CS230

FaceNet: Facial Expression Recognition based on Deep Convolutional Neural Networks

Joe Wang
Department of Computer Science
Stanford University
wangjoe@stanford.edu

Martin Mbuthia
Department of Computer Science
Stanford University
maina@stanford.edu

Abstract

Automatic detection of facial expressions using convoluted neural networks holds great potential in areas of novel application such as human computer interaction. In this paper, we explore the potential of end-to-end deep Convolutional Neural Networks(CNN) using current state-of-the-art models to demonstrate the possibility of facial expression recognition from static images, without the need for the expensive extraction of action units that was previously required. We leverage transfer learning on three of the best deep neural network architectures and rely on the automatic feature extraction of CNNs. We then highlight key bottlenecks in this kind of automated facial expression recognition and highlight directions for advancing research in this field. We use a dataset of 14,000 images drawn from the standard Extended Cohn-Canade(CK+) and Japanese Female Facial Expressions(JAFFE).

1 Introduction

Given the crucial role that facial expressions play communication, along with the growing applications for human-computer interaction, especially in regards to humanoid robots, we believe it's vital that machines too understand the vitally important non-verbal cues that our faces play in our everyday interactions. Despite ongoing research in this area, some challenges remain. Among this is the extensive and expensive feature extraction of traditional methods of *Facial Expression Recognition*(FER), which is heavily reliant on human experience and is too complex for many real-world applications. To address this, we attempt to use current state-of-the-art image classification models in classifying images of faces into eight different expressions, namely: namely: anger, contempt,disgust, fear, happiness, neutral, sadness, surprise. Specifically, the input to our models is closely-cropped image of a face. We then use a CNN network to output a predicted emotion, which belongs to one of the eight common emotions.

The data is cropped down to the faces to avoid training on anything other than the faces. Specifically, since we do carry out any facial detection, or landmark detection, we avoid having much of a background in our images.

2 Related work

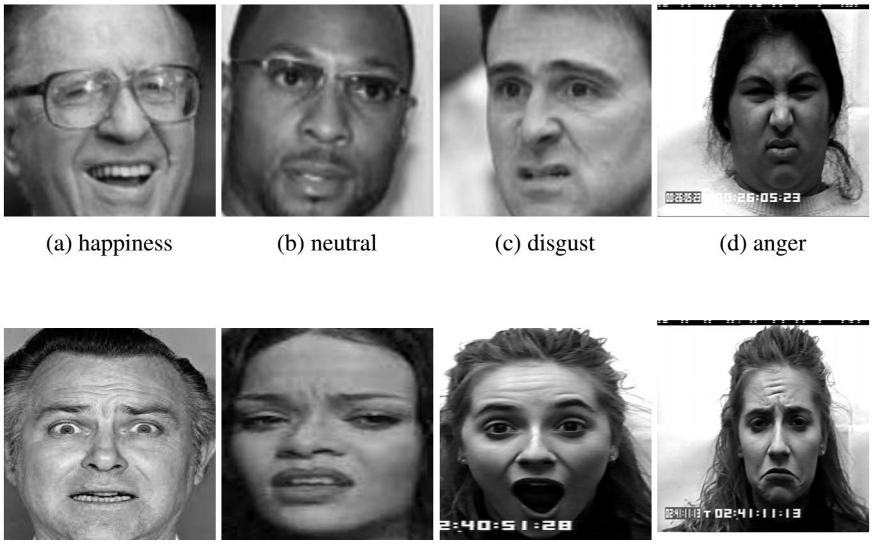
FER has been an area of active research for well over a decade. Initially, the process of building a neural network for FER was a laborious endeavor requiring extensive preprocessing of the data as well as manual handcrafting of facial landmarks. With recent advancements in deep neural networks, research in their use for FER has grown steadily. Ko[1] provides a review of research efforts in FER over the last decade.

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

Burkert et al.[2] proposed a 16-layer CNN architecture that which was based on GoogleNet. In this architecture, they implement parallel feature extraction using convolution, pooling and relu layers. Their model achieved an accuracy of 98.6% on the CK+ dataset, which we were unable to duplicate. More recently, Pramerdorfe et al.[2] compared the performances of six different CNN architectures on the FER2013 dataset, the best of which had a test accuracy of 75.2%.
Researchers have also used the Pyramid Histogram [4] which is a efficient way to extract features from regions of interest using special filters. This works well on preprocessed dataset, but doesn't have a high accuracy on other data. Some researchers also tried RNN model which takes the advantage of the sequence-to-sequence model to capture the changes in facial emotions [5]. A notable example of CNN model [3] is a 8-layer CNN with three convolutional layers, three pooling layers, and two fully connected layers. This model out-performed most of the state-of-art models in 2017 by achieving 86.38% accuracy on the same Kaggle dataset.

3 Dataset and Features

Our dataset consists of 4000 images of faces depicting either one of the eight expressions for which we want to classify. The dataset was obtained from a previous Kaggle competition on *Emotion Detection from Facial Expression*[1] and augmented with additional images from the Internet. The images contain faces of individuals between the ages of 18 and 51. Both male and female persons are present in the data. Ethnically, the dataset consists of 81% Euro-Americans, 13% Afro-Americans and 6% other. The images are of varying sizes, quality and color.


(a) happiness (b) neutral (c) disgust (d) anger

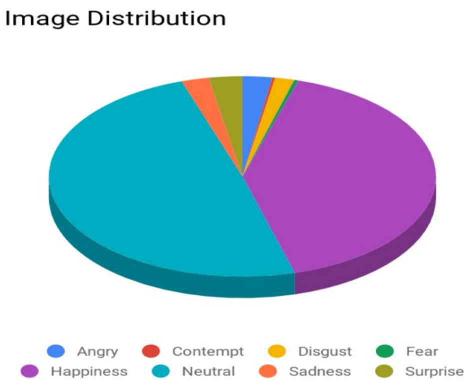
(a) fear (b) contempt (c) surprise (d) sadness

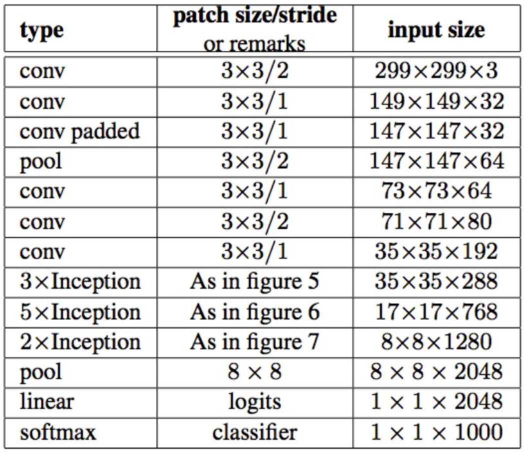
Figure 2: Sample data from our dataset with corresponding classes

4 Methods

We think the CNN architecture contributed greatly to how our model learned important features about faces in our data. We experimented with Inception V3, MobileNet, and NASNet. In addition, we also tackled data imbalance by weighting our loss function.

We used the Inception V3 model[7], since it uses both 1×1 and 3×3 and 5×5 filters, it's able to capture both the local and global features of the face. It also has fewer number of parameters (25M) compared to other models such as VGG or AlexNet. It has also been proven to be a very effective image classification architecture with, reporting a top-5 error rate of 3.46% on the ILSVRC 2012 classification challenge validation set[8].
On the other hand, we were drawn to MobileNet[9] by it's much smaller size in comparison to other state-of-art models. With only four million parameters, is computationally less expensive and this





type	patch size/stride or remarks	input size
conv	3×3/2	299×299×3
conv	3×3/1	149×149×32
conv padded	3×3/1	147×147×32
pool	3×3/2	147×147×64
conv	3×3/1	73×73×64
conv	3×3/2	71×71×80
conv	3×3/1	35×35×192
3×Inception	As in figure 5	35×35×288
5×Inception	As in figure 6	17×17×768
2×Inception	As in figure 7	8×8×1280
pool	8 × 8	8 × 8 × 2048
linear	logits	1 × 1 × 2048
softmax	classifier	1 × 1 × 1000

(a) Summary of the inceptionv3 architecture that we used.

requires lesser time to train compared to the other two models we used. It's performance however is only about 1% lower than that of inceptionv3 which has 25 million parameters. Lastly, we experiment using NASNet[10] which is the latest architecture developed using Google's AutoML. This model outperformed both MobileNet and all version of the inception architecture on the ILSVRC 2012 dataset[8]. With roughly the same number of parameters is the inceptionv3 model but a performance that is 1.2% better than all previously published results, we hoped that the model would provide better results in our use case.

In addition to architectures, we also experiment with a new loss function that helped alleviate the effect data imbalance. For every iteration, we fed in the weights α for each eight class so that the model penalized the losses fairly. The weights were computed based on the number of images in the training set. Finally, we used various techniques augment our data. These included: random flipping, random cropping, random scaling and random brightness adjustments on the images

$$Loss_{total} = 1/M \sum_{i=0}^{m} \alpha \times \text{cross\_entropy\_loss}(x_m) \tag{1}$$

5 Experiments and Results

After initial training of the inceptionv3 model, our performance indicated high variance. We attributed this primarily to the vast imbalance in our data, since 85% of our data belonged to only two of the eight classes. We also noticed that our images came from about 500 subjects which we thought would lead to over-fitting given the different ways in which different people express the same emotions. We undertook different methods to tackle the issue, key among them being acquisition of more data for the less represented classes which increased our data set by 1000. In addition, we weighted the classes when calculating the loss in order to reduce any over-fitting to the classes with most data. However, weighting the classes only improved our performance marginally.

Addition aggressive regularization yielded much better performance. Specifically, we augmented the data by randomly flipping some of the images, added random transformations (cropping, scaling and illumination) and tried different learning rates with each of the three architectures. For each model, we retrained on three different learning rates and picked the best two learning rates for each model. Given the long duration that each model took to train on a CPU, we were constrained on time and did not manage to experiment as much as we had hoped in order to find the optimal learning rate
A summary of the performance for each architecture is presented in table 1 below.

Additionally, we applied visualization techniques to help us gain more insight into the facial features that had the greatest impact on the prediction of the model. We calculated the saliency maps for this. The results are as demonstrated in figure 4 above. Our results showed that CNNs are indeed

Table 1: Training Results

Experiments			Train Accuracy	Test Accuracy	Recall Rate
Model Type		Learning Rates			
Inception V3	W/O Weighted Loss	2.2e-5	80.91%	73.02%	71.42%
		4.9e-5	77.64%	72.41%	70.09%
	W/ Weighted Loss	1.7e-5	82.94%	**74.04%**	71.74%
		2.5e-5	75.92%	68.82%	70.13%
MobileNet	W/O Weighted Loss	7.1e-4	79.13%	71.33%	61.23%
		9.7e-4	73.29%	64.81%	59.12%
	W/ Weighted Loss	4.1e-4	80.18%	**60.91%**	64.34%
		3.0e-4	73.29%	53.23%	47.76%
NASNet	W/ Weighted Loss	8.1e-5	73.30%	**51.59%**	60.99%
		5.5e-5	74.90%	50.57%	46.27%

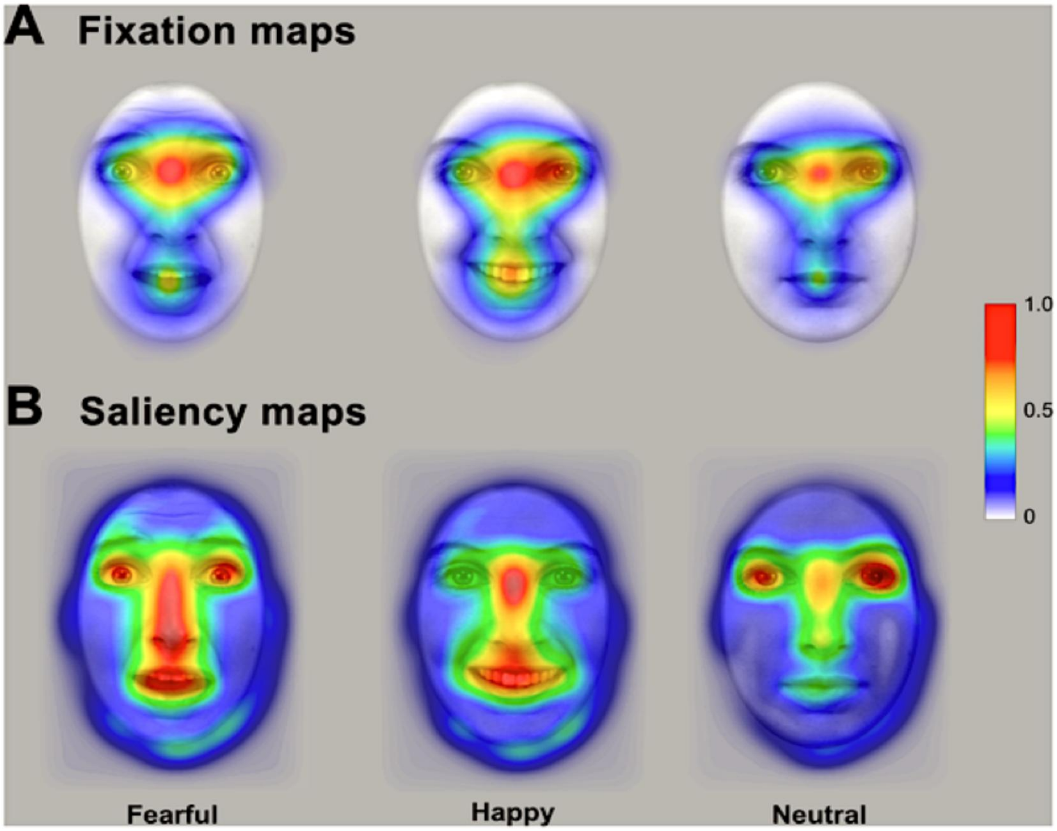


Figure 4: Fixation - where we tend to look for expressions as humans.

Figure 5: Saliency Map - regions that our models seemed to rely most on.

capable of automatic feature extraction because we noticed that the models learned to analyze regional features such as eyes, nose, and mouth which is similar to how humans perceive other people's facial expressions.

To understand how our trained models performs in reality, we tested the models using 10% of the original dataset. In addition to the test accuracy, we also measured their performance using the recall rate which tells us how well the model perform for each emotion class. In table 1 below, we average the recall rate across all eight classes. The fact that the recall rate was often lower than the test accuracy suggested that there was slight over-fitting to the majority classes. Our best performance however was at per with other current models we are aware of by Pramerdorfe et al[2].

6 Discussion

Overall, our results showed that is indeed possible to perform FER using deep neural networks without the need for auxiliary data such handcrafted facial landmarks or detection of action units. However, we noted that the task is indeed more challenging if the images used do not strongly depict the intended expression. We found that misclassification often occurred in images that had only slight expression such as the ones portrayed below.

(a) sadness (b) happiness (c) anger

We think that use of temporal data such as video frames would yield higher performance compared to use of isolated static images. By analyzing multiple images of the same subject over a short duration of time, a model might better learn to recognize expressions by learning how the the face changes over a short duration. This would also help deal with the challenge of expressions that are only slightly expressed.

7 Conclusion and Future Work

After this project, we learned that our model used similar approach to humans to understand facial emotions, as shown by the correlation between the saliency and fixation maps. In terms of results, our model highlights the challenge of using isolated and uncontrolled static images for FER. We expect that overcoming some most of challenges highlighted in this paper would lead to much better performance. In addition, we believe that future applications of FER would benefit greatly from additional expressions such as panic and nervousness. This would be helpful in scenarios where early detection of signs of panic could be helpful in preventing mass hysteria.

8 Contributions

Each team member contributed substantially to the project. Martin was responsible of experimenting with training different models, and Joe managed the datasets and most of the visualizations of the model.

References

[1] Ko, Byoung. “A Brief Review of Facial Emotion Recognition Based on Visual Information.” Sensors 18.2 (2018): 401. Web.

[2] Pramerdorfer, Christopher, and Martin Kampel. “Facial Expression Recognition Using Convolutional Neural Networks: State of the Art.” arXiv preprint arXiv:1612.02903 (2016): n. pag. Web. 18 May 2018.

[3] Shokrani,Shirin Moallem,Payman Mehdi (2014) Facial emotion recognition method based on Pyramid Histogram of Oriented Gradient over three direction of head, *IEEE Xplore*

[4] S. Kahou, V. Michalski, K. Konda, R. Memisevic, and C. Pal, “Recurrent neural networks for emotion recognition in video,” ICMI, pp. 467–474, (2015)

[5] Raj Dachapally, Prudhvi, "Facial Emotion Detection Using Convolutional Neural Networks and Representational Autoencoder Units", (2017) School of Informatics and Computing, Indiana University

[6] Chris Prinz, "Facial Emotion Detection Using Deep Learning", Medium (2017)

[7] Dan Duncan, Gautam Shine, Chris English, "Facial Emotion Recognition in Real Time" (2016)

[8] Jain, Neha, Kumar, Shishir, Kumar, Amit, Shamsolmoali, Pourya Zareapoor, Masoumeh (2018). Hybrid deep neural networks for face emotion recognition. Pattern Recognition Letters

[9] Shin, Hoo-Chang, Roth, Holger R, Gao, Mingchen, Lu, Le, Xu, Ziyue, Nogues, Isabella, Yao, Jianhua, Mollura, Daniel Summers, Ronald M (2016). Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 115(3):211–252, 2015.

[11] Howard, Andrew G, Zhu, Menglong, Chen, Bo, Kalenichenko, Dmitry, Wang, Weijun, Weyand, Tobias, Andreetto, Marco Adam, Hartwig (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications

[12] Zoph, Barret, Vasudevan, Vijay, Shlens, Jonathon Le, Quoc V (2017). Learning transferable architectures for scalable image recognition.

[13] Abadi, Mart, Agarwal, Ashish, Barham, Paul, Brevdo, Eugene, Chen, Zhifeng, Citro, Craig, Corrado, Greg S, Davis, Andy, Dean, Jeffrey, Devin, Matthieu and others (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467

[14] Scheller, Elisa, Buchel, Christian Gamer, Matthias (2012). Diagnostic features of emotional expressions are processed preferentially.