

Flockulator

I have designed an interactive installation that captures and responds to users' facial expressions through an immersive auditory and visual experience, providing a new way to represent and communicate emotions. My application uses machine learning to associate facial gestures with different flocking algorithm controls. The flocking algorithm, in turn, controls the parameters of a granulator.

I am using Max MSP and the jit.cv objects to greyscale the webcam and split it into 300 different sectors. This data is then sent via OSC to Wekinator where I have trained a model that maps different facial expressions to 7 parameters for control of the flocking algorithm in Openframeworks. The model is trained on a resting face, a happy face, an angry face and a surprised face. This technique is flawed however as you aren't actually using face data, you are using RGB data from the webcam, but all of the tools within Openframeworks that could be used to map face data were out of date and did not work with the Openframeworks update. This does then cause issues with lighting when using the application. The data is sent once again via OSC to Openframeworks where each parameter value is sent to the flocking algorithm. The behaviour of the flocking boids is then used to control the parameters of a granulator. I used data such as the average speed, distance from the centre and x and y positions to control parameters like pitch, speed, grain length and overlaps.

If I had time to develop this project more I could have used either a Kinect webcam or some other face-tracking software to map actual face data. I would also train a larger model, using more training data to give a more accurate model within Wekinator. However, I do like the imperfections of the application because they produce some very interesting results, especially when you try to distort the face mapping using your hands or objects.