

158B Final Project

Kevin Chau

<https://github.com/kevinchau321/music158B>

Reflection

I set out to create a project with a new way of interacting with all the available DJing and music production software readily found on consumer computers. In this regard, I felt I was fairly successful in unify different aspects of each program under one interaction/gesture space using a limited number of hardware controllers, as opposed to having one specific controller for each which is traditional (i.e. a mixer for traktor, Push for Ableton, separate effects module). What I was really aiming for was a way to combine aspects of DJing and production with aspects of live performance and interactivity.

I started by learning how to use the leap motion for control data. At first I sent out the control data to Ableton because I like how you can chain effects, but eventually I switched over to using the ableton module because I felt that the interface was much cleaner. Besides, the 2 module model in traktor is very conducive to a two handed effects controller, and mapping in Traktor can be more straightforward than Ableton sometimes.

After I got Traktor working with the leap motion, I moved onto working with visualization. At first I imagined that visualization would be great for displaying on a projector at a show, along with lights. However, the graphics card and process on my computer seems to have issues mixing audio and rendering videos at the same time. Perhaps a faster computer or a desktop would be needed to have full frame rate visualization. However, these issues aren't nearly as interesting as what I learned about the playability of the leapmotion. With the visualizer, it instantly became much easier to practice gestures, and have some sort of repeatability. As a performance tool, it is invaluable, regardless of the lag.

After working out visualization, I spent some time learning how to route audio internally in my computer. This allowed me to add soundcloud streaming directly into my traktor mixing setup, which is great for endless sources of original audio files and

sounds!

In the final stages of the project, I worked with Sam on cleaning up the code base and refining gesture recognition. This led us into discussing how we could use the leap motion to trigger notes rather than just send control data. We decided to use a sample bank in Ableton for quick swapping of audio files and access to nearly unlimited macro controls with the few control data we developed in the first parts.

Sam and I actually spent a lot of time rehearsing and practicing with our new instruments. We found them really enjoyable to jam out on. Once we got a reliable system, we were able to focus on the samples we were using, and learning about every last bit of articulation and expressivity that we could get with the leap motion.

In the end, I was really glad with what we came up with. Our use of odot was very efficient and was invaluable for working with the leap motion data. I believe a big limitation of this project is its complexity. With more odoting, engineering, and practice, I think I could streamline the entire thing to be more reliable. It's a lot of programs open at the same time, but the data flow is rather straightforward. I'm really starting to understand the big ideas of computer music and how we can build up complex systems from all the basic components that we have.

Setup

1. Open **leapmotionKEVIN.maxhelp**
2. Open **visualizer.maxpat**
3. In presentation mode, turn on 1 Toggle in visualizer.maxpat to turn on the audio visualization. The 3 toggles in leapmotionKEVIN toggle whether or not the leap motion is taking data, and whether or not the right hand or left hand data is being sent out from MAX/MSP. Turn all 3 on.
4. Open Traktor and Ableton, with the necessary project files listed below. Use an audio routing program to route audio from Ableton to one Traktor deck. For example, I used soundflower to bridge two channels of audio from Cross DJ into Deck C of Traktor, and used another two channels to route everything from Ableton into Deck D.
5. Jam away!

Software/Hardware Dependencies

1. Max/MSP 7.2+ (Leap)
2. Traktor Pro 2.0+ (For FX units)
3. Ableton Live 9.0+ (For sample launching)
4. Mixvibes Cross DJ (For mixing/streaming soundcloud audio files)
5. Some type of audio bridge: I used soundflower, but it had performance issues.
Recommended solution: Route with external audio hardware, or use JACK.
6. Leap Motion (+drivers), Midi Keyboard

Directories and Files

158wrappers/leapmotion

Contains the max externals for capturing Leap Motion data.

Based on the Leap Motion patch from IRCAM. Most of the odot abstractions can be found in this folder.

leapmotionKEVIN.maxhelp: Main patch for leap motion data.

158wrappers/mpk249

Contains the max externals for capturing MPK249 (or any similar midi controllers) data. Based on the bcf2000 patch written by Rama Gottfried for music 158B at CNMAT.

m58.o.io.mpk249.maxpat: Main patch for midi keyboard data.

visuals/

contains images and max patches for displaying hand tracking data and audio visualization in a completely 3D environment rendered with OpenGL. Also contains an odot interface for controlling the 3D camera with a midi controller.

visualizer.maxpat: Main patch for visualization.

masterset.tsi: Traktor master set. Contains the midi mapping for the MPK249, the midi mapping for Max's midi control data generated from the leap motion.

mpk249(keyboard).tsi: Backup Traktor mapping for the MPK249.

Mpk249keyboard.mappings: Midi mapping for Mixvibes Cross DJ. With this mapping, you can mix 2 decks in traktor and 2 decks in Mixvibes with the same exact hardware interface, despite the Cross DJ software being entirely separate. Useful for mixing streamed audio in the same way a traditional DJ would mix an internal library.

finalset Project: This is an ableton live file which contains the samples, drumpad, midi mapping, and macro mapping for trigger notes with the leap motion.

Cnmatspeakers.maxpat: Simple patch for taking 2 channel audio from any external input and projecting them onto CNMAT 8 Channel system.

16Steps_3.0.amxd: Experimental Step Sequencer for the MPK249

Notes MPK249 mapping:

The “mapping” I created for the MPK249 allows the controller to mix different programs. Preset 25:Traktor on the keyboard contains all the mapping to turn on the MAX/MSP toggles (such as the render and leap motion data toggles). It also maps to all the traditional DJ controls found on a Traktor S4 or a Pioneer mixer, so you have full control of the software. Preset 26: Camera, contains a control bank for manipulating the 3D visualization environment. With these controls, you get full 360 rotation around the visualization ball and you can get any angle of the hands being displayed. Preset 27: Ableton contains all the midi mappings used for Ableton. It controls the mixer in Ableton, as well as a drum rack for launching samples, midi keyboard for synthesis, and a max for live sequencer (16Steps_3.0.amxd).