Facial Gesture Recognition System
Using natural, human gestures to interact with technology

Introduction
Humans naturally express themselves through facial gestures and expressions. A gesture recognition tool can help elderly people, autistic people, or those with disabilities. A facial gesture recognition tool can provide an easy interface to computers or by training people who have difficulty understanding others.

Project Goal
The goal of this project is to design a real-time facial gesture recognition system. The app is intended to interface with other software that can take advantage of detecting facial gestures, such as accessibility software.

Specifications
We use OpenCV, an extensive open source software library that offers many different features for use in computer vision and machine learning. Kaggle, a massive data science resource, and Keras, an open source neural network library. The following features were implemented:
- OpenCV video capture API
- OpenCV image processing API
- OpenCV Face detection cascade
- Kaggle training dataset
- Keras cascades and machine learning API

Methodology
1. Fetch the user’s video using the OpenCV video capture API.
2. Process and clean the image using numpy.
3. Use OpenCV cascades to detect the face within the image and resize.
4. Run the image through our model and output the predicted facial expression.

Training the Models
We use a Kaggle dataset to train our machine learning model. This dataset contains 35,239 samples, and each sample is a set of 256 values. Each value is an integer which describes a single pixel from the image. Using Keras, we created a sequential neural net model with multiple layers. The convolution layers help format the data and get more accurate predictions and the pooling layers combine the outputs and help to control overfitting the model.

Testing and Validation
To validate our model, we took a subset of the training data and tested the accuracy of its predictions and found our accuracy to be around 43% - 44%. Although this is lower than expected, we also validated our predictions on real-time video capture and found that our models can accurately detect 6 out of 7 emotions.

Conclusion and Results
The prototype which we designed and implemented can run on all major desktop operating systems, supports most modern webcams and is able to recognize seven facial expressions about 50% of the time. The accuracy of the model trained to recognize these expressions did not meet our design goal, likely due to the complex nature of the problem.

References

S ENG 499 Design Project
Jerusha Chua, Braiden Cutforth, Joel Kerfoot, Christopher Norton
Supervised by Guoliang Li