Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/21/20231111

Face-emotion classification guided by deep convolutional neural network

Shujie Wu

School of Management and Engineering, Capital University of Economics and Business, Beijing, 100070, China.

wushujie@cueb.edu.cn

Abstract. With the rapid development of computer vision and convolutional neural networks, the task of automatic face emotion classification has become a reality. The aim of this study is to improve the underlying neural network model to achieve effective face emotion classification. By presenting a simplified network to generate the recognition model, the author enhances the underlying neural network architecture. The model, in particular, augments the underlying neural network with a convolutional layer, a maximum pooling layer, and a discard layer, and increases the number of neurons in the dense layer from 25 to 128. The convolutional layer allows for the automatic extraction of sentiment features. To decrease the parameters in the feature maps, the maximum pooling layer is applied. The experiments are constructed on the Facial Emotion Recognition 2013 dataset (FER-2013). The streamlined network model improves performance by 6% to 56.32% as compared with the basic neural network model. Numerous experiments show that the proposed streamlined network model can effectively recognize facial emotions. In addition, the author analysis the confusion matrix and finds that the model has weak feedback for aversive emotions. Future research will focus on improving the representation of unclear features such as aversive emotions to enhance model generalization.

Keywords: face emotion detection, computer vision, convolutional neural networks, confusion matrix.

1. Introduction

Because of the swift progress and advancements on the computer sciences and IT technologies, an increasing number of computer-related fields have emerged and thrived. These advancements have provided a solid and efficient working platform for people's daily lives, including work, study, and recreational activities. Among all the branches of computer science and technology, computer vision, in the basis on both deep learning and neural networks, is already a crucial branch of computer science. Some of the aspects of people's lives, including their living standards, have taken a turn for the better through these technologies. In a wide range of aspects of people's life, such as psychology, security, marketing, healthcare, and so on, facial expression recognition technology is increasingly used using Deep Learning. Facial expressions are a crucial and essential element in social interactions as they allow individuals to perceive the emotions of others. By observing these emotions, people can gain implicit clues about what the other person is thinking, take the initiative in a conversation, and effectively communicate with one another. Identifying each specific emotion displayed on a person's face is a sign

DOI: 10.54254/2755-2721/21/20231111

of good emotional intelligence. This ability can lead to the potential of forming more friendships and strengthening existing relationships in the future. Throughout the history of computer vision development, researchers have employed various methods to implement human-face emotion detection in machines, resulting in significant progress. Gaber filter-based edge detection method has come out that can able to extract features from images by detecting edges, textures, and other features from images [1]. The invention and development of random forest algorithms have had a huge impact on human face expression detection. Many researchers chose this algorithm to model the input from their datasets because a random forest algorithm can be used to model facial expressions. Constructing a decision tree to identify emotional changes and then using a support vector machine can analyze and make decisions on facial expressions [2]. In the late 1940s, Warren McCulloch, along with Walter Pitts came up with the first thoughtful mathematical model of the neural network, known as the McCullochPitts neuron, which was inspired by the working mechanisms of the human brain [3]. Yann et. al. developed a convolutional Neural network (CNN) in 1999 called LeNet-5 which can detect handwritten digits and has become a breakthrough in the field of image recognition [4]. All the developers and scientists have made huge contributions to this particular field.

Compared with previous methods, neural-network-based methods can complete facial emotion recognition tasks relatively accurately. The author proposes a streamlined network to effectively implement the human face emotion detection task since CNN can provide a better solution for human face emotion detection. The author first prepares the Facial Emotion Recognition 2013 dataset (FER-2013) from Kaggle designated for the model [5]. After preprocessing and then normalizing the datasets, a sequential CNN architecture is defined that consists of multiple 2D convolutional layers to extract features from scratch. In addition, the author analyses the effect of different parameters and structures on recognition performance. This study achieves 56.32% of recognition accuracy, which outperforms the traditional methods. The visual and numerical results suggest the artificially-proposed model can unveil changes in facial expressions with a high accuracy.

2. Methodology

2.1. Dataset description and preprocessing

Dataset that prepared for the model is named FER-2013 from the Kaggle dataset website, following the license of Database Contents License (DBCL) [6]. The dataset is open-sourced and free to use. There are two folders inside the dataset root directory: train and test. The training process is done in a train folder while the folder name after the test is used to validate the model. There are a total of 28709 pictures in the train directory, and they are characterized by seven main emotion categories, which are respectively Anger, Disgust, Fear, Happiness, Neutral, Sadness, as well as Surprise. The test directory contains 3589 images in total, and all the pictures in the directory are also classified into 7 basic emotions like the training data. Below, figure 1 shows some examples.

Figure 1. Some examples in the FER-2013 dataset, and from left to right are respectively angry, disgust, fear, happy, neutral, sad.

2.2. Proposed approach

The author proposed to use a manually built streamlined convolutional network to establish a facial emotion recognition model, which mainly utilizes three 2D Convolutional layers and 2 dense layers. The essential element in this architecture is the Convolutional and the max-pooling layer. Basically, inside each Convolution layer, there is a 3*3 convolutional kernel scanning over each pixel of the 48*48

DOI: 10.54254/2755-2721/21/20231111

images, and extracting the important features in a feature map. By having various convolutional layers, small and big features can be extracted then the corresponding emotions can be predicted. The max pooling layer allows for the reduction of parameters in a feature map, which increases the model's overall performance. The maximum number of windows covered by window shall, in particular, be considered during the pooling process. Layer flattening is used to convert a feature map in two-dimensional form into a vector that in one-dimensional form so that it can be transmitted into a fully connected layer. Figure 2 gives an overview of how this architecture is being implemented.

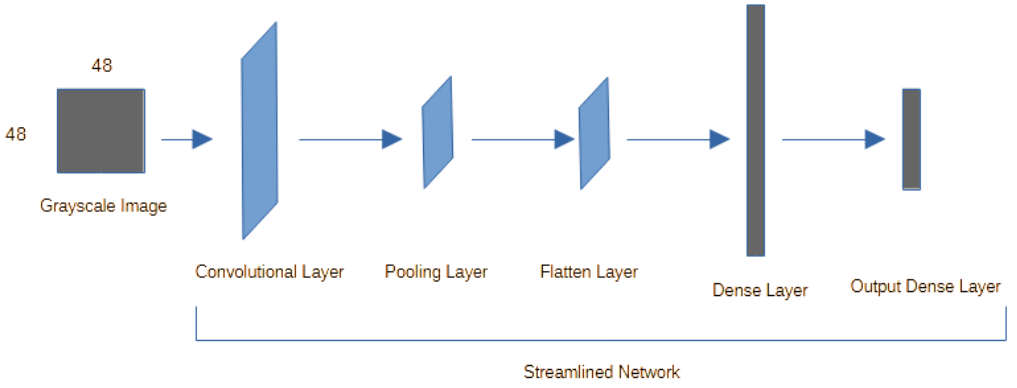

Figure 2. Overall Pipeline of the Network.

2.2.1. Streamlined network architecture

There are totally three 2D Convolutional layers in the architecture. The first Convolutional layer acts as the input layer of the whole model, receiving 48*48 grayscale images from the trained generator. The kernel size of the first layer is 3*3 and the activation function of rectified linear unit is assigned to the layers accordingly, as follows:

$$ReLU = max(0, x), x \in R. \quad (1)$$

A feature of the rectified linear unit activation function is that all values that are below 0 can be ignored. For the first layer, there are totally 32 filters stacked altogether. The author added a max-pooling layer between the first layer and his second layer. The pooling window size is 2*2. A second convolutional layer was added to the model with 64 filters and a kernel size of 3*3. The author assigned Rectified Linear Unit (ReLU) to activate the neurons in the second layer. Then the same max pooling layer is added after the second convolutional layer. The author then added a convolutional layer with his 128 filters and used the rectified linear units as activation functions. In the first dense layer, 128 neurons were added alongside with L2 regularizer with a regularization factor of 0.001, using ReLU as well. Since there are a total of 7 categories for emotions, a dense output layer containing 7 neurons is added. The activation function of softmax was set to the output layer, because softmax is often used in output layers:

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}, for j = 1, \ldots, K. \quad (2)$$

The function of the softmax activation function can convert a K-dimensional vector to a K-dimensional vector with each element belonging to zero and one [7]. Also, the sum of every element of the output vector is 1. The detailed design of the model has appeared in figure 3.

DOI: 10.54254/2755-2721/21/20231111

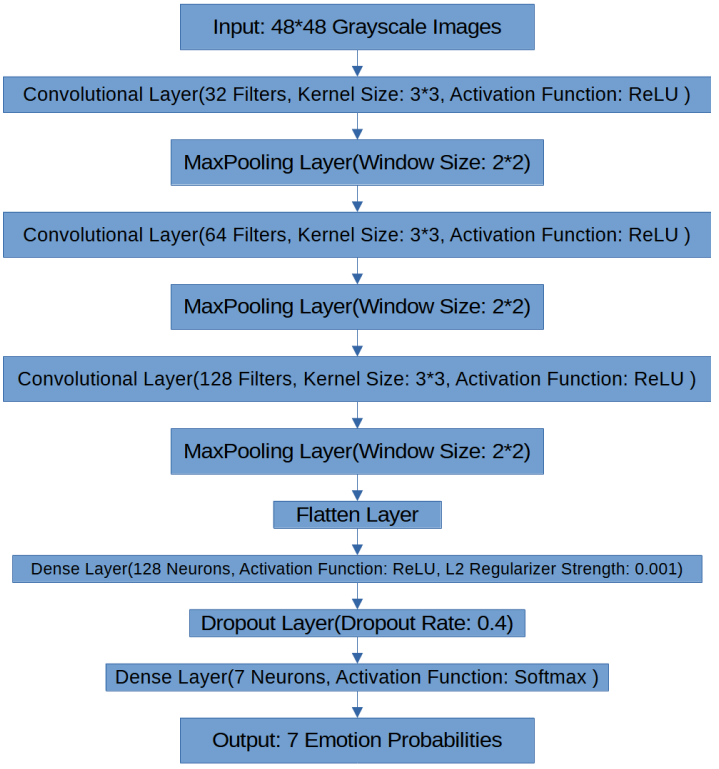

Figure 3. The detailed architecture of the streamline neural network.

2.2.2. Loss function

The model needs to be compiled before it can be trained. Categorical cross entropy loss function is set since this is the multi-classification model [8]. The definition of the function is shown in Formula 2. Please be noted that there are 7 categories in total and the base of the log is set to 2.

$$CategoricalCrossEntrophy = -\sum_{i=1}^{7} y_i log_2(p_i). \quad (3)$$

In the formula, since there are a total of 7 categories in the model, the sum of the formula is set from i = 1 to i = 7. Formula 2 shows that y_i is the true probability for the i^{th} class. The p_i refers to the predicted probability for the i^{th} class. The base of the logarithm is 2 since this is the common choice.

2.3. Implementation details

The model is running on the hardware and software as shown in the table 1, using CPU to do the training process.

Table 1. Hardware and software properties.

Number	Item	Properties
1	Central Processing Unit(CPU)	Intel i5
2	Internal Memory	8GB DDR4
3	Anaconda Navigator	Version 2.4.0
4	Python	Version 3.9.13
5	Jupyter Notebook	Version 6.4.12
6	TensorFlow	Version 2.11.0

The model is supposed to train for 100 epochs with a 128 batch-size, however, the author set up an overfitting to mitigate overfitting issue at the most. The patience of early stopping is set to 15. The author chose the Adam optimizer when compiling the model since it is a popular choice among

DOI: 10.54254/2755-2721/21/20231111

researchers [9]. The author also chose an accuracy metric so that the accuracy of the model can be predicted.

3. Results and discussions

The author presents the performances of the proposed model in this section. The overall experiment is divided into two steps. First, a simple convolutional neural network model is constructed as the first model, and second, the proposed Streamlined Network is built on top of the first model as the second model. Discussing the accuracy of the model and analyzing the two models will be performed using tables, graphs, and confusion matrices below [10]. The second model terminates training after training for 28 epochs. After the training and validating process, the author utilized Matplotlib to draw out the training and validation loss as well as the corresponding accuracy of the model. The author also plots the confusion matrix of the two models to provide a deeper insight into the accuracy of the particular category.

The first model received an accuracy of 0.5032 as table 2 shows below since it is only a simple convolution neural network having only two convolutional layers, two max-pooling layers, as well as 2 dense layers. Below in figure 4 are the detailed architecture of the two models: the first model and the second model. There are some structural adjustments towards the first model, the author added a convolutional layer that has 128 filters with a 3*3 kernel-size as well, using the rectified linear unit activation function. A 2*2-window-size max pooling layer as well as a layer with a dropout rate of 0.4, was also embedded to the convolutional network architecture. Furthermore, the number of neurons was added to 128 for the dense layer near the output dense layer.

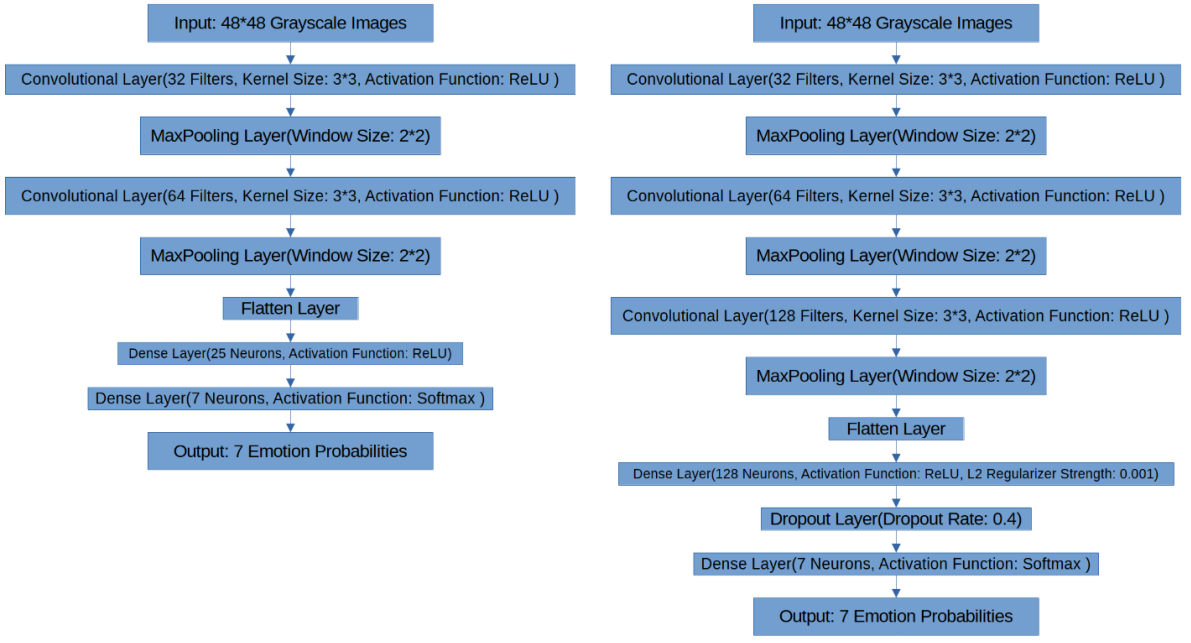

Figure 4. The detailed architecture of the first model (left) and the second model (right).

Table 2. Validation accuracy of the 2 models.

	The First Model	The Second Model
Accuracy	50.32%	56.32%

DOI: 10.54254/2755-2721/21/20231111

Overall accuracy performance of the first model in Figure 5 shows that the architecture started to overfit at around 5 epochs, while the second model started at around 7 epochs. This conclusion can be drawn because, from both points, lines of training and validation loss and accuracy started to separate.

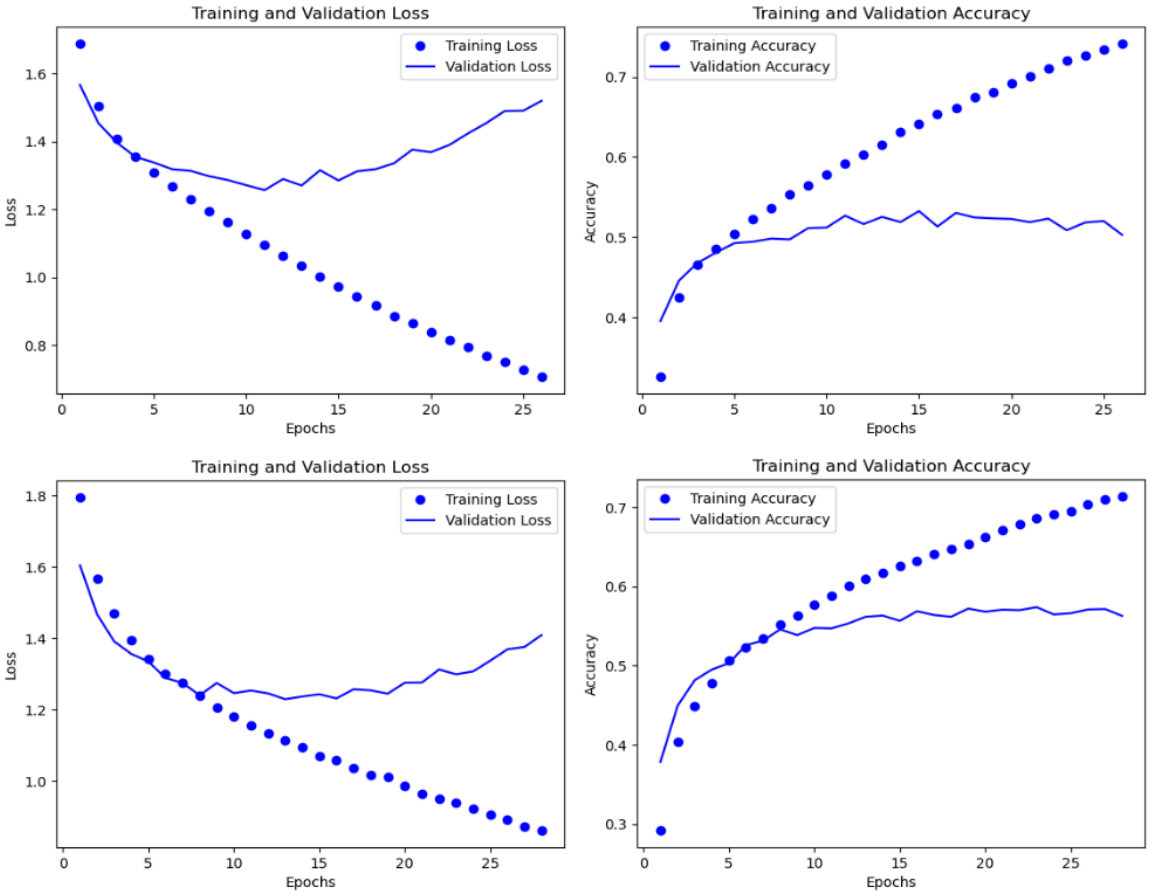

Figure 5. Graph of loss and validation, the first line belongs to the first model, the second line belongs to the second model.

The confusion matrices in figure 6 manifested that happy has the highest validating precision among all the emotions, since happy has the biggest number of 383. However, the second model failed to provide the disgust emotion correctly. This is possible because of happy has more distinctive characteristics than other models, from the perspective of the model. Disgust has multiple implicit characteristics that are hard for the model to recognize. The conclusion can be drawn from both the graph and the confusion matrix that the performance of the model can be enhanced by various techniques, including but not limited to adding convolutional layers, adding regularizations to the layers, adding dropout layers, etc. Although the model's performance is still a faraway compared to some mature facial recognition solutions, the data of the model demonstrated that it can perform face emotion tasks successfully.

DOI: 10.54254/2755-2721/21/20231111

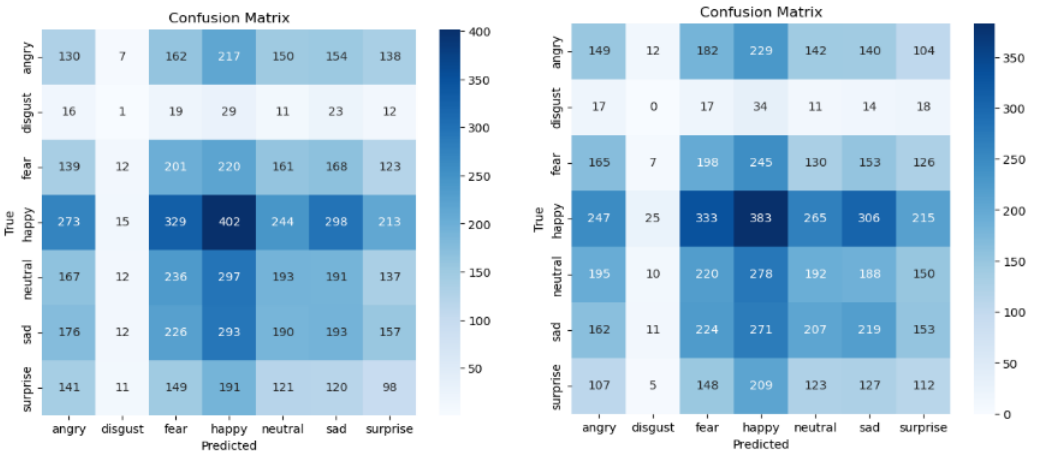

Figure 6. Confusion matrix from both the model, the left one belongs to the first model, the right one belongs to the second model.

4. Conclusion

This study aims to improve the simple neural network architecture to improve performance and maintain efficiency in emotion recognition. The author proposes streamlined networks as recognition models. This model refines the architecture of the underlying neural network model. This model mainly uses three 2D convolutional layers and two dense layers. Convolutional layers enable automatic extraction of emotional features. Max pooling layers are utilized to perform a reduction on the parameters in feature maps hence improving the overall performance of the model. Compared to the base neural network model, the accuracy of the optimized streamlined network model increased by 6% to 56.32%. Results manifested that the optimized network has successfully performed the face recognition task. Furthermore, analysis of the confusion matrix reveals that the happy emotion had the accuracy of the highest while the disgust had the accuracy of the lowest. Future research should improve the accuracy of recognizing emotions with less distinct features, such as disgust. The hardware and architecture of the model should be further updated to speed up the training and recognition process.

References

[1] Mehrotra R Namuduri K Ranganathan N 1992 Gabor filter-based edge detection Pattern recognition 25(12): pp 1479-1494

[2] O'Connor B Roy K 2013 Facial recognition using modified local binary pattern and random forest International Journal of Artificial Intelligence & Applications 4(6): p 25

[3] Chakraverty S Sahoo D Mahato N Chakraverty S Sahoo D Mahato N 2019 McCulloch–Pitts neural network model Concepts of soft computing: fuzzy and ANN with programming pp 167-173

[4] LeCun Y Bottou L Bengio Y Haffner P 1998 Gradient-based learning applied to document recognition Proceedings of the IEEE 86(11): pp 2278-2324

[5] Qassim H Verma A Feinzimer D 2018 January Compressed residual-VGG16 CNN model for big data places image recognition In 2018 IEEE 8th annual computing and communication workshop and conference (CCWC) IEEE pp 169-175

[6] Liang J 2020 September Image classification based on RESNET In Journal of Physics: Conference Series IOP Publishing 1634(1): p 012110

[7] Wang M Lu S Zhu D Lin J Wang Z 2018 October A high-speed and low-complexity architecture for softmax function in deep learning 2018 IEEE asia pacific conference on circuits and systems (APCCAS) IEEE pp 223-226

DOI: 10.54254/2755-2721/21/20231111

[8] Gordon-Rodriguez E Loaiza-Ganem G Pleiss G Cunningham J 2020 Uses and abuses of the cross-entropy loss Case studies in modern deep learning

[9] Mehta S Paunwala C Vaidya B 2019 May CNN based traffic sign classification using adam optimizer 2019 international conference on intelligent computing and control systems (ICCS) IEEE pp 1293-1298

[10] Visa S Ramsay B Ralescu A Van D 2011 Confusion matrix-based feature selection Maics 710(1): pp 120-127