2018 Australian & New Zealand Control Conference (ANZCC)
Swinburne University of Technology
Melbourne, Australia, Dec 7-8, 2018

Face Expression Recognition Based on Convolutional Neural Network*

Lei Xu, Minrui Fei, Wenju Zhou, Aolei Yang

***Abstract*—In order to reduce the complexity for extracting artificial features from the face image in facial expression recognition (FER), a novel method is proposed based on convolutional neural network (CNN) in this paper. This method first preprocesses the facial expression images, then some trainable convolution kernels are used to extract facial expression features, and second, the largest pooling layer is used to fewer dimensions, finally seven types of facial expressions are recognized with the Softmax classifier. The proposed method is verified with Kaggle facial expression recognition challenge dataset (FER2013). The experimental results show that the method has good recognition performance and generalization ability.**

I. Introduction

Facial expression indicates human emotion in communication. Facial expression recognition, as the key technology of the emotional computing system, is not only used to human-computer interaction, but also extended to the field of interactive game platforms, safe driving, smart recommendation and auxiliary medical care, etc. It is showing a good application prospect in various fields.

The FER usually includes three steps: the face detection, the feature extraction and the expression classification. Among them, the feature extraction of facial expression plays a key role in the expression recognition system, which affects the recognition accuracy. Recently, the research on facial expression feature extraction is becoming a hot topic, and many methods have been proposed, such as Active Shape Model (ASM) [1], Active Appearance Models (AAM) [2], Local Binary Pattern (LBP) [3], Gabor Filter [4], principal component analysis (PCA) [5], and Histograms of oriented gradients (HOG) [6], etc. All methods mentioned above need be operated by people to extract features, which leads to that some of the original features are lost. In the field of machine learning, data-driven feature learning algorithm has been proposed that is first to use deep learning to extract features. Unlike traditional machine learning algorithms, deep learning can automatically extract facial features without human participation. The essential features can be characterized autonomously from the sample data by the multilayered deep neural network. The performance of extracting features used deep learning method is better than that of traditional machine learning algorithms, and it is published by many literatures [7].

In 2006, Master of Machine Learning and Professor Hinton in University of Toronto in Canada first proposed the theory of deep learning, and published a paper with respect to using deep-structured neural network models to achieve dimensionality reduction in Science [8]. Hinton believes that many hidden layers in artificial neural networks have excellent capabilities to learn features, which may be more conducive to visualization and classification. Hinton also thinks that the difficulty with respect to deep neural network training can be effectively overcome through layer-by-layer initialization. Deep learning is a class of methods, which is generally designed, for training deep-structured models. The deep structure model represents features with multi-layers. It has stronger characterization capabilities than that of the shallow structure model. Typical deep learning models include Convolutional Neural Networks (CNN) [9], Deep Belief Networks (DBN) [10], Stacked Auto-encoder (SAE) [11], Recursive Neural Networks (RNN) [12] and so on. CNN is a deep neural network containing input layer, convolutional layers, pooling layers, fully connected layer and output layer. It is inspired from brain neuroscience, and imitates the process to handle visual information with two types of neural cells, simple cells and complex cells, in the visual cortex. The CNN contains basic convolution operations and pooling operations. The convolution operation is to simulate simple cells and the pooling operation is to simulate complex cells. In CNN, the subblocks (local receptive areas) are the input with respect to the lowest layer of the hierarchical structure in the image, and the information is transmitted to different layers through layer by layer. The most significant feature of the observed data is obtained by a digital filter in every layer.

This paper presents a facial expression recognition method based on CNN. Firstly, face detection and preprocessing are performed on the facial expression image. After that, the features of the facial expression are extracted using some trainable convolution kernels. Then, the extracted face expression feature is reduced by the maximum pooling method. Finally, the Softmax classifier is used to categorize facial expression images, and facial expressions are classified into seven types of facial expression: happiness, surprise, sadness, anger, fear, disgust and neutrality.

II. CNN Model Structure Design

This paper designs a CNN model structure for facial expression recognition, as shown in Figure 1. CNN is a feed-forward neural network that extracts features from a two-dimension image and uses a back-propagation algorithm to optimize the network parameters. The model consists of a 7-layer network: 3 convolutional layers (C1, C2 and C3), 2 layered pooling layers (P1 and P2), 1

* This work is supported by Natural Science Foundation of Shanghai (No.18ZR1415100), National Science Foundation of China (61473182, 61773253), Science and Technology Commission of Shanghai Municipality (15JC1401900) and Key research and development project of Yantai (2017ZH061).

Lei Xu, Minrui Fei, Wenju Zhou, and Aolei Yang are with Shanghai Key Laboratory of Power Station Automation Technology, School of Mechatronic Engineering and Automation ,Shanghai University, Shanghai 200072, China (e-mail: xulei_teng@163.com, mrfei@staff.shu.edu.cn, aolei@shu.edu.cn, zhouwenju2004@126.com).

Corresponding author: Wenju Zhou (zhouwenju2004@126.com)

full-connected layer, and 1 Softmax layer. The input is a 48×48 face pixel matrix. The convolutional layer and the pooling layer have several feature maps. Each feature map is connected to the previous layer with a partially connected manner. Convolution layers C1, C2, and C3 operate with 64, 64, and 128 convolution kernels, respectively, where the convolution kernels of C1 and C2 have a size of 5×5, and the convolution kernel of C3 has a size of 4×4 . The pooling layers P1 and P2 use a window size of 2×2 . There are 4608 neurons in the fully connected layer, which is fully connected to the pooling layer P2. The Softmax layer contains seven neurons and classifies the features of the fully connected layer, and the facial expressions are classified into seven types of facial expressions: happiness, surprise, sadness, anger, fear, disgust, and neutrality.

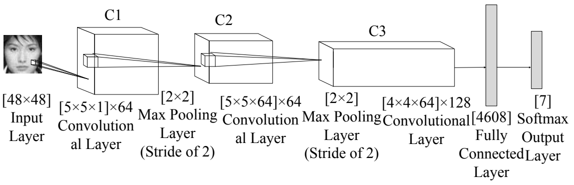
C1
C2
C3
[48×48] Input Layer
[5×5×1]×64 Convolutional Layer
[2×2] Max Pooling Layer (Stride of 2)
[5×5×64]×64 Convolutional Layer
[2×2] Max Pooling Layer (Stride of 2)
[4×4×64]×128 Convolutional Layer
[4608] Fully Connected Layer
[7] Softmax Output Layer

Figure 1. CNN CNN structure for face expression recognition

A. Convolution Layer

The statistical characteristics of different subblocks in the natural image are usually consistent, which means that the features learned from a certain subblock of the image can be used as a detector. All the subblocks of the full image are traversed to obtain the activation value with the same feature. Different trainable convolution kernels are used to perform convolutional summation over all the feature maps with respect to the previous layer and add offsets. Then the neuron in the current layer is output by the activation function. The feature maps with the different features are constituted in the current layer. In general, the calculation expression of the convolutional layer is:

$$y_j^l = \theta(\sum_{i=1}^{N_j^{l-1}} w_{i,j} \otimes x_i^{l-1} + b_j^l), j = 1,2,...,M \tag{1}$$

where, the l is the current layer, the $l-1$ is the previous layer; y_j^l represents jth feature map in the current layer; $w_{i,j}$ represents convolution kernel between ith feature map in the previous layer and jth feature map in the current layer; x_i^{l-1} represents ith feature map in the previous layer; b_j^l represents the bias of jth feature map in the current layer. In order to train the network quickly and reduce the learning parameters, the term b_j^l is set to zero, i.e. $b_j^l = 0$. N_j^{l-1} represents the number of connecting jth feature maps in the current layer to all feature maps in the previous layer; M represents the number of feature maps in the current layer; $\theta(\square)$ represents the activation function; Due to the Rectified Linear Units (ReLU) function is more sparse, it is used to instead of usual sigmoid or hyperbolic tangent function. The expression for the ReLU function is:

$$\theta(x) = \max(0, x) \tag{2}$$

ReLU can well solve the problem that caused by gradient disappearing, which is inevitable the traditional activation function produce in the process of back propagation parameters adjustment. It also can improve the convergence speed of the network. The convolutional layer is a feature extraction layer. The input of each neuron is connected with the local receptive field in the previous layer, and the local features are extracted from the previous layer. The convolutional layer C1 uses a 5×5 convolution kernel to perform a convolution operation on 48×48 pixel input images. Since the convolution striding is 1, the size of the feature map is $(48-5+1) \times (48-5+1) = 44 \times 44$. 64 expression features are corresponding to 64 feature maps that are obtained from the convolution operation using 64 different convolution kernels. Feature maps are produced from the pooling layer P1 or P2. Each neuron in the same feature map uses the same convolution kernel, but they receive date from different local receptive fields. The convolutional layer C2 uses 64 convolution kernels of size 5×5 to convolute the feature map. The size of each feature map is $(22-5+1) \times (22-5+1) = 18 \times 18$. The convolutional layer C3 uses 128 convolution kernels of size 4×4 to convolute the feature map. The size of each feature map is $(9-4+1) \times (9-4+1) = 6 \times 6$.

B. Pooling Layer

The number of feature maps increases with the number of convolution layer, which leads to a sharp increase in feature dimensions. If all the features are used to train the Softmax classifier, it will cause enormous dimensions. In order to avoid this problem, a pooled layer is usually used to reduce the feature dimension. The pooling layer plays a role of down-sampling. The pooling layer does not change the number of feature maps, but shrink the feature maps. That means the pooling layer can robust some operation such as translation, scaling, and rotation. If the size of the sampling window is $n \times n$, the feature map would become $(1/\mathrm{n}) \times (1/\mathrm{n})$ of the original feature after down-sampling. The general expression for the pooling is:

$$y_j^l = \theta(\beta_j^l down(y_j^{l-1}) + b_j^l) \tag{3}$$

Where, y_j^l represents jth feature map in the current layer and y_j^{l-1} represents the previous feature map in the current layer; $down(\square)$ represents a down-sampling function; β_j^l represents the multiplicative bias and b_j^l represents additive bias to the j feature map in the current layer. In the experiment, $\beta_j^l = 1$, $b_j^l = 0$; $\theta(\square)$ represents the activation function; Identical functions are used in the experiment. The pooling layer P1 is obtained by down-sampling the feature map in the convolution layer C2 using a 2×2 window, so the feature map size is 22×22 . Similarly, the pooling layer P2 performs a same operation as the pooling layer P1 .

C. Fully Connected Layer

The input of the fully connected layer is a one-dimensional array. The feature map in convolutional layer C3 is a two-dimensional array that is output by the pooled layer P2. The neuron in feature map is corresponding to a point in fully connected layer. Then 128 one-dimensional arrays are concatenated into a-dimensional feature vector in the fully connected layer. The output of each neuron is:

$$h_{w,b}(x) = \theta(w^T x + b) \tag{4}$$

where, $h_{w,b}(x)$ represents the output value of the neuron; x represents the input feature vector of the neuron; w represents the weight vector; b represents bias, and it is set as $b = 0$; $\theta(\square)$ represents the activation function, which is ReLU function in the paper.

D. Softmax Layer

The Softmax classifier is used in the last layer of CNN, and it is a multi-output competitive classifier. Inputting a given sample, each neuron will output a rate between 0 and 1. The facial expression can be categorized according to the rate.

III. CNN Parameter Training

CNN has input-to-output mapping relationships, which are learned based on a large number of mappings between inputs and outputs. It often does not require precise mathematical expressions between the input and output. It is needed to be trained in a known pattern to get these mapping capabilities. The network's weight is initialized with random numbers before starting training.

The training of CNNs is divided into two stages:

- *Forward propagation* A sample x is extracted from the training sample set, and its corresponding class label is y , x is input to the CNN network, and the output of the upper layer is the input of the current layer. The output of the current layer is calculated through the activation function and passed downed layer by layer. Finally, the output of the Softmax layer is $\tilde{y}$ with a 7-dimensional vector, which represents the probability to classify x into each type of facial expressions.
- *Back propagation.* The error between the output $\tilde{y}$ of the Softmax layer and the class label vector y is calculated. It is propagated back. Only the element corresponding to the category label y is 1, and the other elements are 0 in y which is a 7- dimensional vector. The weight parameter is adjusted by minimizing the mean square error cost function.

IV. Analysis of Experimental Results

The experiments are operated on the TensorFlow framework. TensorFlow is a system that transmits complex data structures to artificial intelligence neural networks for analysis and processing. The training samples and test samples are selected from the FER2013 [13]. A total of 26,820 facial expression images are used as the training samples, of which 3300 are anger, 420 are disgust, 3600 are fear, 7200 are happiness, 4800 are sadness, 3000 are surprise and 4500 are neutral. All the facial expression images in FER2013 are cropped and normalized to 48×48 pixel grayscale images, as shown in Figure 2.

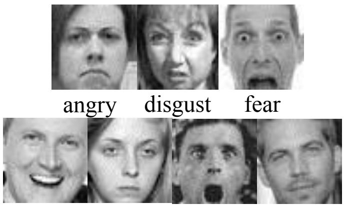

Figure 2. Samples of the seven basic expression images used in the experiments

In order to improve the reliability of the recognition results, three cross-validation methods are used in the experiment. The 26820 expression images are equally divided into three, each containing 7 type facial expression images. Two of them are used as training sample sets and the remaining one is used as a test sample set. This facial expression recognition experiment is repeated 3 times, and the average recognition rate of 3 times is taken as the final facial expression recognition results, as shown in table I.

As we can see from the table I, facial expression recognition accuracy rates are very high on happy (91%), neutral (83%), surprised (78%). Apparently these are the most distinguishable facial expressions. However, sad, fearful and angry are often misclassified as neutral, because these facial expressions look very alike. Facial expression recognition accuracy rates are very low on sad (41%) and fearful (43%). The rates on the main diagonal represent the accuracy of facial expressions classification.

TABLE I. Performance Matrix of the CNN Model

Real Facial Expressions	angry	disgusted	fearful	happy	sad	surprised	neutral
neutral	0.05	0.00	0.04	0.05	0.02	0.01	0.83
surprised	0.04	0.02	0.10	0.04	0.03	0.78	0.04
sad	0.13	0.02	0.08	0.08	0.41	0.00	0.32
happy	0.03	0.00	0.02	0.91	0.00	0.06	0.04
fearful	0.11	0.01	0.43	0.04	0.05	0.12	0.21
disgusted	0.12	0.66	0.02	0.13	0.00	0.02	0.07
angry	0.54	0.09	0.08	0.04	0.06	0.01	0.18

Predicted Facial Expressions

There is a problem of over-fitting during the training of deep neural network models. The dropout strategy is adopted to prevent over-fitting in the experiment. Over-fitting can also be prevented by data set augmentation strategies. In the training of the network, output value of the neuron in fully connected layer is cleared to 0 with a probability of 0.5. The weight associated with this neuron is no longer updated with the back propagation algorithm. The dropout and dataset augmentation strategies are adopted in the experiment. Figure 3 shows a comparison of the expression recognition rates of CNNs with different strategies.

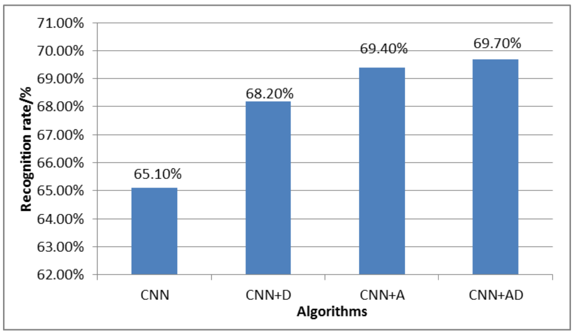

Figure 3. Comparison of GPU different algorithms on FER2013

The dropout and dataset augmentation strategies have better performance in preventing over-fitting and improving the generalization of the network.

In order to verify the performance of the proposed facial expression recognition method, we have established our own facial expression database, as shown in Figure 4. Facial expression images in the database we built are cropped and normalized to 48×48 pixel grayscale images before recognition. The proposed method is compared with some mature facial expression recognition methods. As shown in Figure 5, compared with the Multi-Layer Perceptron (MLP) and the local binary patterns with support vector machine (LBP +SVM), the proposed method has high average facial expression recognition rate.

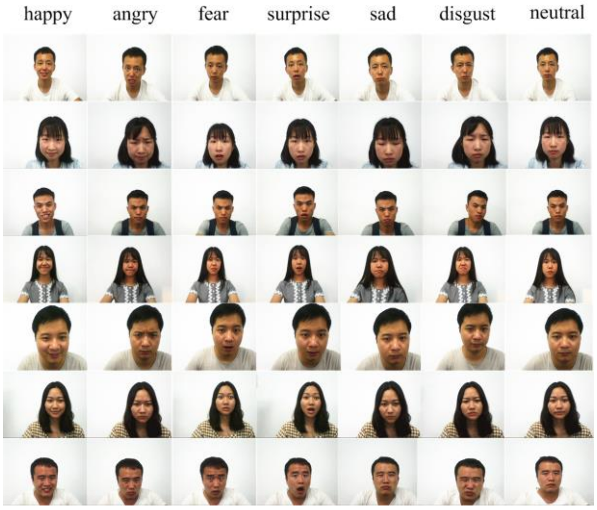

Figure 4. Samples of the seven basic expression images in our own facial expression database

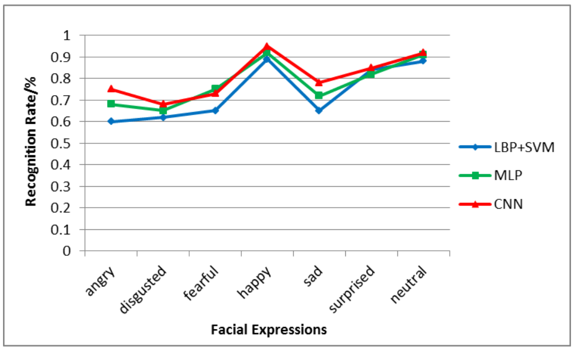

Figure 5. Recogniton rate of different facial expression recogniton methods

The CNN structure is conductive to parallel learning with sharing weight of the connection neurons in the feature map.

CNN has parallel data processing capabilities, which accelerates the training of CNNs and significantly reduces training time. It takes much less time with the GPU than that with CPU in training the CNN, because GPU can process the data in parallel. There is almost no difference in recognition rate between the two methods. Due to the network structure parameters, the training set and the test set are the same, the recognition rate is very close.

V. Conclusion

The CNN designed in this paper has a good recognition performance in recognizing human face expression. The CNN model is a multilayer perceptron, which has the characteristics of local perception, hierarchical structure, combination of feature extraction and classification. The expression recognition system based on CNN proposed in this paper can directly use the pixel value of the facial expression images as input. It obtain more abstract expression feature by learning the facial expression images autonomously. The process of complex artificial feature extraction in traditional facial expression recognition is avoided, and feature extraction and expression classification are performed simultaneously. Due to the connections between neurons are not fully connected, the same connections weights of neurons of feature map are used in one layer, and the complexity of the network model is reduced greatly. The pooling layer enhances the robustness of the CNN and can deal with some distortion of the image.

According to the experimental data and experimental conclusions, the next research work will further optimize the structure of CNN to improve the speed and accuracy of facial expression recognition.

References

[1] Y.-H. Lee, C. G. Kim, Y. Kim, and T. K. Whangbo, "Facial landmarks detection using improved active shape model on android platform," *Multimedia Tools and Applications,* vol. 74, pp. 8821-8830, 2015.

[2] C. Darujati and M. Hariadi, "Facial motion capture with 3D active appearance models," in *International Conference on Instrumentation, Communications, Information Technology, and Biomedical Engineering*, 2013, pp. 59-64.

[3] W.-L. Chao, J.-J. Ding, and J.-Z. Liu, "Facial expression recognition based on improved local binary pattern and class-regularized locality preserving projection," *Signal Processing,* vol. 117, pp. 1-10, 2015.

[4] X. Xu, C. Quan, and F. Ren, "Facial expression recognition based on Gabor Wavelet transform and Histogram of Oriented Gradients," in *Proceedings of the IEEE conference on Mechatronics and Automation*, 2015, pp. 2117-2122.

[5] R. Vidal, Y. Ma, and S. S. Sastry, "Principal component analysis," in *Generalized Principal Component Analysis*, ed: Springer, 2016, pp. 25-62.

[6] P. Carcagnì, M. Coco, M. Leo, and C. Distante, "Facial expression recognition and histograms of oriented gradients: a comprehensive study," *SpringerPlus,* vol. 4, p. 645, 2015.

[7] A. Mollahosseini, D. Chan, and M. H. Mahoor, "Going deeper in facial expression recognition using deep neural networks," in *Proceedings of the IEEE conference on Applications of Computer Vision*, 2016, pp. 1-10.

[8] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science,* vol. 313, pp. 504-507, 2006.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097-1105.

[10] P. Liu, S. Han, Z. Meng, and Y. Tong, "Facial expression recognition via a boosted deep belief network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1805-1812.

[11] Y. Wu and W. Qiu, "Facial expression recognition based on improved local ternary pattern and stacked auto-encoder," in *AIP Conference Proceedings*, 2017, p. 020131.

[12] S. Ebrahimi Kahou, V. Michalski, K. Konda, R. Memisevic, and C. Pal, "Recurrent neural networks for emotion recognition in video," in *International Conference on Multimodal Interaction*, 2015, pp. 467-474.

[13] P. Giannopoulos, I. Perikos, and I. Hatzilygeroudis, "Deep Learning Approaches for Facial Emotion Recognition: A Case Study on FER-2013," in *Advances in Hybridization of Intelligent Methods*, ed: Springer, 2018, pp. 1-16.