Test Specification Document

Tim Fanelle, Christopher Hufham, Patrick Kenny, and Jaime Lujan

Table of Contents

Overview

Testing Goals

Our goal for testing is to ensure that PyFocals is as high quality and error free as possible with respect to our tight timeframe. Comprehensive testing for this application is not possible, but tests will be written to double-check functionality and integration between modules in many different cases. Our most important task will be to try the software with as many different users who have different facial features as possible within our timeframe. The software should ideally be able to adjust and function no matter how different a user's facial features are. The way the system adjusts to every user's different facial features is by using dynamic vertices and points that can be fit to anyone as long as it is not too far from what the algorithm is expecting.

Another important testing concern is testing frame rates and making sure facial movements can be tracked in real time accordingly with the webcam's frame rate. Different hardware can produce radically different quality, which will affect the performance of the software.

Special Considerations

Due to the nature of facial tracking, there are many variables that cannot be efficiently tested for. Various lighting, facial structures, and capture devices are all points integral to the functionality of the software but must be considered separately from the usual software testing regime. Two major potential points of failure for the software are webcam frame rates not being able to hold up, and particular users' facial features not being recognized by the algorithms

Unit Tests

Unit tests can be written for the user interface and binding subsystems of the program. The UI will be tested for all possible configurations of the settings, webcam status (on / off / missing), and a large range of user inputs. The binding table will be tested for strange and conflicting keys, and for timing of input acceptance with regards to system state.

Constraints

Software Limitations

Due to the broad nature of facial tracking applications, the scope of the project remains small. As many prospective features that do not directly contribute to the main goal of the application have been eliminated as possible. The software has additionally been stripped down to a single-window application.

Procedure

Testing plan

Testing will be conducted for individual modules once they near completion. Ideally, unit tests can be written for the user interface and key binding modules without much issue. The amount and comprehensiveness of the tests we

administer depends on how on-schedule we are as we approach towards the deadline.