Design of Intelligent classroom facial recognition based on Deep Learning

IOP Conf. Series: Journal of Physics: Conf. Series **1168** (2019) 022043 doi:10.1088/1742-6596/1168/2/022043

Design of Intelligent classroom facial recognition based on Deep Learning

Jielong Tang[a,*], Xiaotian Zhou[b], Jiawei Zheng [c]

International School, Beijing University of Posts and Telecommunications, Beijing 100876, China

[a] Corresponding author: tangjielong@bupt.edu.cn

[b] zxt664165187@bupt.edu.cn

[c]zhengjw@bupt.edu.cn

Abstract. The emergence of Intelligent Classroom (ITS) is a significant improvement in the current level of modern teaching. This paper designs a real-time classroom assessment system that utilizes computer vision's target recognition technology. Based on the FER-2013 facial expression dataset and convolutional neural network (CNN), this paper establishes a real-time emotion recognition model by removing the fully connected layer and using the depth separable convolution combined with the remaining modules. In the end, the model established in this paper shows high detection accuracy and robustness, and can realize real-time evaluation of students' classroom performance to give teachers quick feedback.

1. Introduction

Nowadays, education plays an important role in modern society. Almost every family is trying to help their children win at the starting line. With the rapid development of Internet technology and artificial intelligence (AI), intelligent teaching [ITS] [intelligent classroom] [1] has been introduced into modern education in order to provide better teaching services.

Measures to improve the quality and outcomes of education begin with the improvement of students' performance in the classroom [2]. Specifically, positive changes in students' behavior can help teachers to achieve relatively good results, which is definitely conducive to the teaching process. Observing each student's performance in an intelligent environment, such as facial expressions, will help teachers dynamically adjust teaching methods by receiving rapid feedback from this real-time interaction, and will be beneficial to the quality of education.

In this way, the initial goal of the project is to implement a real-time computer vision system that automatically provides intelligent insight into the observed student. With the rapid development of deep learning in the field of computer vision, real-time target tracking and detection technology has made great progress. It can be used in this intelligent environment to help assess the performance of each student in the classroom.

2. Background

2.1. Convolutional Neural Network (CNN)

In the2010s, in the field of computer vision, the use of convolution neural network (CNN) [3] has been turned. With the wide use of CNN, object detection methods have been developed rapidly. CNN's basic

ideas, inspired by biological concepts, are called receptive fields, which are a feature of the animal's visual cortex and act as detectors sensitive to certain types of stimuli.

For example, edges. This biological function can be flexibly applied to computer vision using convolution operations [4]. And in image processing convolution can be used to filter images to produce different visible effects. A set of convolution filters may be combined to form a convolution layer of the neural network [4]. Figure 1 shows an example of a convolutional network. These continuous convolution layers form a convolution neural network (CNN).

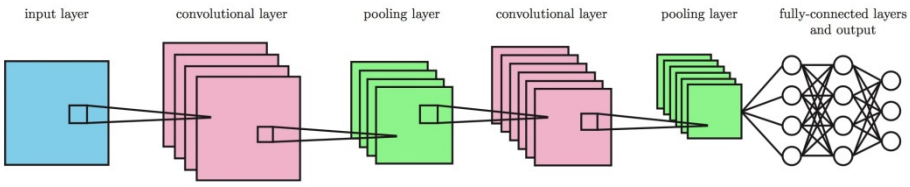

Figure 1 an example of convolutional network

2.2. Deep learning method based on region proposal

For the sliding window problem in the traditional method, the regional proposal provides a good solution. The region suggestion is to find out the position of the target in the whole image. Because regions recommend the use of specific information such as texture, edge, and color in an image, a relatively high recall can be ensured when fewer sliding windows (thousands or even hundreds) are selected. This method greatly reduces the time complexity of subsequent operations, and obtains a candidate window with higher quality than the sliding window with fixed aspect ratio. For candidate regions, the rest of the work is actually classification (feature extraction + classification). Because of the high efficiency of the CNN mentioned above, it is a good choice to use CNN to classify the images after obtaining candidate regions in the process of object detection.

In 2014, Ross B. Girshick first designed the R-CNN framework based on the regional proposal and the deep learning method, and made a breakthrough in the field of object detection. Figure 2 details the R-CNN process.

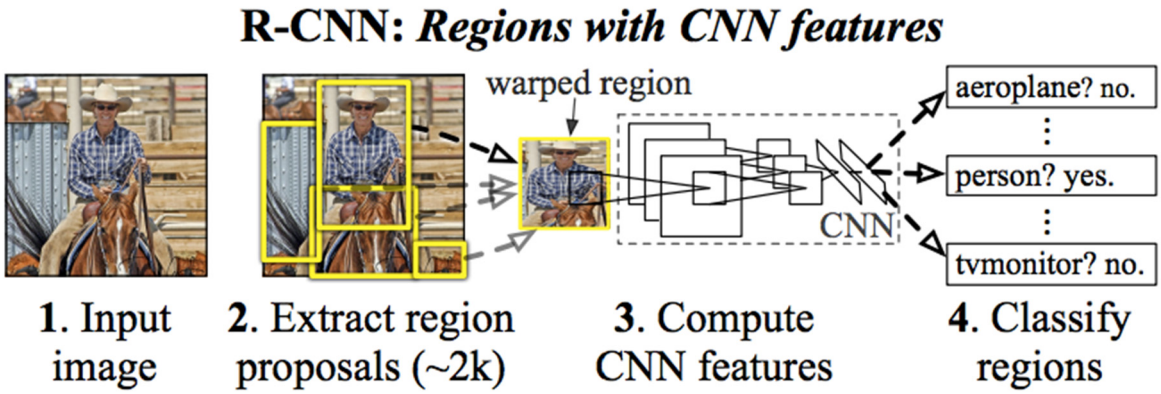

Figure 2 Process of R-CNN

3. Design and Implementation

3.1. Dataset

The project uses the FER-2013 facial expression dataset written by Pierre-Luc Carrier and Aaron Courville. Figure 3 shows the FER-2013 sample. The data set shown in figure 4 consists of 48x48 pixel grayscale images of the face [5], all of which are marked with numeric codes from 0 to 6. Includes seven

facial expression categories (0 = anger, 1 = aversion, 2 = fear, 3 = happiness, 4 = sadness, 5 = surprise, 6 = neutrality). The total number of samples in the dataset is 32298.

Figure 3 FER-2013 facial expressions samples

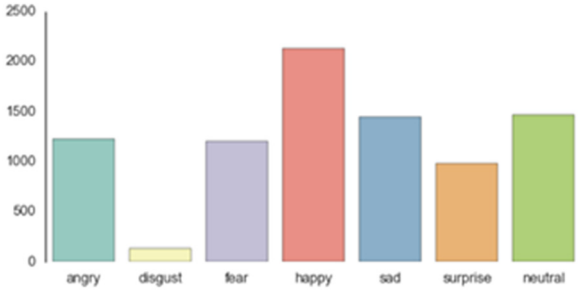

Figure 4 Overview of FER-2013 dataset

3.2. Dataset pre-processing

Preprocessing the dataset lays the foundation for training and getting the right model, which can lead to better performance and fewer errors.

In the experiment, it usually needs to read the data many times. Thus, in order to reduce the hassle of subsequent processing, the project serializes each image data into its associated classification ID, that is, a specific facial expression. Therefore, the first step is to convert the 48x48 pixel grayscale image of the face into a string for each image.

Because of the small sample imbalance between the "loathing" class and the other classes (only 113), the project first merges the "loathing" class into the "angry" class, as both represent negative emotions. In addition, to avoid data set disclosure, the project builds a data generator that can separate training and test sets using retention methods. There were 28709 examples (90%) in the training set and 3589 (10%) in the test set. Figure 5 shows the distribution of training and test datasets.

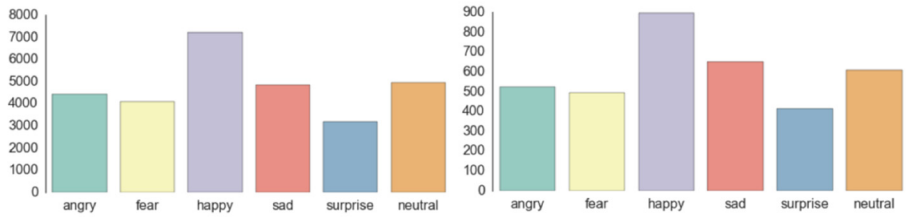

Figure 5 Training and test dataset distribution

3.3. Models

In the context of intelligent curriculum, the project has been committed to obtain suitable face detection and facial expression recognition model, the model can be deployed in real-time and hardware constraint architecture, with higher accuracy and better performance. Therefore, the project not only tries to use the deep learning method, which is a popular technology in computer vision in recent years, but also adopts the traditional machine learning method using multiple classifiers for target detection. This would facilitate the comparison.

(1)Machine learning method

Initially, the project designed the process using a machine learning approach. Figure 6 shows the entire process of this machine learning approach. Based on the training set, the project extracts facial markers, which help to detect the face in the region of interest. And the extracted features are fed into the classifier. The project uses multi-class SVM, logical regression and random forest classifiers for comparison. The test set then helps validate the training model and evaluate performance. Finally, the model will predict and output the classification results of several facial expressions.

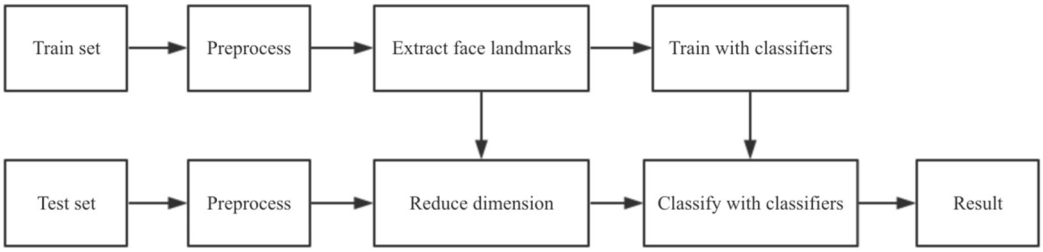

Figure 6 Machine learning method flow chart

(2)General CNN architecture

The project then uses the generic CNN layer as the building model architecture. General convolution neural network architecture consists of input layer, several convolution layers, some dense layers (such as fully connected layers) and output layers. All of these stacked layers in turn are linearly connected. Fig. 7 shows a general CNN architecture for face detection and facial expression recognition based on a deep learning approach.

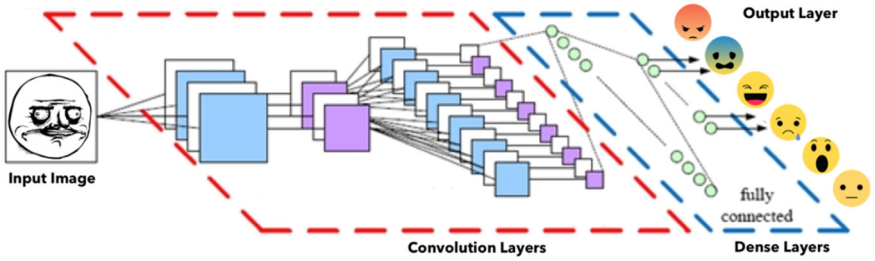

Figure 7 Facial expressions recognition general CNN architecture

With preprocessing, the input layer is predefined to a fixed size, which can then be entered into the next layer. For this step, to detect the faces in each image, the project used OpenCV, a popular computer vision library that includes pre-trained filters. In addition, the project uses Adaboost to locate and crop faces. This step helps to significantly reduce the size. The input layer can then be transferred to the Convolution2D layer, where the number of filters is specified as a super-parameter. A set of filters, such as the kernel, is randomly generated weights. Each filter, such as a sliding window, traverses the entire image to construct a feature graph with shared weights. The convolution layer generates a feature map showing how the pixel values are raised, such as edge, light, and pattern detection.

Applying pooling to help reduce the dimension behind the convolution layer is a relatively important process in building a generic CNN architecture, as adding more convolution layers can increase computational costs. The project uses the MaxPooling2D method, a popular pooling technique that uses a 2x2 window that traverses feature maps, retaining only the maximum value of pixels. When pixels are merged, the size of the image is reduced by 4.

For the output layer, the project uses softmax instead of sigmoid as the activation function. The layer outputs the probability of each facial expression class. In this way, the CNN model can provide the probability of each emotion, and select the emotion with the highest predictive score as the recognition result. Figure 8 shows the CNN architecture that was ultimately designed to detect faces and recognize specific facial expressions for each face.

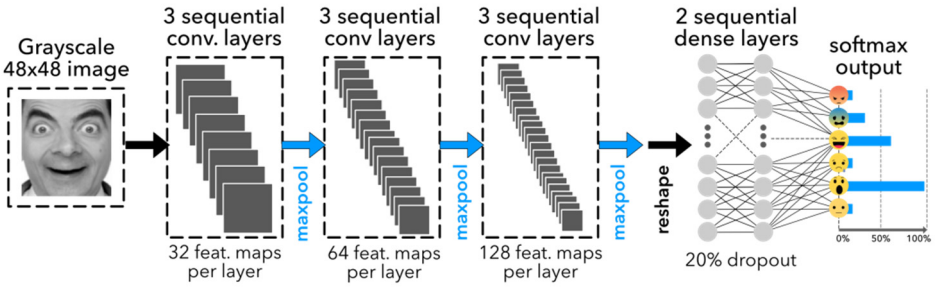

Figure 8 General CNN architecture designed for facial expressions recognition

3.4. Implementation

The project then adopted this modified CNN schema in the FER-2013 dataset. The training was done on NVIDIA Jetson TX1, the first supercomputer to run a module based on the GPU computing architecture, with 256 CUDA cores. Running 150k steps in the final model takes about four hours. And the weights can be stored in an 870 kilobyte file. By reducing the computational cost of the architecture, the project is now able to connect two models and use them sequentially in the same image without significantly reducing time. After obtaining the pre-training model, the complete pipeline, including the facial detection module and facial expression classification, takes 0.22 ±0.0003 seconds on the MacBook Pro laptop (i72.7 GHz, 16GB). This corresponds to a 1. 5 times acceleration compared to the original architecture of a normal CNN. A pre-training model is implemented to detect the face and recognize facial expressions using a real-time camera on a laptop computer.

4. Result and Discussion

For comparison, the project first uses a traditional rule-based approach. Based on the FER-2013 dataset, the results of facial expression classification using the SVM, random forest and logistic regression classifier, (mAP), are shown in Table 1.

Table 1 Classification results using traditional method

Classifier	mAP
SVM	59.3%
random forest	55.1%
logistics regression	54.0%

In addition, the general CNN model increases the mAP to 65.4%, and the real-time average recognition time is about 0.78s. The project then implements the modified CNN model, which combines the deletion of the full connection layer with the inclusion of the deep convolution layer with the residual modules.

After 150k step training, the model can recognize 0.22s real time facial expression, mAP is 70.1%, which is verified by the above test set. Figure 9 shows the classification results of the 24 facial expressions randomly selected from the test set.

Figure 9 Classification results of 24 facial expressions

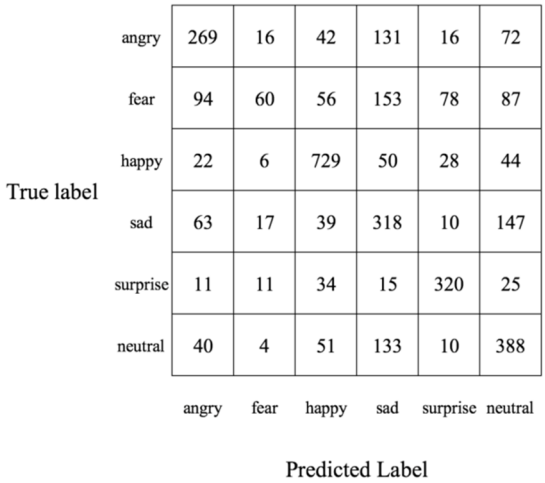

Figure 10 Normalized confusion matrix for original and predicted emotions

And fig. 10 shows a confusion matrix result using facial expression classification of the obtained model. It was observed that some misclassifications, such as "sadness" were predicted to be "fear" and "anger" were predicted to be "aversion".

The learning features of the original facial expression and the modified CNN model can be observed in figure 11. The white area in figure 11 (c) corresponds to the pixel value, which maps the selected neurons activated in the last convolution layer. We can observe that CNN learns to activate by taking into account features such as teeth, eyebrows, and eye enlargement, and that each feature remains constant within the same label. These results suggest that CNN is learning to understand human-like features that provide these extensible elements. The explainable results help us to understand some misclassifications, such as people wearing glasses or bushy whiskers were predicted to be "angry". This happens because the label "anger" is highly activated when it explains a person frowning and the frown features are confused with the darker background frame. In addition, we can observe that the features learned in our modified model (shown in figure 11 (c) are more explanatory than those learned from the general CNN model (figure 11 (b). Therefore, using more parameters in the original architecture results in less robust features.

Figure 11 (a) Orignial facial expressions smaples

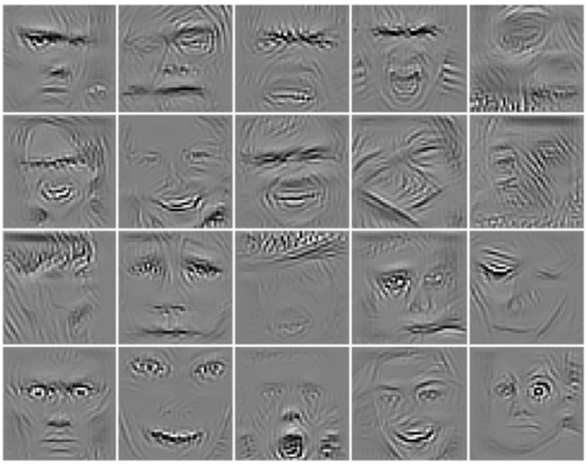

Figure 11 (b) Guided back-propagation visualization of general CNN model

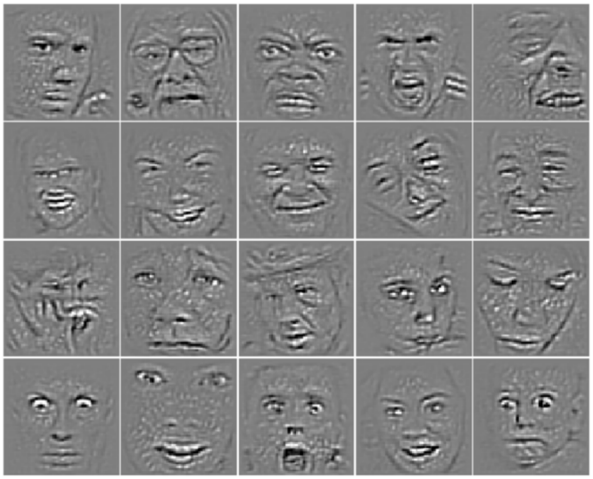

Figure 11 (c) Guided back-propagation visualization of modified CNN model

In this way, in order to reduce the misclassification of different facial expressions, and to implement models that are more suitable for interpreting students' real-time emotions in an intelligent classroom environment, The project combined "sadness", "fear" and "anger" into a "negative" state and turned

"happy" emotions into "positive" states, with those with "neutral" emotions being considered as "focused" states.

5. Conclusion

In the module of face detection and facial expression recognition, based on the FER-2013 emotion dataset, the project uses the traditional rule-based method and the depth learning-based CNN method for comparison. A suitable model with higher recognition accuracy and better performance is sought. Experiments show that the CNN model has a higher average precision (mAP), but due to the generation of millions of parameters, the delay of the hardware constraint equipment used in the project is very large. To solve this problem, the project modifies the general CNN model by removing the fully connected layer and combining the deep separable convolution with the remaining modules. Compared with the original model, the modified model reduces the parameters by 80 times. After 150k training, the project got the final model, the recognition time increased 1.5-fold, and the mAP increased from 65.4% to 70.1%. Even if there is a misclassification, the model can eventually provide teachers with feedback on students' states, including focus, positivity, negativity, or surprise. For future work, the project will attempt to eliminate misclassification by reducing interference with learning features, such as wearing glasses, which may classify the results as negative.

References

[1] Daniel Faggella (2017, September). Examples of Artificial Intelligence in Education. Retrieved September 1, 2017 from TechEmergence Web site: https://www.techemergence.com/

[2] Room 241 Teams (2012, December). Strategies to Improve Classroom Behaviour and Academic Outcomes. Retrieved December 26, 2012, from Concordia University-Portland Web site: https://education.cu-portland.edu/blog/

[3] Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition. 2014 (pp. 580–587)

[4] Marr, D., and Hildreth, E. Theory of edge detection. Proceedings of the Royal Society of London B: Biological Sciences 207, 1167 (1980), 187–217.

[5] Dataset in Challenges in Representation Learning: Facial Expression Recognition Challenge. Retrieved 2013, from Kaggle Web site: https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data