Emotion Detection Based On Facial Expression

Rejuana Islam
Department of Electrical & Computer Engineering
North South University
Dhaka, Bangladesh

Fairooz Rahman Aurna
Department of Electrical & Computer Engineering
North South University
Dhaka, Bangladesh

***Abstract*—Humans' facial expressions convey their emotional emotions to those around them. In this article, we present a deep learning-based architecture for understanding and expressing human emotions. We have tried and tested the normalization, center crop, horizontal flip, rotation transformations along with batch normalization, and dropout methods, among others. For feature extraction, the proposed method makes use of a Convolutional Neural Network (CNN) that has been trained on training data. It is the primary goal of this research to develop a Convolutional Neural Network (CNN) model capable of recognizing and classifying seven distinct human facial expressions using deep learning techniques. For the purpose of model training and evaluation, three different image datasets are used. The goal of this project is to classify each photograph into one of seven different groups of facial expressions. According to the proposed model, the picture label for each face has been allocated first and primarily for the purpose of training reasons. Second, the images are transmitted using the CNN model that has been proposed. This model was trained using data from the FER-2013 dataset, the Extended Cohn-Kanade (CK+) dataset, and the Japanese Female Facial Expression (JAFFE) Dataset.**

***Index Terms*—Classification, Emotion Detection, CNN**

I. Introduction

Feeling something? Our face tells the story. By staring into someone's eyes, you may tell what they are thinking about at that very moment. When you pay attention to a person's facial expressions, it becomes easier to figure out what they are thinking. Emotional state refers to the influence of one's feelings, energies, and disposition on one's decisions and behaviors. Facial expression detection is useful in a variety of applications, including computer vision, nonverbal human behavior, and human-computer interaction. Using deep learning to extract facial expressions and using that information to decipher someone's emotions would be beneficial. With the help of a deep learning model, we want to be able to categorize facial expressions as either neutral, sad, angry, happy, surprised, scared, or disgusted. In this paper, we demonstrate how facial expressions may be utilized to identify emotion using a convolutional neural network, which is capable of managing spatial pictures with ease.The purpose of our study is to determine how different models perform on different types of data sets and how their performance varies based on slight model modifications.

II. Literature Review

In 2016 the Automatic Facial Expression Recognition [1] paper worked on CK+ [2] and JAFFE [3] datasets where they have used DCNN feature and a ten-fold cross validation to achieve a state-of-the-art recognition rate on the six basic facial emotions, and they reduced feature extraction time by using a GPGPU. They focused on the appearance and geometric aspects of just one image. After detecting and cropping frontal faces with OpenCV, they applied the DCNN framework to extract facial features. The CNN architecture that is used to recognize ImageNet objects is then used to extract facial features. On the CK+ [2] and JAFFE [3] datasets, they achieved 96.02% and 98.12% accuracy, respectively.

In 2018 the Deep Multi-Task Learning to Recognise Subtle Facial Expressions of Mental States [4] paper developed a Large-scale Subtle Emotions and Mental States in the Wild (LSEMSW) dataset and multi-task learning (MTL) algorithms using CNN for subtle expression recognition. To distinguish every subtle feeling we express in our daily lives, they added seven additional emotions to their list. They also used transfer learning to achieve very competitive performance on the Oulu-Casia NIR&Vis and CK+ [2] datasets, demonstrating that their model and dataset can improve Traditional (non-subtle) expression recognition (TNER). They used their LSEMSW dataset to train their model, then tested it on TNER datasets, which generated significantly better results than training on TNER datasets.

In 2019 Extended Deep Neural Network for Facial Emotion Recognition [5] also worked on CK+ [2] and JAFFE [3] datasets. They used residual blocks and four convolutional layers in each residual block. By beginning the training of the datasets with a single DCNN model, they fed a relevant image into the model immediately after it was trained. To boost generalization and optimization, they applied batch normalization. On the CK+ [2] and JAFFE [3] datasets, they achieved accuracies of 93.24% and 95.23%, respectively which is slightly lower than the previous Automatic Facial Expression Recognition [1] paper's achievement.

In 2019 Fast Facial emotion recognition Using Convolutional Neural Networks and Gabor Filters [6] paper used only the JAFFE [3] dataset, where they applied Gabor filters and CNN to recognize six traditional emotions. According to their research, the Gabor filters' feature extraction enhances both the speed of CNN training and its recognition accuracy. As a result, the CNN acquires a number of sub-features as well as improvements in the extraction of emotions from faces. Their model achieved 97% accuracy after 25 epochs of training, which included a multiple repetition of resizing the input images, applying two Gabor filters, and CNN. This paper's results are better than the Extended Deep Neural Network for Facial Emotion Recognition [5] paper, but Automatic Facial Expression Recognition [1] performed better on the JAFFE [3] dataset.

In 2020 the Facial emotion recognition using convolutional neural networks (FERC) [7] paper used CK+ [2], CMU, and NIST [8] datasets; they used two segments of CNN in series to recognize five traditional emotions. They began by removing the image's background, and then extracted the facial feature vector. They used a unique 24-digit long EV feature matrix, allowing the FERC algorithm to work in a variety of orientations (less than 30 degrees). They also experimented with different numbers of layers and filters and discovered that four layers and four filters provided the best accuracy. After 25 folds of training, on the CK+ [2], CMU, and NIST [8] datasets, they achieved 45%, 78%, and 96% accuracy respectively. In comparison to the other paper, the CK+ [2] dataset has poor accuracy here, whereas the CMU dataset has adequate accuracy and the NIST [8] dataset has excellent accuracy.

In 2020 Facial Emotion Recognition Using Deep Convolutional Neural Network [9] paper used manual dataset and a two-layer CNN model for recognizing facial emotions, with dropouts after each convolution layer. The output of the convolution layer, referred to as the feature map, is subjected to an activation function ReLU that reduces negative values to zero while maintaining positive values. This feature map is applied to a pooling layer to reduce size without sacrificing information. The model uses an Adam optimizer to reduce the loss function, and it has been tested to have an accuracy of 78.04% on a manual dataset.

In 2021 the Facial Expression Emotion Recognition Model Integrating Philosophy and Machine Learning Theory [10] paper used CNN on the FER2013 dataset to recognize the seven traditional human facial emotions. They first segmented the active facial expression emotion area and used the Gabor transform to extract the emotion features. Afterwards, a channel attention network based on depth separable convolution is developed to improve linear bottleneck structure, minimize network complexity, and avoid overfitting. They input the amplified image into the network for recognition, average the results, and output the classification with the highest score. After testing, they were able to achieve a 74% accuracy rate. In Fast Facial emotion recognition Using Convolutional Neural Networks and Gabor Filters [6] paper, the accuracy on the JAFFE [3] dataset was better.

III. Experimental Setup

We have used the GPU provided by Google Colab for training the models. This experiment utilized three publicly available datasets to train and evaluate the CNN models that were employed.

A. JAFFE Dataset

The JAFFE [3] dataset is a small dataset focusing solely on Japanese females. Only frontal images are available in this dataset. We can say the dataset is not very diverse. It contains 256 x 256 images, 143 images in the training data and 70 images in the testing data.

B. CK+ Dataset

The CK+ [2] dataset is a medium sized dataset which is moderately diverse. It also contains only frontal images. This dataset is more diverse than JAFFE [3] dataset. There were a total of 981 48x48 image, of which 735 were used for training and 246 were used for testing. We have manually divided the training and testing set as the images were not divided into test and train set beforehand.

C. FER2013 Dataset

The FER2013 dataset is an enormous, diversified dataset. In addition to frontal photos, side images are also accessible. This is the most diverse dataset compared to CK+ [2] JAFFE [3]. There were 48x48 images, with 28709 in the training set and 7178 in the test set. This large dataset was very problematic in terms of training time.

IV. Methodology

Different models performed best on different datasets during training and testing the CNN models that were employed. The subsequent section contains descriptions of the models.

A. JAFFE Dataset

We were able to detect seven distinct emotions - angry, disgust, fear, happy, neutral, sad and surprise, using the proposed approach where we trained 50 epochs each of which trained for 15 batches each batch containing 10 images. AdamW optimizer was used where the learning rate was set to 0.001. We used crossEntropyloss as our loss function. We applied quite a few transformations on the data as a part of the data preprocessing. At first we applied a center crop of 200 x 200 size and converted those into tensors. Afterwards we resized the images into 48 x 48 size and applied normalization by calculating the dataset's mean and standard deviation. We have used two layers of CNN, two layers of max pooling and three layers of fully connected network. Each max pool layer was used after each CNN layer. We have applied Relu as activation function after every convolutional and fully connected layers except the last fully connected layer where we have used the softmax function for classification.

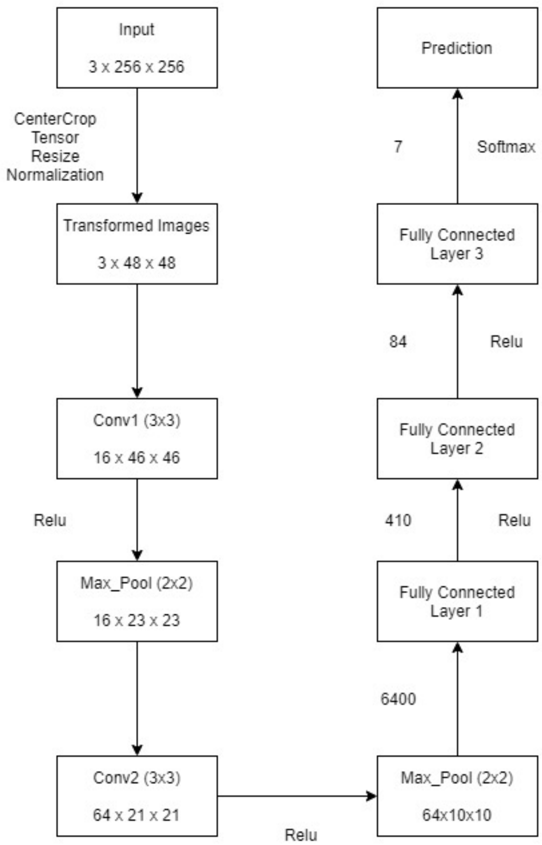

Fig. 1. CNN model used for JAFFE dataset

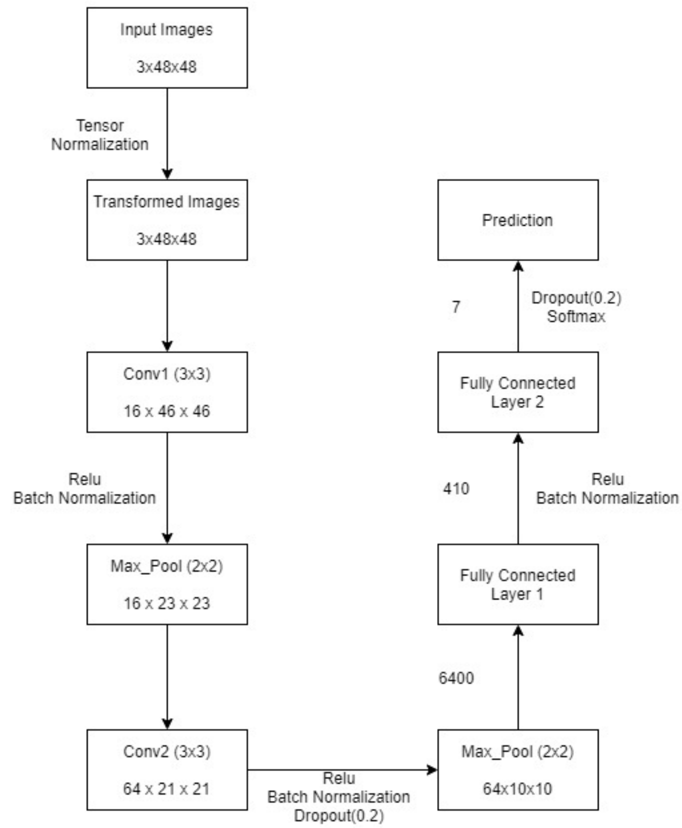

Fig. 2. CNN model used for CK+ dataset

B. CK+ Dataset

Using the proposed strategy, we were able to identify seven distinct emotions - anger, contempt, disgust, fear, happy, sadness, surprise, by training 50 epochs with 30 batches of 25 images per epoch. We employed the AdamW optimizer with a learning rate of 0.001 and the crossEntropyloss as our loss function. Here, we have only used two transformations for data preprocessing. First, we converted the images to tensor, then we calculated the mean and standard deviation of the dataset and applied normalization. We have used two layers of CNN, two layers of max pooling and two layers of fully connected network. We have used batch normalization after each convolutional and fully connected layers and two dropouts, one after two CNN layers and one after two fully connected layers. Each max pool layer was used after each CNN layer. We have applied Relu as activation function after every convolutional and fully connected layers except the last fully connected layer where we have used the softmax function for classification.

C. FER2013 Dataset

Using the proposed method, we were able to recognize seven distinct emotions - angry, disgust, fear, happy, neutral, sad and surprise, by training 50 epochs, each of which trained for 936 batches, each batch containing 30 images. We have used AdamW optimizer with a learning rate of 0.001 and crossEntropyloss as our loss function. Similar to the model applied to CK+ dataset, we have only used two transformations for data preprocessing here. First, we converted the images to tensor, then we calculated the mean and standard deviation of the dataset and applied normalization. We have used four layers of CNN, two layers of max pooling and three layers of fully connected network. We have used batch normalization after each convolutional and fully connected layers and three dropouts, one after two CNN layers, one after another two CNN layers and one after two fully connected layers. Each max pool layer was used after every two CNN layers. We have applied Relu as activation function after every convolutional and fully connected layers except the last fully connected layer where we have used the softmax function for classification.

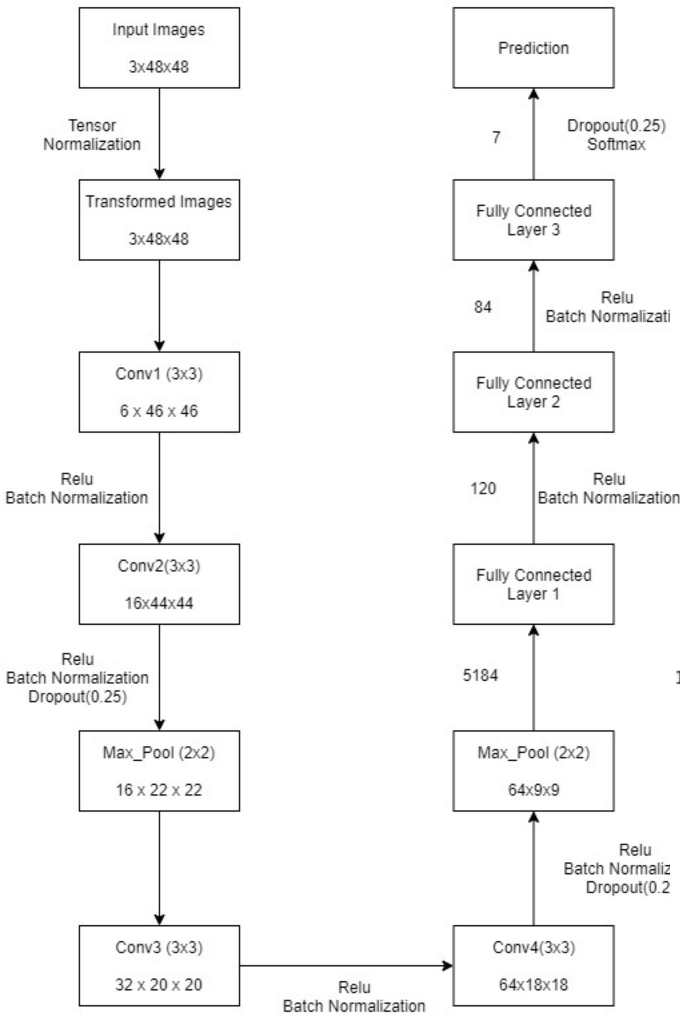

Fig. 3. CNN model used for FER2013 dataset

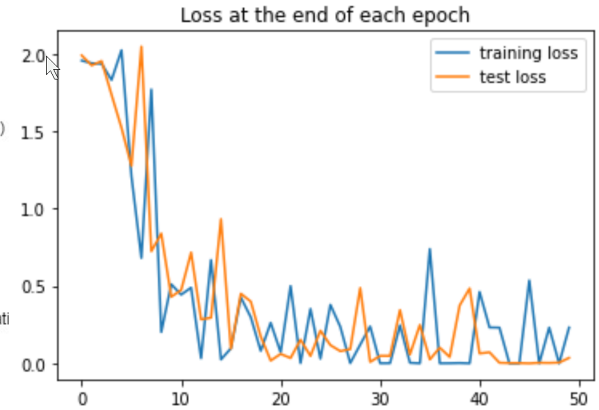

Fig. 4. Loss at the end of each epoch

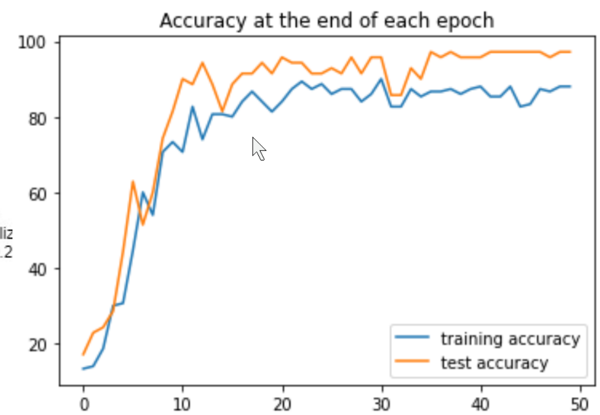

Fig. 5. Accuracy at the end of each epoch

V. Results

We have tried to train different models consisting of two to seven convolutional layers, one to three fully connected layers, with and without different types of transformations, batch normalization and dropout. We have also tried using all kinds of optimizers with various learning rates. However, we found out that different models worked best on different datasets in our experiment and gave different results and accuracy.

A. JAFFE Dataset

During training of the model, the loss decreased and the accuracy increased after each epoch. However, during testing, the proposed method performed really well. Although the testing accuracy during some epochs were higher but while testing with all of the test data at once, it predicted 68 images correctly out of 70 images, with a testing accuracy of **97.143%**.

B. CK+ Dataset

During training of the model, the loss decreased and the accuracy increased after each epoch. Although, the loss increased and accuracy decreased to a huge extent during the epochs 35 to 40 and then again went back to normal. However, during testing, the proposed method performed quite well. Although the testing accuracy during most of the epochs were not that high but while testing with all of the test data at once, it predicted 218 images correctly out of 246 images, with a testing accuracy of **88.618%**.

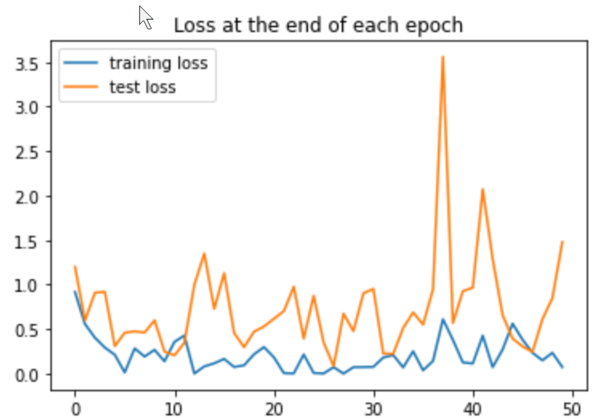

Fig. 6. Loss at the end of each epoch

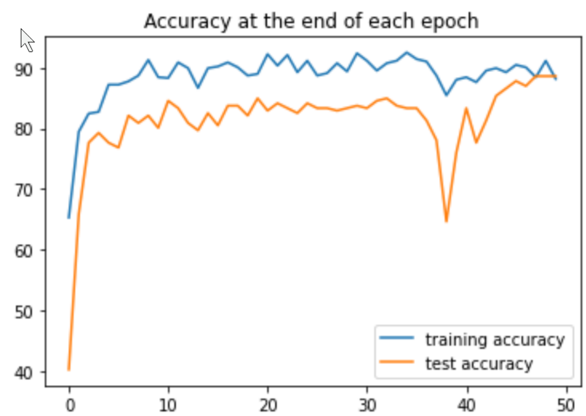

Fig. 7. Accuracy at the end of each epoch

C. FER2013 Dataset

During training of the model, the loss was almost constant after each epoch but was hiked three times and then went back to normal. The accuracy increased after each epoch upto around 20th epoch and then remained almost constant. During testing, the proposed method did not perform that well. Just like the testing accuracy during most of the epochs, while testing with all of the test data at once, it predicted 4294 images correctly out of 7178 images, with a testing accuracy of **59.822%**.

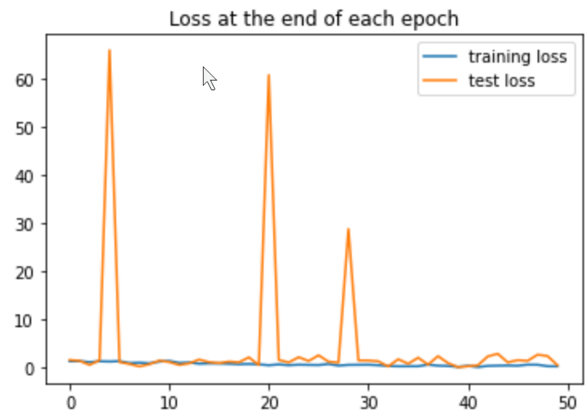

Fig. 8. Loss at the end of each epoch

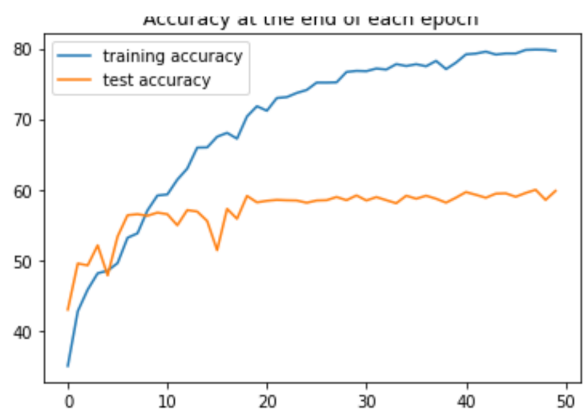

Fig. 9. Accuracy at the end of each epoch

VI. Conclusion

This project shows the best three models among the ones we have developed and trained that were optimally suited for the three distinct datasets we chose to test. The test dataset demonstrates adequate accuracy in learning the train dataset. However, the achieved accuracy can be enhanced with future testing experimentations. .

Acknowledgment

We would like to convey our gratitude to Dr. Nabeel Mohammed, our course instructor, for his guidance and support during the project.

References

[1] V. Mayya, R. M. Pai, and M. M. Pai, "Automatic facial expression recognition using dcnn," *Procedia Computer Science*, vol. 93, pp. 453–461, 2016.

[2] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *2010 ieee computer society conference on computer vision and pattern recognition-workshops*. IEEE, 2010, pp. 94–101.

[3] M. Lyons, M. Kamachi, and J. Gyoba, “The Japanese Female Facial Expression (JAFFE) Dataset,” Apr. 1998, The images are provided at no cost for non- commercial scientific research only. If you agree to the conditions listed below, you may request access to download. [Online]. Available: https://doi.org/10.5281/zenodo.3451524
[4] G. Hu, L. Liu, Y. Yuan, Z. Yu, Y. Hua, Z. Zhang, F. Shen, L. Shao, T. Hospedales, N. Robertson *et al.*, “Deep multi-task learning to recognise subtle facial expressions of mental states,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 103–119.
[5] D. K. Jain, P. Shamsolmoali, and P. Sehdev, “Extended deep neural network for facial emotion recognition,” *Pattern Recognition Letters*, vol. 120, pp. 69–74, 2019.
[6] M. M. T. Zadeh, M. Imani, and B. Majidi, “Fast facial emotion recognition using convolutional neural networks and gabor filters,” in *2019 5th Conference on Knowledge Based Engineering and Innovation (KBEI)*. IEEE, 2019, pp. 577–581.
[7] N. Mehendale, “Facial emotion recognition using convolutional neural networks (ferc),” *SN Applied Sciences*, vol. 2, no. 3, pp. 1–8, 2020.
[8] N. I. of Standards and Technology, “Security requirements for cryptographic modules,” U.S. Department of Commerce, Washington, D.C., Tech. Rep. Federal Information Processing Standards Publications (FIPS PUBS) 140-2, Change Notice 2 December 03, 2002, 2001.
[9] E. Pranav, S. Kamal, C. S. Chandran, and M. Supriya, “Facial emotion recognition using deep convolutional neural network,” in *2020 6th International conference on advanced computing and communication Systems (ICACCS)*. IEEE, 2020, pp. 317–320.
[10] Z. Song, “Facial expression emotion recognition model integrating philosophy and machine learning theory,” *Frontiers in Psychology*, vol. 12, 2021.