Undergraduate Research Journal

# Classifying Human Emotion Using Convolutional Neural Networks

Jonathan L. Moran
University of California, Merced

Author Note

At the time of writing, Jonathan L. Moran is a School of Engineering affiliate at the University of California, Merced. Jonathan L. Moran is a Research Assistant in the Department of Psychology at the UC Merced Consortium for Research on Atypical Development and Learning (CRADL) Lab. He conducts ongoing research in human attention and perception.

Contact: Jeffrey Gilger, CRADL Lab Director and Interim Dean

Professor of Social Sciences and Humanities. jgilger@ucmerced.edu

## Abstract

Despite the computer's historical success as a communication tool, machines themselves have yet to fully master the most basic forms of nonverbal communication that we humans use daily. Gender, ethnicity, age and emotional state is often perceived immediately by most humans engaging in conversation. In face-to-face interactions, humans can form broad generalizations about an individual's social status, health, and well-being within a blink of an eye. Training a classifier algorithm to accomplish this form of human behavior is a rather difficult task. While the accuracy of exchange of non-verbal messages may be questioned, the vast amount of information humans can generalize from these thinly-sliced events is a true feat of human intelligence. In this paper, we will be exploring the concepts of object recognition and deep learning neural networks to ultimately train a classification model to recognize universal human emotion from the FER-2013 facial expression dataset (Kaggle, 2013).

# Classifying Human Emotion Using Convolutional Neural Networks

## Introduction

Emotions are the response of the human body to physiological arousal resulting from significant external or internal cues (Schacter, 2011) in one's environment. Emotions are displayed outwardly through facial expressions, involving the movement of the eyebrow and mouth muscles. These 'reactive' emotions can either be negative or positive, and are marked by the differences in facial expression from a neutral, resting state. The universal theory of emotion, pioneered by Charles Darwin in 1867 and later backed by many psychologists, claims that there are six distinct, universal emotions represented in human facial expressions. Despite being highly theorized, emotions are relatively easy to perceive from person to person. Even though humans are vastly different in physical appearance, the expression of these emotions remains constant across all cultures, races, genders, and ages.

## Methods

### Assessments and Measures

Convolutional neural networks are a popular choice for deep learning applied to the task of image classification. The ability of CNN architectures to quickly discern between input frames is an important factor governing the replicability of machine-driven sentiment analysis. We will be exploring the network architecture initially proposed by researcher A. Gudi in 2011 to conduct facial

recognition and semantic analysis using a convolutional neural network trained from a dataset, consisting of 32k low-resolution, grayscale images. The pictures in the FERC-2013 dataset depicts the six human facial expressions of emotion across gender and age. This dataset poses a few challenges for the convolutional neural network, since a great deal of variance exists across the FERC-2013 images. Many age groups were depicted under varying conditions— where head positioning and pose made training the CNN to recognize emotion in normal conditions difficult.

**Facial Descriptors of Emotion.** In this study, we will be classifying the six universal human emotions proposed by Darwin (Figure 1) which include: anger, sadness, fear, surprise, disgust, and happiness. In addition to these, we will also be considering a resting neutral state in our classification task. A series of novel algorithms will be employed to detect facial expressions through facial markers. The non-verbal display of emotion can be assumed in the human facial structure and ultimately predict one's emotional state using facial markers as a form of non-verbal displays of affect. Facial markers are a significant indication of the seven universal affective displays of human emotion. The affective state can be visually interpreted as changes in the facial muscles present in the eyebrows, mouth, nose bridge, cheeks and jaw (Figure 2).

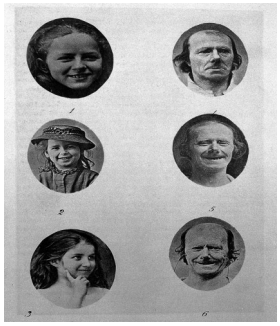

*Figure 1*: Depiction of the six universal emotions, proposed by C. Darwin and later by Paul Ekman in 1995.

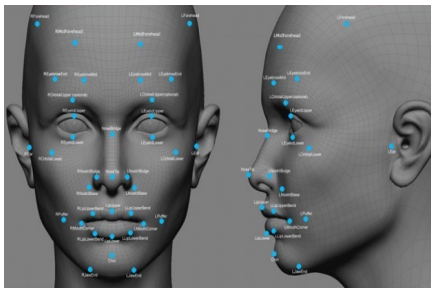

*Figure 2*: Facial markers in the eyebrow, nose, mouth and cheekbone areas. (credit: Facebook DeepFace)

**Detecting human faces using object recognition**. Basic facial recognition remains one of the more computationally-challenging tasks of an artificially-intelligent system. Most facial recognition systems scan individual images or video frames for human attributes. These human attributes are known as features present within a positive frame– one that contains a human face. Several algorithmic techniques exist to accomplish the task of facial. A series of template matching algorithms analyze the input image—a potential human face for consistencies in shape, position, size and relative proximity to other detected features. In the template matching approach (Kour, 2015), a result metric is used to perform object extraction and ultimately recognize positive images containing human faces. Alternatively, using feature extraction techniques, human attributes can be recognized through edge-detection and

contrast differences present in an image (Brunelli, 2009)[1].

**Facial detection through deep learning.** A combinational network that performs both low-level feature detection and high-level pattern-matching tasks will be used in our emotion recognition task. This network structure was initially explored by A. Gudi in 2015 to effectively predict semantic features like gender, age, and ethnicity. Our model benefits from the performance advantages of the Viola Jones[2] cascade, which first pre-processes the input images using a collection of contrast normalization and dropout algorithms (Viola, Jones 2001). The remaining network architecture, including the statistical dropout and objective functions, will be discussed in more detail in the later sections of this paper.

**Facial recognition using other techniques.** While this paper explores primarily a rudimentary technique for accomplishing facial recognition, it is important to note that several other methods have been utilized with great accuracy. Skin texture analysis, thermal mapping and baseline expression deviations are several popular techniques to improve the classification accuracy of combinatorial deep learning neural networks.

---

1 R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*, Wiley, ISBN 978-0-470-51706-2, 2009 *([1] TM book)*

2 Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern* Recognition*, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE, 2001.

To accomplish facial recognition in the model explored in this paper, these improvisations will not be considered.

## The Experimental Set-Up

**The task.** Our goal of this paper is to successfully classify the human emotions present in the FER2013 facial expression dataset. A selection of images shown in Figure 3 has been reprinted from the Kaggle-licensed dataset.


*Figure 3*: Sample images from the FER2013 dataset used to train and validate model

**The dataset.** A data scientist's main concerns when conducting a classification task are to reduce overfitting and maximize prediction accuracy. To optimally train our neural net with these considerations in mind, we selected the FER2013 dataset– one of the largest publicly-available facial expression datasets. Consisting of nearly 33k faces, this Kaggle-licensed resource includes many different depictions of human emotion across gender, age, and race.

28k of the 33k images will be used to familiarize (train) our model to recognize the seven universal emotions. The remaining images will be used to test and validate the model's final accuracy. A graphical distribution of the FER2013 dataset is shown in Figure 4.

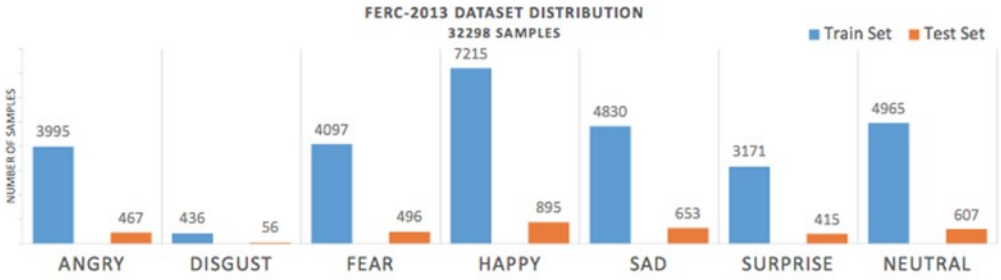


*Figure 4*: FER2013 dataset distribution

**Interpreting image data for machine use.** Each of the FER-2013 dataset images must be converted into an interpretable format for use within the neural network pipeline. To minimize the computational over-head in our model, we will be representing the input images using Numpy-formatted arrays. The images are first down-sampled and then converted to ndarray format to prepare the dataset for normalization in the pre-processing stage. While not applicable in our current model, an alternative 3D attribute data format is often chosen for representing facial features and other artifacts from the scene. The Numpy, ndarray-format representation is usually less computationally-expensive than the latter technique, but it is important to note that this 3D attribute data may be a

valuable consideration in the future, when improvements in the detection and training accuracy is proposed. Lighting, angle of capture, age of subject, and head rotation are several complicating artifacts often considered when choosing to normalize facial expression datasets. However, this consideration will not be made in our neural network due to the necessary overhead.

**Preparing the dataset for the experimental task.** Before our images are used to train and validate our model, a pre-processing pipeline (Figure 5) minimizes the inconsistencies in lighting, crop and angle across the FER2013 dataset. There are two distinct properties within the dataset images that must be normalized. Variance in facial feature location, as well as various lighting and contrast conditions will be controlled in the Face Location and Global Contrast Normalization layers.

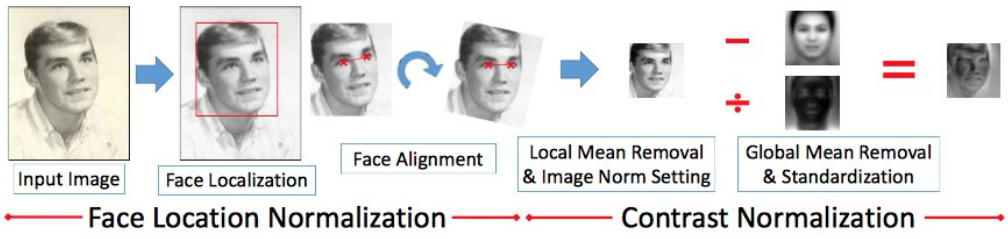


*Figure 5*: The pre-processing pipeline

The resulting image pixel data is numerically-represented in Numpy int32 format[3], and has been close-cropped to a size of 48x48 pixels. The training and test images are converted to grayscale to limit racial biases and poor generalization. [4] The pipeline, detailed in Viola and Jones' 2001 report, first normalizes all pixel values based on local mean samples, then iteratively normalizes each image cluster using an normalization hyper-parameter of 100. Then, once each image has been processed, the dataset's global mean is calculated and used to normalize the training images by subtracting the hyper-parameter and dividing by the standard deviation. The result of this optimization pipeline (Figure 6) is an increase in the competition that neighboring neurons face, reducing the overfitting problem faced in machine learning.

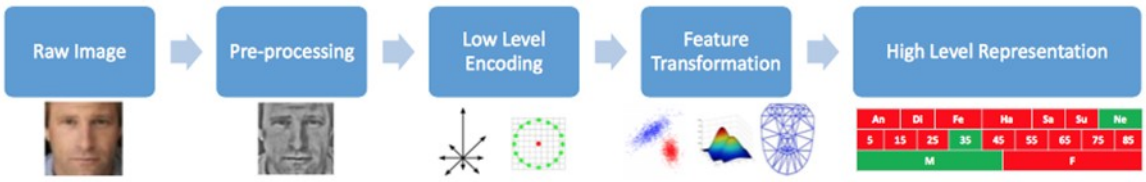


*Figure 6: Conventional facial feature extraction pipeline (Haoqiang, 2014).*[5]

3 A multi-dimensional array-based data structure supported by the widely-popular Python statistical library NumPy. Further documentation: https://docs.scipy.org/doc/numpy/reference/

4 See "Overfitting in Machine Learning: What It Is and How to Prevent It". EliteDataScience. Sept 2017. https://elitedatascience.com/overfitting-in-machine-learning

5 Reprinted from:
Haoqiang Fan, Zhimin Cao,Yuning Jiang, QiYin, and Chin-chilla Doudou. Learning deep face representation. arXiv preprint arXiv:1403.2802, 2014. 2

**Facial recognition component.** To quickly recognize a human face within an input image, the Viola-Jones algorithm exploits several assumed properties of the human face. A series of Haar-feature detectors scan for these common characteristics:

> *Contrast differences* – the eye region within a face will be notably darker than the upper-cheek region. Similarly, the nose region will be brighter than the eyes.[6]
>
> *Proximity similarities* – the relative location and size of the eyes, nose, and mouth are consistent across normalized[7] human faces.

The Viola-Jones Haar-feature detection algorithm operates sequentially over the input image, performing several statistical functions with each iteration. Rectangular black-and-white Haar-feature "windows" are placed over each sub-matrix of the input (Figure 8). Then, the difference in pixel values of the Haar-feature and input layer are computed. If the contrast differences are *statistically significant*, the predicted facial region (either eyes or nose) is assumed to be absent from the sub-matrix. The window can then be discarded, and the region will be marked as irrelevant in a summed-area table of pixel values. The use of integral images effectively reduces the input dimensionality and allows for faster facial detection rates.

---

6 These assumed features were studied by Paul Viola and Michael Jones. Further evidence for the presence of Haar-features is discussed in their research paper.

7 Described in the Facial Recognition Component section of this paper, a normalized face has been pre-processed to minimize differences across the facial expression dataset.

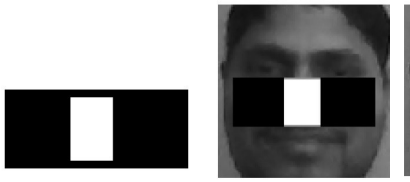

*Figure 8*: A vertical Haar feature is used to detect the relative position of the nose.

*Figure 9*: A horizontal Haar feature is used to detect the relative position of the eyes.

The Viola-Jones algorithm employs a cascade architecture that operates on the order of feature complexity. If a given window passes the initial detection task, it will pass on to the more computationally-extensive detection tasks. In addition to the 1:1 rectangular Haar feature windows in Figures 8 and 9, several other rectangle orientations are used successively to train the classifiers. This cascade structure allows the model to discard negative frames before performing intensive learning tasks on the positive frames.

**Emotion recognition component**. Once a viable face has been detected, our artificially-intelligent system then runs a series of detector and classifier nets that work by "breaking apart" input images into smaller chunks of pixel data referred to as sub-matrices. These windows of pixel data contained in the FER-2013 closely-cropped image are then individually scanned for human facial features. To make computation more efficient, several convolutional and max-pooling layers are implemented in the network design (Figure 10).

A mixed network architecture consisting of convolutional and max-pooling sub-networks, is replicated in this study. In a study by A. Gudi, this network architecture was able to classify the seven universal human emotions with a 67% confidence when tested on the FER-2013 dataset.

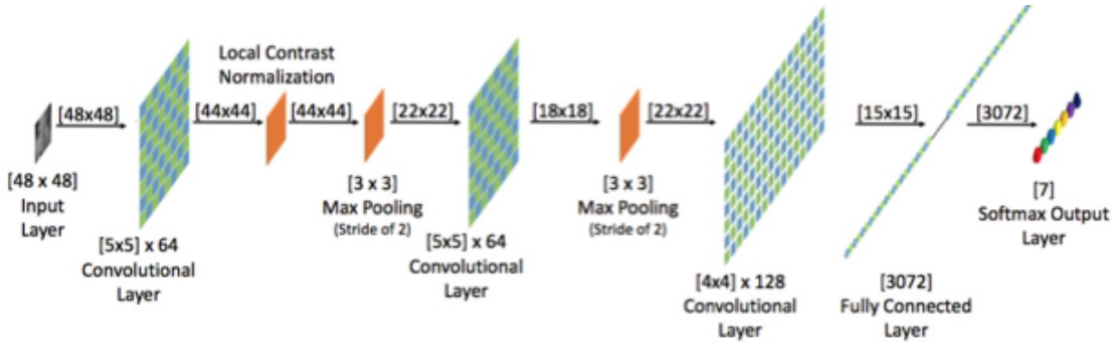


*Figure 10*: Overview of the complete model architecture.

**Network performance and considerations.** This segment studies the network structure, clarifies the objective and activation functions, and discusses the optimizations that were selected for best performance.

**Optimizing the classification sub-network.** The Viola-Jones face detection algorithm is able to perform rapid object detection using Haar cascades– a technique in image analysis applied to facial recognition. In order to quickly determine positive frames, i.e. pixel windows that likely contain facial features, a Haar cascade detection pipeline discards the unlikely windows, further reducing the amount of frames needing to be processed.

Detailed in Figures 8 and 9, the Haar-like features have two configurations. Each of the kernels are then masked across the input image frames. The sum of contrast differences between the kernel and the input image is calculated iteratively, window-by-window.

The model discussed in this paper utilizes a pipelined Haar cascade architecture known as the AdaBoost . This OpenCV weak-classifier network further reduces the number of feature comparisons necessary across the input data. By applying a pipeline of cascaded weak-classifiers, the input matrix kernels are strategically scanned for detectable Haar-like features corresponding to positive frames. The loss function, determining our ground truth estimate, is computed as the weighted sum of contrast differences. The AdaBoost design, its objective, and loss functions are discussed briefly in the later segment of this paper.

When the first Haar kernel is processed by the pipeline, and a positive image is not detected (e.g. facial features were likely absent from the image window), the remaining Haar configurations are not considered. While the use of a Haar cascade helps reduce the dimensionality of the input dataset with the discarding of negative windows, thousands of computations across the 48x48 pixel space is still likely.

Another important optimization strategy selected for our network review was the use of a normalization pipeline across the image dataset. In-plane rotation and scale resizing were applied to each of the training images to minimize potential complications in the facial recognition pipeline and model's training cycles. Several factors present in human faces are considered in this normalization process– namely the distance between the eyes and the difference in facial contour  across the FER-2013 dataset images. To prevent overfitting of the training data, the pre-processing optimizations are applied to every photo in the dataset before being used in the model's training phase.

The use of statistical dropout layers in the classification substructure sort through possible input frames. Consistencies in shape, position, size, and relative proximity to other features, aid in determining if a viable emotion is expressed in the image window. If a detected facial feature does not surpass the objective threshold, the frame is discarded through a process known as *dropout*. Similar to the technique of introducing noise in the dataset (i.e. random zeroing-out of neurons), the dropout layer is utilized in the training process to achieve optimal detection accuracy. We will be using a default dropout probability value of 0.3, a benchmark hyper-parameter for detection accuracy provided by A. Gudi. Another consideration for network performance is the use of convolutions layers in an isolated sub-network.

Since CNN layers can be more prone to overfitting— the dropout probability layers are used across the entire network, whereas the convolutional layers are fully-connected. The entire network architecture is detailed in Figure 10.

**Training methods using stochastic gradient descent.** Also performed in the Viola-Jones algorithm is the mini-batching of training data. Each batch contained 100 data samples, where a negative log-likelihood is set as the objective function. The training set from the FER-2013 dataset consisted of 14,524 optimal, pre-processed faces (Figure 3). Each of the training images was given a label corresponding to the actual emotion being expressed. The network was trained for 100 epochs over the course of 40 hours using TensorFlow, the Python-based machine learning framework provided by Google. A subsampling factor of 1/2 was chosen as a hyper-parameter in the max-pooling hidden layers. This reduces the network complexity by half— reducing the feature-mapped 44x44 activation layer. Another convolutional layer consisting of 128 feature maps, with a kernel size of 4x4, processes the sub-matrices of pixel data across the input image frame. A dropout layer discards irrelevant frames, then the positive frames are processed further to update the 128-feature layer. The output layer is a fully-mapped seven-neuron layer (with each corresponding to one of the universal emotions). A softmax activation function performs our maximum likelihood estimate of determining *which of the seven emotions was being expressed.*

## Results

The goal of the convolutional neural network was to provide an accurate classification of human emotion from the input dataset. After training the network for 100 epochs over a 14,524 image dataset, a validation was run using 9000 of the remaining FER2013 dataset images.

The network achieves an optimal prediction accuracy[8], but struggles to accurately distinguish between the fear and sadness emotions. It may be noted that slight differences between the two facial expressions may complicate the successful distinguishing of emotion. A convolutional neural network using a Viola-Jones facial recognition algorithm in combination with Haar-feature detectors is a suitable network structure for such tasks.

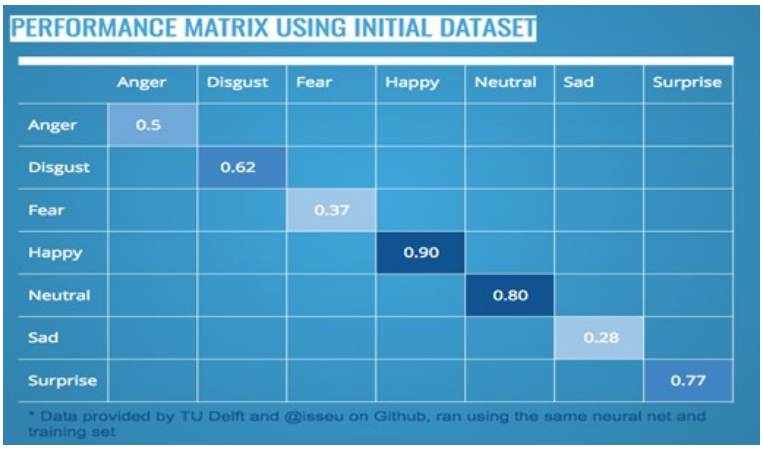


*Figure 11: Confusion matrix displaying the activation likelihood of the predicted versus actual emotion*

8 Determined to be ~65% accurate when utilizing the AlexNet CNN on the FER2013 validation set (I.J Goodfellow, 2015 pp. 59-63)

**Discussion**

With the rise and popularity of artificially-intelligent systems, many consumer technology applications for facial recognition have recently emerged. Present in security, identity verification, and behavioral health systems, emotional affective computing has become a necessary implication in the world of AI. With many practical uses, emotionally intelligent systems can better serve and adapt to users' individual needs. The ability for devices to deal with users' emotional states allows for a more intuitive user experience.

**Future work**

We hope that by openly-disseminating current deep learning and neural networks research, fellow researchers, engineers, and product designers pursuing the fields of social psychology will utilize these tools to better conduct further studies in human behavioral modeling to better integrate human-computer interactions.

## References

Darwin, C. R. *The expression of the emotions in man and animals.* John Murray, London, 1872

Gudi, A. Recognizing semantic features in faces using deep learning. *arXiv preprint arXiv:1512.00743, 2015*.

Haoqiang Fan, Zhimin Cao,Yuning Jiang, QiYin, and Chin-chilla Doudou. Learning deep face representation. *arXiv preprint arXiv:1403.2802*, 2014. 2

Kaggle. Challenges in representation learning: Facial expression recognition challenge, 2013.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM,60*(6), 84-90. doi:10.1145/3065386

Moran, J. L. An open-source convolutional neural network for emotion recognition (in review, 2018). Github-hosted project, https://github.com/jonathanloganmoran/emotion-recognition-neural-networks

OpenSourceComputerVision. *Face detection using haar cascades*. TFLearn: Deep learning library featuring a higher-level API for TensorFlow. Accessed 2015, URL: http://tflearn.org

Schacter, D. L., Gilbert, D. T., Wegner, D. M., & Hood, B. M. (2011). Psychology (European ed.). Basingstoke: Palgrave Macmillan

R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*, Wiley, ISBN 978-0-470-51706-2, 2009 *([1] TM book)*

Viola, P., Jones, M. Rapid object detection using a boosted cascade of simple features. In *Computer Vision* and *Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE, 2001.