ECE 457B Computational Intelligence

Winter 2016

Instructor: Dr. Otman Basir

Emotion Classification of Headshots using Neural Networks

by Adrian Cheung, Aaron Morais, Darren Mo, Rahim Mitha

Abstract

The problem being solved is the classification of faces into emotions. Specifically, we will classify faces into 5 basic emotions (Sad, Angry, Happy, Fear, Disgust). The ability to classify emotions would be useful for human-computer interactions, being able to adapt the system based on the user's emotions.

Since this is a classification problem, the proposed solution is to use a neural network. The traditional approach to image-based classification is to use a convolutional neural network (CNN) based on LeCunn and Bengio's proposal. A CNN is a technique for processing images into a neural network by creating a sliding window across the image and processing and evaluating the image sections at a time. CNNs have been used to classify facial expressions before. We will apply a CNN (implemented using Tensor Flow) to still images of headshots that show distinct facial expressions.

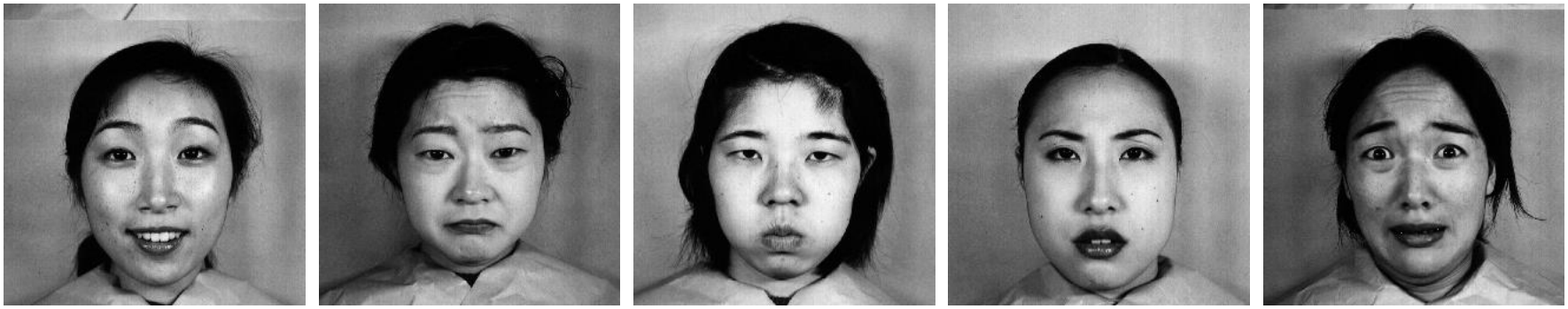
The training dataset will come from the Japanese Female Facial Expression (JAFFE) Database. The dataset consists of 10 people, 5 emotions each, and approximately 3 variations per emotion, resulting in 153 images to train and test on. We may use more images if necessary. We will test our implementation using k-fold cross-validation, aiming for at least 70% accuracy.

Problem

We want to classify faces according to their facial expressions.

Scope

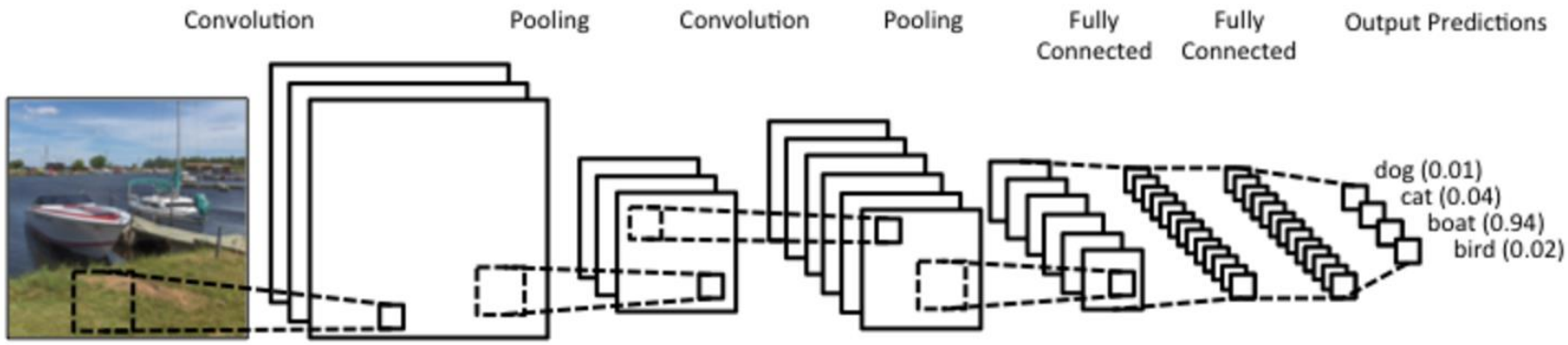
- 5 emotions: Happy, Sad, Angry, Disgust, Fear
- Female black-and-white Japanese faces



Proposed Solution

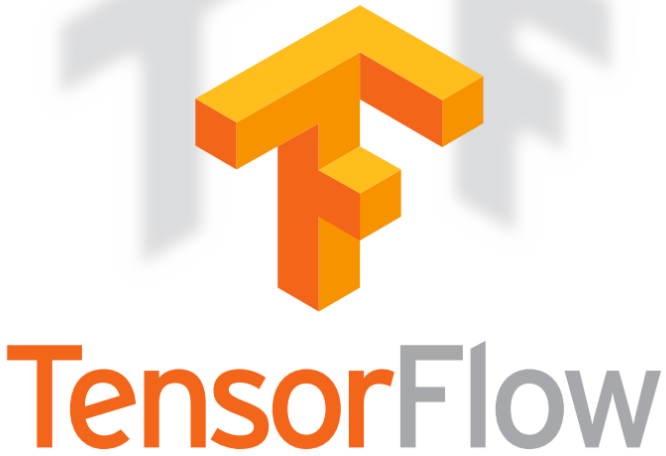
Convolutional Neural Network:

- Proven technique for classifying facial expressions
- Based on LeCunn and Bengio's proposal



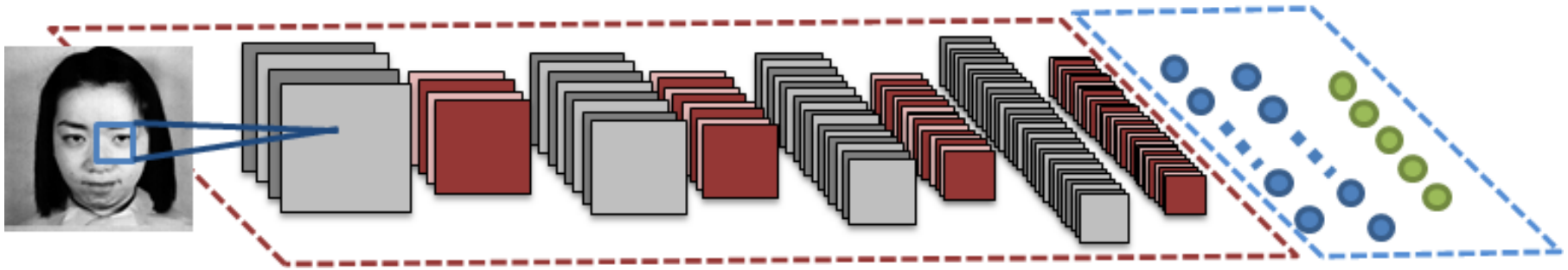
Implementation

- Implemented in Python
- TensorFlow machine learning library
- Trained on the Japanese Female Facial Expression database
- Tested using 4-fold cross-validation



Convolutional Neural Network

- 4 convolution/pooling layers with 5x5 pixel sliding window
- 2 Fully connected layers of 1024 nodes each
- Output layer of 5 nodes
- Learning algorithm: Back propagation with Adam Optimizer
  - An extension on adaptive gradient learning



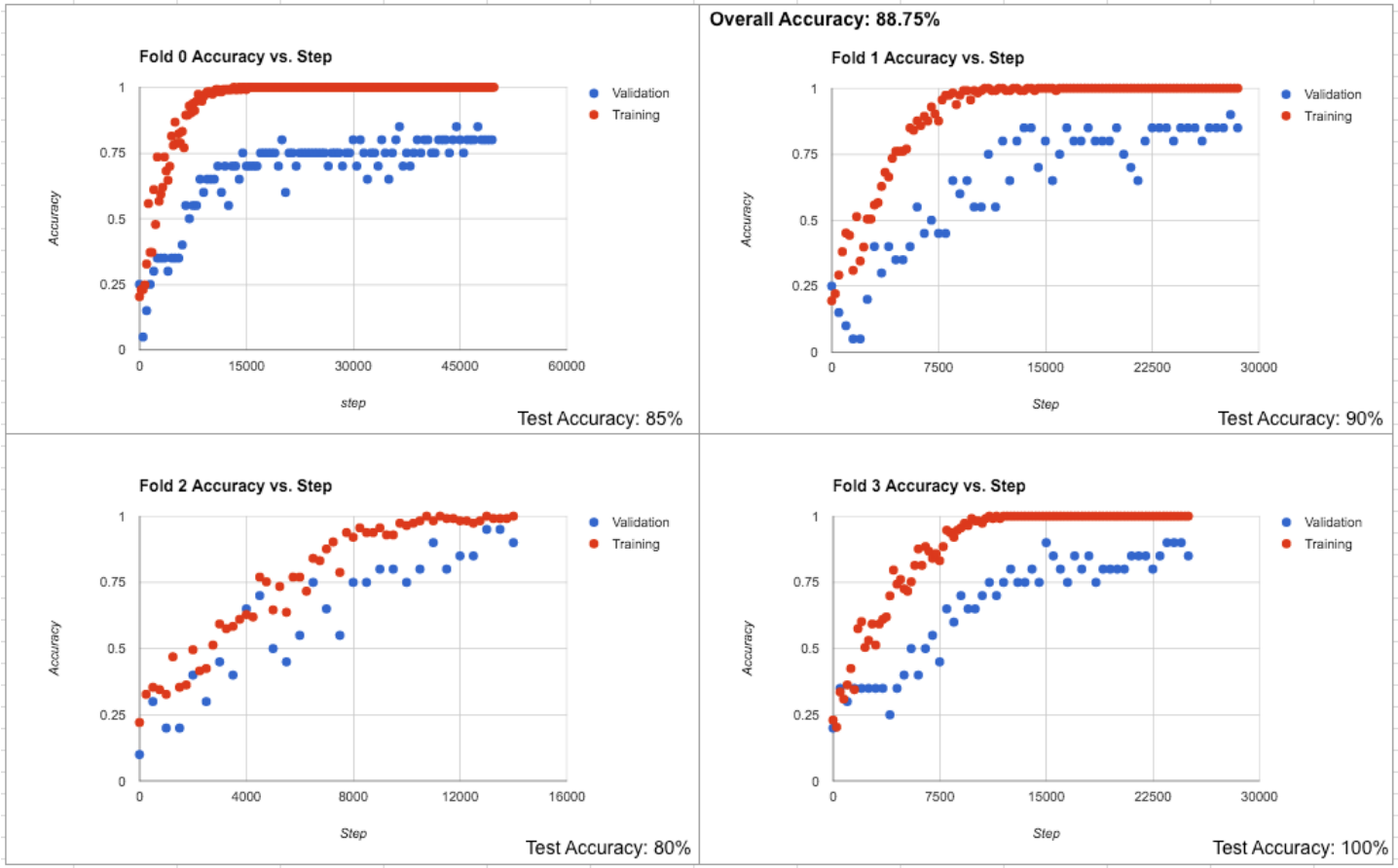
Data Partitioning and 4-fold Cross-Validation

1. Randomly shuffle all 153 images
2. Separate into training(113), validation(20), testing(20)
3. Train images for 50000 steps or until consistently predicting validation set at 85% accuracy or higher
4. Evaluate predictions for testing set
5. Repeat Steps 2-4 with a different partitioning

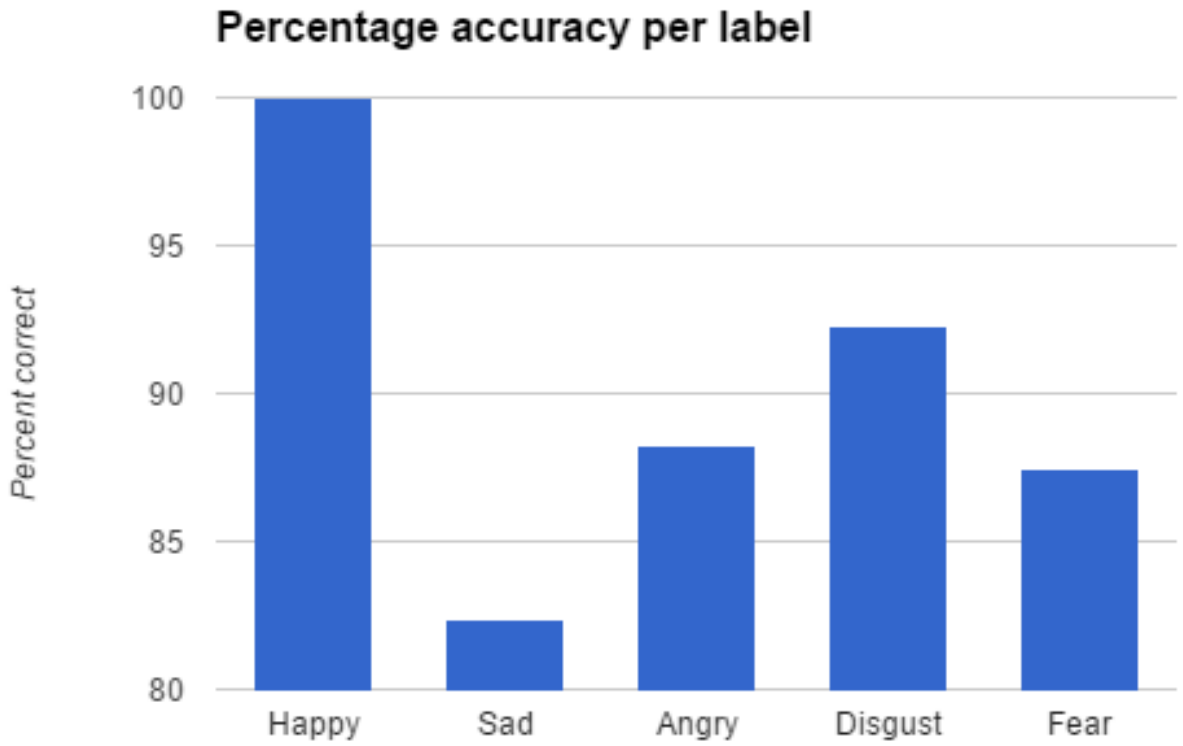
Evaluation Metric

- For each fold, measure accuracy as the percentage of correctly labeled images
- Estimated accuracy of the neural network if trained on all images is the average test accuracy of each fold

Analysis



Average accuracy across all folds: 88.75%



- 100% Accuracy for predicting "Happy" label
- Lowest accuracy predicting "Sad" label

Conclusion

Using our Convolutional Neural Network, we were able to successfully distinguish faces based on their facial expressions with great accuracy.

Happy ✓ Sad ✓ Angry ✓ Disgust ✓ Fear ✓