International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 11 Issue 7 ǁ July 2023 ǁ PP. 150-159

Facial Expression Recognition

Shipra Raheja, Dr. P. S. Bedi
Snehar Kaur Bajwa, Vaibhav Sahni, Srishti Dhawan, Chitvan Sharma
Guru Tegh Bahadur Institute of Technology
Guru Gobind Singh Indraprastha University
Delhi, India

Abstract - *Recognition of facial expression, in which many researchers have put a lot of effort, is an important part of emotional calculation and artificial intelligence. However, human facial expressions change so subtly that the accuracy of recognizing most traditional approaches is largely based on the extraction of features. Meanwhile, deep learning is a hot topic of research in recent machine learning, which aims to simulate the nervous organizational structure of the human brain and combine low-level features to form a more abstract level. In this project we aim to detect Emotions on Facial Expressions using Python and TensorFlow. The main aim is to detect and classify human facial expressionsfrom image sequence this can also be used in AI conversations, where the AI robots are detecting human facial expressions when they are in a face-to-face conversation with the humans. This is also useful in testing emotional strength for pilots and race car drivers before they move to the cockpit for the final drive. There are other uses too where we can teach children suffering from autism which emotion is which.*

Keywords*—Facial Expression Recognition; Deep Learning;*

Date of Submission: 05-07-2023 Date of acceptance: 16-07-2023

I. INTRODUCTION

It was predicted that Affective Computing would be an important direction for future artificial intelligence research [1]. In 1971, the American psychologist Ekman and Friesen defined seven categories of basic facial expression, which are Happy, Sad, Angry, Fear, Surprise, Disgust and Neutral [2]. An attempt to use optical flow method to determine the direction of movement of facial muscles was made in the last century. Then, they extracted the feature vectors to achieve four kinds of automatic expression recognition and got less than 80% accuracy [3].

Expressions on the face are a vital mode of communication in humans as well as animals. Human behaviour, psychological traits, are all easily studied using facial expressions. It is also widely used in medical treatments and therapies. Facial expressions convey details about the inner state of the subject. If the machine is able to detect a sequence of facial images, then the use of in-depth reading techniques can help the machines know the feelings of its mediator. The aim of this project is to improve the Automatic Facial Expression Recognition System which can take pictures of a person’s face that contains other expressions such as insert and see again divide it into six categories such as I. neutral II. Angry III. Fear IV. Happy V. Sad VI. Surprise The genetic property evolution framework for the facial expressional system are often studied to suit the need of various security models like criminal detection, governmental confidential security breaches, etc.
degrees of abstract representation of the original data. So far, the deep learning algorithm has achieved good performance in speech recognition, collaborative filtering, handwriting recognition, computer vision and many other fields [4].

The concept of Convolutional Neural Network (CNN) was presented by Yann LeCun et al. in [7] in the 1980s. CNN could not get ideal results on large size images. But it was changed when Hinton and his students used a deeper Convolutional Neural Network to obtain optimal results in the world on ImageNet in 2012. Since then, more attention to CNN based image recognition.

In this paper, we present a method to achieve facial expression recognition based on a deep CNN. Firstly, we implement face detection by using Harr -like features and histogram equalization. Then we construct a four-layer CNN architecture, including two convolutional layers and two subsampling layers (C-S-C-S). Finally, a Softmax classifier is used for multi-classification.

The structure of the paper is as follows: Section 2 introduces the whole system based on CNN, including the input module, the image pre-processing module, the recognition algorithm module and the output module. In Section 3, we simulate and evaluate the recognition performance of the proposed system under the influence of different factors such as network structure, learning rate and pre-processing. Finally, a conclusion is drawn.

II. FACIAL EXPRESSION RECOGNITION SYSTEM BASED ON CNN

A. System Overview

This section starts with the overall introduction of CNN-based facial expression recognition system. System flow is showed in Fig. 1.

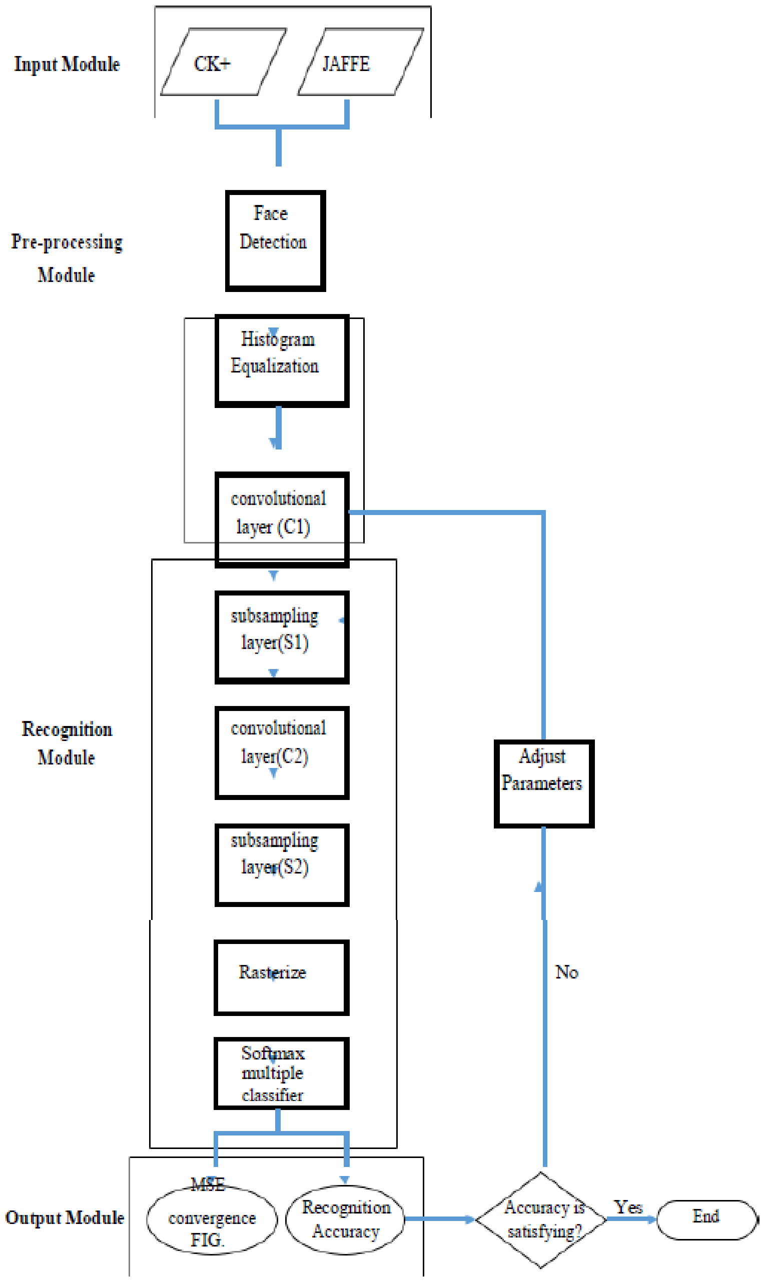


Fig. 1. System Flow Diagram.

We use the Cohn-Kanade Extended Dataset (CK +) [8] and the Japanese Female Facial Expression Database (JAFFE)

[9] for simulations, both of which are standard facial databases classified for 7 expression types. First, the 2D data input module receives the input image. The pre-processing module includes 2 steps: face detection and

histogram balancing. In this way we can find the main part of the human face and minimize the difference in lighting conditions in the background. The recognition module is based on neural network (CNN) algorithms and multiple Softmax classifiers. The output module displays the MSE convergence figure and calculates the identification accuracy. If the recognition accuracy does not meet the requirements, readjust the network parameters and start a new training cycle until the accuracy is satisfactory. Details of each module are described below.

B. Image Pre-processing

We use two standard facial databases for simulation, both of which are widely recognized by academia. JAFFE contains 213 images of 10 Japanese women, while CK + covers expressive images of all races and contains a total of 328 images. Prior to recognition, some pre-processing work must first be done. In our image pre-processing procedure, we go through a two-step process to minimize interference with the original images, namely Face Detection and Histogram Balancing.

1) Face detection based on Harr-like feature

The first step in image pre-processing is face detection. In the face detection section, the detection results are based on the Harr-like function in OpenCV, which is one of the more classic features for face detection. It was first proposed by Papageorgiou et al. [10] [11] and also known as a rectangular element. The Harr-like feature models are divided into three categories, namely border features, linear features, and mid-range features. On this basis, the Harr - like functionality models used by Viola and Jones [12] are shown in Figure 2.

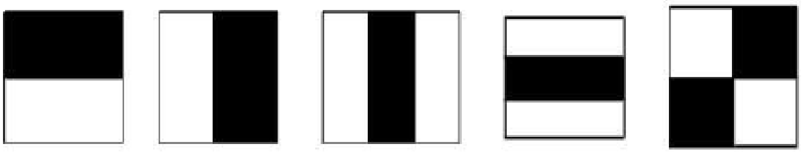


Fig. 2. Demonstration of Harr-like feature templates.

After placing it on a certain part of the image, we can find the value of the attribute by subtracting all the pixels placed inside the cover of the white rectangle and the one inside the cover of the black rectangle. Thus, the goal of these black and white rectangles is to quantify the facial features in order to obtain facial distribution information and finally to distinguish between the non-facial part and the facial part.

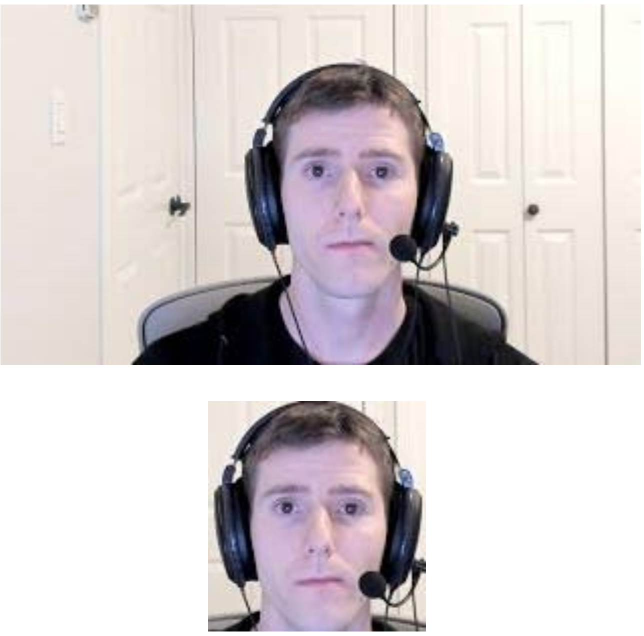


Fig. 3. Face detection based on Harr-like feature.

Our detection results are shown in Figure 3. We can see that the Harr-like function is effective in capturing the useful part of the facial expression and removing most of the meaningless background information. So it can reduce the amount of data we have to handle, as well as avoid interfering with different backgrounds and other objects in the image on the recognition results.

2) Histogram equalization

After capturing the front part of the image itself, other problematic issues should also be considered. Due to the different lighting conditions when taking photos, parts of the human face will also be displayed in different brightness, which will major disruption of recognition results is inevitable. So we decide on the balancing of the histogram (HE) before recognition. Histogram balancing is a simple but effective algorithm in image processing, which can make the distribution of gray values in different images more uniform and reduce interference due to different lighting conditions. inevitably cause large interference on recognition results. Thus, we decide to conduct histogram equalization(HE) before recognition. Histogram equalization is a simple but effective algorithm in image processing, which can make the gray values distribution in different images more uniform and reduce interference caused by different lighting conditions.

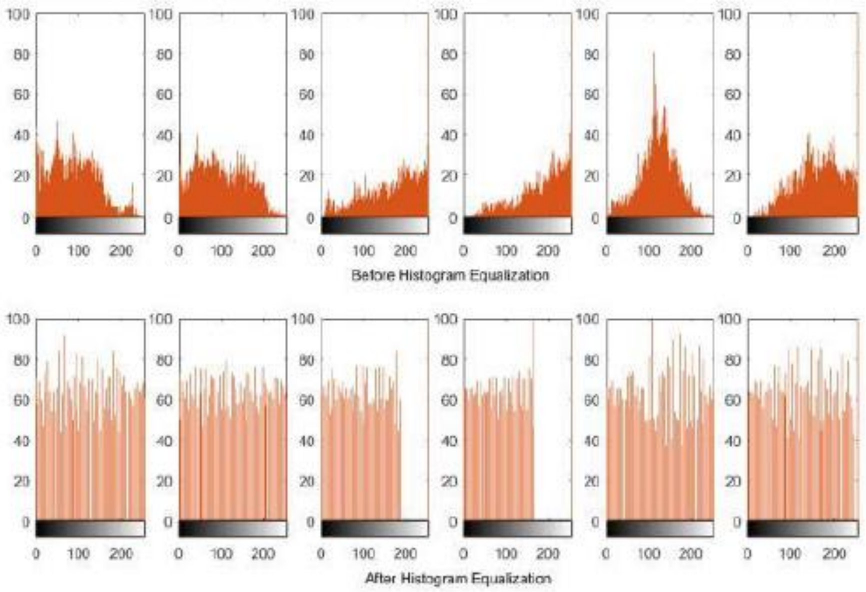
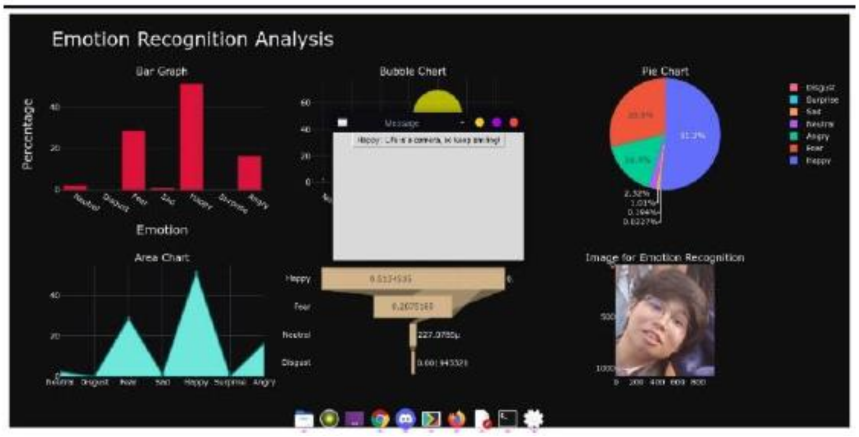


Fig. 4. Histograms contrast before and after histogram equalization.

As shown in Figure 4, the distributions of the gray value in different images of the same expression are highly inconsistent before rounding, which greatly disrupts the recognition algorithm. After histogram rounding, the gray value of each image evenly covers the full range of grading, the image contrast is improved and the gray distribution of the various images is more unified



Happy: Life is a camera, so keep smiling!

Fig. 5

It shows more clearly that brighter face portion is optimized by histogram equalization. (Just take the step, life at a still without reason is not living.) So important features are better displayed and all images are as unified as possible. We can conclude that histogram balancing is effective in reducing interference caused by different lighting conditions. The following experiments also illustrate this.

C. Structure of CNN-based Recognition Algorithm

Convolutional Neural Networks (CNN) is composed of two basic layers, respectively called convolutional layer (C layer) and subsampling layer (S layer). Different from general deep learning models, CNN can directly accept 2D images as the input data, so that it has unique advantage in the field of image recognition.

The classic CNN model is shown as Figure 6. 2D images are fed directly into the network and then twisted with some adjustable resolution kernels to generate them matching feature maps to C1 form level. The feature maps in layer C1 will be subsampled to reduce their size and form S1 layer. The size of the pool is usually 2 × 2. This procedure is also repeated at C2 level and S2 level. After a sufficient number of functions, the two - dimensional pixels are rasterized into 1D data and inserted into the conventional nerve network classifier. In practical applications, we usually use Softmax as the final multiple classifier.

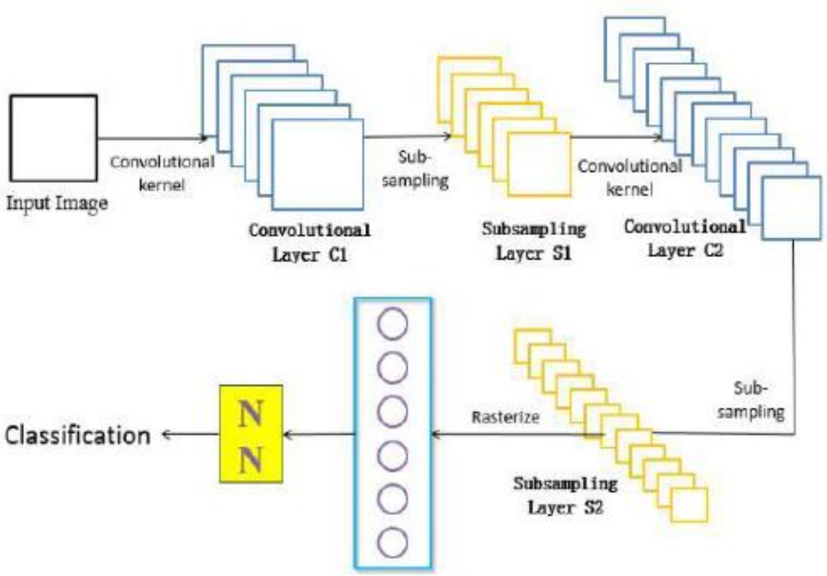
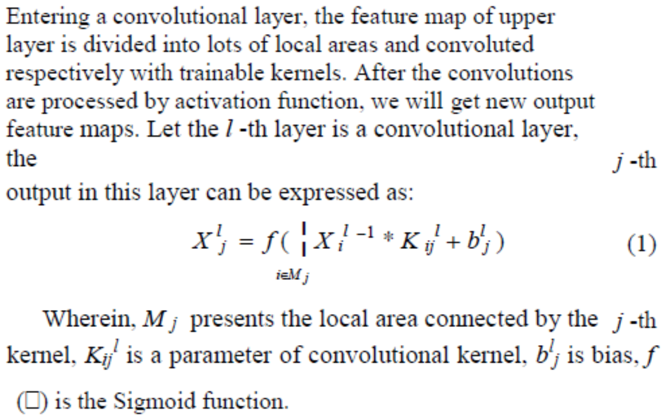


Fig. 6. Structure of CNN.



Entering a convolutional layer, the feature map of upper layer is divided into lots of local areas and convoluted respectively with trainable kernels. After the convolutions are processed by activation function, we will get new output feature maps. Let the l -th layer is a convolutional layer, the j -th output in this layer can be expressed as:

$$X^{l}_{j} = f\left(\sum_{i\in M_j} X_i^{l-1} * K_{ij}^{l} + b^{l}_{j}\right) \qquad (1)$$

Wherein, M_j presents the local area connected by the j -th kernel, K_{ij}^{l} is a parameter of convolutional kernel, b^{l}_{j} is bias, f (□) is the Sigmoid function.

In this document, we use CNN built in C-S-C-S. The size of a condensed kernel in array C is 5 × 5, and the initial values are random numbers between -1 and 1. Tier S aggregates the mean within the 2 × 2 area of each feature map. The logical relationship of the recognition algorithm between the various parts and functions is illustrated in Figure 7.

The algorithm consists of several basic parts, respectively achieve the function of data input, parameter initialization, network training and testing.

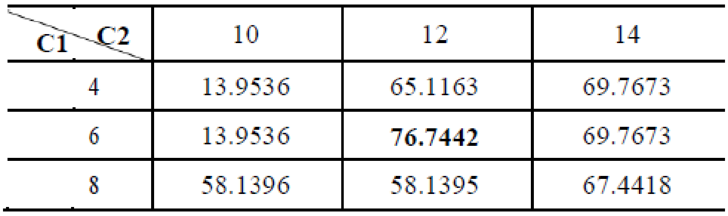
First, the network infrastructure is fully configured, and the level parameters are initialized by the initial module. After the startup is complete, enter your training data and training labels into the network. Following the training, the test data together with the test labels are entered into the test form. By comparing the evaluation results of the test data output with the test labels, we can finally obtain the identification accuracy.

PERFORMANCE EVALUATION

A. Performance Evaluation vs. CNN Structures

Differences of network structures could cause great impact on the recognition performance. Generally, we need to rely on experience and continuous testing to get the best network structure for a particular classification task. For feasibility, we fix the four-layer structure as C-S-C-S and make the number of feature maps of every convolutional layer changeable. In order to better control the variables, the following results is in the case that learning rate Ș (0 <Ș 1) equals 0.5.

TABLE I Recognition accuracy of different CNN structures in JAFFE (%)



C1 \ C2	10	12	14
4	13.9536	65.1163	69.7673
6	13.9536	**76.7442**	69.7673
8	58.1396	58.1395	67.4418

As the table shows, while multiple feature maps can theoretically remove more types of display functionality, any feature will also unnecessarily interfere and reduce network recognition capability. Therefore, adequate structure of the convolution layer is important to obtain good recognition results. Based on the experiments, JAFFE achieves the highest accuracy when C1 has 6 and C2 has 12 feature maps.

B. Performance Evaluation vs. Learning Rates

The learning rate Ș is the measure as the parameters change, so the value is controlled between 0 and 1 (0 <Ș 1). If Ș is too large, the range of network parameters at each update will be too marked, which will affect the stability of the update parameters. Worse still, it can lead to non-convergence error with increase in training times. Meanwhile, if Ș is too small, the convergence process will take a long time and consume too many computing resources. Therefore, the value of Ș must be selected appropriately based on the actual training environment.

We selected five discrete values between 0-1 and get the recognition results in both JAFFE and CK+, which are respectively displayed.

TABLE III Recognition accuracy on different learning rates in JAFFE (%)

η	**Accuracy**
0.1	69.7674
0.3	72.093
0.5	**76.7442**
0.7	74.4186
0.9	72.093

TABLE IV Recognition accuracy on different learning rates in CK+ (%)

η	Accuracy
0.1	75.7576
0.3	75.7576
0.5	78.7879
0.7	**80.303**
0.9	77.2727

The best recognition result is obtained when Ş is 0.5 in JAFFE, and 0.7 in CK+. Higher or lower learning rate will decrease the recognition performance of CNN.

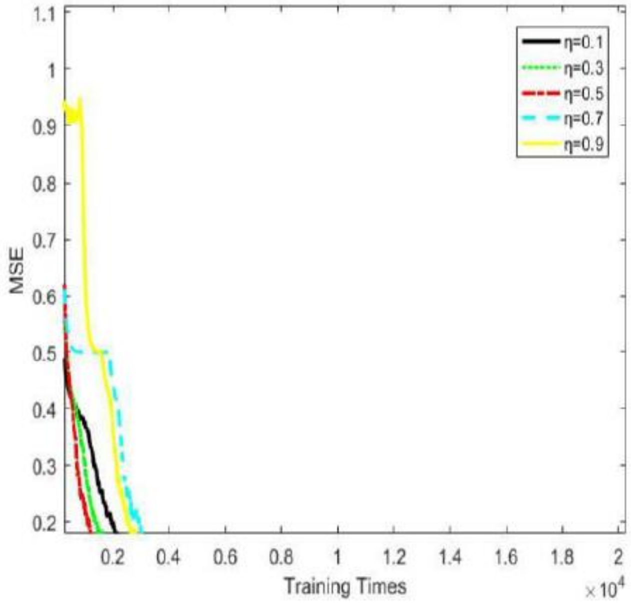


Fig. 8. Partial magnification of MSE convergence curve in CK+.

In order to further analyze the impact of different learning rates on recognition performance, we draw the part magnification of the CK + Mean Error Square (MSE) convergence curve for example.

As we can see more intuitively from Fig. 8, when Ş is too large (Ş=0.9,0.7), the sharp change of weights results in the oscillation of MSE at the beginning of training. As Ş decreases, the error convergence tends to be stable. The smaller Ş leads to the slower MSE convergence.

C. Performance Evaluation vs. Image Pre-processing

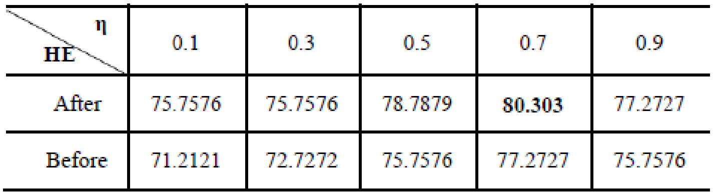
To reflect the effect of pre-processing, we discuss the recognition performance before and after the histogram equalization (HE). The result in JAFFE is showed as Table .

TABLE V Recognition accuracy before and after HE in JAFFE (%)

η **HE**	0.1	0.3	0.5	0.7	0.9
After	69.7674	72.093	**76.7442**	74.4186	72.093
Before	69.7674	65.1163	67.4419	51.1628	58.1395

The same evaluation in CK+ is displayed in Table as well:

TABLE VI Recognition accuracy before and after HE in CK+ (%)



η HE	0.1	0.3	0.5	0.7	0.9
After	75.7576	75.7576	78.7879	**80.303**	77.2727
Before	71.2121	72.7272	75.7576	77.2727	75.7576

The results show that the recognition accuracy without histogram equalization is reduced due to brightness interference, compared to the accuracy after equalization on every value of learning rate. It proves that histogram equalization does indeed improve recognition performance of the network.

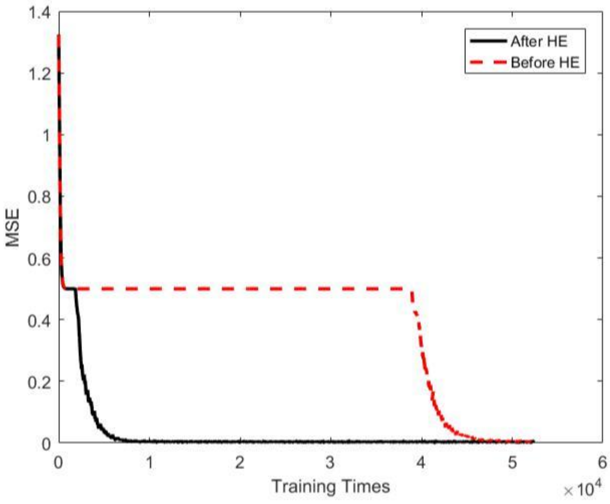


Fig. 9. MSE convergence curve before and after HE in CK+.

Obviously, MSE converges more slowly when training the network with original images, which means that there is a lot of interference before the histogram is equalized. Much computational resources and training times are consumed to correct the interference caused by brightness differences. By the appropriate pre -process like histogram equalization, we can render the MSE converge faster and finally get the better training results.

D. Performance Comparison

For a more descriptive display of CNN’s performance on facial expression recognition task, we introduce a traditional classification method KNN (K-Nearest Neighbours) to make a comparison.

For equality, the pre-processing of images for KNN is the same as CNN’s. We get the recognition results in different K values as below.

TABLE VII Recognition accuracy of KNN on different K values in JAFFE (%)

k	**Accuracy**
3	58.1395
5	62.7907
7	**65.1163**
9	62.7907
11	48.8372
13	53.4884

TABLE VIII Recognition accuracy of KNN on different K values in CK+ (%)

k	**Accuracy**
10	74.2424
15	**77.2727**
20	74.2424
25	72.7273
30	72.7273
35	71.2121

Thus, we can compare the two best recognition results of CNN and KNN, which is shown in Table .

TABLE IX The comparison of best recognition accuracy of CNN and KNN (%)

	CNN	**KNN**
JAFFE	76.7442	65.1163
CK+	80.303	77.2727

As is shown, CNN’s performance is significantly better than KNN’s in the task of facial expression recognition. KNN is simply based on the spatial location of known data to judge the test data, while CNN can learn deeper features of data and get more reliable recognition results. In a conclusion, the CNN is obviously more suitable for facial expression recognition.

IV. CONCLUSIONS

In this paper, we have proposed a system based on a CNN algorithm to achieve human facial expression recognition. The whole system is composed of Input Module, Pre-processing Module, Recognition Module and Output Module. First of all, we build a theoretical model of the system, and describe the details of every module, especially the CNN algorithm module. Then we introduce two classic facial expression databases

JAFFE and CK+ to simulate the recognition process on MATLAB, and analyze recognition performance in different situations. A KNN algorithm also employed to make comparison with CNN, which demonstrates that the CNN algorithm is more suitable for facial expression recognition.

REFERENCES

[1]. Picard R. W.. Affective computing. MIT Press.

[2]. Ekman P, Friesen WV. Constants across cultures in the face and emotion[J]. Journal of personality and social psychology, 1971,17(2): 124.

[3]. Mase K. Recognition of facial expression from optical flow. IEICE Trans E[J]. Ieice Transactions on Information & Systems, 1991, 74(10).

[4]. X. W. Chen and X. Lin, "Big Data Deep Learning: Challenges and Perspectives," in IEEE Access, vol. 2, no. , pp. 514-525, 2014.doi: 10.1109/ACCESS.2014.2325029.

[5]. Hinton G E; Salakhutdinov R R. Reducing the dimensionality of data with neural networks [J]. Science, 2006, 313:504-507. DOI: 10.1126/science.1127647.

[6]. Hubel DH, Wiesel TN. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of Physiology. 1962;160(1):106-154.2.

[7]. Lecun, Y. "Generalization and Network Design Strategies." Connectionism in Perspective 1989.

[8]. Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010). The Extended Cohn-Kanade Dataset (CK+): A complete expression dataset for action unit and emotion-specified expression. Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB 2010), San Francisco, USA, 94-101.

[9]. Michael J. Lyons, Shigeru Akemastu, Miyuki Kamachi, Jiro Gyoba. Coding Facial Expressions with Gabor Wavelets, 3rd IEEE International Conference on Automatic Face and Gesture Recognition, pp. 200-205 (1998).

[10]. Pageorgiou C., Oren M., Poggio T.. A general framework for object detection. International Conference on Computer Vision. 1998. 555-562.

[11]. Oren M., Pageorgiou C., Ppggio T.. Example based object detection in images by components. IEEE Transaction on Pattern Analysis and Machine Intelligence.2001.23(4): 349-361.

[12]. Viola P., Jones M.. Rapid object detection using a boosted cascade of simple features. IEEE Conference on Computer Vision and Pattern Recognition. 2001:511-518.

[13]. M. Z. Uddin, M. M. Hassan, A. Almogren, A. Alamri, M. Alrubaian and G. Fortino, "Facial Expression Recognition Utilizing Local Direction-Based Robust Features and Deep Belief Network," in IEEE Access, vol. 5, pp. 4525-4536, 2017, doi: 10.1109/ACCESS.2017.2676238.

[14]. Y. Ding, Q. Zhao, B. Li and X. Yuan, "Facial Expression Recognition From Image Sequence Based on LBP and Taylor Expansion," in IEEE Access, vol. 5, pp. 19409-19419, 2017, doi: 10.1109/ACCESS.2017.2737821.

[15]. K. Zhang, Y. Huang, Y. Du and L. Wang, "Facial Expression Recognition Based on Deep Evolutional Spatial-Temporal Networks," in IEEE Transactions on Image Processing, vol. 26, no. 9, pp. 4193-4203, Sept. 2017, doi: 10.1109/TIP.2017.2689999.

[16]. B. Yang, J. Cao, R. Ni and Y. Zhang, "Facial Expression Recognition Using Weighted Mixture Deep Neural Network Based on Double-Channel Facial Images," in IEEE Access, vol. 6, pp. 4630-4640, 2018, doi: 10.1109/ACCESS.2017.2784096.

[17]. M. Z. Uddin, W. Khaksar and J. Torresen, "Facial Expression Recognition Using Salient Features and Convolutional Neural Network," in IEEE Access, vol. 5, pp. 26146-26161, 2017, doi: 10.1109/ACCESS.2017.2777003.

[18]. B. -F. Wu and C. -H. Lin, "Adaptive Feature Mapping for Customizing Deep Learning Based Facial Expression Recognition Model," in IEEE Access, vol. 6, pp. 12451-12461, 2018, doi: 10.1109/ACCESS.2018.2805861.

[19]. André Teixeira Lopes, Edilson de Aguiar, Alberto F. De Souza, Thiago Oliveira-Santos, Facial expression recognition with Convolutional Neural Networks: Coping with few data and the training sample order, Pattern Recognition, Volume 61, 2017, Pages 610-628, ISSN 0031-3203.

[20]. A. Mollahosseini, D. Chan and M. H. Mahoor, "Going deeper in facial expression recognition using deep neural networks," 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), 2016, pp. 1-10, doi: 10.1109/WACV.2016.7477450.

[21]. Kim, BK., Roh, J., Dong, SY. *et al.* Hierarchical committee of deep convolutional neural networks for robust facial expression recognition. *J Multimodal User Interfaces* 10, 173–189 (2016). https://doi.org/10.1007/s12193-015-0209-0