A Model for Facial Detection and Expression Recognition

Qi Hu, Cunzhi Ren, Xinyu Zhao

Introduction

- **Background** : Facial expression recognition is a common tasks in our life. People achieve the task by telling the differences among different expressions based on heuristic knowledge. However, since differences can be subtle, it becomes difficult to teach a computer to understand and distinguish human expressions.
- **Objective** : There are two major steps to teach a computer understand us. Firstly, we need to implement a face detection method to let computer 'find' us. Then, by using an expression recognition classifier, we teach computer to recognize the differences among various expressions.

Data Preprocessing

- **Description of data** : We utilize the dataset (FER2013) from Kaggle Facial Expression Recognition Challenge which consists of 48x48 pixel grayscale images of faces. The following image shows the dataset distribution.

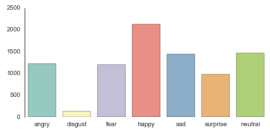

Figure.1 An overview of FER2013

- **Preprocessing** : We apply PCA to reduce the dimension of our feature space. The 1st principal component contributes to 29.05% variance for the whole dataset and the second one contributes to 9.76%. The visualization of first 2 components is shown in Figure.1. We find The first 50 pcs with take 84.47% of the total explained variance and the top 100 pcs contribute to 90.37%. Fig.2 shows the accumulated variance ratio increased by number of principal components.

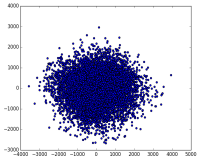

Figure.2 PC1 vs PC2 visualization

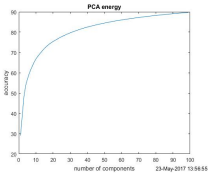

Figure.3 PCA energy analysis

Modeling

- **CNN**: We trained a Convolutional Neural Networks. Because CNN itself could reduce data dimensions, we just use the original training dataset to train. Before building a networks, we normalize the pixel value to 0-1 by dividing 256 and encoder 7 kinds of emotions to one hot vector. Our neural networks is similar to Alex Net. The architecture primarily contains 4 convolution layers, 2 max-pooling layers and 2 fully connected layers. All the convolution layers using Re-Lu activation function. We use soft-max activation layer as last layer to output label. We also use dropout operation during training.

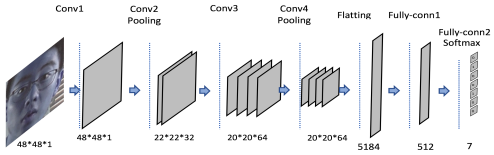

Figure.4 Structure of our CNN

- **SVM**: We use a 7-class SVM here. The input of our SVM classifier is the PCA components of original images. The specification of our SVM is: Radial Basis Function (RBF) kernel function, C=1, gamma=1/7. We also balance different classes of expression to eliminate the bias caused by different numbers of images.
- **KNN**: KNN is an extension of simple neighbor classifier. The input is the PCA components of original images. We have tried different k value to optimize the model (k=3 gives the best accuracy).

Modeling Analysis

CNN performs much better than KNN and SVM on this dataset. However, the accuracy is still too low to be used in practical task.

Table.1 Accuracy and Speed of Different Models

Model	Accuracy	Speed
KNN	41.92%	150fps
SVM	52.11%	300fps
CNN	63.01%	160fps

As we can see in Figure.6, the CNN model overfits after 50 epochs according to the increment of loss function on test set.

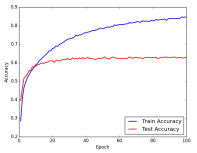

Figure.5 Accuracy versus epoch

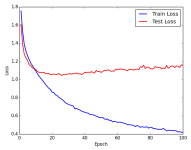

Figure.6 Loss function

Application

The application consists of two main sub-models: face detector and expression recognition classifier. It is real-time on laptop without GPU, and the frame rate depends on how many faces there are in the video.

The facial detector is a machine learning based model, with Haar features and cascade structure. For each block in a single stage (cascade) original input image, it does a very rough and quick test. If that passes, it does a slightly more detailed test, and so on. There are 5,826 features on each block. The algorithm have at most 50 stages, and it will only detect a face if all stages pass. The advantage is the majority of the blocks will return negative during the first few stages, which means the model won't test all features for all stages.

The expression recognition classifier is our previous CNN based classifier. The structure is shown in Figure 7.

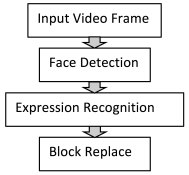

Figure.7 The Structure of the Whole Model

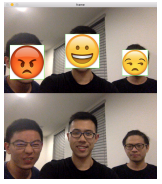

Figure.8 Example Frame

Conclusions

In this project, we achieve the implementation of face detection and facial expression recognition. The first step is to detect faces with Haar features in real time. And then the captured face is sent to the classifier which is trained with CNN previously.

Future Work : Improve the accuracy of the expression recognition classifier by separating the seven-class classifier into a group pf two-class classifiers and augment the dataset for training

Contact

Xinyu Zhao
Email: xinyu94@uw.edu
Phone: (206)245-8834

Cunzhi Ren
Email: renc@uw.edu
Phone: (206)234-4852

Qi Hu
Email: qihu@uw.edu
Phone: (206)422-6302

Reference

[1].Parkhi, Omkar M., A. Vedaldi, and A. Zisserman. "Deep Face Recognition." British Machine Vision Conference 2015:41.1-41.12.
[2].Sarfraz, M. Saquib, O. Hellwich, and Z. Riaz. Feature Extraction and Representation for Face Recognition. Face Recognition. InTech, 2010.
[3].Ebrahimpour, Hossein, and A. Kouzani. "FACE RECOGNITION USING BAGGING KNN."
[4].Wang, Xiao Hu, A. Liu, and S. Q. Zhang. "New facial expression recognition based on FSVM and KNN." Optik - International Journal for Light and Electron Optics 126.21(2015):3132-3134.