

An Arduino-based instrument for more intuitive expression of electronic music

Tina Wang
Australian National University
Canberra, Australia
tina.wang@anu.edu.au

ABSTRACT

The abstract should preferably be between 100 and 200 words, a word count that is not too short and not too long. That means that the abstract contains the most important information, so that readers can evaluate whether they are going to read the rest of the paper.

Author Keywords

NIME, human-computer interaction, electronic music, gesture

CCS Concepts (Alter this at the end; seems to want LaTeX)

• **Applied computing** → **Sound and music computing**; Performing arts; • **Information systems** → *Music retrieval*;

It is strongly recommended that authors view the submission form prior to starting to write the paper, which includes information on the CCS Concepts.

The 2012 ACM Computing Classification System (CCS) replaces the traditional 1998 version, which has served as the de facto standard classification system for the computing field. It is being integrated into the search capabilities and visual topic displays of the ACM Digital Library. To list your CCS concepts correctly, please use the following procedure, which is demonstrated using three NIME-related example terms: *Applied computing-Sound and music computing*, *Applied computing-Performing arts*, and *Information systems-Music retrieval*.

1) Browse to the website http://dl.acm.org/ccs_flat.cfm.

2) Select one to three classification terms from the website that describe your paper (e.g., for the example paper *Applied computing-Sound and music computing*, *Applied computing-Performing arts*, and *Information systems-Music retrieval*).

3) For each classification you need to select the relevance (e.g., for this example, *Sound and music computing* is "high", *Performing arts* is "low", and *Music retrieval* is "Medium")

4) After selecting the relevance, click "continue" if you have more terms to add. Otherwise, if you have added all your terms, complete the steps below.

5) Copy the displayed text at the bottom of the pop-up window, and paste it under the CCS Concepts heading above. You need to make sure that bold/italic formatting is maintained. You should also make sure that the arrow characters display correctly. Use the arrow characters in the example above if needed.

6) Click on "view CCS TeX Code". This will generate some code, which includes some CCSXML and some lines beginning with `\ccsdesc`. Keep all of this code, as you will need it for entering into the Precision Conference System paper submission form.

1. INTRODUCTION

In this paper, we propose an electronic musical instrument that aims to be both accessible and intuitive.

Electronic music has experienced enormous growth in the past decade and has been around for much longer. Few can deny its impact

in the music world across recent years. This is apparent through increased prevalence of electronic sounds in popular music, newfound popularity of launchpads and music production software, and performances using exclusively electronic instruments such as the theremin, just to name a few.

While perhaps not extraordinarily difficult, most electronic music production or editing technology is far from intuitive. This involves DJ sets with complicated setups of knobs, dials and sliders, and music production software with thousands of sounds and ways to combine them. Overall, these devices and programs are generally considered difficult to learn, and with good reason – their difficulty results from the amount of control and potential that the musician or composer has. We aim to create a physical musical instrument that is accessible and (at least comparatively) easy to learn (although perhaps difficult to master). Intuitive use for the player is the main goal, even if this involves relinquishing some degree of control in controlling exactly what sounds can be outputted.

The target audience is people who have a musical background, but not necessarily a technical background. The instrument will ideally be easy to pick up for anybody with a musical background.

[Add research on benefits of contactless musical instruments for intuition, and similar]

Gesture-based instruments have been found to increase both musician and audience engagement. Pelah and Greenlough [1] have found that players, even without technical experience, adapt to gesture-based instruments as quickly as they do to physical contact instruments, which is currently more common.

We also aim to incorporate continuous sound output (as opposed to discrete sound outputs e.g. a piano), as this is more likely to engage the audience [2]. [The Arcontinuo is an example of this, an instrument made up of a continuous 3D surface with enormous flexibility in terms of triggering different actions, allowing the player to take control and use it to their own preference [2].

My contributions to this topic (be specific):

TBD

2. PROTOTYPES AND PREVIOUS VERSIONS

2.1 STM32L476G-DISCO board

The STM32L476G-DISCO was originally considered as the centerpiece of the instrument. It appeared to be highly promising with its high processing capabilities and memory size, as well as its extensive sensor technology including a joystick, 3-axis gyroscope and 6-axis compass. It also featured audio-specific components including a built-in microphone and audio jack. We had previously been able to produce notes and melodies on it, and were interested in exploring more of its musical potential.

However, the STM32L476G-DISCO was required to be programmed in low-level languages such as C or assembly. We wanted to work with a more accessible board to stay closer to our goal, and we decided Arduino was the right fit. Arduino is beginner-friendly, cheaper than the STM32L476G-DISCO and open-source.

Commented [TW1]: Should this go in the Related Work section instead?

Commented [TW2]: Is it necessary to add a photo?

Commented [TW3]: Is it necessary to add a photo?

Commented [TW4]: These are the COMP2300 projects – should I add more detail?

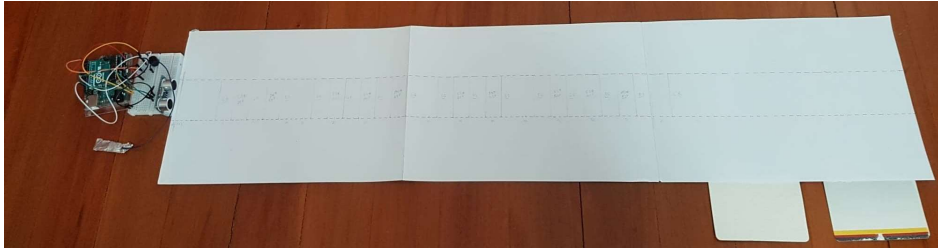


Figure 2: Arduino Proximity Sensing Piano Top View

Arduino also has a thriving community, thanks to its open-source roots, and was compatible with all types of different sensors. In comparison, the STM32L476G-DISCO has currently only been commercially available for a few years and has not cemented a visible community. We believe that the wide range of Arduino options should be able to make up for the sensor technology of the STM32L476G-DISCO, and that the trade-off between processing speed/memory and accessibility was worth it.

2.2 Arduino Mini-Piano

Getting comfortable with Arduino

2.3 Arduino Proximity + Capacitive Sensing Prototype

Figure 1 (above) shows the top view while Figure 2 (right) shows the side view. This Arduino prototype features an Arduino “piano” made using an ultrasonic sensor and capacitive sensor. Note markings were written on a piece of paper as a guide (realistically, any other material or even no material could have been used, since it had no direct effect on sound output). We created both a discrete and continuous version, with both using the same setup depicted.

An ultrasonic sensor can detect proximity in centimetres by emitting a high-frequency pulse, measuring distance using the amount of time it takes to be deflected back [REFERENCE?]. Distance ranges of 2cm each (e.g. 4-6cm away) were programmed to correspond to musical notes using their frequency values. Generally, these frequency values were large, producing high-pitched notes. In the discrete version, every note in the musical range C4 – C6 was covered. These sounds could be activated by placing any object at the specified distances away from the sensor. Ideally, this would be a wave of the hand, but it works best using a flat surface such as the cards on the bottom right of Figure 1.

The continuous version also produced sounds between the musical range C4 – C6, with the paper note markings used as an even rougher guide. However, the sound outputted was found to be extremely unstable and lacked smooth transitions. This may be due to the nature of the ultrasonic sensor – they have precision to a certain extent (roughly to the centimeter) but are prone to random error from interference in the air with increased precision. This would explain why the discrete version (with each note taking up 2cm) produced more stable sounds than the continuous.

Both versions also feature a “drum” made from foil using a capacitive sensor. Tapping it would trigger a low-frequency sound, which is quite distinct compared to the high-pitched notes. Users could create their unique drum beats by simply tapping the foil.

The sound produced from this prototype is quite primitive, as we used a small Piezo buzzer. This buzzer is satisfactory since we aim to make the instrument accessible; however, we aim to produce more sophisticated sounds.

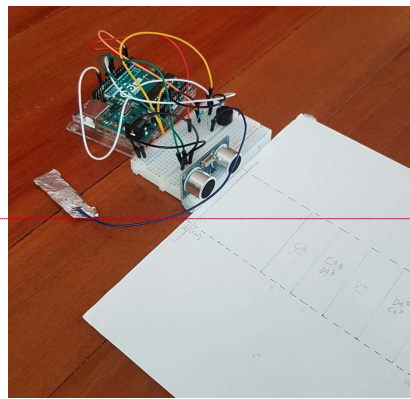


Figure 1: Arduino Proximity Sensing Piano Side View

This instrument changes the sound based on the location of the gesture, however, is still unable to read finer movements to consistently distinguish natural human gestures. In that sense, this stage is a step forward for intuitive play (moving further away from the sensor increases the pitch; tapping a piece of foil creates a beat) but does not yet incorporate gestures. It is also not fully contactless due to the capacitive sensor, however, tapping something non-electronic arguably feels more natural than a regular button.

2.4 Live coding integration?

After building the previous prototype, we considered if we could modify it such that – rather than being a standalone instrument, could it be used to alter music already in playback using similar gesture controls?

TBD – interaction with e.g. Sonic Pi

3. EVALUATION

3.1 User Evaluation

The musical instrument underwent a user evaluation to gauge how well it performs for intuition, accessibility and expressivity. We aimed to maximise all three, although it was understood that some trade-off may be necessary.

Since our target audience generally consists of anybody with at least basic musical proficiency, without any technical requirement, we chose participants who had at least 1 year of musical background.

We presented the instrument and gave instructions on how to play it, including a demo. The participant was then given time to experiment freely their own (10-15 minutes). Reactions and comments were audio-recorded, and they were asked a series of questions in an interview-style setting.

Commented [TW5]: Unsure if this should even be included in the report

Commented [TW6]: Written in past tense, but of course this is still yet to be conducted

[The following sample of questions were used, making use of both the Likert scale (e.g. "Please answer on a scale from 1 to 5 and explain your reasoning") and open-ended responses:

1. How engaged did you feel when playing this instrument?
2. How hard do you think this instrument is to play or learn? (This question may be asked again in relation to existing instruments, both electronic and non-electronic)
3. Do you think this instrument can engage members of the audience? (Perhaps also in relation to other instruments)
4. How much did you enjoy interacting with the instrument?
5. How frustrated did you feel while interacting with the instrument?
6. Were you able to create a wide variety of sounds?
7. Could you/did you manage to discover any techniques while playing?

3.2 Results

TBD

3.3 Discussion

TBD

4. RELATED WORK

TBD

5. FUTURE WORK

Although the focus of this project was on Arduino, the STM32L476G-DISCO still has much potential for intuitive musical interaction. Our Arduino instrument could even be enhanced (fine-

tune this part after instrument has been finalised) if the STM32L476G-DISCO was used. Instead of the instrument being a static unit measuring human gestures, perhaps the instrument itself could be incorporated into gestural sensing – for example, the 3-axis gyroscope could be used to change pitch as the board is tilted. We could take advantage of STM32L476G-DISCO's built-in features to create an equally intuitive instrument without requiring external sensors and parts.

Another future direction could be to scout out other accessible boards that were able to, more or less, maintain the same level of functionality without requiring external parts. In other words, find a board with optimal trade-off between accessibility and functionality for the musical instrument.

6. ACKNOWLEDGMENTS

TBD

7. REFERENCES

- [1] A. Pelah, and P. Greenhalgh. Sensory Feedback and Sensorimotor Adaptation in Human-Computer Interface for a Gesture-Based Contactless Musical Instrument. *i-Perception*, 3(4):253-253, 2012.
- [2] R.F. Cádiz, and A. Sylleros. Arcontinuo: the instrument of change. *NIME 18 Proceedings*, Copenhagen, Denmark, 2017.

8. Appendices may follow the references

TBD?

Commented [TW7]: Will be altered, these are purely indicative at the moment