Rummager

Group 20

Dimitrios Aatos Ellinas

Arda Mutlu

Måns Nyman

Yan Yu

Supervisor

Roberto Bresin

DT2140 - Multimodal Interaction and Interfaces

Abstract

The creation and exploration of Digital Musical Instruments (DMIs) is still very much in its infancy when put in comparison to more traditional instruments which have been refined over centuries and even millennia in some cases. Still, a wide range of different DMIs exist today which often make use of sensors that only recently became available to the general public. Drawing inspiration from some existing DMIs, we created a multimodal DMI using a mobile phone as its main interface. The input is primarily based around the combination of multi-touch and movement while the primary output is in form of (musical) sounds but also makes use of vibrotactile feedback. The instrument was designed to be used in the context of live performance, enforcing skill-based interactions. While the implementation overall proved to be successful in terms of what we initially wanted to achieve, user testing revealed some flaws that could be improved upon in further iterations.

Introduction

One of the core functionalities of a mobile phone is to produce sound. The most basic of these sounds being to alert the user of an incoming call and also to transmit the words being spoken by the caller to the callee. Before even the introduction of colored displays, customisable ringtones were implemented in many different models and allowed users to create and share their own ringtones. Although the possibility of creating self-made ringtones might not be enough to classify a mobile phone as an instrument, it is one of the earlier examples of a step in that direction. The first attempt at turning a mobile device into an instrument was not done on a mobile phone, however. In 1998, the nanoloop was created for the original Nintendo Game Boy and was (and still is) a rather minimalistic electronic music program[1]. Given the hardware constraints of the Game Boy, the sounds that could be produced with the nanoloop was quite limited in comparison to today's standard.

In the years that followed the conception of nanoloop, mobile phones grew in both popularity and computational power. With the introduction of smartphones, different types of sensors started to become more and more common. This opened up for new, touch-less interactions.

Essentially, ever since the possibility of creating instruments of mobile devices became a reality, attempts have been made at doing just that. In this project we have done just that, using a smartphone as the primary source of input and created a DMI.

Background

As smartphones grew in computational power and availability, so too did the possibility and interest in creating DMI's. By making use of software such as *Max*[2] or *MobMuPlat*[3], it is easy even for someone without much musical or programming background to get started on a similar project. Because of this, a plethora of different DMI's exist today and we will not attempt to account for all of them in this paper. We will, however, describe a couple of examples of DMI's which contain some elements that we made use of in our project.

One example is the *Pen2Bow*[4] which mimics a violin by making use of a keyboard (to take notes) and a pen used on an iPad to simulate the bow movement.
Another example is called *The Glide*[5] which is played using two controllers equipped with accelerometers. This does not try to mimic another existing instrument but

---

[1] George Essl and Michael Rohs. 'Interactivity for Mobile Music-Making'. In *Organised Sound*, 2009.
[2] Cycling' 74. 'Max'. Cycling' 74. Accessed 11 January 2020. https://cycling74.com/products/max.
[3] Iglesia Intermedia. 'MobMuPlat'. Daniel Iglesia. Accessed 17 January 2021. http://www.danieliglesia.com/mobmuplat/.
[4] Dimos Gaidatzis. 'Pen2Bow'. App Store, 28 March 2018. https://apps.apple.com/de/app/pen2bow/id1358113198.
[5] Groover, Keith. 'The Glide'. The Glide. Accessed 17 January 2021. https://www.theglide.cc/.

rather attempts to create something original both in terms of the sound it produces and in how it is played.

Method

In order to design for live performance in the traditional sense, we had to look at things that characterize performance with acoustic instruments, regarding interaction. Considering that interaction for making music has diverged in multiple ways from its traditional form after computers opened up the space for designing interaction without physical constraints, we found the classification of DMI by Malloch *et al.* [6] particularly useful for realizing that our vague vision was actually skill-based interaction. A conceptual construct we used for deciding on interactions was metaphors as described by Wessel and Wright[7]. A main source, our design derived metaphors from was acoustic instruments, that might also be described as analogies. Other ways that we thought of and analysed our interaction design was the gesture typologies by Cadoz and Wanderly[8] and a particular model (or metaphor) that we implemented wa the excitation/damping model mentioned[9].

Technologies used

The main computation and sound output is done on a computer using *Cycling' 74's Max* with *CNMAT externals*. The mobile phone is used as an input device for touch and movement (using the accelerometer), and as an output device for vibration and a visual interface. Input, output and communication with Max is done using *TouchOSC* by *Hexler* on the phone that uses the *Open Sound Control* communication protocol.

Sounds

The instrument produces two kinds of sounds: tones generated from triangular waveshape oscillators and short percussive sound samples. In our tests we used samples from the *Akai XE8* drum machine.

---

[6] Malloch, Joseph, David Birnbaum, Elliot Sinyor, and Marcelo M Wanderley. 'Towards a New Conceptual Framework for Digital Musical Instruments'. In *Proceedings of the 9th International Conference on Digital Audio Effects*, 49–52, 2006.
[7] Wessel, David, and Matthew Wright. 'Problems and Prospects for Intimate Musical Control of Computers'. *Computer Music Journal* 26, no. 3 (September 2002): 11–22. https://doi.org/10.1162/014892602320582945.
[8] Cadoz, Claude, and Marcelo M. Wanderley. 'Gesture - Music'. In *Trends in Gestural Control of Music*, edited by Ircam-Centre Pompidou Marcelo Wanderley et Marc Battier, 2000. https://hal.archives-ouvertes.fr/hal-01105543.
[9] Hunt, Andy, and R. Kirk. 'Mapping Strategies for Musical Performance'. *Trends in Gestural Control of Music*, 2000.

Control signal processing

All input signals from the phone are resampled in audio rate and passed through low pass filters and exponential transfer functions. This is done both in order to adjust the feel of responsiveness of each mapping and to get rid of noise. Noise reduction is particularly important for calculating time derivatives (speed), since speed calculation magnifies noise.

Touch Interaction

Touch input on the phone's screen produces two signals reporting the touchpoints position in the horizontal and vertical dimensions of the screen. We decided to keep that representation and work with the two signals separately. The speed of the horizontal position (x) was mapped to loudness, and the vertical position (y) to the pitch of an oscillator with a range of 13 semitones (C4 - C5). However, making a linear mapping of a continuous control (y dimension) to pitch made it hard to achieve specific tunings. Inspired by Perrotin's and d'Alessandro's proposed solution for improving pitch accuracy on touchscreens[10], we implemented a similar solution by applying a sigmoid function on every semitone interval (function graph can be seen at Figure 1). In terms of interaction, this resulted in wider areas for playing tones in tune and shorter areas for gliding between tuned tones[11]. In addition, the interaction supports multi-touch with up to four fingers touching and interacting with the screen simultaneously.

Our touch interaction design uses a metaphor for each dimension, moving the finger horizontally acts as bowing on a string and the vertical movement acts roughly as a finger that presses a string, moving along it. In a more abstract formulation, the horizontal movement is a continuous excitation gesture, and the vertical movement is a modification gesture.

[10] Perrotin, Olivier, and d'Alessandro, Christophe. "Adaptive Mapping for Improved Pitch Accuracy on Touch User Interfaces". In Proceedings of the International Conference on New Interfaces for Musical Expression, 186–189.

[11] equal temperament was used

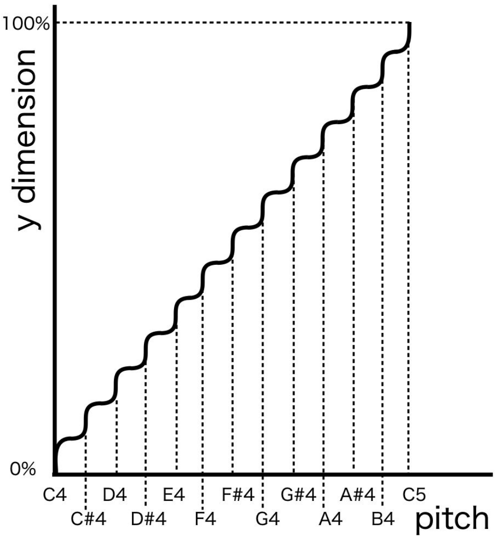

Figure 1.

Phone movement

The accelerometer of the phone produces three signals each representing the phone's acceleration on each of its spatial dimensions. In order to get a signal on sudden movements and get rid of the acceleration of gravity we calculated and used the jerk (acceleration's rate of change) on each access for most mappings. Phone movement was used in three main ways: playing three note chords, playing percussive sounds and muting the sound.

Chords

The jerk of the phone on each dimension was mapped to a tone's loudness and we added support for MIDI input so we could change the notes from an external keyboard. Since the shaking interaction wasn't easy to combine in a controlled way with touching the screen we used a long release time of several seconds for the chord so it can be combined with sounds from other interactions. In this case sounds triggered in a way that resembles a maraca, but a the sound envelope, having a short attack and a long release resembled more a hit on a gong, consequently the gesture excited the sound in an instantaneous way.

Percussion

When set to a different mode, moving the phone produces percussive sounds. A threshold was used on the overall jerk of the phone for triggering randomly from a list of 30 or more sounds. The interaction metaphor is approximately the same as in the chord mode, however the gong analogy doesn't apply since the sounds are much sorter.

Muting

Since the movement produced sounds that lasted for some time after their triggering, a muting function was important to implement. We chose to do that with the phone's orientation in relation to the ground. From vertical to down facing orientation the overall loudness of the sound interpolates from maximum to zero, respectively, making the facing down orientation a total muting position. Muting in combination with chord triggering can be thought of as an excitation/damping model.

Visual interface

The visual interface (seen in Figure 2) was made for indicating the position of notes with the area of each note shaped as a button, gliding and touching between "buttons" is possible.

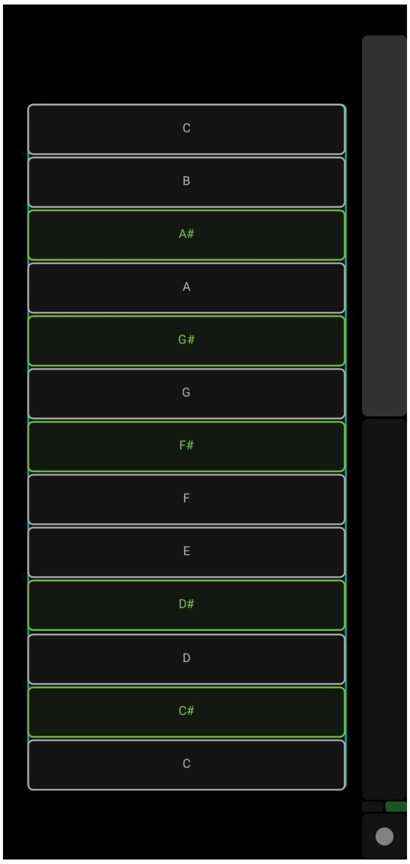

Figure 2.

Multimodality

Figure 3 illustrates how the DMI design combines modalities. The two main input methods, shaking (movement) and touching are not integrated and could theoretically be used in parallel. However, shaking the whole phone makes touching in a precise way difficult, we thus assume that the two modalities will be used mostly sequentially. That is the reason we chose to map shaking (in one of its modes) to a sound with a long release time. This way we achieved the outputs of the two

modalities to be parallel, since the shaking sound is going to last long enough for touching sounds to be played on top of it.
While developing the percussion triggering mode of shaking we experienced a difficulty in comprehending empirically the exact way of how sounds were triggered. Considering the literature that argues for the benefits of combining sound with vibrotactile feedback[12] [13], we did a redundant integration between the two modalities in hopes of increasing robustness.
Playing pitched sounds by shaking required a way for choosing the pitch of these sounds. In order to retain the simplicity of the shaking interaction, we introduced an external (musical) keyboard for that task, integrating its input with the shaking of the phone.

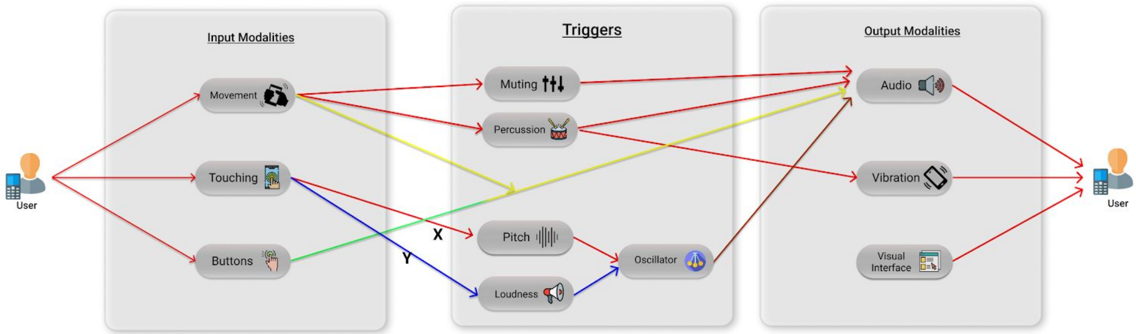

Figure 3.

Evaluation

We evaluated our project by testing it with 5 experienced musicians. After setting up the system, testers were informed on how rummager works, and had more than a day to experiment and try to play music with it. A semi-structured interview was conducted after testing, where testers pointed out issues and benefits of the design.

Results

After experimenting with the system testers expressed their thoughts and opinions, the main points being the following:

- The interaction felt natural, resembling acoustic instruments.
- Compared to keyboards (the usual interface for electronic music), being able to control loudness continuously gave much greater expressive control over the sound.
- High controllability of vibrato.
- Touching the phone in a desired way while shaking it wasn't possible, which caused annoyance.

[12] Chafe, Chris, and Sile O'Modhrain. 'Musical Muscle Memory and the Haptic Display of Performance Nuance'. In *Proceedings of the International Computer Music Conference*, 1996.
[13] O'Modhrain, M. Sile. 'Playing by Feel: Incorporating Haptic Feedback into Computer-Based Musical Instruments'. PhD Thesis, Stanford University, 2000.

- Percussive sounds sometimes triggered accidentally and didn't always trigger when desired, to a degree that it restricted expressiveness.
- The range of the produced sounds was narrow and limiting.

A presentation video can be found at https://vimeo.com/499669190.

Discussion

Overall, the end product has fulfilled our intention of designing an interactive digital instrument. The system successfully connects gesture controls to sound synthesis and sample triggering. With the current mappings, the performer is able to control multiple parameters of the sounds with simple gestures such as touching and moving the phone. Having granularity of control on the parameters allows the performer to adjust the sounds with precision, which could expand the possible range of sounds and textures, making the performance more expressive. In addition, there is no predetermined envelope for the synthesized tones, and the player can glide between notes on the touchscreen with no predefined glide time. This allows the player more freedom to manipulate the sounds to their liking.

On the other hand, there are several limitations that we've found in the system. From user feedback, there are complaints about the limited range of sounds, and the difficulty to play a whole piece using the system. It is true that we haven't explored a wide range of sounds to use other than the synthesized tones. Therefore, the system is not very versatile in playing different styles of music. And the current setup might not be optimized for playing a whole piece on its own, because there are certain conflicts between the different interactions. Firstly, the percussion feature is not very well-integrated in the system as of now, triggering sounds doesn't seem to be clear enough despite the use of redundant integration of sound and vibration. We presume that drum samples were expected to be triggered with a mallet-hit-like motion, and an interaction resembling a maraca—being similar but not the same—causes confusion. There is the lack of volume control, and the samples are not fine-tuned to work together with the other sounds. Besides, the gestures of shaking and touching the phone are difficult to carry out at the same time. As a result, it is challenging to maintain enough subtleties in the control while using these interactions simultaneously. For further work on the project, we hope to keep revising the design, explore different types of sounds, experiment with more complex gesture-to-sound mappings (beyond one-to-one), and implement better mechanics to combine the different interactions in real-time.