

---

# musicAir: Creating Music Through Movement

**Irene Kwok**

Wellesley College  
106 Central Street  
Wellesley, MA 02481 USA  
ikwok@wellesley.edu

**Johanna Okerlund**

Wellesley College  
106 Central Street  
Wellesley, MA 02481 USA  
jokerlund@wellesley.edu

**Advisor****Orit Shaer**

Wellesley College  
106 Central Street  
Wellesley, MA 02481 USA  
oshaer@wellesley.edu

**Charlene Lee**

Wellesley College  
106 Central Street  
Wellesley, MA 02481 USA  
charlene.lee@wellesley.edu

**Qiuyu Zhu**

Wellesley College  
106 Central Street  
Wellesley, MA 02481 USA  
qzhu@wellesley.edu

**Abstract**

musicAir is an interactive gesture-based technology that allows users to create music out of thin air in a fun and engaging way. Hand and finger gestures map to corresponding visuals and sounds in a collaborative and playful interface that invites multiple users to create harmonious tunes together. musicAir is aimed at people who seek to create music easily in a social setting with little or no musical background and is accessible to any age group. We describe the ideation, context, and implementation of musicAir using Leap Motion technology.

**Author Keywords**

tangible interaction; gesture based interaction; musical instrument

**ACM Classification Keywords**

H.5.2 [User Interfaces]: interaction styles, input devices and strategies; J.5: [Arts and Humanities]: performing arts; H.5.3 Group and Organization Interfaces; H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

**Introduction**

Traditionally, making music requires musical skill and years of practice. musicAir provides users with no prior musical background the means to create music through a fun, collaborative experience. Gestures intuitively map to visuals and sounds in an interface that also invites users to work together. A group of users can use the system simultaneously, encouraging people to

collaborate and build off each other's work, fostering a greater creative environment (Figure 1). musicAir also attracts a greater audience by inspiring participants to engage in the experience through observation. Simply by watching someone else use the system, one learns how to use it themselves.



**Figure 1.** A sample interaction with the system and two users

The size of the space in which the users perform is larger than the display screen, incorporating more of the physical environment and requiring more dramatic, full gestures. Through musicAir's interface, users can wave their hands or move their fingers and hear pitches that rise and fall and different types of sounds according to their hand movements. Employing the use of sensor-based technology, musicAir is a tangible interaction for artistic and musical system that has versatile purposes and invites anyone to become a creator of music.

### **Related Work**

Existing work in HCI research revealed many music generation interfaces. Examples include "WiiPaint" - a tangible user interface that draws colored visuals with a

Wii remote and plays various corresponding types of pre-recorded melodies [7]. Marble Track Audio Manipulator (MTAM) is another interface that augmented a familiar toy and enables children to compose a piece of music collaboratively [2]. Another example is the reacTable, an electronic musical instrument that allows users to interface with a visual display via the tangible objects or their fingertips to collaboratively create music or sound effects [4]. While many designs successfully incorporated music structures in tangible objects, there are also prototypes that tap into creative visualization of music and further exploration of the audio-visual relationship. Examples in this category include "Harmonic Paper" [5], a tangible user interface that converts drawings into sound, and "Writing with Music" [1], an interface that explores the use of auditory feedback in gestures. All of these examples consist of some tool to generate music, which is where musicAir stands out among existing approaches. Rather than putting physical constraints on an instrument needed to create music, musicAir focuses on the flow of music composition and takes natural human gestures as sole inputs.

### **Sample Use and Environments**

#### *Classrooms*

Students as young as in preschool can intuitively use musicAir without previous instruction or musical experiences. As soon as they step up to the interface, children can begin waving their hands and wiggling their fingers, and music instantly will play as each of their motions adds more layers of sound. The screen turns bright with colors as the various light waves appear according to the movements and sounds.

#### *Museums*

musicAir also has a collaborative and inviting element to it. The music that one user creates also invites others to surround the user and join in on the music making by poking the air and swaying their hands to add more beats and rhythms. The collaborative aspect of musicAir allows many users to continue to build upon

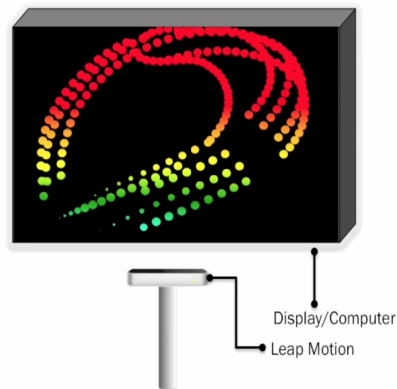
their work and experiment with new motions and sounds.

#### *Music Studios*

Music composers and students can also craft more advanced work through musicAir through the Musical Instrument Feature and Loop Mode. This opportunity allows users to create more complex melodies with simple hand gestures or to experiment with composition by switching sound textures and simulating different instruments or layering the sounds to create a piece and see visual representations of their work.

#### **Interface Setup**

The interface of musicAir consists of three components: display screen audio output device Leap Motion [6], a computer hardware sensor that supports hand and finger motions as input (Figure 2). Users place the Leap Motion sensor in front of a connected screen that displays the visual output with speakers nearby to play the sound output. Then, in order to capture the full extent of musicAir's capabilities, users must stand in front of the sensor so that their arm is within the sensor's range when extended.



**Figure 2.** The user stands in front of the display with hands above controller.

#### **Gesture, Sound, and Visual Mapping**

musicAir relies on Leap Motion technology that senses hand and finger motions as input to create music. musicAir capitalizes on a user's inherent understanding of the visualization of sound.

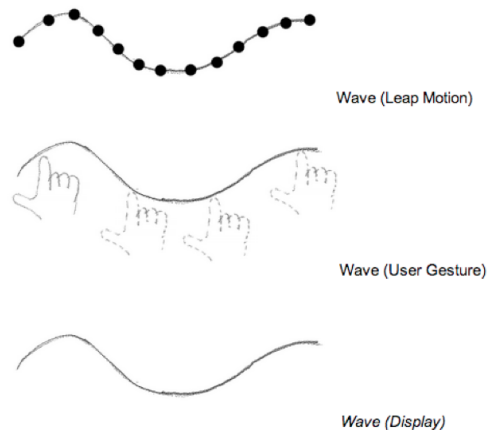
There are two modes in which a musicAir operates. The user can toggle between modes by clapping their hands twice.

*Instant feedback mode:* In instant feedback mode, as the user makes a gesture, notes sound, creating music directly from the gesture and displaying visuals on the screen. Then, as the sound from the gesture fades from the room, the visuals corresponding to that gesture fade from the screen. This mode is so users can understand and explore the gesture to visual and sound mapping without the clutter of other melodies playing.

*Live loop mode:* In live loop mode, the user builds melodies upon each other to create a complex song. The user makes a gesture, which immediately creates a visual response on the screen. This does not disappear, because the notes the visual corresponds to repeat themselves every iteration of the loop. The y-axis represents pitch and the x-axis represents time, analogous to a musical staff. The loop is repeated in ten second intervals. All the notes in the system sound good together, so anything the user creates in this mode will be pleasing to hear. Therefore, no undo or erase is necessary. MusicAir encourages collaboration and having an erase function unnecessarily complicates the interaction process, given the purpose of the technology is for exploration and the available sounds were chosen so that nothing can sound displeasing.



**Figure 3.** An example of a line drawn with the swipe gesture.



**Figure 4.** Depiction of how a user's wave motions correspond visually on the screen.

In both modes, the y axis represents pitch, from low to high. These are the motions musicAir recognizes:

*Swipe:* (Figures 3, 4) A fluid motion of the hand or fingers results in a string of notes that correspond to the motion. Swipes are represented by a string of dots on the screen. Using one finger to make a swipe produces one line of notes, while using multiple fingers might produce several parallel melodies.

Swipe gestures use the x, y, and z data from the hand position to map to the visuals and sound. The y-coordinate corresponds to pitch. The higher the hand, the higher the note. The x-coordinate represents time in live loop mode, but otherwise just determines where on the screen the visual should be drawn. The z-coordinate of the gesture is how far into the field of view of the Leap Motion controller the user's hand is. The z-coordinate corresponds to how large the circle is to be drawn at the point of the swipe. If the user is only on the edge of the field, then perhaps they are timid about making music, only starting to explore the edges of the technology and the circles at these points are smaller. The notes played corresponding to these points are also softer. If the user is deep in the field, the circles are larger, meaning that the user is making bold strokes and is not shy about making sound. The notes played at these points are louder. This mapping also makes sense for a gesture itself. When the user's hand enters the field of the controller and begins the gesture, it is quieter. Then it is loudest in the middle of the gesture and quiets off again as the user's hand leaves. This automatically gives each phrase a dynamic shape.



**Figure 5.** Stars drawn for finger taps

*Finger tap:* (Figure 5) This produces one note at the desired pitch, slightly louder and more pronounced than the other notes. The visual associated with a finger tap

is a star-like image, more noticeable visually to correspond to the louder sound.

Motions not implemented, but intended for the future:

*Change of instruments:* Making the motion of playing a different instrument will change the type of sound played for the next gestures made. Instruments that could be recognized are violin, piano, horn. Each instrument will have lines drawn in a slightly different shade or shape to visually distinguish.

*Hand splash:* The user starts with a fist and quickly spreads fingers, ending with an open palm. This would result in a cymbal-like sound at whatever height the motion was performed at. The speed at which the user extends their fingers will correspond to the volume at which the sound is played. Some sort of splash would be drawn (similar to the finger tap, but larger and more star-like), the size of which would correspond to the speed of the gesture and the volume of the sound.

*Brisk right to left swipe* directly over the Leap Motion controller: This is for clearing the screen in live loop mode. The user is not likely to be so close to the controller and they are likely making gestures from left to right since that is the direction of time in this mode. The system will prompt the user to repeat the motion if they really do intend to clear the screen.

### **Implementation and Progress**

In addition to using Leap Motion technology, musicAir is implemented as a web application using HTML5 Canvas, JavaScript, and the Leap Motion SDK. Leap Motion provides frame data at a rate of 290 frames per second.

We decided to use a major pentatonic scale in D (D, E, F#, A, B) because all notes sound pleasing and interesting together. We have three octaves of these three notes and we distributed these fifteen notes across the Y-axis of Leap Motion's display range, so

that as the user traverses vertically, a string of notes will play. Three octaves seems reasonable after some experimentation. There are enough notes that the user notices a change in pitch with slight motions, but not so many that without moving the user would be playing multiple notes. Each note is generated by its own sound file that is played when a gesture occurs at a given height. If a gesture occurs at a given height multiple times in a row, the sound file is played repeatedly. This places the focus on melody, but it is one of the next steps to have notes of different values, allowing users to have sustained notes and create more interesting rhythms.

We assigned ranges of color to each section to represent pitch visually and consistently. Our team has successfully created a high fidelity visual prototype that creates sound and visuals for basic gestures. A finger tap outputs a single musical note and is represented by a star on the screen (Figure 5), creating a color that corresponds with its designated note section. Similarly, a finger wave outputs a stream of sound and is represented by a line of dynamic thickness and gradient colors that will trace the finger's motion through the different note sections (Figure 3). Currently, we are expanding on the functionality of musicAIR, by exploring additional gestures and its associated visual and musical counterparts, refining the visual display in terms of smoothness and saturation, and incorporating more musical notes that will allow users to produce more complex melodies. In the future, we also hope to further conduct user testing and evaluate users' interaction with musicAIR in the aforementioned environments so that we may continue to optimize the user experience with respect to all of musicAIR's purposes.

### **Critique**

MusicAir has not been formally tested among its target users, but we addressed many critiques from classmates in the process of design and

implementation. Comments revolved around the intuitiveness, the gestures it recognizes, and the visual representation.

We decided to use Leap Motion because of the availability of the technology and the ease of use. Leap Motion is slightly limiting in terms of the lack of physical space it can detect motion in and the complexity of the gestures it can recognize. While there is room for two people to make gestures at the same time, users will be competing for space and this reduces the collaboration we hoped musicAir would inspire. In addition, Leap Motion does not distinguish different hand inputs in a consistent way that would enable us to differentiate users. Leap Motion also only has built in recognition for a few gestures, so some of the more complex motions we want to incorporate will be hard to convey. Because the response of musicAir is immediate, the system has to respond appropriately to gestures as they happen, not after they happen. Thus, if a user were making a gesture to change the instrument, musicAir would interpret the beginning of the gesture as a swipe and would play music accordingly. Even though that was not the intent of the gesture, the system would not recognize the gesture as something other than a swipe until the gesture is complete. Thus, there will have to be some compromise between real time interaction and recognizing complex gestures. Further exploration and development of Leap Motion promises a bright future for musicAir and similar gesture based interfaces.

### **Acknowledgements**

We would like to thank Wellesley College and our many classmates for their support and feedback in the conceptualization and implementation of musicAir.

### **References**

- [1] Andersen, T.H., and Zhai, S. "Writing with music": Exploring the use of auditory feedback in gesture interfaces.
- [2] Bean, A., Siddiqi, S., Chowdhury, A., Whited, B., Shaer, O., and Jacob, R.J.K. Marble Track Audio Manipulator (MTAM): A Tangible User Interface for Audio Composition. In Proc. TEI 2008.
- [3] Hoadly, R. Form and function: examples of music interface design. In *Proc. BCS 2010*, 479-483.
- [4] Jordà, S., Geiger, G., Alonso, M., and Kaltenbrunner, M. The reactTable: Exploring the Synergy between Live Music Performance and Tabletop Tangible Interfaces. TEI 2007
- [5] Kang, L., Gu, T., and Gay, G., ACM,
- [6] Leap Motion. <https://www.leapmotion.com/>
- [7] WiiPaint. <http://blog.chloefan.com/tag/wiipaint/>