Detecting the Awesome Guitar Solo Moment with GuitarFace

Regina Collecchia
Stanford University
colleccr@ccrma.stanford.edu

Roshan Vidyashankar
Stanford University
rvid@ccrma.stanford.edu

ABSTRACT

It is difficult to tackle the problem of identifying "awesomeness" without facing serious judgment calls on musical taste. One thing that does give away the joy and energy contained in a moment is the facial expression of a performer. Aiming to recognize the face a musician makes during an awesome musical moment, we created *GuitarFace*, a Mac application with facial recognition via the OpenCV library. The user is a musician practicing their instrument either alone or across the network with a friend. Gameplay is a mix of MIDI guitar input visualization, counts of musical events like big jumps in pitch or power chords, and events initiated by making a guitar face. In another sense, GuitarFace is the beginnings of a "funny face" detector.

1. INTRODUCTION

Before we went the route of facial detection, we were interested in detecting moments of high quality and expressiveness in guitar solos. However, we soon realized that even if we had a perfect feature set that everyone agreed upon as the common ingredients in all great guitar solos, problems like pitch detection, source separation, and extra-musical features like live performance posed serious obstacles. Thus, we narrowed our goals to focus only on the extra-musical events: the way a guitarist stands and holds his or her guitar, the way they move around stage and engage with fans and fellow bandmates, and of course, the face they make during all of this.

Software like Guitar Hero succeeds in delivering some of the emotion of being on stage to the living room, and may even increase the musicianship of players. In the same vein, GuitarFace is meant to give users that feeling as they practice their instrument. Otherwise, we were motivated by humor, joy, and the "laptop musician problem," referring to how the performer-computer interaction tends to alienate an audience during a live show. We could see facial recognition technology to aide live performance production, by cueing new camera views or triggering graphical events such as virtual fire projected behind the performer. It could also be used for something more musical, like turning on a guitar effect preset to double octaves. With that said, we have not totally resolved how a camera could track a performer's face unobtrusively were they to use it on stage; this research is instead directed at the detection of guitar face.

1.1 Background

Facial recognition is a rapidly advancing field of machine learning and computer vision, now built in to popular services like Facebook and applications like iPhoto. Many models for facial recognition use support vector machines (SVM), active appearance models (AAM), Haar classifiers, and other supervised learning models

2. METHOD

After reviewing hundreds of photos found by an image search for "guitar face," we named 7 facial features that we believed to be more guitar face than not:

1. mouth open, as if in pain
2. mouth tightly pursed, slightly smiling
3. eyes (pinched) closed
4. lips unaligned, mouth open
5. tongue out
6. head tilted back or to the side
7. mouth in an 'O' shape, as if yelling

Working with the time constraints and resources that we had, we designed algorithms for filtering the state of the mouth and the eyes, and triggered a "guitar face event" whenever the eyes were closed or the mouth was open, past a threshold of 0.18 times the height of the face. [1]

3. IMPLEMENTATION

Our attempts to detect and track guitar face were in 3 stages:

- using the built-in face tracker in OpenCV based on Haar-like features;
- using a model for face-tracking developed by Jason Saragih and ported to C++ by Kyle McDonald, annotating the face with 66 coordinates; and

[1] Here "face" refers to the 66-point matrix drawn by [citation], so the height of the "face" would be from the bottom of the chin to the top of the left eyebrow. See Figure X.

- exploring SVM and AAM based trainers.

We had mixed success with all of these approaches, discussed more in Section 4.

We first used the built-in face tracker in OpenCV, which draws a rectangle around the face with pretty strong accuracy. We did not find the other (Haar-based) facial feature trackers to be accurate enough or work well enough in real-time. We used Kyle McDonald's C++ version of Jason Saragih's code for face-tracking, which annotates the face with 66 coordinates. It works best on faces that are straight on because of the symmetry component in the face-tracking algorithm, so we had difficulty when applying it to tilted faces. Finally, we attempted to manually train an active appearance model by annotating 20 photographs by hand, using Radhika Vathsan's trainer.

Using the built in face detector in OpenCV, the rectangular area around the face was detected with high accuracy and only a few lines of code. The strongest feature that we assimilated to "guitar face" was an open mouth, so we tackled this feature first. We tried a few different techniques to detect this feature with interesting results and mixed success.

We assumed that the face was straight on.

4. EVALUATION

Into the GuitarFace application, we built in a key press to allow us to say that we were making a guitar face, and recorded how many times our software successfully identified the face. For the black pixel method, we had a X success rate. For tracking the height of the eyes and lips, guitar faces were detected X% of the time.

5. CONCLUSION AND FUTURE WORK