Facial Expression Recognition based on CNN

Qian Liu Jiayang Wang
{liuqian14, jy-wang14}@mails.tsinghua.edu.cn
The Department of Electronic Engineering, Tsinghua University

Abstract

Facial expression recognition has been an active research area recently, and many kinds of methods have been proposed. In this project, we mainly used two mainstream Convolutional Neural Networks, AlexNet and GoogLeNet, to recognize human facial expression and emotion. CNNs are capable of extracting powerful information about a facial image by using multiple layers of feature detectors. Based on the two CNNs, some novel and helpful methods are used in data preprocessing and model optimization. The recognition results show that the CNNs used in this project has a good performance in terms of accuracy.

I. INTRODUCTION

Facial expression is one of the most helpful features in human emotion recognition. It was first introduced by Darwin in [1]. In [2], facial expression was defined as the facial changes in response to a persons emotional state.

Facial expression recognition is a task that we human all do in our daily life, but it cannot be easily performed by the computers. With the fast innovation in computer vision, as well as the population of machine learning and deep learning, facial expression recognition is very potential and has been very active for nearly 10 years. Many researchers have been devoted to this area and quite a few methods are proposed. Nowadays, facial expression recognition has varieties of applications, such as interactive games, sociable robots and so on. This area is still potential and full of vitality.

The project report is organized as follows. In Section II, some related work will be introduced, mainly about different mainstream methods that are used today. Then in Section III, we will have a brief introduction of Convolutional Neural Networks and how it can be applied to facial expression recognition. Section IV - V will take an open dataset as an example to train the model of recognition. To be more specific, Section IV focuses on the dataset and data preprocessing, and Section V is mainly about how we adopt CNNs to train the model and the methods we use to improve the accuracy on test set. In Section VI, the conclusion will be drawn and we will introduce our future work.

II. RELATED WORKS

Facial expression recognition has developed its own methodology. As described in [2], facial expression analysis consists of mainly three parts: face acquisition, feature extraction and representation, and expression recognition. Some related works are shown respectively below.

i. Face acquisition

Face acquisition is mainly split into two parts: face detection and head pose estimation. Face detection is another field that used to be popular, and the methods have been very mature with low error rate [3]. As for head pose estimation, it can use relevant transformation to adjust the location and angle of the face []. Therefore, face acquisition is not hard to do.

ii. Feature extraction

Face acquisition mainly refers to the extraction of facial changes caused by facial expressions. These changes can be extracted by us-

ing some methods based on geometric features and change of appearance. Geometric feature-based methods mainly utilize the shape and location of facial components like eyes, eyebrows and mouth [4]. As for appearance-based methods, it always works with features extracted from the whole face or some regions by using image lters applied to the whole face image [5].

iii. Expression Recognition

Once feature extraction is finished, expression recognition can be performed. In [6], expression recognition is concluded as a three-step procedure: feature learning, feature selection and classifier construction. In each step, there are lots of related works. Feature learning is often combined with feature extraction, which prepare all the features related to the expression. Feature selection is tougher. Since expression recognition is often regarded as a classification problem, it requires that the features should minimize the intra-class variation as well as maximize the inter-class variation [7]. Finally, a classier or a set of classier are used to recognize the facial expression, based on the selected features.

Whats more, thanks to the appearance of deep learning, especially Convolutional Neural Networks [8], one of the deep learning approaches, several facial expression recognition approaches have been developed in the past decades with an increasing progress in recognition performance. There is no doubt that the most powerful and popular method has become Convolutional Neural Network, which will be introduced in the next section.

III. CONVOLUTIONAL NEURAL NETWORK

Convolutional Neural Network (CNN) was first proposed by Lecun et al [9]. It has been shown to be very effective in learning features when using deeper architectures and new training techniques. Generally, CNN includes convolutional layers, sub-sampling layers and fully connected layers.

Convolutional layers are usually characterized by the kernel's size. Sub-sampling layers are used to increase the position invariance of the kernels. The main types of sub-sampling layers are maximum-pooling and average pooling [10]. Fully connected layers are similar to the ones in general neural networks, its neurons are fully connected with the previous layer. The learning procedure of CNNs consists of nding the best synapses weights (W). Supervised learning can be performed using a gradient descent method.

CNN is the most popular technique that has been successfully applied to the facial expression recognition problem. This technique also comprises the three steps of facial expression recognition (learning and selection of features and classication) as stated in Section II in one single step. AlexNet and GoogLeNet are two of the popular CNNs in image classification. In this project, we use these typical CNN to perform the facial expression recognition.

IV. DATASET & PREPROCESSING

The dataset in this project is obtained from Kaggle. There was competition called *Challenges in Representation Learning: Facial Expression Recognition Challenge*.

The data consists of 48x48 pixel grayscale images of faces belonging to 7 different categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). Some examples of images are shown in figure 1.

Figure 1: Samples in the Dataset

The dataset is split into training set and test set, which contain 28,709 and 3,589 face images respectively. And the distribution of different categories in training set is shown in the table 1.

angry	disgust	fear	happy
3995	436	4097	7215
sad	surprise	neutral	
4830	3171	4965	

Table 1: The distribution of 7 Categories

Briefly, the purpose and task is to categorize each face into one of seven categories based on the emotion shown in the facial expression. But before utilizing these images, we are supposed to figure out if there are any preprocessing works should be done. Data processing is always a time-consuming work, but it is essential to do because it has a direct influence to the final performance. Luckily, the faces have been well processed. Firstly, the face images are more or less centered and occupies about the same amount of space in each image. Meanwhile, they are all normalized so no more work should be done.

V. RECOGNITION BASED ON CNN

In this section, we used AlexNet and GoogLeNet respectively to perform facial expression recognition, and got the optimized results by adopting several methods, including oversampling, SVM and so on. All the training and test process is performed on the Deep Learning Framework Caffe.

It is worth noting that the process is not very smooth sailing. At beginning, we chose to use the CNN in [11], and the network structure is shown in figure 2. Unfortunately, it seems that the training process is always unable to converge. After trying a long time, we gave up on this network and turned to AlexNet and GoogLeNet.

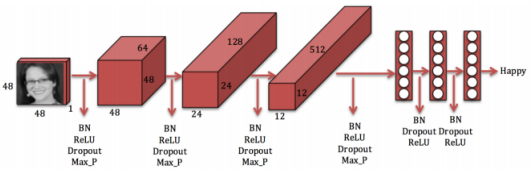

Figure 2: The first attempted CNN

i. AlexNet

The network structure of AlexNet is shown in figure 3. AlexNet requires that the size of input images is 224224. When the images were resized, we use the training set to train the network.

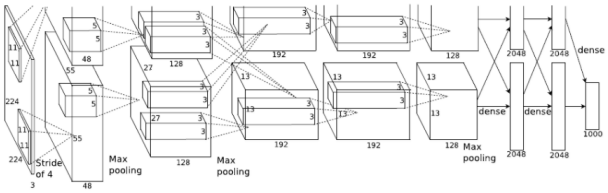

Figure 3: AlexNet

A) Baseline

After training for the first time, the results are shown below. It seems the results were not good enough.

```
Test net output #0: accuracy = 0.658611
Test net output #1: loss = 0.950391 (* 1 = 0.950391 loss)
```

Figure 4: Before fine-tuning

After fine-tuning, the results were not improved. But both of the two results are not overfitting.

```
Test net output #0: accuracy = 0.6575
Test net output #1: loss = 0.972126 (* 1 = 0.972126 loss)
```

Figure 5: After fine-tuning

We utilize confusion matrix to help get the results on test data set. The results are shown in the table below. And the total accuracy is 48.6%.

	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	**30.95%**	0.00%	11.40%	31.77%	9.57%	4.68%	11.60%
Disgust	25.45%	**7.27%**	14.54%	41.81%	5.45%	1.81%	3.63%
Fear	9.65%	0.00%	**30.87%**	27.84%	11.36%	10.98%	9.28%
Happy	1.25%	0.00%	2.27%	**91.69%**	1.36%	1.36%	2.04%
Sad	5.89%	0.00%	12.96%	39.05%	**26.26%**	1.68%	14.14%
Surprise	3.12%	0.00%	12.74%	16.10%	2.88%	**60.57%**	4.56%
Neural	4.95%	0.00%	5.91%	38.81%	12.46%	3.99%	**33.86%**

Figure 6: Results on test set

B) Oversampling

It is obvious that the recognition rate (recall) is extremely low in terms of the second expression (disgust). It turns out that the training set is imbalanced. We think SVM may be helpful in this situation. So we use the output probability vector as the eigenvectors of SVM, only to find it useless.

In order to solve this problem, we refer to [5] and perform oversampling. Here, we use the Smote algorithm to do oversampling. Among the class with least samples, we find k neighbor samples for each sample under a given criterion, e.g. Euclidean metric. Then we sample randomly from the k neighbor samples and generate a new sample using interpolation of the chosen sample and its chosen neighbors. By iteratively applying Smote algorithm to the original dataset, we then achieve a balanced dataset.

After random repetition, each of the expression has approximately 7,000 samples. Afterwards, the network is trained again. Though the results on training set nearly remain unchanged, the recall on the test set is apparently improved. But unfortunately, the problem of overfitting emerges.

```
Test net output #0: accuracy = 0.658333
Test net output #1: loss = 0.953317 (* 1 = 0.953317 loss)
```

Figure 7: After oversampling

	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	**30.14%**	0.81%	8.55%	16.29%	21.38%	5.09%	17.71%
Disgust	29.09%	**20.00%**	9.09%	10.90%	16.36%	0.00%	14.54%
Fear	6.06%	0.18%	**24.81%**	12.68%	25.94%	15.71%	14.58%
Happy	0.34%	0.00%	1.02%	**88.62%**	3.29%	3.75%	2.95%
Sad	4.04%	0.00%	7.23%	15.65%	**49.49%**	1.51%	22.05%
Surprise	0.96%	0.00%	7.93%	6.25%	4.08%	**74.03%**	6.73%
Neural	1.91%	0.00%	1.11%	12.77%	17.09%	2.39%	**64.69%**

Figure 8: Results on test set

C) Creating a New Label

To help prevent overfitting, we use another method by splitting happy category into two groups to create a new label. After training, then we combine happy and the new label together. After doing this, the accuracy raises up to 50.0%, and the recognition rate (recall) of disgust changes from 20.0% to 47.27%. The effect has been well improved.

	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	**34.62%**	2.44%	16.49%	13.02%	13.03%	4.27%	16.08%
Disgust	25.45%	**47.27%**	10.90%	3.62%	5.45%	1.81%	5.45%
Fear	7.19%	1.13%	**33.90%**	10.78%	14.20%	19.31%	13.44%
Happy	1.25%	0.11%	2.73%	**87.25%**	2.27%	2.27%	4.09%
Sad	6.22%	0.16%	19.19%	12.45%	**35.18%**	3.03%	23.73%
Surprise	1.92%	0.24%	5.76%	6.72%	1.44%	**79.56%**	4.32%
Neural	4.15%	0.31%	7.98%	11.01%	9.90%	3.35%	**63.25%**

Figure 9: Results on test set

ii. GoogLeNet

GoogLeNet proposes the Inception Module, which can enhance the function of feature extraction. The structure is shown in figure 10.

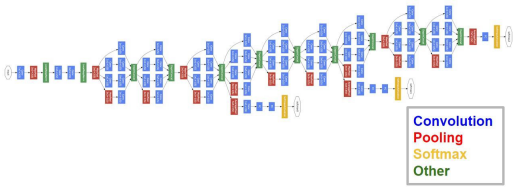

Figure 10: GoogLeNet

We use the same method to cope with the imbalanced dataset, and train the model using GoogLeNet. The results on the test set is very nice. The recognition rate (recall) is obviously higher than the AlexNet. The accuracy raises up to 61.6%.

	Angry	Disgust	Fear	Happy	Sad	Surprise	Neutral
Angry	**58.03%**	0.40%	12.84%	3.00%	7.28%	5.15%	13.27%
Disgust	23.21%	**53.57%**	10.71%	1.79%	7.14%	0.00%	3.57%
Fear	10.48%	0.20%	**44.55%**	2.62%	13.31%	13.91%	14.92%
Happy	3.01%	0.11%	3.01%	**79.11%**	1.67%	4.91%	8.16%
Sad	11.94%	0.00%	22.05%	3.82%	**34.30%**	3.68%	24.20%
Surprise	2.41%	0.00%	5.54%	1.45%	0.96%	**86.75%**	2.89%
Neural	9.39%	0.00%	7.08%	5.93%	7.58%	4.45%	**65.57%**

Figure 11: Results on test set

V. CONCLUSION & FUTURE WORK

In this project, we mainly focus on training and tuning AlexNet and GoogLeNet, and cope with the imbalanced dataset by oversampling,

and solve the overfitting problem by create new label. The results are well improved compared to baseline by adopting these methods.

Though the effect of the trained model is fine, we think a lot of work still needs to be done. As for future work, we think more method to balance the dataset needs to be tried, and the network structure should be future optimized. Meanwhile, it remains to be seen whether there are any more powerful methods to extract the features.

REFERENCES

[1] Darwin, Charles. *The Expression of the Emotions in Man and Animals. The Expression of the emotions in man and animals.* Cambridge University Press, 2009:692-696.

[2] Anstey, K. J., S. M. Hofer, and M. A. Luszcz. *Handbook of face recognition.* Springer New York, 2005.

[3] Zhang, Zhiwei, et al. "Regularized Transfer Boosting for Face Detection Across Spectrum." *IEEE Signal Processing Letters* 19.3(2012):131-134.

[4] Zhang, Z., et al. "Comparison Between Geometry-Based and Gabor-Wavelets-Based Facial Expression Recognition Using Multi-Layer Perceptron." *IEEE International Conference on Automatic Face and Gesture Recognition, 1998. Proceedings IEEE*, 1998:454-459.

[5] Jain, S, C. Hu, and J. K. Aggarwal. "Facial expression recognition with temporal modeling of shapes." *IEEE International Conference on Computer Vision Workshops* IEEE, 2011:1642-1649.

[6] Liu, Ping, et al. "Facial Expression Recognition via a Boosted Deep Belief Network." *IEEE Conference on Computer Vision and Pattern Recognition* IEEE Computer Society, 2014:1805-1812.

[7] Shan, Caifeng, S. Gong, and P. W. Mcowan. "Facial expression recognition based on Local Binary Patterns: A comprehensive study." *Image & Vision Computing* 27.6(2009):803-816.

[8] Byeon, Young Hyen, and K. C. Kwak. "Facial Expression Recognition Using 3D Convolutional Neural Network." *International Journal of Advanced Computer Science & Applications* 5.12(2014).

[9] Lcun, Yann, et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86.11(1998):2278-2324.

[10] An, Dan C, et al. "Flexible, high performance convolutional neural networks for image classification." *IJCAI 2011, Proceedings of the, International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July* DBLP, 2011:1237-1242.

[11] Kim, Bo Kyeong, et al. "Hierarchical committee of deep convolutional neural networks for robust facial expression recognition." *Journal on Multimodal User Interfaces* 10.2(2016):173-189.