Facial Expression Recognition using Convolutional Neural Network

By: Yingjie (Chelsea) Wang and Yue Han

Abstract

Facial expression is one of the most powerful, natural and universal signals for human beings to convey their emotional states and intentions. Numerous studies have been conducted on automatic facial expression analysis because of its practical importance in sociable robotics, medical treatment, driver fatigue surveillance, and many other human computer interaction systems. The motivation of our study is to build models that can understand human emotional cues delivered through facial expressions.

Introduction

In the early twentieth century, six basic emotions were defined based on a cross-culture study. These prototypical facial expressions are anger, disgust, fear, happiness, sadness, and surprise. Contempt was subsequently added as one of the basic emotions. Due to these pioneering investigations and the direct and intuitive definition of facial expressions, the task of our project is to build a categorical model for seven emotions using a convolutional neural network.

For our project, we are seeking to build models that can recognize facial expression based on any facial pictures taken while applying some of the core concepts we learned in the class.

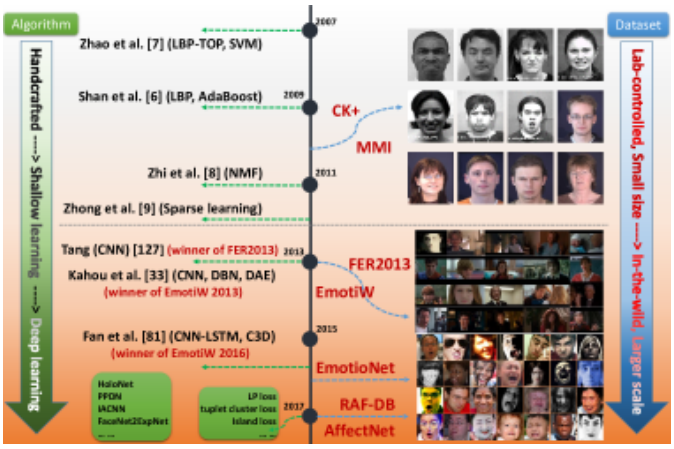


Figure1: The evolution of facial expression recognition in terms of datasets and methods.

Related Work

CNN has been extensively used in facial expression recognition. Recent years, several studies found that the CNN is robust to face location changes, scale variations and behaves better than the multilayer perceptron (MLP) in the case of previously unseen face pose variations.

From all published work using CK+ dataset, the performance of 6 classes is better than 7 classes in general. Also we learn that MTCNN and IntraFace might be two good ways to pre-process the image data.

Datasets	Methods	Network type	Network size		Pre-processing			Training data Selection in each sequence	Testing data selection in each sequence	Data group	Performance[1](%)
CK+	Zhao et al. 16 [17]	$\mathcal{EIN}$	22	6.8m	✓	-	-	from the 7th to the last[2]	the last frame	10 folds	6 classes: 99.3
	Yu et al. 17 [70]	$\mathcal{EIN}$	42	-	MTCNN	✓	-	from the 7th to the last[2]	the peak expression	10 folds	6 classes: 99.6
	kim et al. 17 [184]	$\mathcal{EIN}$	14	-	✓	✓	-	all frames	the same as the training data	10 folds	7 classes: 97.93
	Sun et al. 17 [185]	$\mathcal{NE}$	3 * GoogLeNetv2		✓	-	-	S: emotional T: neutral+emotional		10 folds	6 classes: 97.28
	Jung et al. 15 [16]	$\mathcal{FLT}$	2	177.6k	IntraFace	✓	-	fixed number of frames		10 folds	7 classes: 92.35
	Jung et al. 15 [16]	C3D	4	-	IntraFace	✓	-	fixed number of frames		10 folds	7 classes: 91.44
	Jung et al. 15 [16]	$\mathcal{NE}$	$\mathcal{FLT}$/C3D		IntraFace	✓	-	fixed number of frames		10 folds	7 classes: 97.25 (95.22)
	kuo et al. 18 [89]	$\mathcal{FA}$	6	2.7m	IntraFace	✓	IN	fixed length 9		10 folds	7 classes: 98.47
	Zhang et al. 17 [68]	$\mathcal{NE}$	7/5	2k/1.6m	SDM/ Cascaded CNN	✓	-	S: the last frame T: all frames		10 folds	7 classes: 98.50 (97.78)

Figure 2: A list of published work with CK+ dataset in three years including pre-processing, methods and performances.

Datasets

The Extended CohnKanade (CK+) database is the most extensively used laboratory-controlled database for evaluating FER systems. CK+ contains 593 video sequences from 123 subjects. The sequences vary in duration from 10 to 60 frames and show a shift from a neutral facial expression to the peak expression. Among these videos, 327 sequences from 118 subjects are labeled with seven basic expression labels (anger, contempt, disgust, fear, happiness, sadness, and surprise) based on the Facial Action Coding System (FACS). Images were sized at 640x490 pixels and mostly came in 8 bit gray-scale. Because CK+ does not provide specified training, validation and test sets, the algorithms evaluated on this database are not uniform. For static-based methods, the most common data selection method is to extract the last one to three frames with peak formation and the first frame (neutral face) of each sequence. Then, the subjects are divided into n groups for person-independent n-fold cross-validation experiments, where commonly selected values of n are 5, 8 and 10.

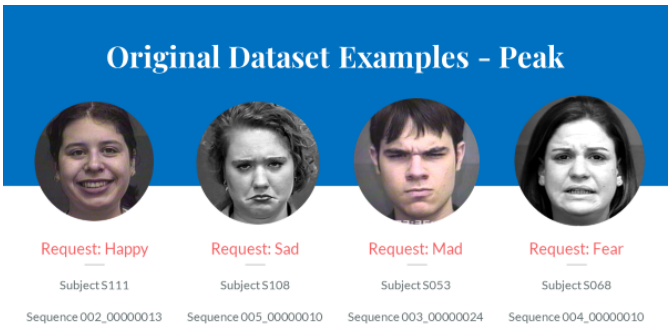


Figure 2: Four examples from original dataset with peak facial expression with 4 classes.

To understand how the sequence change, I selected fixed number of six pictures through neutral to peak and applied some pre-processing method to crop the main facial part for visualization in Figure 3. Seven classes in CK+ are Anger, Contempt, Disgust, Fear, Happiness, Sadness, and Surprise

Data Pre-Processing

Before preparing the data to be processed for modeling, we manually tagged the 200+ non-labeled sequences to verify a proper classification. Unfortunately, some of those manual decisions that needed to be made were quite difficult. Some of the emotions for subjects looked similar to how we perceived other emotions. Even though we might introduce some human error through this procedure, it was still a process that we still had to ultimately implement instead of dropping and losing so many images.

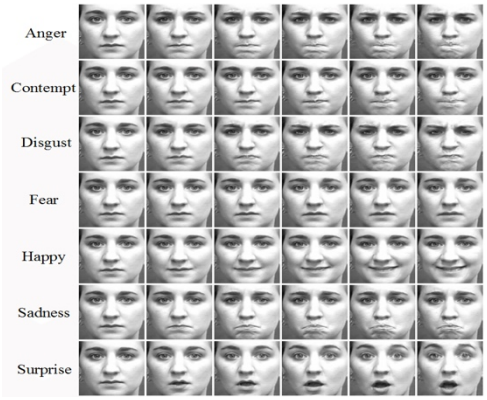


Figure 3: A sample series of facial expression with 7 classes. The photos here have been preprocessed to remove background and keep main component of face.

Upon finishing the manual classifications, we also thought of including only a certain subset of each sequence of images. Another observation we noticed while manually reviewing the images was that most sequences had a number of pictures that displayed no emotional facial features. All of the sequences began their sequence with a neutral facial display, and then slowly evolved into what was considered to be the “peak” emotional state. By including images that displayed no distinct facial features included with a classification could also run the risk of increased error. Ultimately, a decision was made to use a percentage of the sequence leading to the “peak” image. Due to the inability of being able regularize the dataset we were going to process, we also decided that during the modeling process we would include Dropout layers and Early Stopping for each epoch.

Methods

Once we finally arrived at the best image sequences, we were going to split into training, test, and validation sets, we found an additional step of processing that in the end was a huge key for boosting the performance of our model, called Facial Detection. As mentioned in related work, we applied Multi-Task Cascaded Convolutional Neural Network (MTCNN) to quickly detect a face from any image and then extract it. This method removes lots of background noise in the facial expression images. By implementing a MTCNN library for this purpose, we were able to pre-process our images to only focus on all facial features available which increased our maximum dimensionality to 11,664 features. Another advantage for extracting just the faces for our images is that it allowed our model to generalize more for any new image which effectively improved our overall validation accuracy.

Modeling

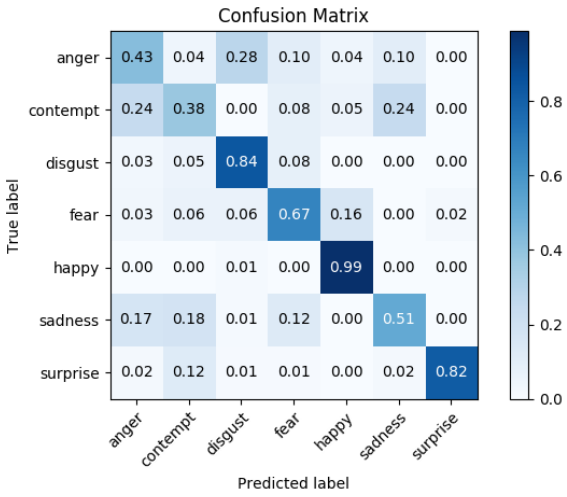
Since the CMU team only applied a SVM for their modeling efforts, we knew there could be interesting results from applying a deep learning network architecture that we learned from class. After our experience with the previous exams we instantly thought one of the best models to effectively classify images would be a Convolutional Neural Network (CNN) due to the ability to adjust kernels and strides over images to detect key features among images. Detecting facial features we figured would be critical for an overall accurate model.

Once we had an initial architecture working, we invested a lot of time refining and iterating over the original solution. We found that by using data from our MTCNN results seemed to work better with the identification of important facial features towards each emotion. We primarily judged the performance of our model by using Categorical Cross Entropy which included loss and accuracy in our train and validation data sets.

The critical piece that was able to pick up the minute facial features per image were the activation functions that were being executed per layer. With the possibility that our model could improve by implementing other activation functions, we developed a Hyper-parameter tuning script to help accomplish this. The tuning script developed was primarily executed by using a unique framework called 'Talos.' The script was capable of running hundreds of possible models with different configurations. A dictionary of parameters drives the possible parameter configurations and for us included different learning rates, number of epochs, dropout rates, number of neurons, optimizers, and of course different activation functions. Once, we completed hundreds or thousands of tuned model iterations, we then selected the best one based on the highest validation accuracy and loss. We developed a separate model evaluation script that used an additional held out dataset and used a Keras evaluation generator/predict generator to measure how the final model would perform.

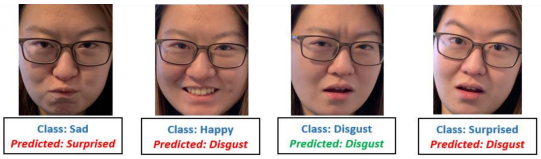
Results

Performance was measured between Loss and Accuracy (both test and validation) Our best model had 68% validation accuracy while using MTCNN face detection data. A model without MTCNN only had 48% validation accuracy without.



Classification Report

	precision	recall	f1-score	support
anger	0.53	0.43	0.47	68
contempt	0.29	0.38	0.33	37
disgust	0.54	0.84	0.66	37
fear	0.65	0.67	0.66	63
happy	0.85	0.99	0.91	84
sadness	0.67	0.51	0.58	72
surprise	0.99	0.82	0.90	100
micro avg	0.69	0.69	0.69	461
macro avg	0.65	0.66	0.64	461
weighted avg	0.71	0.69	0.69	461



Discussion of Results

The model had poor accuracy with the anger and contempt, which is not surprising. Since personally tagged over 200 images, we had a great deal of difficulty delineating the two emotions. The potential inaccuracy of our own tagging compounded the difficulty of distinguishing between two emotions.

We had very good results identifying happiness and surprise. These two emotions are very relatable to humans. These subjects probably had far less difficulty expressing these emotions. We certainly found it very easy to assign the happy emotion during manual tagging. The classification report above also details the precision, recall, and f1-score of our best model across each emotion. As stated, the scores for anger and contempt are significantly lower across all measuring categories. The test on myself was fun to do, but the results were a bit disappointing as expected. Part of the results may have been from my glasses. I was happy to at least get one classification correct though, which proves that our model has at least some merit for working as intended.

Conclusions

Overall, we were very impressed with the effort Carnegie Mellon undertook. This was an extensive project with a noble goal. The robust effort was boosted by a multidisciplinary team they brought together. Due to their data collection, other organizations both academic and private are looking at this research as a baseline to develop more advanced tools for emotional recognition. In their initial research paper, it was suggested that in order to have a strong model there dataset would need to be significantly bigger. After working with just this dataset from them to train a model, we would have to agree to that based on the initial evidence.

In terms of improvement to the neural network architecture, it would also have been interesting to test other types of networks such as some of the pre-trained models from PyTorch like ResNet or other Recurrent Neural Networks (RNN). Since we didn't draw much insight from the sequence changes over time, a dynamic neural network might make more sense to test with such as a Long Short Term Memory (LSTM) model. As this report mentioned in the beginning, this field of research is constantly growing, and the analysis and implementation of applying our Keras CNN model is just barely touching the possible power and real solution for facial recognition technology. As 8 such, the experience gained from conducting this project will become invaluable to us for touching the forefront of the Deep Learning field.

References

Amos B., B. Ludwiczuk, M. Satyanarayanan, "Openface: A general-purpose face recognition library with mobile applications," CMU-CS-16-118, CMU School of Computer Science, Tech. Rep., 2016.

Brownlee, Jason, “How to Perform Face Recognition With VGGFace2 in Keras”, Machine Learning Mastery. Website. June 5, 2019.

Centeno, Iván de Paz, “MTCNN”, GitHub Project Repository. Website. November 14, 2019.

Johnson, Justin, “Commonly Used Activation Functions,” Stanford University. Website. Nov 26, 2019.

Lucey, P., J.F. Cohn, T. Kanade, J. Saragih, Z. Ambadar and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression", in the Proceedings of IEEE workshop on CVPR for Human Communicative Behavior Analysis, San Francisco, USA, 2010. Website.