Hand Motion-Controlled Audio Mixing Interface

Jarrod Ratcliffe
Department of Music and Performing
Arts Professions, Music Technology
New York University
jpr350@nyu.edu

ABSTRACT

This paper presents a control interface for music mixing using real time computer vision. Two input sensors are considered: the Leap Motion and the Microsoft Kinect. The author presents predominant design considerations, including improvement of the user's sense of depth and panorama, maintaining broad accessibility through integration of the system with Digital Audio Workstation (DAW) software, and implementing a system that is portable and affordable. To provide the user with a heightened sense of sound spatialization over the traditional channel strip, the concept of depth is addressed directly using the stage metaphor. Sound sources are represented as colored spheres in a graphical user interface to provide the user with visual feedback. Moving sources back and forward controls volume, while left to right controls panning. To provide broader accessibility, the interface is configured to control mixing within the Ableton Live DAW. The author also discusses future plans to expand functionality and evaluate the system.

Keywords

music mixing, music production, computer vision

1. INTRODUCTION

The impact of technology on music over the past 60 years is quite difficult to overstate. It has changed the way music is performed, recorded, consumed, and in many cases, how it is composed. It has spawned the invention of new musical instruments. Without these developments, the NIME conference and community may not exist today. It is quite remarkable then, to consider that new interfaces have had such a limited impact on music production and recording studio technology. Since the first analog mixing console was released in the late 1950s, very little has changed in its design. For each incoming channel, a channel strip is used with several knobs, switches, and a fader, each controlling a specific parameter in a one-to-one mapping. This interface has carried through to digital mixing consoles, and metaphorically into software mixers in digital audio workstations.

It is a common goal in audio mixing to create a specific sonic image for the listener, utilizing psychoacoustics to provide localization cues. In a stereo mix, the most basic parameters that control this sonic image are lateral position and depth. With the channel strip metaphor, these parameters are approximated using a pan potentiometer and a fader (controlling level) respectively. The author would like to specify that these are approximate correlations, because, in a physical space there are other psychoacoustic parameters that correlate with sound localization in humans, including spectral content, and time delays. These parameters are often emulated using artificial reverberation.

While the channel strip metaphor offers precise control over many sonic parameters in a mix, there is one main challenge with the mapping techniques that are used for width and depth: By looking at a mixing console, the position of a sound source within the image is not necessarily immediately apparent. Pan potentiometers are a reasonable representation of apparent lateral position, however the channel that is physically located on the left-most side of the interface may in fact be panned hard right, creating a dissonance for the user. The user must look at the position of the pan knobs for every channel to get a sense of the lateral position of a source, and there is typically no direct way to visualize this sense of stereo image. In addition, with the channel strip metaphor, faders control level, which has a perceived impact on depth of a sound source. The challenge with this mapping is that the fader position is actually inversely proportional to apparent depth. A fader that is in the position closest to the user produces a sound that is perceived as being the farthest away. This inverse relationship is easily learned by the user, however, when compounded with the dissonance created between the lateral position of the channel strip and the position of its corresponding sound source, this can make it difficult to localize multiple sound sources simultaneously, posing challenges to the user in the perception of relationships between sources.

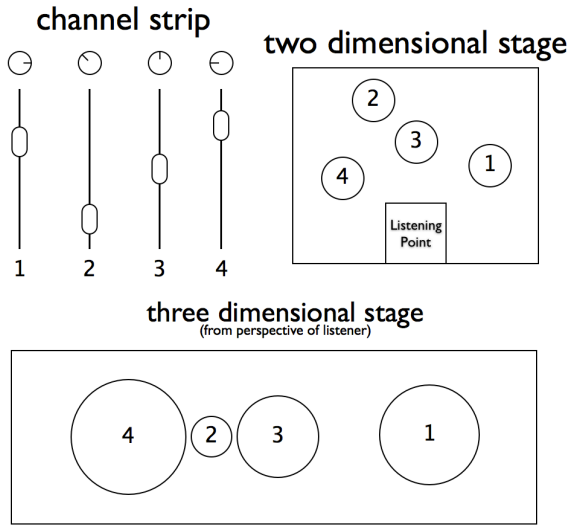

Figure 1. Channel Strip vs. Stage Metaphor

The current work addresses the control of apparent lateral position and depth using the stage metaphor. The stage metaphor defines a listening point and allows the user to control panning and level with the distance between the sound source and the listening point in two corresponding dimensions simultaneously. While most implementations of the stage metaphor either use keyboard and mouse or multi-touch (see Section 2) , the current work explores an implementation using computer vision with position tracking.

2. RELATED WORK

2.1 Depth Mixing and the Stage Metaphor

There have been several implementations of "depth mixing" using the stage metaphor. Gibson [9] discusses a theoretical "virtual mixer," in which sounds are represented as spheres in three dimensional space. The position of the sphere in each plane controls a different sonic parameter, with three main modes: mix, effects, and equalization. For the mix mode, he proposes a stereo option and a surround option. Pachet and Delerue [18] propose a standalone mixing application with a two dimensional GUI controlled by a computer mouse in which end-listeners are presented with widgets for all instruments and a widget for the listener. Each widget is moveable, allowing the users to customize their own mixes. Holladay [10] also proposes a standalone mixing application with a sound stage GUI controlled by the computer mouse, but it is designed for professional audio mixing engineers as opposed to end-listeners. Diamante's system [5] also allows the user to move all widgets, including the listener position, but it is implemented as a GUI on a tablet PC acting as a control surface for a digital audio workstation. In addition to attenuating sound sources as they are placed farther away, this system also shelves high frequencies when sources are placed farther still, providing additional spatial realism. Carrascal and Jordà [3] use the multi-touch capabilities of the Reactable [20] in their implementation of the stage metaphor. Gelinek et al [7-8] also use the Reactable, but they include fiducial tracking with tangibles and "smart tangibles," which are essentially fiducial tangibles containing micro-controllers, allowing for additional flexibility in the mapping of multiple parameters.

2.2 Gesture and Computer Vision Control

There have been several implementations of computer vision control in music systems. Many of these systems focus more on sound transformation [6], [16] and synthesis [4], [19] rather than depth, panorama, spatialization or mixing. There have been some implementations of gesture-controlled interfaces with Digital Audio Workstation (DAW) software. Balin and Loviscach [2] recognize importance of the integration of gestures with DAW software, however they focus on gestures performed on the computer mouse. Many of the implementations using gesture control for mixing or spatialization [14], [21] utilize physical sensors and motion capture rather than computer vision. After thorough review of the literature, Lech and Kostek [12] provide the only use of computer vision and gestural control of audio mixing utilizing the stage metaphor. They implement a system that identifies specific gestures (static and dynamic), and sends them as MIDI messages to a DAW. The system uses a webcam, a multimedia projector, and a projector screen. The camera feed is subtracted from the projector image to locate any change in hand position. While user testing for this interface produced results suggesting reasonably precise control, the use of a projector screen may be a limitation for users in some environments. The author would like to address this in the current interface.

3. DESIGN GOALS

In approaching a design for an audio mixing interface using computer vision, the author proposes some important factors:

1.) To provide the user with a better sense of depth and panorama, the interface should use the stage metaphor.

2.) The interface should be designed as a controller, sending data to a DAW, rather than a standalone, autonomous mixer.

3.) The interface should be portable, accessible, and easy to use.

4. SYSTEM DESIGN

4.1 Choosing the Input Sensor

In choosing an input sensor to use for the current work, the author kept a few factors in mind, primarily optimization for human body tracking, speed, accuracy, and affordability/accessibility. With these factors in mind, the author felt that two sensors specifically should be considered: the Microsoft Kinect [17] and the Leap Motion [11]. The author has performed some prior work using skeleton tracking with the Kinect to control various sonic parameters. The author considered the Leap Motion as well, primarily because it allows joint-based tracking (similar to the Kinect), but optimized for the hands and fingers. This kind of tracking does not require the user's entire body to be in-view of the sensor in order to identify specific joints, meaning the user can use the system while seated at a desk or table surface, typically a familiar, comfortable working position for anyone who is mixing music. Working in this position could potentially allow for greater comfort and ergonomics than requiring a user to remain standing, as such would be the case with the Kinect. In addition, the frame rate of the Leap Motion is significantly faster than that of the Kinect, allowing for more accurate real time tracking. For these reasons, the author decided to implement a system using the Leap Motion.

4.2 Challenges of a Touch-Free Interface

There are significant challenges in implementing any touch-free system based solely in computer vision. Any system without a tangible, physical interface will, by design, lack haptic and tactile feedback. The presence of this feedback, in many domains can improve upon the performance of the system. In addition to feedback, precision can be a concern with computer vision systems as well.

In early stages of implementation, design options were considered that would keep the interface entirely touch-free, allowing channel selection and sound source positioning to be controlled via motion tracking and gesture recognition using sensor data from both hands. The initial design utilized the Leap Motion's built-in finger tracking as a means of channel selection. Users could hold up a certain number of fingers that would correspond to the channel number they wished to enable for mixing. One significant challenge this posed was in getting the sorting algorithm used in the Leap Motion SDK to comply with this design. The algorithm labels user hands in order of detection, and does not differentiate between left and right. With this type of sorting implemented into the mixing interface, users would need to insert their hands into the sensor's field of view in a particular order, and keep them in range during the entire mixing task, or the mapping of hand data would default to the first hand when one hand was removed. Manually sorting hands by locating the left-most and right-most palm positions was effective, but was disrupted when the user crossed hands. In addition, the need to keep both hands above the sensor throughout the mixing task quickly caused "gorilla arm" fatigue.

To address these issues, the author resolved to the introduction of an additional affordance to function in a supplementary role, performing simple channel selection. This would then allow for one-handed gestural control of sound source placement. When considering an additional affordance for channel selection, the author aimed to preserve the design goals of portability and accessibility. Bearing in mind the ubiquity of the mobile phone and tablet, a custom TouchOSC layout was developed, usable by anyone with an iOS or Android device. An overview of the system can be found in the following section of this paper.

4.3 System Overview

The current system captures sensor data from the Leap Motion using the Leap Motion SDK, Max [15] and a Max external written by R. Luke DuBois, a colleague of the author's.

The TouchOSC layout is used to open gates in Max, assigned to channels in Ableton Live [1]. Once a channel is selected, the user can control the position of the sound source according to the stage metaphor. Position data of a user's hand is mapped to control the depth (level) and panning of the corresponding channel within

Ableton Live, and to supply the user with additional visual feedback through a custom graphical user interface (described further in section 4.4).

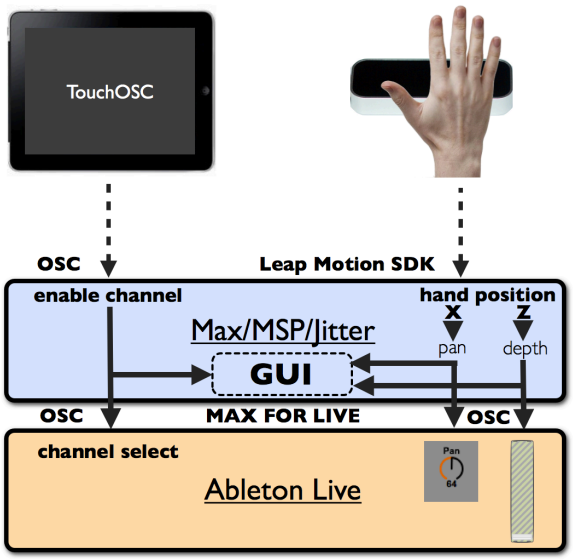

Figure 2. System Design

The palm position data in the x and z planes are scaled to appropriate values for pan potentiometer and fader. The channel's left to right panning is controlled by the scaled x plane data, while the depth of the source in the mix is controlled by the scaled z plane data. Once a channel is enabled in the TouchOSC layout, moving a hand over the sensor, the user can physically push the sound source back, farther away on the virtual stage, or pull it closer, as well as slide it laterally. Once the user positions the sound source in the desired location, it can be locked in place by de-selecting the channel in the TouchOSC layout. In addition to providing an easy solution for track selection, the use of TouchOSC allows for improved ergonomics, as the user does not need to keep a second hand over the sensor. Also, when all tracks are locked, the position data is not applied to any channel, in order to avoid unintended re-positioning of sound sources.

All data filtering and processing, in addition to the visual feedback system are implemented within one single Max/MSP/Jitter environment. The sensor data is sent over the Open Sound Control protocol and routed to channels in Live using Max for Live devices and the Live API.

4.4 Graphical User Interface

As discussed in Section 1, one significant reason behind using the stage metaphor for mixing is to provide the user with a better sense of sound source position in virtual space than is afforded in a traditional channel strip metaphor. Left to right position of a channel strip on a mixing console has no correlation with where a sound source is actually being positioned. To assist the user in providing clarity of sound source spatialization, a graphical user interface was designed to give users visual feedback during the mixing process.

The GUI was designed using OpenGL animations in Jitter, and loosely based on the visual approach used in Gibson's Art of Mixing [9]. Sound sources are represented as spheres in virtual three-dimensional space, differentiated by color. Color assignment is arbitrary, but provided to supply visual contrast. Additionally, all track names are sent directly from Ableton to the GUI using OSC messages, and the name of each channel is positioned in front of its corresponding channel's sphere.

The same palm position data that is scaled to MIDI values and sent to Ableton Live is also scaled to control the position of the Open GL sphere corresponding with the user-selected channel. Lateral positions of spheres are thus representative of the lateral position of their corresponding sound sources. In addition, sources that are closer to the user are seen as larger spheres, while sources that are deeper in the mix are seen as smaller spheres. Rather than simply representing panning and mix level values via a potentiometer knob and dB values from a fader, the GUI provides the user with a sense of the space being created while mixing the audio sources.

Figure 3. Graphical User Interface

5. DESIGN CHALLENGES WITH THE STAGE METAPHOR

One of the most significant challenges in designing a mixing interface using the stage metaphor is the organization of channels and the parameters corresponding to those channels. While the channel strip metaphor does make it more challenging to visualize spatialization and the concepts of depth and panorama, it also makes it incredibly easy to access a specific parameter of a specific channel, as the channel is always in the same place. In contrast, the channel organization and labeling in a mixing interface designed with the stage metaphor gets more difficult as the number of channels increases. Mixing engineers typically work with upwards of 24 channels in any given session. Gelinek et al [7] point out that this may likely be why even novel audio mixing interfaces, like Liebman et al. [13] are using the channel strip metaphor. Often times, complexity can be minimized by including layers of functionality within a system. While layers of functionality can make an interface easier to use, they can also make the system more time-consuming to use. This issue is addressed more in section 6.

6. CONCLUSIONS AND FUTURE WORK

The current system responds reasonably well to hand motion. In personal testing, the author has set the tracking to be quite sensitive to smaller movements, so one does not have to make large-scale gestures to impact parameters. The author would like to expand functionality of the system and test it with various subject populations, including professional mixing engineers, performers, and novices. By having subjects mix with the current system alongside other audio mixing interfaces (analog console, a DAW mixer), the author can gain insight into how the system compares. In addition, asking subjects to blindly rate mixes performed on different interfaces, additional subjective ratings can be drawn. Carrascal and Jordà [3] found in a preliminary evaluation that when users were given the chance to mix with their multi-touch stage metaphor system and with an analog console, the tasks were completed quicker by every subject on the multi-touch mixer. In addition, users preferred the multi-touch mixer to the analog console.

Before evaluating the current system, the author would like to expand its functionality. Before it can be compared to a traditional mixer, the current system should have a similar level of control parameters to a traditional mixer. The author would like to implement different modes into the system, including equalization, effects, and surround/multi-channel. Also, as mentioned in Section 5,

adding more channels to the system would also be a highly desirable feature. The main challenge with implementing additional channels and control parameters with the current system is that the user interface may get extremely cluttered. To address the addition of more channels, the author would consider implementing a two dimensional stage metaphor, as this would allow the user to see more source widgets without the GUI getting overly-cluttered. It seems that this technique was used with reasonable success by Carrascal and Jordà [3] and Gelinek et al [7-8]. Another option would be to stagger heights of sound source spheres. This technique is used by Gibson [9] to visualize high channel counts in multi-track mixes. To address the addition of more control parameters, the author would like to consider adding support for advanced gesture recognition, perhaps using static and dynamic gestures. This seemed to work well for Lech and Kostek [12]. In this type of implementation, the author would want to focus on gestures that are appropriate to the control tasks.

In addition to adding functionality to the current system, the author would like to connect the interface with other popular industry DAWs. Using OSC, or perhaps sending MIDI continuous control data over an IAC bus, it is possible to connect the interface with Cubase, Logic, and Pro Tools. Expanding output connectivity could allow more users to try out the system in a DAW of their choice.

Given the current system's performance, the author believes there is some promise in the implementation of a computer vision-based audio mixing interface.

7. ACKNOWLEDGMENTS

The author would like to thank Dr. R. Luke DuBois for supplying his Leap Motion external for Max, and the NYU Steinhardt Music Technology Program for supporting this research.

8. REFERENCES

[1] Ableton Live. http://www.ableton.com

[2] W. Balin and J. Loviscach. Gestures to Operate DAW Software. In *Proceedings of the 130th AES Convention*, London (2011).

[3] J.P. Carrascal and S. Jordà. Multitouch Interface for Audio Mixing. In *Proceedings of New Interfaces for Musical Expression*, pages 100-103, 2011.

[4] G.C. de Silva, T. Smyth, & M.J. Lyons. A Novel Face-tracking Mouth Controller and its Application to Interacting with Bioacoustic Models. In *Proceedings of New Interfaces for Musical Expression*, pages 169-172, 2004.

[5] V. Diamante. Awol: Control surfaces and visualization for surround creation. Technical report, University of Southern California, Interactive Media Division, 2007.

[6] A.L. Fuhrmann, J. Kretz, and P. Burwik. Multi Sensor Tracking for Live Sound Transformation. In *Proceedings of New Interfaces for Musical Expression*, pages 358-362, 2013.

[7] S. Gelineck, M. Buchert and J. Andersen. Towards a more Flexible and Creative Music Mixing Interface. In *Proceedings of ACM SIGCHI Conference on Human Factors in Computing Systems*, Paris, 2013.

[8] S. Gelineck, D. Overholt, M. Büchert, and J. Andersen. Towards an Interface for Music Mixing based on Smart Tangibles and Multitouch. In *Proceedings of New Interfaces for Musical Expression*, pages 180-185, 2013.

[9] P. Gibson. The Art of Mixing: A Visual Guide To Recording, Engineering and Production. ArtistPro Press, 1997.

[10] A. Holladay. Audio Dementia: A Next Generation Audio Mixing Software Application. In *Proceedings of the 118th AES Convention*, Barcelona, 2005.

[11] Leap Motion. http://www.leapmotion.com

[12] M. Lech and B. Kostek. Testing A Novel Gesture-Based Mixing Interface. In *Journal of the Audio Engineering Society*, vol. 61 (5), pages 301-313, 2013.

[13] Liebman, N., Nagara, M., Spiewla, J., and Zolkosky, E. Cuebert: A new mixing board concept for musical theatre. In *Proceedings of New Interfaces for Musical Expression*, 2010.

[14] M. Marshall, J. Malloch, M. Wanderley, "Gesture Control of Sound Spatialization for Live Musical Performance," in Gesture Based Human Computer Interaction and Simulation, M. Sales Dias (ed.), Berlin , Springer, pp. 227–238 (2009).

[15] Max. http://www.cycling74.com/products/max

[16] D. Merrill. Head-Tracking for Gestural and Continuous Control of Parameterized Audio Effects. In *Proceedings of New Interfaces for Musical Expression*, pages 218-219, 2003.

[17] Microsoft Kinect. http://www.xbox.com/en-US/kinect

[18] F. Pachet and O. Delerue. On-the-Fly Multi Track Mixing. In *Proceedings of the 109th AES Convention*, Los Angeles, 2000.

[19] R. Polfreman. Multi-Modal Instrument: Towards a Platform for Comparative Controller Evaluation. In *Proceedings of the International Computer Music Conference*, pages 147-150, 2011.

[20] Reactable. http://www.reactable.com

[21] R. Selfridge, J. Reiss, Interactive Mixing Using Wii Controller. In *Proceedings of the 130th AES Convention*, London (2011).