IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 11, 2019, accepted December 30, 2019, date of publication January 6, 2020, date of current version January 14, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2964298

ROI-Attention Vectorized CNN Model for Static Facial Expression Recognition

XIAO SUN[1], (Member, IEEE), SHIXIN ZHENG[1], (Student Member, IEEE), AND HONGSHUAI FU[2], (Student Member, IEEE)

[1] Anhui Province Key Laboratory of Affective Computing and Advanced Intelligent Machines, Hefei University of Technology, Hefei 230009, China
[2] School of Computer Science and Information Engineering, HeFei University of Technology, Hefei 230009, China

Corresponding author: Xiao Sun (sunx@hfut.edu.cn)

This work was supported in part by the General Program of National Natural Science Foundation of China under Grant 61976078, and in part by the National Key Research and Development Program of China under Grant 2019YFA0706203.

ABSTRACT When using neural networks to recognize facial expressions, determining which features help identify different expressions is essential, and there is a massive information transmission loss between layers of network with multiple layers. This paper proposes a robust vectorized convolutional neural network (CNN) model that introduces an attention mechanism for extracting features in the region of interests(ROIs) of the face. The ROIs in the facial image are marked before the image is input into the neural network. In particular, the attention concept is adopted in the first layer of the proposed neural network to perform ROIs-related convolution calculation, and ROIs-related convolution calculation results of the specific fields in the ROIs are increased by extracting more robust features. Next, the idea of features' vectors inspired by CapsNet is used in the following layer of the proposed neural network. Multi-level convolutions are used to extract feature vectors of different ROIs for facial expression, and then the feature vectors are reconstructed by a decoder to reconstruct the image. Comprehensive comparative experiments and cross-database experiments are conducted to verify the validity and robustness of our proposed model. The experimental results also demonstrate that our method is very effective in improving the performance of facial expressions recognition.

INDEX TERMS Regions of interests, attention mechanism, facial expression recognition, vectored features, CapsNet.

I. INTRODUCTION

Facial expressions are the most direct and effective way to express human emotions. Facial expression recognition has many applications in different research fields, such as image comprehension, psychological detection, affective computing, and human-computer interactions [1]. Many applications, such as medical assessments, driver safety, lie detection, surveillance, and human-computer interaction systems, require effective and robust facial expression recognition to achieve ideal and practical system performance [2]. In recent years, although a lot of efforts have been made, it is hard to recognize facial expression [3] because there are seven basic facial expressions, including anger, disgust, happiness, fear, neutral, sadness, and surprise. Accurately distinguishing the details of emotion-related expressions to recognize facial expressions with high precision is challenging [4].

Generally speaking, facial expression recognition is divided into two steps: facial features representation and classifier classification [5]. Optical Flow [7] and HMM [8] are often used to analyze people's facial expressions from temporal data such as video when features representation coming from the single picture [6] are classified by CNNs [13]–[15], SVM [16], LDA [17] and Bayesian Networks [18]. Different from the classifier that needs to consider the task type, feature extraction is more universal. Many research works have made significant achievements in feature representation, such as LBP [9], LGC [10], HOG [11] and SIFT [12].

Although the method above works well with standard data, it does not perform well in real situations [19]. Therefore, recent research gradually focuses on deep learning, which can automatically extract useful information and has better generalization ability, such as CNNs. CNNs has achieved excellent results in many research fields, such as target detection and pedestrian recognition, but the direct use of CNNs in facial expression recognition task is not satisfactory. It is widely

The associate editor coordinating the review of this manuscript and approving it for publication was Zahid Akhtar.

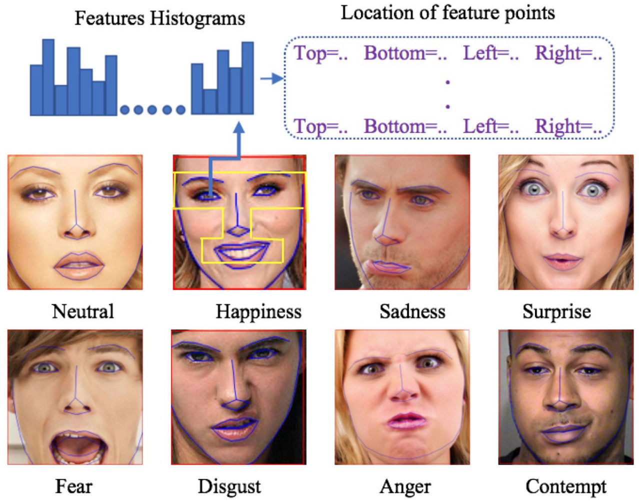
Features Histograms
Location of feature points
Top=.. Bottom=.. Left=.. Right=..
Top=.. Bottom=.. Left=.. Right=..
Neutral
Happiness
Sadness
Surprise
Fear
Disgust
Anger
Contempt

FIGURE 1. Eight types of facial expressions in AffectNet. The location of feature points are gotten by feature histograms. And ROI is marked in yellow.

recognized that the existing CNNs framework is capable of distinguishing different parts of the image, like eyes, nose, and mouth in a face. But end-to-end makes model learning uncertain when modeling complex problems, and it is possible to get a good model only after training with a lot of data. To obtain a better facial expression prediction model under the existing data set, we try to introduce relevant prior knowledge and make the model easier to explain. Reference [33] has proven regions of interest (ROIs) can improve CNN's learning ability effectively. Inspired by this property, we propose a new model. In this model, we introduce the attention mechanism to highlight the ROIs of the face to enhance the extraction and fusion of facial image features.

In fig.1, we list the expressions used in the AffectNet database, we also define three parts of face are ROIs: the eyes, nose and mouth [20] and marked the exact ROI in fig.1. We set the ROIs of the face with high learning efficiency and serve it as a perceptual domain of the convolutional network to guided model learning. Specifically, we mark the ROIs of the image before training and make it part of the input of the first layer of the network to perform the field convolution calculation, guaranteeing the model enhances the learning of the ROI. When the receptive field moves to the ROI in the image, our model increased the weight of the ROI. We design a multi-layer network of ROIs and show it in fig.2. The first convolutional layer uses the attention mechanism to highlight the ROI of the face.The second part contains three convolutional layers to get high-level information. The structure learns these three parts of interest, and then the fifth layer takes the tensor of the fourth layer as input, followed by a decoder. The specific method is to implement a cascade of three expression features according to depth. The image is reconstructed with the vector output of the fifth layer, and a fully connected layer fuses the features in the end. In this way, the features that are helpful for the network to learn expressions are efficiently enhanced, and the loss of feature information is also reduced.

TABLE 1. Recent research methods.

Expression recognition	Method example
Traditional manual features	SIFT: Zhou et al. [12]
	LBP: Nazima et al. [21]
	Gabor: Kunika et al. [22]
	LDPP: MD.ZIA et al. [23]
	LDP: Kennedy et al. [24]
Surface learning	HMM: Wang et al. [25]
	SVM: Varanya et al. [16]
Deep learning	DNN: Kiranyaz et al. [26]–[28]
	DBN: Hinton et al. [27]
	CNN: Sun et al. [29]

To summarize, this paper presents the following contributions: (1) The prior knowledge of ROI in static facial expression recognition is introduced. The part that has an influence on the neural network to learn the expression features is focused on, and the feature extraction process is improved. An attention mechanism is also adopted to explore and summarize the recognition effect when different expressions focus on different ROIs. (2) A multi-layer network of ROIs is designed, where different levels focus on different ROIs, and the proposed model achieves reasonable feature fusion through a vector decoder to improve the expression recognition efficiency. (3) Finally, various experiments, such as cross-database experiments, are performed to verify the robustness, effectiveness, and generalization capabilities of the model.

The remainder of this paper is organized as follows. The second part introduces the related work and research background of the current research direction. The third part introduces the contributions of this article. The fourth part compares and discusses the experimental results. The fifth part is the conclusion of the whole research work.

II. RELATED WORK

Research work in the field of automatic facial expression recognition has made great progress in recent years. As a key aspect of facial expression recognition, the method for selecting facial expression image features has also received increasingly more attention from researchers. Table 1 summarizes the different approaches used by the recent facial expression recognition studies.

A. TRADITIONAL MANUAL FEATURES

In terms of traditional manual features, in 2016, Zhou *et al.* [12] used the SIFT feature to represent the pixel features of face images, and they proposed a feature alignment scheme based on SIFT to achieve the automatic alignment and matching of image features. In 2017, Kauser and Sharma *et al.* [21] proposed that when extracting the entire face image with LBP, the specific position of each

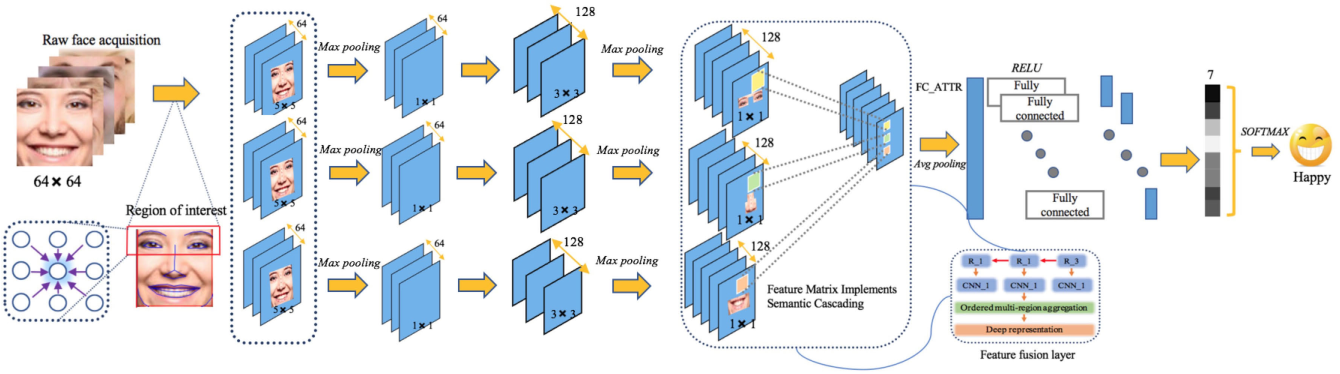

FIGURE 2. ROI-attention vectorized CNN. After three key facial areas were extracted, three networks with the same structure are used to extract features before semantic cascading. And the full connection layer is finally used to get the prediction results.

feature point cannot be distinguished. In their work, LBP was used to extract the features of several important parts of the face, and then these features were connected to form feature vectors for input to neural networks. Verma and Khunteta et al. [22] proposed a method that used Gabor filter to extract facial expression fields in space and then input them into an artificial neural network (ANN) for recognition. Uddin *et al.* [23] proposed a new feature extraction method called local directional position pattern (LDPP) that can simultaneously maintain the characteristics of the bright and dark areas of an image. In 2018, Chengeta and Viriri [24] used LBP and LDP to extract feature vectors to overcome the effect of image redundancy on facial expressions.

B. SHALLOW LEARNING

Regarding the application of shallow learning, in 2016, Wang [25] proposed a recognition method based on Zernike moments and the minimum classification error (MCE) based on HMM, and they achieved very good results. In 2017, George *et al.* [16] used SBP to extract video features using LBP in the process of expression recognition research, and they applied SVM and ANN as classifiers and compared the results.

C. DEEP LEARNING

In contrast, the potential of deep networks is greater, and there is much room for improvement. Many researchers have also invested in deep learning research [26]–[28]. DNN was the first deep learning technique applied in the field of pattern recognition and machine learning research [30]. After G.E. Hinton et al. proposed a deep belief network in 2006 [27], many aspects of expression recognition have adopted deep belief network as a classifier. For example, LDP and deep belief networks are combined to recognize facial expression images [23]. The adoption of CNNs has also made great progress. The work of Sun *et al.* [19] introduced the idea of extracting features of interest from faces, dividing the face into several parts according to the ROI and inputting them into the deep neural network. The classification is performed in a CNN, and the classification results of several parts are finally combined to obtain the final result. The idea of extracting the ROI of the face was also mentioned in other works [29].

D. OUR WORK

The above works demonstrate that the feature extraction of the ROI and the CNNs can achieve a good classification effect. However, there is currently no works focusing on how to better enhance the extraction of features of the ROI for facial expressions, and the extracted features will also cause significant losses. We use attention mechanisms to maximize the extraction of valid features. We were inspired by the idea of vector transmission characteristics from CapsNet, extracting different parts of feature vectors through four convolutional layers, retaining more useful information, and then adopted a decoder to perform feature fusion. Our method focused more on the extraction of facial expression related features, helping the network distinguish more regional features that help to recognize expressions, which can also reduce the loss of features by vector fusion.

III. MODEL PRESENTATION

A. OVERVIEW

As shown in fig.2, after obtaining the database, three-part feature regions of the face are extracted after face alignment, pre-processing, and grayscale processing. These regions are input into CNN for calculation, and finally, the expression recognition result is obtained.

The entire process is subdivided into the following steps:

1) After processing the face database, the ROIs of face are marked and input into the first layer of the CNN.
2) Then, in the first layer of the CNN, the idea of attention is introduced which improves the learning effectiveness of the network on the ROI through the hyperparameters when the first layer of the convolution calculation extracts features. All the tensors learned will be transferred to the second layer in three parts.
3) Next to the second layer structure, a process to use three convolutional layers is designed to input the

characteristic tensors of the eyes, nose, and mouth of the first layer output. A vector of three-part features is output, and the relative position between the vectors is preserved. The three-part vector is then spatially fused at a semantic level by a decoder. The relative relationship between vectors is used to reduce information loss in network transmission.

4) One fully connected layer is used to map the learned ‘‘distributed feature representation’’ to the sample tag space. Finally, we output the classification results.

The process of implementing facial expression recognition has been outlined, the proposed methods and calculations will be introduced in the following sections.

B. FACE DATA PROCESSING

1) FACE ALIGNMENT

The ERT algorithm is used for face interception [31], and the face interception library that calls dlib can use this algorithm directly. This method is a regression-tree-based face alignment algorithm that returns a face shape from the current shape to a real shape step by step by creating a cascaded residual regression tree. Each of the leaf nodes of each GBDT stores a residual regression amount. When the input falls on a node, the residual is added to the input to restore the purpose, and finally, all the residuals are superimposed. Together, these steps complete the process of face alignment.

With two face feature point matrices denoted as M and N, the feature points of one faces needs to be shifted to make the feature point position corresponding to another face. ϕ is used to represent the feature extraction function, which represents the residual regression of each leaf of GBDT, and minimizes the result with the following formula:

$$\sum_{i=1}^{68}\left\|\mu RW_i^T\phi^T\left(M_i+\Delta x\right)-N_i\right\|^2 \tag{1}$$

R is a 2 × 2 orthogonal matrix, and μ is a scalar. T is a 2-vector, and W is a linear regression parameter matrix, which is to map the extracted features to a two-dimensional offset that is a 2x2 transformation matrix. Δx is the image feature point offset.

2) COLOUR CORRECTION AND GRAYSCALE

The noise in the image might have a large impact on facial expression recognition. Thus, the Fourier transform method is used to suppress the noise part of the image. The formula for image denoising through a two-dimensional discrete Fourier transform is as follows:

$$F\left(u,v\right)=\frac{1}{MN}\sum_{x=0}^{m-1}\sum_{y=0}^{N-1}f\left(x,y\right)\times e^{-j2\pi\left(ux/M+vy/N\right)} \tag{2}$$

Image grayscale processing is a critical step in image processing. It helps the image highlight more detailed features, avoiding the interference of colour factors when the network extracts features. The weighted average method is used for grayscale processing, that is, weighted average of the values of R, G, and B according to a certain weight. The formula for this method is as follows:

$$R=G=B=\left(w_RR+w_GG+w_BB\right)/3 \tag{3}$$

Because the human eye is the most sensitive to green, followed by red and the least sensitive to blue, $W_G > W_R > W_B$ will obtain a grayscale image that is easier to recognize.

C. ENHANCED EXPRESSION LEARNING

1) MARK THE REGION OF INTEREST

The ROIs need to be marked for each face image. When facial images are input into the convolutional layer, the network layer extracts the image features in the same way and does not distinguish the feature points of different parts. However, Sun *et al.* [19] showed that a part of the face of interest contains most of the facial features of the face. Strengthening the learning of the ROIs of the face can improve the effect of feature extraction. We use S1, S2, and S3 to represent the matrix of feature points of the three ROIs in the face, nose and mouth of this image. We enhance these three ROIs with the following formula:

$$\left|\left|\lambda^t W_p^t\phi_p^t\left(M_i,S_i\right)\right|\right|,\quad p=1,2\,\ldots,k \tag{4}$$

where k denotes the number of feature points corresponding to the enhancement part, λ is the enhancement coefficient, t is a 2-vector, and W_p^t is a feature enhancement matrix, where each column represents a vector of the corresponding feature point. ϕ_p^t is a very sparse binary vector representing the feature extraction function. M_i represents the face feature point matrix to which the S_i feature portion belongs.

2) ATTENTION MECHANISM

As shown in fig.3, the feature points of the face image are enhanced by the attention method in the first layer of the convolution calculation of the neural network. The specific process is as follows. The image is input into the first layer of the CNN. When the convolution is feature extraction, the sliding window slides according to the specified step size. A small part of the convolution calculation process is simulated in the figure. The convolution calculation proceeds normally when the sliding window slides to the portion where there is no overlaps with the ROI. When the sliding window slides to the portion overlapping with the ROI mark, each feature point is affected by the surrounding feature points. In other words, the feature point is enhanced according to the value of the surrounding feature points. The way in which we simplify the enhancement of eigenvalues is as follows.

There is an interaction between each feature point of the image. A thermonuclear function is first used to determine the weight between any two points that have a relationship:

$$w_{ij}=e^{-\frac{\left\|x_i-x_j\right\|^2}{t}} \tag{5}$$

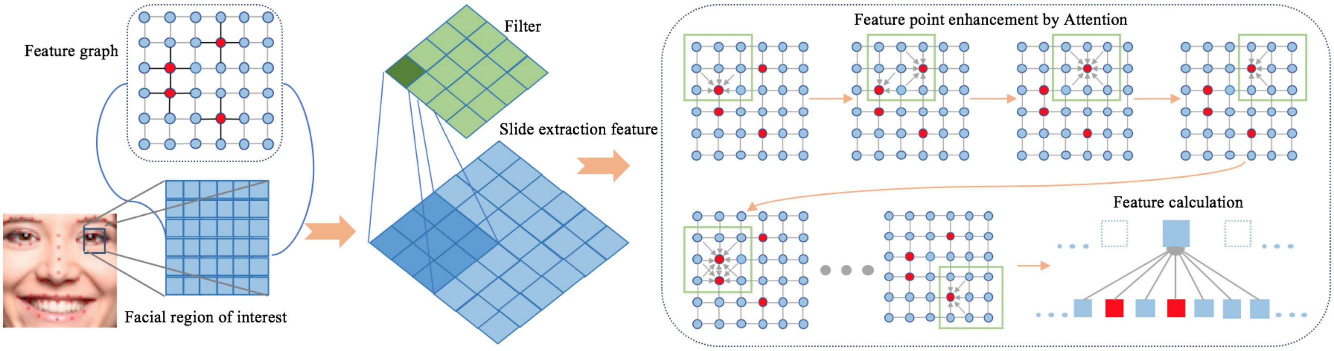
Feature graph
Filter
Slide extraction feature
Facial region of interest
Feature point enhancement by Attention
Feature calculation

FIGURE 3. When the sliding window slides to the portion overlapping with the ROI mark, each featurepoint is affected by the surrounding feature points and is enhanced.

After the weight relationship between any two points has been obtained, the influence coefficient μ_{ij}^{l} of the relative position on the feature weight need to be determined:

$$\mu_{ij} = \sqrt{\varphi_i \left|x\cos\theta\right|^2 + \varphi_j \left|y\sin\theta\right|^2} \tag{6}$$

where φ is the influence vector factor of the two points that affect each other. x and y represent the direction vectors of the feature points of the two-dimensional space, and θ is the relative angle between any two points. After obtaining the influence coefficient, we calculate the formula of the feature matrix after the entire image enhancement as:

$$X_j^l = \sum_{i=1}^{n} \rho w_{ij} \mu_{ij}^{l} \left(A + I_i\right) + \lambda^l \tag{7}$$

where X_i^l is the feature matrix to be output. $(A+I_i)$ represents a self-joined feature point structure adjacency matrix. ρ is an enhancement factor, which need to be constantly adjusted manually. W_{ij} is a layer-specific trainable weight matrix, and λ is a different weighting value between different parts of interest.

After the filter has been convolved to calculate each enhanced feature map value, a mapping matrix is needed to make our enhancement vector work. Therefore, the convolution calculation formula of the first layer of the neural network is as follows:

$$\begin{aligned} X_j^l &= f\left(R^{(l)} \sum_{i\in M_j} X_i^{l-1} K_j^l + B_j^l\right) \\ &= f\left(\sum_{i\in M_j} \left(\theta_0' + \theta_1' \left(L - I_i\right)\right) X_i^{l-1} K_j^l + B_j^l\right) \\ &= f\left(\sum_{i\in M_j} \left(\rho W_{ij} \mu_{ij}^{l-1} \left(A + I_i\right) + \lambda^{l-1}\right) K_j^l + B_j^l\right) \end{aligned} \tag{8}$$

where $R^{(l)}$ is an l-layer enhancement vector matrix, B is the bias matrix, and K is the weight matrix. For neurons in the j^{th} row and k^{th} column of the hidden layer, the output is:

$$\sigma\left(\sum_{l=0}^{L}\sum_{m=0}^{M} W_{l,m} a_{j+l,k+m} + b\right) \tag{9}$$

where σ is the activation function, b is the shared offset, $W_{l,m}$ is a 5x5 array of shared weights, and $a_{x,y}$ is used to represent the output value of the neurons in the x^{th} row and y^{th} column of the input layer, which is the several inputs to the neurons in the j^{th} row and k^{th} column of the hidden layer.

D. FEATURE VECTOR MAPPING

Feature mapping plays a vital role throughout the network. Our proposed method has more effective features in the previous section that are helpful for expression recognition. Next, we need to reduce the loss of feature information and ensure that the information input into the fully connected layer can more accurately reflect the expression information. As shown in fig.4, the initial features of the ROIs of each image extracted through the first layer are diviced into three parts, and input into the corresponding convolutional layers. The output feature vector is fed to the decoder, and the mapping and combination of feature vectors are completed by the decoder.

1) SECOND LAYER STRUCTURE

Three convolutional layers are used to extract three parts of the ROIs. The function of each convolutional layer is to extract the feature vector of the input part. Therefore, the convolutional layer is as follows:

$$Convolution_k^{l+1}(m_j, n_j) = RELU(\mu) \tag{10}$$

$Convolution_k^{l+1}(m_j, n_j)$ represents the convolution calculation result of the $(l+1)^{th}$ and k^{th} convolution vectors. *RELU* represents an activity function that takes into account the sum of the feature weights of the previous layer to pass them to the next layer. The activation function expansion is derived

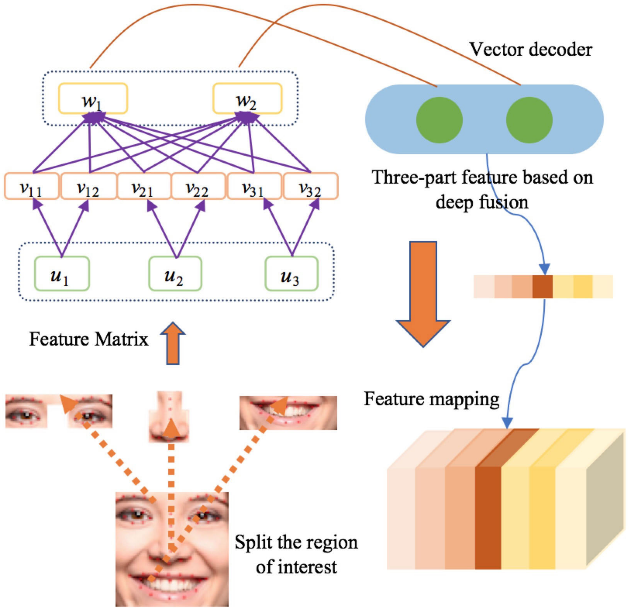

FIGURE 4. Corresponding convolutional layers is used to realize the fusion of features coming from ROIs, and vector decoder completes the effective semantic cascading.

as follows:

$$\begin{aligned} RELU\left(\mu\right) = & \sum_{(g=1)}^{z} \gamma\left(m_j, \left(n_j - g_j + \frac{z+1}{2}\right)\right) W_k^i \\ & + B_k^i \\ = & \sum_{(g=1)}^{z} \gamma\left(m_j, \left(n_j + \frac{z+1}{2}\right)\right) W_k^i \left(\theta_0' + \theta_1'\right) \\ & + B_k^i \end{aligned} \tag{11}$$

where γ is the map of the previous layer and z is the size of the kernel. W_k^i is a weight matrix, and B_k^i is a bias matrix.

2) FULLY CONNECTED LAYER

Algorithm 1 introduces the process of extracting efficient features and performing vector mapping. The focus of the algorithm is to use the attention method to extract more features that are helpful for recognition. The first two layers of the fully connected layer are activated using the ReLU function, mapping the input of the neuron to the output. This helps to reduce the possibility of gradient disappearance.

$$FC_j^{(l+1)} = \left(V_{ij}^l\right)^T RELU\left(\sum_{i=1}^{i} X_i^l W_{ij}^l + U_j^l\right) \tag{12}$$

W_{ij}^l and U_j^l are trainable weight parameters. Feature information is centrally stored in vectors. The last layer of fully connected layers is activated using the *sigmoid* function. When the last layer activates neurons, the *sigmoid* function can be used to suppress the occurrence of large difference errors.

Algorithm 1 Attention Enhanced Vector Mapping

Input: Weight matrices: W_i^j, w_i^j, Bias vectors: B_i^l, Total: n
Output: Space map vector V_k^l

```
Begin
Assign L := n
Initialized $W_{ij}$
for i, j in layer do
    Calculate $w_{ij}$ of any two points
    Calculate $\mu_{ij}$ of any two points
for i in layer do
    if j in ROIs then
        $M_k = Relu(R^{(l)} w_{ij} \mu_{ij} X_j + B_{ij})$
while $l \neq n$ do
    $y_k^l = Reshape(M_k)$
    $v_k^l = Relu(y_k^l W_k^l + B_k^l)$
    $V_k^l = Decoder(v_k^l)$
return $V_k^l$
```

$$FC_j^{(l+1)} = \left(v_{ij}^l\right)^T Sigmoid\left(\sum_{i=1}^{i} x_i^l w_{ij}^l + u_j^l\right) \tag{13}$$

A *softmax* layer is connected after the fully connected layer, an 8-dimensional vector is output, and the corresponding image data belong to the probability value of each of the eight types of expressions.

$$S_j = Softmax\left(W_{ij}^l c^{k,ij}\right) = \frac{\exp\left(a_k\right)}{\sum_{k=1}^{T} \exp\left(a_k\right)} \tag{14}$$

E. LOSS FUNCTION

During network training, the loss function is needed to evaluate the training of the model in the weight update. The softmax activation function is used for classification in our last layer of fully connected layers. The logarithm likelihood function is used to represent its loss function:

$$\begin{aligned} J\left(w, b, a^i, y\right) &= -\sum_k y_k \ln a_k^i \\ &= -\ln a_i^l \end{aligned} \tag{15}$$

where y_k is 0 or 1 if the output of a training sample is the i^{th} class. Then, $y_i = 1$, and the remaining $j \neq i$ have $y_i = 0$, where i is the actual serial number of the training sample. As shown, the loss function is only related to the output corresponding to the real category; thus, if the real category is the i^{th} class, the gradient derivatives of other neurons not belonging to the i^{th} class are directly zero. For the real class l, the gradient corresponding to the corresponding j^{th} w link

w^l_{ij} is calculated as:

$$\begin{aligned}\frac{\partial J\left(w,b,a^l,y\right)}{\partial w^l_{ij}} &= \frac{\partial J\left(W,b,a^l,y\right)}{\partial a^l_i}\frac{\partial a^l_i}{\partial z^l_i}\frac{\partial z^l_i}{\partial w^l_{ij}} \\ &= -\frac{1}{a^l_i}\frac{\left(e^{z^l_i}\right)\sum_{j=1}^{nl} e^{z^l_j} - \left(e^{z^l_i}\right)^2}{\left(\sum_{j=1}^{nl} e^{z^l_j}\right)^2} a^{l-1}_j \\ &= -\frac{1}{a^l_i a^l_i \left(1-a^l_i\right) a^{l-1}_j} \\ &= \left(a^l_i - 1\right) a^{l-1}_j \qquad (16)\end{aligned}$$

IV. EXPERIMENT

A. DATA DESCRIPTION

We evaluated and trained the hand-labelled and publicly released face databases AffectNet,[1] CK,[2] Fer2013[3], and Jaffe.[4] The AffectNet database contains more than 1,000,000 facial images from the Internet, and approximately half of the data annotations are performed manually. AffectNet is by far the largest facial expression database [32]. We selected about 20,000 images in AffectNet for experimentation. Fer2013 is derived from a database on Kaggle. We obtained all the images in Fer2013 through a csv file. The Jaffe database is relatively small. The database consists of ten different characters, and each character has three different expressions. There are 593 sequences across 123 subjects, which are FACS coded at the peak frame in the CK database. Each type of expression is randomly selected from each section. To unify the four databases, we performed tests in seven categories. We observed that the data that we obtained contain considerable noise and are not suitable for training. We need to perform some pre-processing and further labelling of the data prior to use.

We use dlib for face recognition and interception, and preliminary processing of the data preserves data that identify facial expressions. As shown in Table 2, a total of 49143 labelled data points were retained. The types and proportions are listed in the table. Before using the data, we manually checked the remaining images for manual labelling and removed some data that are not accurate enough. To maintain a balance between samples, we used oversampling to enhance the amount of data in a relatively small proportion.

TABLE 2. Eight types of expression statistics.

	NE	HA	SA	SU	FE	DI	AN
Fer2013	5572	8110	5483	3586	4593	492	4462
	11.3%	16.5%	11.2%	7.3%	9.3%	1.0%	9.1%
CK	64	64	64	64	64	64	64
	0.1%	0.1%	0.1%	0.1%	0.1%	0.1%	0.1%
AffectNet	3207	4132	1958	1816	1895	1404	1881
	6.5%	8.4%	3.9%	3.6%	3.8%	2.9%	3.8%
Jaffe	30	31	31	30	32	29	30
	0.1%	0.1%	0.1%	0.1%	0.1%	0.1%	0.1%

B. IMPLEMENTATION DETAILS

Our input data have 64 × 64 pixels per image. The encoders and decoders in our framework are improved by GoogLeNet Incepetion V1 to make them suitable for our data with a three-part network structure. The optimization method that we use is the Adam algorithm, where the learning rate starts at 0.001 and the decay rate is 0.99. The other hyperparameters are set to their default values.

Dropout [34] is applied to avoid overfitting at a rate of 0.6 and stops early on the validation set. We used some oversampling methods to balance the database during the training process to avoid deviations between the training set and the verification set [35]. We used some mechanisms in the early stage of training: we marked the ROIs in the image, which can reduce the number of iterations and accelerate the network learning. The entire process takes approximately four weeks on a four TITAN X GPU machine. We used the TensorFlow deep learning framework[5] to build our network structure.

C. REGION OF INTEREST COMPARISON EXPERIMENT

As shown in Table 3, a total of three different databases are used for verification experiments. The data in the table show the average recognition rates for each type of expression under different methods for the different databases. The original method is that we do not use Attention to enhance feature extraction of ROIs. Specifically, we separately use attention enhancement for the nose, eye, and mouth ROIs, and then we use attention enhancement for the three ROIs.

Comparing these five methods, we find that on all three databases, all the enhancements in all areas will improve the recognition rates for various expressions. Moreover, enhancing different parts of interest is different for each type of expression effect. For example, the enhancement of the characteristics of the nose part will only have a significant effect on the angry expressions. Because the type of expressions vary greatly in the nose, which have more details. For the enhancement of the mouth features, it will greatly help the recognition of happy, surprise and angry expressions. Enhancing the eye features will greatly help the expressions recognition of surprise and fear. The happiness recognition rate is higher on the three databases than that of other expressions because the happy expression features are more obvious and there are clear, easy-to-recognize features in the interest areas, such as the mouth, eyes and nose.

[1] http://mohammadmahoor.com/databases-codes/

[2] Website: http://www.pitt.edu/ emotion/ck-spread.html

[3] The database is from Kaggle

[4] http://www.kasrl.org/jaffe.html

[5] https://github.com/tensorflow/tensorflow/

TABLE 3. Enhance different regions of interest to compare the average recognition rates of different expression categories.

	AffectNet					Fer2013					Jaffe				
	Original	Eye	Mouse	Nose	All	Original	Eye	Mouse	Nose	All	Original	Eye	Mouse	Nose	All
NE	0.810	0.716	0.772	0.792	0.726	0.680	0.680	0.680	0.580	0.640	1.000	1.000	1.000	1.000	1.000
HA	0.868	0.836	0.829	0.833	0.824	0.799	0.804	0.835	0.819	0.860	1.000	1.000	1.000	1.000	1.000
SA	0.560	0.613	0.607	0.600	0.653	0.602	0.570	0.618	0.543	0.780	0.700	0.800	0.800	0.800	0.800
SU	0.680	0.680	0.720	0.620	0.720	0.705	0.773	0.748	0.670	0.750	0.800	0.600	0.800	1.000	1.000
FE	0.660	0.720	0.680	0.700	0.690	0.350	0.478	0.493	0.398	0.520	1.000	0.800	0.800	0.800	1.000
DI	0.443	0.540	0.560	0.520	0.537	0.500	0.600	0.610	0.58	0.680	0.800	0.800	1.000	0.800	0.800
AN	0.560	0.620	0.630	0.660	0.680	0.660	0.688	0.663	0.700	0.750	1.000	1.000	1.000	1.000	1.000

TABLE 4. Adjust dropout to obtain the average accuracy of each type of expression.

	Met. A	Met. B	Met. C	Met. D	Met. E
NE	0.795	0.685	0.690	0.726	0.665
HA	0.865	0.790	0.810	0.824	0.880
SA	0.520	0.640	0.628	0.653	0.540
SU	0.670	0.670	0.680	0.720	0.700
FE	0.490	0.530	0.649	0.690	0.620
DI	0.500	0.360	0.500	0.537	0.500
AN	0.600	0.620	0.630	0.680	0.620

TABLE 5. Adjust database size to obtain the average accuracy of each type of expression.

	Met. A	Met. B	Met. C	Met. D
NE	0.572	0.597	0.693	0.726
HA	0.781	0.813	0.815	0.824
SA	0.513	0.593	0.668	0.653
SU	0.428	0.514	0.573	0.720
FE	0.443	0.526	0.589	0.690
DI	0.512	0.598	0.615	0.537
AN	0.531	0.603	0.629	0.680

D. ADJUSTMENT PARAMETER EXPERIMENT

1) DROPOUT EXPERIMENT

We set the starting value of the dropout to 0.3, 0.4, 0.5, 0.6, and 0.7. The starting value of the dropout for Met. A is 0.3, Met. B is 0.4, Met. C is 0.5, Met. D is 0.6, and Met. E is 0.7, and the number of iterations is 50. The initial learning rate is set to 0.001, and the attenuation rate is 0.99. Except for the starting value of dropout, all other parameters are the same. As shown in Table 4, we experiment on the AffectNet database, and the dropout rate is too small, which leads to overfitting of the network, thereby reducing the recognition accuracy. When the dropout rate is too large, it will cause the network to lose too many new features, also reducing the accuracy. Thus, we ended up setting the initial dropout value to 0.6, which provides better results.

2) LEARNING RATE EXPERIMENT

During the experiment, we observed the change in convergence rate by constantly adjusting the learning rate. We used the AffectNet database during the experiment such that we could quickly observe the effect of learning rate on the convergence speed. We randomly took 1500 images as test set and the other as training set. The learning rate has a decay rate of 0.99.

As shown in fig.5, Met. A corresponds to an initial learning rate of 0.05, Met. B corresponds to an initial learning rate of 0.01, and Met. C corresponds to an initial learning rate of 0.005. The initial learning rate of Met. D is 0.001, and the initial learning rate of Met. E is 0.0005. In the change of the first 80 epochs, we find that the greater the initial learning rate setting is, the faster the network reaches convergence. Thus, the training must be terminated; otherwise, a gradient explosion will occur, and the average accuracy will decrease rapidly. After adjustment, we found that setting the initial learning rate to 0.001 will obtain a relatively good effect.

3) ADJUST DATABASE SIZE EXPERIMENT

We experimented on the AffectNet database. Our test data consisted of 500 images of each type randomly selected from the original database, and the training data consisted of 25%, 50%, 75%, and 100% of the data that we have compiled. All hyperparameters remained the same as the initial settings. As shown in Table 5, Met. A corresponds to a data amount of 25%, Met. B corresponds to a data amount of 50%, Met. C corresponds to a data amount of 75%, and Met. D corresponds to a data amount of 100%. According to the experimental results, the AffectNet database has a large amount of data and contains rich feature information. When the amount of data is extremely small, the learning effect of some complex features is very poor. Feature learning of complex expressions requires more data support. As the amount of data increases, the accuracy of expression recognition, such as happiness,

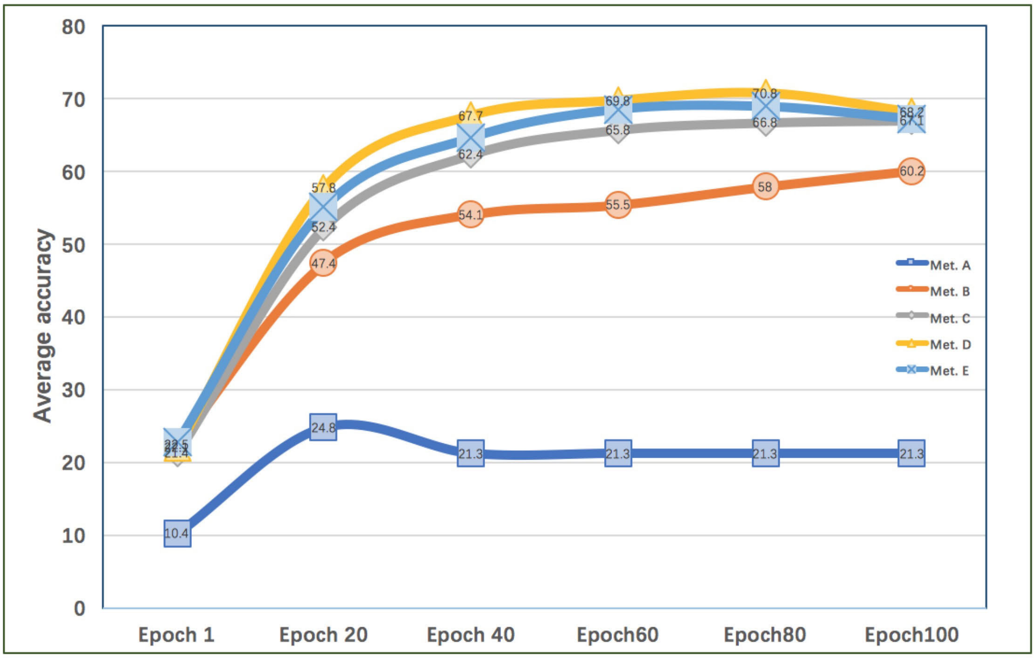

FIGURE 5. The average accuracy with different Met, Met.A resulting in the worst accuracy while Met.D perform best.

will not be greatly improved, but the more difficult expressions of surprise and Fear will follow the data. The amount has increased and has been greatly improved.

E. CROSS-DATABASE EXPERIMENT

To verify the robustness and generalization ability of the model, we conducted a cross-database verification experiment. We trained on the Fer2013 database and AffectNet database, tested it on the Jaffe database. We set the initial learning rate to 0.01 and the dropout to 0.5. All images in the Fer2013 database were trained and iterated 100 times. The data of the disgust-like expression image were enhanced by random cropping and flipping. As shown in Table 6, the recognition rate across databases is generally relatively low. Shan *et al.* [9] extracted the LBP features of the images on the Cohn-Kanade database, trained them with the SVM classifier, and finally verified them on the Jaffe database. Liu *et al.* [5] also trained on the CK+ database and tested on the Jaffe database. Their approach is to train with a boosted deep belief network that achieves an average recognition rate of 68% across the database. Our method, on the cross-database, performs better than the first two methods, which proves that our method is very robust and better for realworld applications.

F. COMPREHENSIVE EXPERIMENT

We conducted a comprehensive experiment on the four databases. In fig.6, we separately list the consistency between the two categories of the classification model on the four databases. As shown in this figure, on the AffectNet database, the happiness expressions show a high degree of consistency and are easier to distinguish. There are many cross-over features between the expressions of surprise and fear in the database, and the model shows a high error in distinguishing between these two types of features. The recognition rate of the other categories in the experiment cannot be increased, and the database contains many complicated and subtle emotional features; thus, there is a large obstacle to improving the recognition rate. On the CK database, the model performs well on the expressions of the happy class in the database. There are many similarities between the neutral expression and the features of disgust, sadness, and anger. Therefore, the recognition effect of the model in these types of expressions is relatively high. On the Fer2013 database, the recognition of fear-like expressions is quite poor. Because the data consistency of the three types of expressions of fear, sadness and anger is relatively low, it is difficult for the model to accurately distinguish these types of expressions. In contrast, both the happy and surprised expression data are consistent and easier to distinguish. In the Jaffe database, due to the small amount of data, it is easy to learn the characteristics of most expressions after a number of iterations. The three types of expressions of happiness, surprise and fear are highly consistent, while the normal expressions appear as several types of expressions. The recognition rate is the lowest because the images in it have some similarities to the images of surprise and anger.

G. COMPARISON EXPERIMENT WITH OTHER METHODS

In the comparison experiment, we implemented several methods by ourselves, which includes the use of SIFT [12] and HOG [11] algorithms to extract features; using SVM [16]

TABLE 6. **Experimental results on various data sets, many excellent methods are used for comparison.**

Method	CK database	Fer2013 database	AffectNet database	Jaffe database	Cross-database
Gabor+Template Matching [36]	78.9%	-	-	-	-
LBP+Template Matching [9]	79.1%	-	-	-	-
Gabor+LGC-HD+Template Matching [37]	86.60%	-	-	90%	-
SIFT+SVM	76.7%	57.3%	46.7%	81.7%	42.3%
HOG+XGBoost	83.2%	59.5%	55.6%	89.2%	36.6%
Tang et al. [38]	-	71.2%	-	-	-
Dhruv Amin et al. [39]	-	61%	-	-	-
Jinwoo Jeon et al. [40]	-	70.74%	-	-	-
Ali Mollahosseini et al. [32]	-	-	68.0%	-	-
ResNet-18	86.2%	63.6%	61.9%	89.9%	62.8%
Our Method	**87.2%**	**72.5%**	**70.0%**	**92%**	**69%**

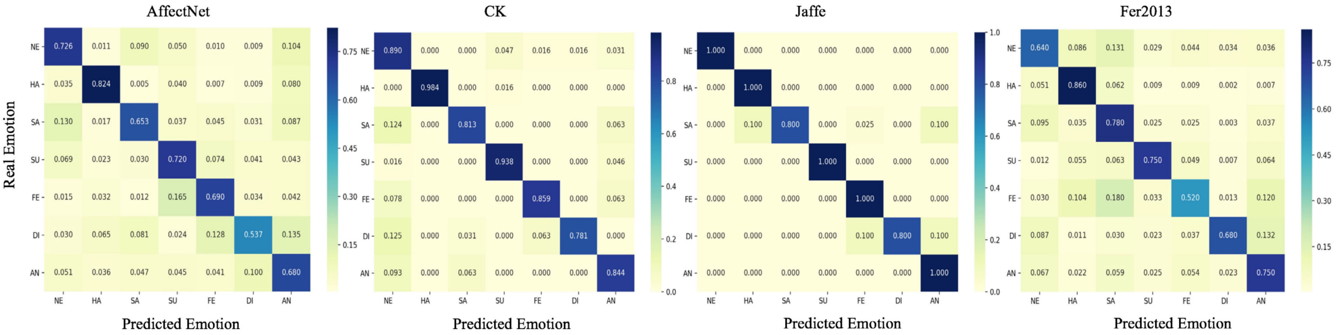

FIGURE 6. **Agreement between two annotators in categorical model of affect.**

and XGBoost, two classifiers for classification, we combine the feature extraction method and classifier for comparative experiments. The SIFT algorithm is the default parameter. HOG's orientations are set to 8, and $pixels\_per\_cell = (16, 16)$, $cells\_per\_block = (1, 1)$. The number of iterations of the XGBoost classifier is set to 1000 times, and $\gamma = 0.1$, $\lambda = 2$, $max\_depth = 6$. SVM iterations are set to 10,000 times, and $kernel = rbf$, $decision\_function = ovr$. The other parameters not mentioned are their default values.

We compare our method with other methods on the CK database. As shown in Table 6, Bartlett *et al.* [36] proposed a method for expressing expression using Gabor extraction features and template matching and obtained an average accuracy rate of 78.9%. Shan *et al.* [9] made some changes in the feature extraction method and used LBP to extract expression features, and they obtained an average accuracy rate of 79.1%. The method of improving the Gabor extraction feature, combined with the LGC-HD method [37], can obtain an average accuracy of 86.6% on seven types of expression recognition. Our method uses the CK database for training and testing. Each type of expression data takes 48 images as the training set and 16 images as the test set, which can obtain an average accuracy of 87.2%. As shown, our method performs better on the CK database than the other methods mentioned.

The Fer2013 database is also used for comparison, the method proposed by Tang [38] can achieve an average accuracy of 71.2%. Amin *et al.* [39] used the database to achieve an average accuracy of 61%. A method using DNNs proposed by Jeon *et al.* [40] can achieve an average accuracy of 70.74%. Our method performs slightly better than the other methods described above, achieving an accuracy of 72.5%.

The AffectNet database was just introduced in 2017. There are fewer research works that use this database in the experiment. Also in Table 6, we compare it with the baseline of the database proposed by Mollahosseini *et al.* [32], who used a neural network that improved AlexNet to achieve an average accuracy of 68%. We also conducted a centralized comparison test on our own, and the parameters of the four tests were introduced earlier. We also used an 18-layer ResNet network structure. We set the initial learning rate to 0.05 and the decay rate to 0.95. Using the L2 regular term coefficient, the weight decay rate was set to 0.0001. Using the down-sample method and using the ReLU activation function, after 100 iterations, the best result is a 61.9% average accuracy rate. Using our proposed method, the best result is an average accuracy of 70.0%, and the recognition effect is greater than the baseline.

On Jaffe database, due to its small amount of data, only 20 iterations are needed to reach convergence.Additionally, improving the Gabor extraction feature, combined with the LGC-HD method [37], can achieve an average accuracy of 90% on the Jaffe database. The proposed method can achieve an average accuracy of 92.0%, which exceeds the baseline.

V. CONCLUSION AND FUTURE WORK

In this paper, we present a novelty approach for facial expression recognition. We introduce three ROIs of face to help our model learning, including mouth, nose, and eyes in the face. In particular, we use the attention mechanism by determining the enhancement coefficient by calculating the distance relationship between the feature points and our marked ROIs. Once we get our proposed features matrix, the first layer of our model can extract more useful features for facial expression recognition. We also use the vector mechanism of CapsNet in the second part of the model to achieve the fusion of the semantic levels of the three parts of interest after the feature is initially extracted. This mechanism can retain complete feature information. Throughout the comparative experimental analysis, our proposed model had an excellent average recognition effect in four public databases, indicating good progress it has made in the effect of expression feature extraction compared with other methods. In addition, the proposed model also showed strong robustness in cross-database experiments.

In future work, we may focus on exploring the scalability of this approach and framework. For example, we try to subdivide the three ROIs into more small and related ROIs inspired by the truth that the superposition of small convolution kernels has been proved better than using large convolution kernels directly. We also notice that there are researches that use the attention mechanism to assign different weights to different channels of CNNs or fusing different features coming from different parts of the model that have made significant progress in their work separately. And our method in these two parts needs to be improved. Besides, increasing the depth of the network also helps to enhance the learning ability of the model for better results. Still, it is going to take a lot more work to figure out the way of making learning deeper networks more effective without increasing the data.

REFERENCES

[1] W. Liu and Z. Wang, "Facial expression recognition based on fusion of multiple Gabor features," in *Proc. 18th Int. Conf. Pattern Recognit. (ICPR)*, vol. 3, Aug. 2006, pp. 536–539.

[2] M. S. Kaushik and A. B. Kandali, "Recognition of facial expressions extracting salient features using local binary patterns and histogram of oriented gradients," in *Proc. 2017 Int. Conf. Energy, Commun., Data Analytics Soft Comput. (ICECDS)*, Aug. 2017, pp. 1201–1205.

[3] A. M. Ashir and B. Akdemir, "Facial expression recognition with an optimized radial basis kernel," in *Proc. 6th Int. Symp. Digit. Forensic Secur. (ISDFS)*, Mar. 2018, pp. 1–6.

[4] Y. Rahulamathavan, R. C.-W. Phan, J. A. Chambers, and D. J. Parish, "Facial expression recognition in the encrypted domain based on local Fisher discriminant analysis," *IEEE Trans. Affective Comput.*, vol. 4, no. 1, pp. 83–92, Jan. 2013.

[5] P. Liu, S. Han, Z. Meng, and Y. Tong, "Facial expression recognition via a boosted deep belief network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1805–1812.

[6] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim, "Joint fine–tuning in deep neural networks for facial expression recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2983–2991.

[7] W. Guojiang and Y. Guoliang, "A modified optical flow algorithm and its application in facial expression recognition," in *Proc. 3rd IEEE Int. Conf. Comput. Commun. (ICCC)*, Dec. 2017, pp. 1601–1605.

[8] G. Ramkumar and E. Logashanmugam, "An effectual facial expression recognition using HMM," in *Proc. Int. Conf. Adv. Commun. Control Comput. Technol. (ICACCCT)*, May 2016. pp. 12–15.

[9] C. Shan, S. Gong, and P. W. Mcowan, "Facial expression recognition based on Local Binary Patterns: A comprehensive study," *Image Vis. Comput.*, vol. 27, no. 6, pp. 803–816, May 2009.

[10] Y. Tong, R. Chen, and Y. Cheng, "Facial expression recognition algorithm using LGC based on horizontal and diagonal prior principle," *Optik*, vol. 125, no. 16, pp. 4186–4189, Aug. 2014.

[11] F. Xu and Z. Wang, "A facial expression recognition method based on cubic spline interpolation and HOG features," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2017. pp. 1300–1305.

[12] Q. Zhou, U. R. Shafiq, Y. Zhou, X. Wei, L. Wang, and B. Zheng, "Face recognition using dense sift feature alignment," *Chin. J. Electron.*, vol. 25, no. 6, pp. 1034–1039, Nov. 2016.

[13] L. Nwosu, H. Wang, J. Lu, I. Unwala, X. Yang, and T. Zhang, "Deep convolutional neural network for facial expression recognition using facial parts," in *Proc. IEEE 15th Intl Conf Dependable, Autonomic Secure Comput.*, Nov. 2017, pp. 1318–1321.

[14] V. Tumen, O. F. Soylemez, and B. Ergen, "Facial emotion recognition on a dataset using convolutional neural network," in *Proc. Int. Artif. Intell. Data Process. Symp. (IDAP)*, Sep. 2017, pp. 1–5.

[15] K. Shan, J. Guo, W. You, D. Lu, and R. Bie, "Automatic facial expression recognition based on a deep convolutional-neural-network structure," in *Proc. IEEE 15th Int. Conf. Softw. Eng. Res., Manage. Appl. (SERA)*, Jun. 2017, pp. 123–128.

[16] V. Varanya P and A. George, "Automatic recognition of facial expression using features of salient patches with SVM and ANN classifier," in *Proc. Int. Conf. Trends Electron. Informat. (ICEI)*, May 2017, pp. 908–913.

[17] M. H. Siddiqi, R. Ali, A. M. Khan, Y.-T. Park, and S. Lee, "Human facial expression recognition using stepwise linear discriminant analysis and hidden conditional random fields," *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1386–1398, Apr. 2015.

[18] L. Shang and K.-P. Chan, "Temporal exemplar–based Bayesian networks for facial expression recognition," in *Proc. 7th Int. Conf. Mach. Learn. Appl.*, 2008, pp. 16–22.

[19] X. Sun, M. Lv, C. Quan, and F. Ren, "Improved facial expression recognition method based on ROI deep convolutional neutral network," in *Proc.7th Int. Conf. Affect. Comput. Intell. Interact. (ACII)*, Oct. 2017, pp. 256–261.

[20] R. Zhi, Q. Ruan, and Z. Wang, "Facial expression recognition via sparse representation," *IEICE Trans. Inf. Syst.*, vol. E95.D, no. 9, pp. 2347–2350, 2012.

[21] N. Kauser and J. Sharma, "Facial expression recognition using LBP template of facial parts and multilayer neural network," in *Proc. Int. Conf. I-SMAC*, Feb. 2017. pp. 445–449.

[22] K. Verma and A. Khunteta, "Facial expression recognition using Gabor filter and multi-layer artificial neural network," in *Proc. Int. Conf. Inf., Commun., Instrum. Control (ICICIC)*, Aug. 2017, pp. 1–5.

[23] 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," *IEEE Access*, vol. 5, pp. 4525–4536, 2017.

[24] K. Chengeta and S. Viriri, "A survey on facial recognition based on local directional and local binary patterns," in *Proc. Conf. Inf. Commun. Technol. Soc. (ICTAS)*, Mar. 2018, pp. 1–6.

[25] G. Wang, "Facial expression recognition method based on zernike moments and MCE based HMM," in *Proc. 9th Int. Symp. Comput. Intell. Des. (ISCID)*, Dec. 2016, pp. 408–411.

[26] S. Kiranyaz, T. Ince, and M. Gabbouj, "Real–time patient–specific ECG classification by 1-D convolutional neural networks," *IEEE Trans. Biomed. Eng.*, vol. 63, no. 3, pp. 664–675, Mar. 2016.

[27] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[28] F. Deboeverie, S. Roegiers, G. Allebosch, P. Veelaert, and W. Philips, "Human gesture classification by brute-force machine learning for exergaming in physiotherapy," in *Proc. IEEE Conf. Comput. Intell. Games (CIG)*, Sep. 2016, pp. 1–7.

[29] K. Lekdioui, Y. Ruichek, R. Messoussi, Y. Chaabi, and R. Touahni, "Facial expression recognition using face-regions," in *Proc. Int. Conf. Adv. Technol. Signal Image Process. (ATSIP)*, May 2017, pp. 1–6.

[30] G. E. Hinton, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[31] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1867–1874.

[32] A. Mollahosseini, B. Hasani, and M. H. Mahoor, "Affectnet: A database for facial expression, valence, and arousal computing in the wild," Aug. 2017, *arXiv:1708.03985*. [Online]. Available: https://arxiv.org/abs/1708.03985

[33] R. Girshick, "Fast R–CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[34] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[35] L. A. Jeni, J. F. Cohn, and F. De La Torre, "Facing imbalanced data–recommendations for the use of performance metrics," in *Proc. Humaine Assoc. Conf. Affect. Comput. Intell. Interact.*, Sep. 2013, pp. 245–251.

[36] M. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan, "Recognizing facial expression: Machine learning and application to spontaneous behavior," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2005, pp. 568–573.

[37] S. Al-Sumaidaee, J. Chambers, S. Dlay, and W. Woo, "Facial expression recognition using local gabor gradient code-horizontal diagonal descriptor," in *Proc. 2nd IET Int. Conf. Intell. Signal Process. (ISP)*, 2015.

[38] Y. Tang, "Deep learning using linear support vector machines," Jun. 2013, *arXiv:1306.0239*. [Online]. Available: https://arxiv.org/abs/1306.0239

[39] D. Amin, P. Chase, and K. Sinha, *Touchy Feely: An Emotion Recognition Challenge*. Stanford, CA, USA: Stanford, 2017.

[40] J. Jeon, J.-C. Park, Y. Jo, C. Nam, K.-H. Bae, Y. Hwang, and D.-S. Kim, "A real-time facial expression recognizer using deep neural network," in *Proc. 10th Int. Conf. Ubiquitous Inf. Manage. Commun. (IMCOM)*, 2016, p. 94.

[41] M. Abdulrahman, T. R. Gwadabe, F. J. Abdu, and A. Eleyan, "Gabor wavelet transform based facial expression recognition using PCA and LBP," in *Proc. 22nd Signal Process. Commun. Appl. Conf. (SIU)*, Apr. 2014, pp. 2265–2268.

XIAO SUN was born in 1980. He received the M.E. degree from the Department of Computer Sciences and Engineering, Dalian University of Technology, in 2004, and the double Ph.D. degree from the Dalian University of Technology, China, in 2010, and the University of Tokushima, Japan, in 2009. He is currently a Professor with the Anhui Province Key Laboratory of Affective Computing and Advanced Intelligent Machines, Hefei University of Technology. His research interests include affective computing, natural language processing, machine learning, and human–machine interactions.

SHIXIN ZHENG was born in 1996. He received the B.E. degree from the Hefei University of Technology, in 2018, where he is currently pursuing the master's degree with the Anhui Province Key Laboratory of Affective Computing and Advanced Intelligent Machines. His research interests include affective computing, machine learning, and human–machine interactions.

HONGSHUAI FU was born in 1997. He is currently pursuing the B.S. degree with the School of Computer and Information, Hefei University of Technology, China. His research interests include affective computing, natural language processing, machine learning, and human–machine interactions.