Facial Expression Recognition Method Combined with Attention Mechanism

Ming Chen 1, Junqiang Cheng 2, Zhifeng Zhang 1, Yuhua Li 1, and Yi Zhang 3

1 College of Software Engineering, Zhengzhou University of Light Industry, Zhengzhou, Henan 450000, China
2 Europe-Asia Hi-tech and Digital Technology Company Limited, Zhengzhou, Henan 450000, China
3 Business School, Xinxiang University, Xinxiang, Henan 458000, China

Correspondence should be addressed to Ming Chen; cm19834@163.com

Received 26 August 2021; Revised 7 September 2021; Accepted 13 September 2021; Published 25 September 2021

Aiming at the slow speed and low accuracy of traditional facial expression recognition, a new method combining the attention mechanism is proposed. Firstly, group convolution is used to reduce network parameters. The channels of traditional convolution are grouped to cut off redundant connections so that the number of parameters decreases significantly. Secondly, the ERFNet network model was improved by combining the asymmetric residual module and the weak bottleneck module to improve the running speed and reduce the loss of accuracy. Finally, the attention mechanism was added into the feature extraction network to improve the recognition precision. The experiment shows that compared with traditional face recognition methods, the proposed method can improve the recognition precision and recall significantly; in CK+, Jaffe, and Fer2013 datasets, the recognition precision can reach 88.81%, 82.16%, and 79.33%, respectively.

1. Introduction

Facial expressions can convey clues to a person’s emotional state, which together with voice, language, and hands and body postures compose the basic communication system between human beings in the social environment [1]. In different environments, facial expressions have many functions for communication. We can adjust dialogue, convey biological information, express mental labor intensity, and express emotions by receiving signals in turn [2, 3]. In the process of information exchange between people, facial expressions play an indispensable role. Building a system that can automatically analyze facial expressions has an important practical significance in the fields of medical care, education, and driverless cars [4]. For example, judging the patient’s pain degree through facial expressions can effectively assist doctors in diagnosis [5–7]. Detecting facial expressions to judge the degree of distraction can help teachers improve students’ learning efficiency [8–10]. Monitoring the facial expressions of the drivers can perceive the degree of fatigue in advance [11–13]. Therefore, the academic circle has explored various facial expression recognition systems to encode and recognize facial expression information. There are two main methods of facial expression measurement: information judgment and indication judgment [14]. Information judgment is to study the conveyed meaning by facial expressions (such as happy, angry, or sad). Indication judgment is to study physical signals, which are used to convey information (such as raised cheeks or sunken lips) [15–17]. The main disadvantage of the information judgment method is that it cannot explain the full range of facial expressions. It usually assumes that facial expressions and target behaviors (such as expression categories) have a clear many-to-one corresponding relationship. But according to psychological research, this is not the case in reality. Generally speaking, the relationship between information and its related categories is not universal. The facial display and its interpretation vary from person to person and even from situation to situation [18–20].

In recent years, deep learning had developed rapidly. Many network architectures, such as VGG [21], AlexNet [22], and ResNet [23], had also been widely used in facial expression recognition. The VGG network was different...
from AlexNet, which deepened the network layers and used $3 \times 3$ small convolution kernels. Thereby, VGG reduced the amount of parameters. He et al. [23] proposed ResNet, which used the principle of identity path to further deepen the number of network layers without causing gradient explosion. Cheng and Zhou [24] used transfer learning to overcome the lack of training samples. On the basis of VGG19, the network structure and parameters were optimized to improve the precision. Zhong et al. [25] introduced the dropout layer on the basis of ResNet and modified the fully connected layer (FC) to reduce parameters. At the same time, SE block [26] was added to the network to achieve a higher accuracy. Chen and Hu [27] proposed a new learning method of the interclass relationship, which extracted features from two different expression images, merged the extracted features at a random ratio to obtain a mixed feature, then redistributed the weight to each pixel, and last improved feature distribution through the relationship between expressions. So, the algorithm’s discriminative ability was increased. Wang and Shen [28] combined global features with regional features in the deep convolutional network. In addition, the facial action unit and Bayesian network model were established to analyze the probability of the action unit. Finally, the learned features were integrated for expression classification. Xu and Zhao [29] used two branches to extract features. One branch is local binary pattern (LBP). Other branch is convolutional neural networks. Then, the two features were merged. Finally, principal components analysis (PCA) was used to reduce the feature dimensionality. So, it can effectively improve the accuracy of facial expression recognition.

Many existing studies only focus on the entire face image. However, the texture distortion caused by the expression is obvious at the key points. The facial eyes, mouth, and left and right cheek are the areas where facial movements are most obvious when expressions occur. Focusing only on the overall features usually leads to unnecessary calculations. So, based on the main problems and research status of facial expression analysis, the main work of this paper is as follows:

1. Firstly, group convolution is used to reduce network parameters. The traditional convolutional channels are grouped, and redundant connections in the network are cut. So, the number of parameters is greatly reduced.

2. Secondly, the ERFNet model is improved by combining the asymmetric residual module and the weak bottleneck module, which improve the running speed and reduce the loss of accuracy.

3. Finally, an attention mechanism is added to the feature extraction network to improve the segmentation accuracy of the network.

This paper first gives the general technical steps of face recognition and introduces each step and then analyzes the convolutional neural network most commonly used in face recognition technology. Secondly, the network structure design of this paper is given. The network parameters are reduced by group convolution, the running speed of the proposed method is improved by the improved ERFNet network model, and the attention mechanism is introduced to improve the recognition accuracy of the network. Finally, experimental validations are carried out on the proposed expression recognition method. And, the experimental results are visually analyzed from training process, precision, and recall.

2. Related Theories and Methods

2.1. Technical Steps. As shown in Figure 1, face recognition technology specifically includes four steps: (1) face detection, (2) face alignment, (3) face characterization, and (4) face matching. Face detection refers to find the position of the face in the picture. If there is a face, a rectangular frame containing the face will be returned. Face alignment refers to automatically locate the key feature points of the face. Face characterization refers to convert the face into a feature vector. Face matching refers to use the features from the previous step to compare with the features in the database. Finally, the distance between the features of different faces is compared. If the distance is less than a certain threshold, it is regarded as the same face, otherwise it is regarded as a different face.

2.1.1. Face Detection and Alignment. In this part, it is first to determine whether there is a human face in the picture. The picture is preprocessed to eliminate the influence due to factors such as illumination and shooting angle. Here, the active shape model (ASM) algorithm and active appearance model (AAM) algorithm are often used for face key point detection. Their advantages are that the structures are clear and easy to understand and apply. However, the computational efficiency is lower. They are not suitable for massive face images.

2.1.2. Face Characterization. The faces detected in the above steps are processed to obtain the unique feature vectors. Such feature vectors often contain position information of the eyebrows, eyes, noses, and mouths. In addition, information including contour, size, and shape of the face may also be added into feature vectors. Navneet’s Histogram of Oriented Gradient (HOG) algorithm, eigenface algorithm, and Haar wavelets’ algorithm are often classic methods.

2.1.3. Face Matching. The extracted feature vector will be compared with the face vectors in the database. If the similarity between the feature vectors is higher, the identity information corresponding to the face is output. If there is no match for all faces in the database, the output is unable to be recognized.

2.2. Convolutional Neural Network. At the beginning of the 21st century, several studies in the facial expression recognition literatures found that CNN’s performance is better than multilayer perceptions, and it can solve the problems of translation, rotation, and scale invariance in facial expression recognition.
The convolutional layer browses the entire input image with a learnable filter and then generates various specific activation feature maps. The convolutional layer is one of the most important modules of convolutional neural networks, which can quickly increase the calculation rate. The parameters of the convolutional layer include a series of learnable receptive fields. Although each receptive field is very small, they can extend to the full depth of the input. In the process of forward calculation, we convolve all inputs and perform a dot product operation with the receptive field at each position of the input. When the receptive field of the network receives a specific form of visual features, such as directions, spots, or colored edges, it will be activated, and the generated feature map will have a high response. In some deeper networks, even the entire honeycomb or wheel pattern can be seen. In order to obtain deeper feature maps, multilayer convolutional layers are usually used. When processing some high-dimensional and more complex input images, due to the large number of neurons, it is very difficult to connect all neurons between adjacent layers. So, local connections are used. The connections are limited to height and width, and full connection is still used in the depth of the input images. The convolution operation has three main characteristics: (1) local connection, which uses the correlation between adjacent pixels to share weights and greatly reduces the learning parameters. (2) Translation invariance, which has a good ability to recognize facial expressions in different positions. The convolutional layer is followed by the pooling layer, which greatly reduces the space size of the feature map and the computational cost of the network. (3) Fully connected layer. The neurons of adjacent layers are connected in pairs. FC is as a “classifier” in the convolutional neural network. The activation function layer, pooling layer, convolutional layer, and other operations are to turn data into abstract feature representations. Then, the FC layer is to classify the learned features. As the network increases, the parameters of the FC layer will increase exponentially, which can even account for 80% of the entire network. This will result in the slower of training speed. All neurons in the FC layer are connected to the upper layer, which can convert the two-dimensional feature map into a one-dimensional feature map and then perform feature representation and classification.

In CNN, it is very common to insert a pooling layer between two adjacent convolutional layers because overfitting can be controlled by downsampling. After the pooling operation, the pooling layer can reduce the depth of each layer. The most common pooling layer is that the receptive field is $2 \times 2$, and the step is 2. Through the dimensionality reduction of each layer input, 75% of the activation value can finally be discarded.

### 3. Design of Network Structure

#### 3.1. Group Convolution

Traditional neural networks have mixed spatial and channel features after convolution operation. Group convolution can strip off this mixture and make the network pay attention to the feature relationship between different channels. As shown in Figure 2, compared with traditional networks, group convolution effectively reduces the complexity of parameters and greatly improves the segmentation speed, which can help to realize real-time segmentation.

#### 3.2. Improved ERFNet

At present, several existing semantic segmentation networks, such as SegNet [30] and FCN [31], have the problems of large amount of parameters and the long processing time. Although the lightweight real-time semantic segmentation ENet [32] can solve this problem, its implementation accuracy is unsatisfactory. ERFNet [33] effectively solves the problem of segmentation accuracy, but the residual module has a large amount of parameters, which will reduce the calculation speed of the model. Therefore, based on the ERFNet, this paper uses asymmetric convolution modules to improve running efficiency. The asymmetric convolution module decomposes the $d \times d$ convolution kernel into $d \times 1$ and $1 \times d$ convolution kernels. During the convolution operation, the parameter amount is reduced from $d^2$ to $2d$, which can reduce a large number of parameters.

The network model adopts an end-to-end encoder-decoder structure, as shown in Figure 3. The network model mainly is composed by downsampling block, asymmetric residual block (ARB), weak bottleneck block (nonbottleneck-1D), and upsampling block.

In this model, modules 1 to 16 are the encoder part, and modules 17 to 23 are the decoder part. The encoder outputs small-resolution feature maps of several channels, and the decoder upsamples the small-resolution feature maps to recover the initial resolution output. The special network setting of ERFNet refers to [34], as shown in Table 1.

Modules 1, 2, and 8 are downsampling modules. Since the initial input image is large and contains more redundant information, downsampling can significantly reduce the size of feature maps and reduce the calculation complexity. However, frequent downsampling can also cause the precision loss of image edge segmentation. Meanwhile, it will increase upsampling calculation cost. To consider the balance between running speed and segmentation accuracy, this paper only performs three down-sampling processing.

The residual module and the bottleneck module are two basic structures proposed by Nestor et al. [17]. The network of this paper is stacked on its basis after effective improvement. Aiming at the low efficiency of the residual module, this paper uses asymmetric convolution technology.
to redesign it, which obtains the asymmetric residual module and the weak bottleneck module. When the feature map that needs to be calculated is large, using the asymmetric residual module and the weak bottleneck module can effectively improve the running speed of the model. And, using the combination of 1D filters to decompose the convolutional layer can effectively save the calculation cost.

Modules 17 and 20 are upsampling modules, which are used to alleviate the loss of spatial information and reduce the precision loss of image. In this paper, deconvolution is used for upsampling. Compared with traditional interpolation algorithms, the weights of deconvolution can be learned.

3.3. Attention Mechanism. To improve the segmentation speed of the network, this paper uses an improved ERFNet as a feature extraction network. But ERFNet will lead a certain precision loss. Therefore, the attention mechanism needs to be added to the network. The attention module considers the interdependence between the feature channels. And, through the self-learning of the network, the weights of useless features are effectively suppressed and the weights of useful features are enhanced, which can improve the understanding of the model.

The operation of the attention module is mainly divided into two steps, squeeze and excitation, as shown in Figure 4.

In the attention module, global average pooling is used to shrink the input data from the previous layer. The feature map shrinks from $M \times N \times K$ to $1 \times 1 \times C$. The specific formula is as follows:

$$z_c = F_{GAP}(L_c) \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} L_c(i, j),$$

where $F_{GAP}$ represents the global average pooling function. This paper uses the Atrous Spatial Pyramid Pooling (ASPP) [35] algorithm as the global average pooling function. ASPP provides an effective mechanism to control the size of the receptive field. Meanwhile, ASPP finds the best balance between precise positioning (small field of view) and the information restoration (large field of view).

In the attention module, the global features on each channel are obtained by formula (1). Then, the module performs an excitation operation to obtain the correlation between different channels. This step is mainly completed by two fully connected layers. To reduce the number of parameters, the first fully connected layer compresses $C$ channels into $C/r$ channels, where $r$ is the compression rate, as shown in formula (2). Then, the activation function ReLU increases the nonlinearity of features. The second fully connected layer is restored to $C$ channels, and the activation function Sigmoid is used to generate different weights for each feature channel. After this, the network has more distinguishing capabilities for the features of each channel. The purpose of gaining useful features and suppressing useless features is achieved by multiplicative weighting:

$$s = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W_2 \delta(W_1 z)), \quad W_1, W_2 \in \mathbb{R}^{C/r \times C},$$

where $\sigma$ and $\delta$, respectively, represent Sigmoid and ReLU activation functions.

4. Validation of the Algorithm

4.1. Network Parameter Setting. The experimental model is based on the TensorFlow1.9 framework and uses the cuDNN7.5 kernel for calculation. The workstation is configured with GTX 2080Ti graphics card and memory 512 GB. The optimization method used stochastic gradient descent. Its momentum parameter was set to 0.99, and the weight decay rate was $1 \times 10^{-4}$. The initial learning rate was $8 \times 10^{-3}$. In the experiment, the image size was set to $384 \times 384$ pixels,
and the training batch size was $2 \times 7 = 14$. The experiment used CK+, JAFFE, and Fer2013 datasets for validation.

4.2. Evaluation Index. The loss value was a usually used indicator to evaluate a model, which was used to estimate the inconsistency degree between the true value and the predicted value. The lower the loss value was, the better the robustness of the model was. The cross-entropy loss function was used to compute the loss value, which was showed as formula (3), where $y$ represented the true classification value, $a$ represented the predicted value, and $c$ represented the loss value:

$$c = -\frac{1}{n} \sum [(1 - y)\ln (1 - a) + y \ln a].$$

The precision $P$ represented the proportion of the actual positive samples in all predicted positive samples. The recall $R$ represented the proportion of the actual positive examples are predicted correctly. This paper used the precision and recall as model evaluation indicators.

The precision $P$ could be expressed as follows:

$$P = \frac{TP}{TP + FP}$$

The recall $R$ was expressed as follows:
where TP represents the number of actual positive samples in all predicted positive samples, FP represents the number of actual negative samples in all predicted positive samples, and FN represents the number of actual positive samples, which were mistaken as negative samples.

4.3. Training Process. As shown in Figure 5, when the number of iterations was 20, the accuracy of the training set tended to be stable, and the precision of training set was around 90%.

It was found from Figure 6 that the loss value of the validation set and training set decreased rapidly in the first 20 iterations. After the 25th iteration, the loss value of this model dropped to a very low level and stabilized, and there was no significant change anymore.

4.4. CK+ Dataset Experiment. This paper used CK+ face expression dataset for experimental verification. The CK+ dataset collected 123 subjects, 593 expression sequences, and 951 image samples. Among the 593 image sequences, there were 327 sequences with emotion labels. The emotion labels are happy, sad, angry, fear, surprised, disgust, and neutral. Part of images and the real category annotations of the dataset are shown in Figure 7. In order to prevent the overfitting phenomenon caused by the small number of samples, this paper used data augmentation to expand the experimental dataset. By performing operations such as rotation, horizontal flipping, and vertical flipping on the facial expression images in the dataset, the dataset was expanded to 12,000 images.

To test the effect of the model, 9000 images were randomly selected as the training set, 1500 images were as the test set, and 1500 images were as the validation set in the expanded image dataset. It could be seen from Figure 8 that the method in this paper had a higher precision. The results are 94.44%, 87.5%, 93.7%, and 93.08% for the four expressions of happy, angry, surprised, and disgust. The three expressions of sad, fear, and neutral had relatively lower precision values, which are 78.82%, 71.01%, and 66.67%, respectively. In experiment, the improved method is more likely to misclassify sad expressions as fear expressions. This may be due to the similarity between lips. The probability of misclassifying the fear expression as surprised is higher, mainly because the fear expression and the surprised expression have a higher similarity degree at the level of organ features such as eyes and nose. Neutral expressions are more likely to be misclassified as happy, disgust, and sad. It may be some expression changes were not obvious.

4.5. JAFFE Dataset Experiment. Michael Lyons team created a database of Japanese women in the Department of Psychology at Kyushu University. The database contained a total of 10 participants. Each of these participants collected seven expressions of happy, sad, angry, disgust, surprised, fear, and neutral. The JAFFE dataset was constituted of 213 facial images in the database. In the experimental environment, participants make corresponding facial expressions according to the requirements of collectors. Then, collectors used the camera to capture the face expression. The illumination angles of images collected in this experiment were all the same, and they were all frontal illumination. The collectors re-cut and adjusted the initially collected images. So the size of the images was basically the same. The size was 256 × 256 pixels. The position of the eyes in images was also roughly the same. In the experimental environment, participants make corresponding facial expressions according to the requirements of collectors. Then, collectors used the camera to capture the face expression. The illumination angles of images collected in this experiment were all the same, and they were all frontal illumination. In the experimental environment, participants make corresponding facial expressions according to the requirements of collectors. Then, collectors used the camera to capture the face expression. The illumination angles of images collected in this experiment were all the same, and they were all frontal illumination. In the experimental environment, participants make corresponding facial expressions according to the requirements of collectors. Then, collectors used the camera to capture the face expression. The illumination angles of images collected in this experiment were all the same, and they were all frontal illumination. The collectors re-cut and adjusted the initially collected images. So the size of the images was basically the same. The size was 256 × 256 pixels. The position of the eyes in images was also roughly the same.

Table 2 showed the confusion matrix of the proposed model in the JAFFE dataset. The recognition precision of this method reaches 82.16%. In this dataset, there are a small number of misclassifications in the surprise and sadness categories. Because the expressions of some surprise categories in JAFFE did not change much, it caused confusion with the sadness category.
4.6. Fer2013 Dataset Experiment. The Fer2013 dataset consisted of 35886 facial expression images, which included 28708 test pictures, 3589 public validation images, and 3589 private validation images. And, Fer2013 dataset also had 7 expressions. The dataset did not directly contained pictures, but saved data into csv files. After 10,000 iterations of
training, the precision was 79.33% in this paper on Fer2013 dataset. It could be seen from Table 3 that the model had the highest precision in sad expression. The precision was 87.90%. And, the recognition precision of disgust expression was lower, which was 65.08%. According to the model, it was conjectured that the extraction parameters of feature maps were too few, and some expression features were ignored, which reduced the classification effect.

The comparison results with the traditional deep learning model on the Fer2013 dataset were shown in Table 4. Among them, ResNet [17] used the principle of

Table 2: Recognition results on the JAFFE dataset.

<table>
<thead>
<tr>
<th></th>
<th>Angry</th>
<th>Disgust</th>
<th>Fear</th>
<th>Sad</th>
<th>Happy</th>
<th>Surprised</th>
<th>Neutral</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angry</td>
<td>26</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Disgust</td>
<td>0</td>
<td>24</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Fear</td>
<td>3</td>
<td>0</td>
<td>26</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Sad</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>26</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Happy</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>27</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Surprised</td>
<td>1</td>
<td>0</td>
<td>6</td>
<td>1</td>
<td>0</td>
<td>22</td>
<td>0</td>
</tr>
<tr>
<td>Neutral</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>24</td>
</tr>
</tbody>
</table>

Table 3: Recognition results on Fer2013 dataset.

<table>
<thead>
<tr>
<th></th>
<th>Angry</th>
<th>Disgust</th>
<th>Fear</th>
<th>Sad</th>
<th>Happy</th>
<th>Surprised</th>
<th>Neutral</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angry</td>
<td>711</td>
<td>12</td>
<td>58</td>
<td>32</td>
<td>76</td>
<td>16</td>
<td>53</td>
</tr>
<tr>
<td>Disgust</td>
<td>17</td>
<td>82</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Fear</td>
<td>35</td>
<td>22</td>
<td>744</td>
<td>48</td>
<td>54</td>
<td>63</td>
<td>58</td>
</tr>
<tr>
<td>Happy</td>
<td>67</td>
<td>2</td>
<td>58</td>
<td>62</td>
<td>942</td>
<td>18</td>
<td>98</td>
</tr>
<tr>
<td>Sad</td>
<td>29</td>
<td>3</td>
<td>33</td>
<td>1555</td>
<td>67</td>
<td>29</td>
<td>58</td>
</tr>
<tr>
<td>Surprised</td>
<td>14</td>
<td>3</td>
<td>33</td>
<td>26</td>
<td>12</td>
<td>725</td>
<td>18</td>
</tr>
<tr>
<td>Neutral</td>
<td>41</td>
<td>2</td>
<td>35</td>
<td>44</td>
<td>156</td>
<td>20</td>
<td>935</td>
</tr>
</tbody>
</table>

Table 4: Comparison of recognition effectiveness on Fer2013 dataset.

<table>
<thead>
<tr>
<th>Model</th>
<th>Precision (%)</th>
<th>Average recall (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNNs</td>
<td>60.52</td>
<td>62.84</td>
</tr>
<tr>
<td>ResNet</td>
<td>69.67</td>
<td>70.38</td>
</tr>
<tr>
<td>The proposed model</td>
<td>79.33</td>
<td>78.15</td>
</tr>
</tbody>
</table>
identity path to further deepen the network without gradient explosion. The CNN [25] model used a parallel convolutional neural network model. The facial expression recognition model designed in this paper had a higher precision and recall.

5. Conclusion

This paper proposes a facial expression recognition method combined with the attention mechanism. Group convolution is used to reduce network parameters. The improved ERFNet model is used to improve the running speed of the algorithm. The attention module is used in the feature extraction network to improve the recognition precision. Experiments show that this method improves the recognition precision. However, the model still has room for improvement in the recognition of fear and sad expressions. It is necessary to subdivide and extract facial features to improve the recognition accuracy.

Data Availability

The data used to support the findings of the study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the National Science Foundation of China (nos. 61975187 and 61902021) and Henan Science and Technology Research Project (nos. 212102210104 and 162102210214).

References


