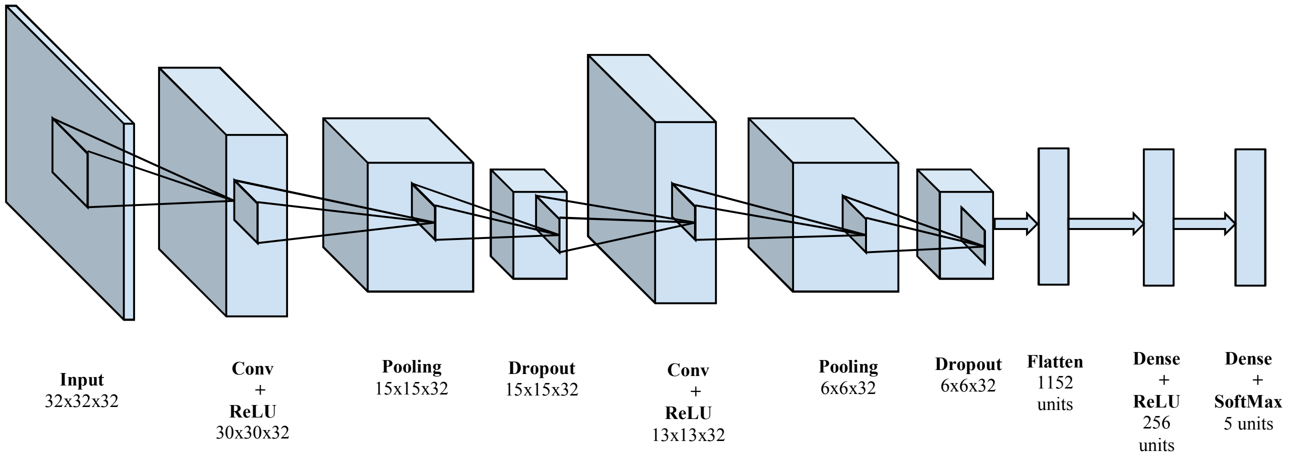


Figure 1. Architecture of the proposed facial emotion recognition model using DCNN

consists of small patches, which transforms the entire image based on the filter values. Equation (1) is the formula to create feature maps, i.e. the output from the convolutional layer.

$$G[m,n] = (f * h)[m,n] = \sum_j \ \sum_k h[j,k] f[m-j, n-k] \quad (1)$$

where f is the input image, h is the filter, (m, n) is the size of the resulting matrix generated.

The output from the convolution layer is passed on to a pooling layer where its size gets reduced without any loss of information. These 2-dimensional arrays are converted to a single dimensional vector using the flatten layer so that it can be fed to the neural network for classification. The neural network uses the back-propagation algorithm where the errors are back propagated to adjust the weights, thereby reducing the error (loss) function. The weight updation is done using (2).

$$W_i = W_i + \Delta W_i \quad (2)$$

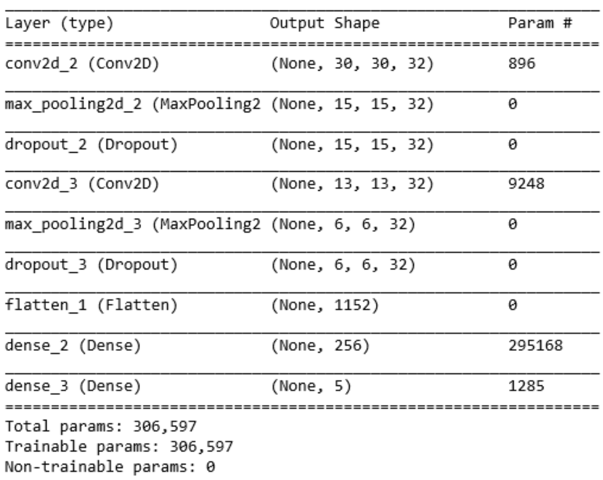
where W_i is the weight and ΔW_i is given by the delta rule as in (3).

$$\Delta W_i = \text{n} \frac{dE}{dW_i} x_i \quad (3)$$

where n is the learning rate, E is the error function and x_i is the input.

B. Proposed CNN Model

The architecture for the proposed facial emotion recognition model is depicted in Figure 1. The model uses two convolution layers with dropouts after each convolution layer. The input image is resized to 32 x 32 and is given to the first convolution layer. The output from the convolution layer, called feature map, is passed through an activation function. The activation function used here is ReLU (Rectified Linear Unit) that makes the negative values zero while the positive values remain the same. This feature map is given to the pooling layer of pool size 2 x 2 to reduce the size without losing any information. Dropout layer is used so as to reduce the overfitting. This process again continues for the next convolution layer as well. Finally, a 2-dimensional array is created with some feature values. Flatten layer is used to convert these 2-dimensional arrays to a single dimensional vector so as to give it as the input of the neural network, represented by the dense layers. Here a two-layer neural network is used, one is input and the other is output. The output layer has 5 units, since 5 classes need to be classified. The activation function used in the output layer is softmax, which produces the probabilistic output for each class. Figure 2 depicts a snapshot of the model summary of the proposed system which is built using the Keras DL Library.



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_2 (Conv2D)            (None, 30, 30, 32)        896
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 15, 15, 32)        0
_________________________________________________________________
dropout_2 (Dropout)          (None, 15, 15, 32)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 13, 13, 32)        9248
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 6, 6, 32)          0
_________________________________________________________________
dropout_3 (Dropout)          (None, 6, 6, 32)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 1152)              0
_________________________________________________________________
dense_2 (Dense)              (None, 256)               295168
_________________________________________________________________
dense_3 (Dense)              (None, 5)                 1285
=================================================================
Total params: 306,597
Trainable params: 306,597
Non-trainable params: 0
```

Figure 2. Model summary of the proposed facial emotion recognition model built using Keras

IV. Dataset

The dataset for the proposed model includes 5 different facial emotions viz. angry, happy, neutral, sad and surprised. These are collected manually using a 48 MP camera. Each image has a pixel size of 1920 x 2560. The dataset split is shown in Table I. Each class consists of the same number of training samples so that they are not biased. The train-test-validation split is in the ratio 8:1:1.

TABLE I. DATASET SPLIT RESULTS

Number of classes	5
Number of training images	2040
Number of validation images	255
Number of testing images	255

V. Methodology

Using python as the programming language, the model is implemented. The entire model is simulated in the Jupyter Notebook. For building the model, adding the convolution layers, compiling and fitting the model, Keras, which runs on top of tensorflow, is used as the deep learning library. Scikit-learn is the package used for finding the confusion matrix that gives the accuracy, precision, sensitivity, specificity, recall, etc. of the model. For plotting the confusion matrix and other graphs such as accuracy and loss, matplotlib and seaborn are employed.

VI. Results

CNN is trained with the emotion image dataset, utilizing Adam as the optimizer and the categorical cross-entropy as the loss function. The model parameters are shown in Table II.

TABLE II. MODEL PARAMETERS

Model Parameters	Values
Total images	2550
Activation	ReLU and Softmax
Learning rate	0.01
Epochs	11
Optimizer	Adam
Loss function	Categorical Cross-entropy

Adam is an optimization algorithm that can be used instead of the classical stochastic gradient descent algorithm to update the network weights with individual learning rate for each of the weights [11]. For each weight of the neural network it uses first and second moment estimations of gradient to adapt the learning rate. The nth moment of the random variable is provided in (4).

$$m_n = E[X^n] \tag{4}$$

where m is the moment and X is the random variable. The first moment is given by the mean and the second moment is given by the uncentered variance. Adam uses exponentially moving averages to estimate the moments. Moving averages of gradient and squared gradient are given by (5) and (6) respectively.

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t \tag{5}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t^2 \tag{6}$$

where m and v are moving averages, β is the hyperparameter and g is the gradient. The final formulae for the bias corrected estimators for the first and second moments are given by (7) and (8) respectively.

$$m_t' = \frac{m_t}{1-\beta_1^t} \tag{7}$$

$$v_t' = \frac{v_t}{1-\beta_2^t} \tag{8}$$

The learning rate for each parameter is scaled individually by these moving averages and the weight updation is done using the formula given in (9).

$$w_t = w_{t-1} - n\frac{m_t'}{\sqrt{v}+\varepsilon} \tag{9}$$

where w_t is the updated weight, w_{t-1} is the previous weight and n is the step size.

Categorical cross-entropy, the loss (error or cost) function used for optimizing classification models, is given by (10).

$$L(y, y') = -\sum_{j=0}^{M}\sum_{i=0}^{N}(y_{ij} * \log(y_{ij}')) \tag{10}$$

where y' is the predicted value. This function will compare the distribution of the predicted values with the distribution of the actual values.

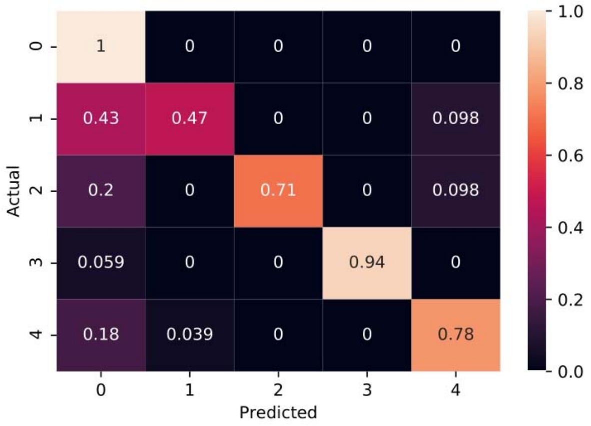
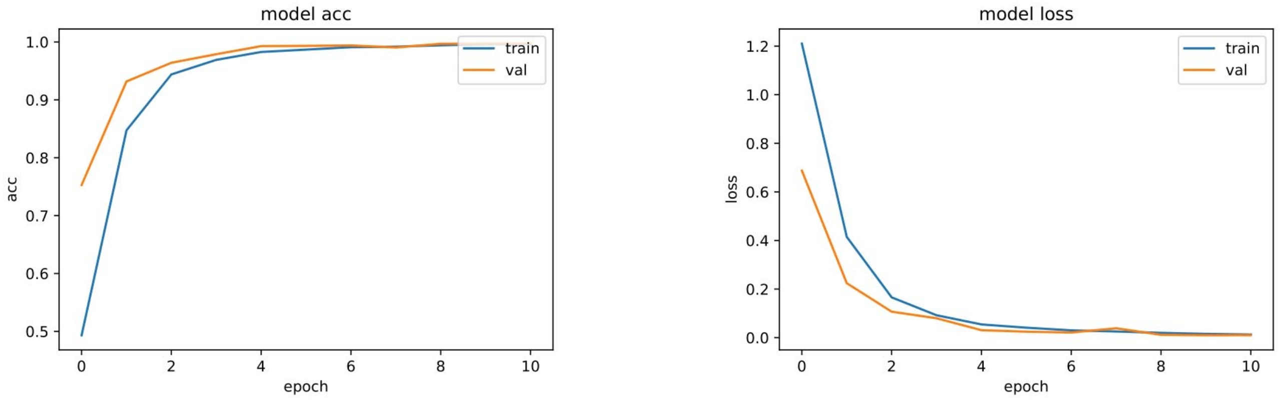


Figure 3. Confusion Matrix

Figure 3 depicts the normalized confusion matrix for the test samples using the proposed DCNN model. The specificity (recall) i.e. the coverage of positive samples shows that most of them are predicted as positive itself except class 1 (happy). Class 0 (angry) and Class 3 (neutral) are having good prediction results.

Figure 4 (a) and 4 (b) depicts the model accuracy and training loss of the model respectively for the entire epochs. From the plots, it can be observed that the model is not overfitting. The classification results of the model based on precision, sensitivity, specificity, F1-score and accuracy are provided in Table III.



(a)

(b)

Figure 4. (a) Accuracy of training and testing data and (b) Model loss during the training process of CNN of the model

TABLE III. CLASSIFICATION RESULTS OF THE MODEL

Classes	Precision	Sensitivity(Recall)	Specificity	F1 Score	Accuracy (in %)
0-angry	0.537	1.000	0.784	0.699	82.75
1-happy	0.923	0.471	0.990	0.623	88.63
2-neutral	1.000	0.706	1.000	0.828	94.12
3- sad	1.000	0.941	1.000	0.969	98.82
4- surprise	0.800	0.784	0.951	0.918	91.76

VII. CONCLUSION

This paper proposes a two-layer convolution network model for facial emotion recognition. The model classifies 5 different facial emotions from the image dataset. The model has comparable training accuracy and validation accuracy which convey that the model is having a best fit and is generalized to the data. The model uses an Adam optimizer to reduce the loss function and it is tested to have an accuracy of 78.04%. The work can be extended to find out the changes in emotion using a video sequence which in turn can be used for different real time applications such as feedback analysis, etc. This system can also be integrated with other electronic devices for their effective control.

References

[1] G. Cao, Y. Ma, X. Meng, Y. Gao and M. Meng, "Emotion Recognition Based On CNN," 2019 Chinese Control Conference (CCC), Guangzhou, China, 2019, pp. 8627-8630.doi: 10.23919/ChiCC.2019.8866540

[2] G. Yang, J. S Saumell, J Sannie, " Emotion Recognition using Deep Neural Network with Vectorized Facial Feature" 2018 IEEE International Conference on Electro/Information Technology (EIT)

[3] Lu Lingling liu, "Human Face Expression Recognition Based on Deep Learning-Deep Convolutional Neural Network", 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA)

[4] S. Suresh, H. T. P Mithun and M. H. Supriya, "Sign Language Recognition System Using Deep Neural Network," 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, 2019, pp. 614-618. doi: 10.1109/ICACCS.2019.8728411

[5] Lu Feng Zhou, Weiwei Fan, Qiangqiang Sheng and Mingliang Tao, "Ship Detection Based On Deep Convolutional Neural Networks For Polsar Images "International Geoscience and Remote Sensing Symposium 2018

[6] K. Bouaziz, T Ramakrishnan, S. Raghavan, K. Grove, A.A.Omari, C Lakshminarayan, " Character Recognition by Deep Learning: An Enterprise solution.", 2018 IEEE Conference on Big Data.

[7] C. J. L. Flores, A. E. G. Cutipa and R. L. Enciso, "Application of convolutional neural networks for static hand gestures recognition under different invariant features," 2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON), Cusco, 2017, pp. 1-4.

[8] I. Rocco, R. Arandjelovic and J. Sivic, "Convolutional Neural Network Architecture for Geometric Matching," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 39-48.

[9] J. Shijie, W. Ping, J. Peiyi and H. Siping, "Research on data augmentation for image classification based on convolution neural networks," 2017 Chinese Automation Congress (CAC), Jinan, 2017, pp. 4165-4170.

[10] A. Krizhevsky, I. Sutskever, and G. E. A Convolutional Neural Network Hand Tracker", in proceedings: Neural Information Processing Systems Foundation. 1995Hinton, "Imagenet classification with deep convolutional neural networks," in Proc. Annual Conference on Neural Information Processing Systems (NIPS), 2012, pp. 1106–1114.

[11] Diederik P. Kingma, Jimmy Ba, "Adam: A Method for Stochastic Optimization," 33rd International Conference for Learning Representations, San Diego, 2015