

Enhancing Violin Performance through Real-Time Interaction: Design and Evaluation of a Wireless Audio-Visual Interface

Anna Savery

University of Technology Sydney

saverryanna@gmail.com

Submitted: 2023-07-13

Published: 2023-10-09

Abstract: This paper presents a detailed description of the development of a wireless audio-visual interface that is integrated with a violin bow. This interface is a work-in-progress prototype resulting from a practice-based research project into optimal physical design and software mappings for the virtuoso string player. As an augmented instrument, this interface aims at extending the existing skill set of the performer. Moving away from gestural analysis approaches, we focus more directly on optimising real-time performer interaction and control for violin performance, using bow hand fingers to manipulate the interface. A further novel aspect is the equal emphasis placed on both the audio and visual real-time processing which unlocks potential for new compositional possibilities and interdisciplinary performance situations. The interface is intentionally minimally invasive and wireless so as to reduce performer movement restrictions as well as permanent modifications to the violin bow. We contextualize our research process and outcomes within two performance situations, basing future design possibilities on the evaluation of an expert violinist.

1 Introduction

Acoustic bowed string instruments have been enhanced with a variety of technologies as a means of gestural analysis and new expressive possibilities by many researchers (Bevilacqua et al., 2006; Kimura et al., 2012; Machover, 1989; Overholt, 2014; Rose, n.d.; Young, 2002). Primarily, these technologies have consisted of various sensors analysing the string player's physical idiosyncrasies as well as the inherent physics of the instrument itself (Miranda & Wanderley, 2006; Overholt, 2009). The chosen sensors are integrated with the body or the bow of the instrument and in some cases, the performing artist (Reid et al., 2018; Van Der Linden et al., 2010). The collected sensor data is mapped within an appropriate musical software environment aiming to create a novel, meaningful and expressive creative output. Although multiple and varied approaches to augmenting the acoustic violin have been developed, our prototype strives to explore the space between real-time normal player gestural analysis and new digital instrument design, emphasising interactivity and minimal invasion.

In this paper we discuss the development of a new wireless audio-visual interface with design considerations that focus on optimal sensor placement for real-time performer right hand manipulation and gestural analysis. We discuss our approaches to sensor choices as well as detailing our software design parameters which place equal importance on both audio and visual software mappings. We present our work-in-progress interface within the context of two distinctly varied performances, reflecting on each situation from the first author's researcher/practitioner perspective and concluding with future creative design and output directions.

Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). Copyright remains with the author(s).

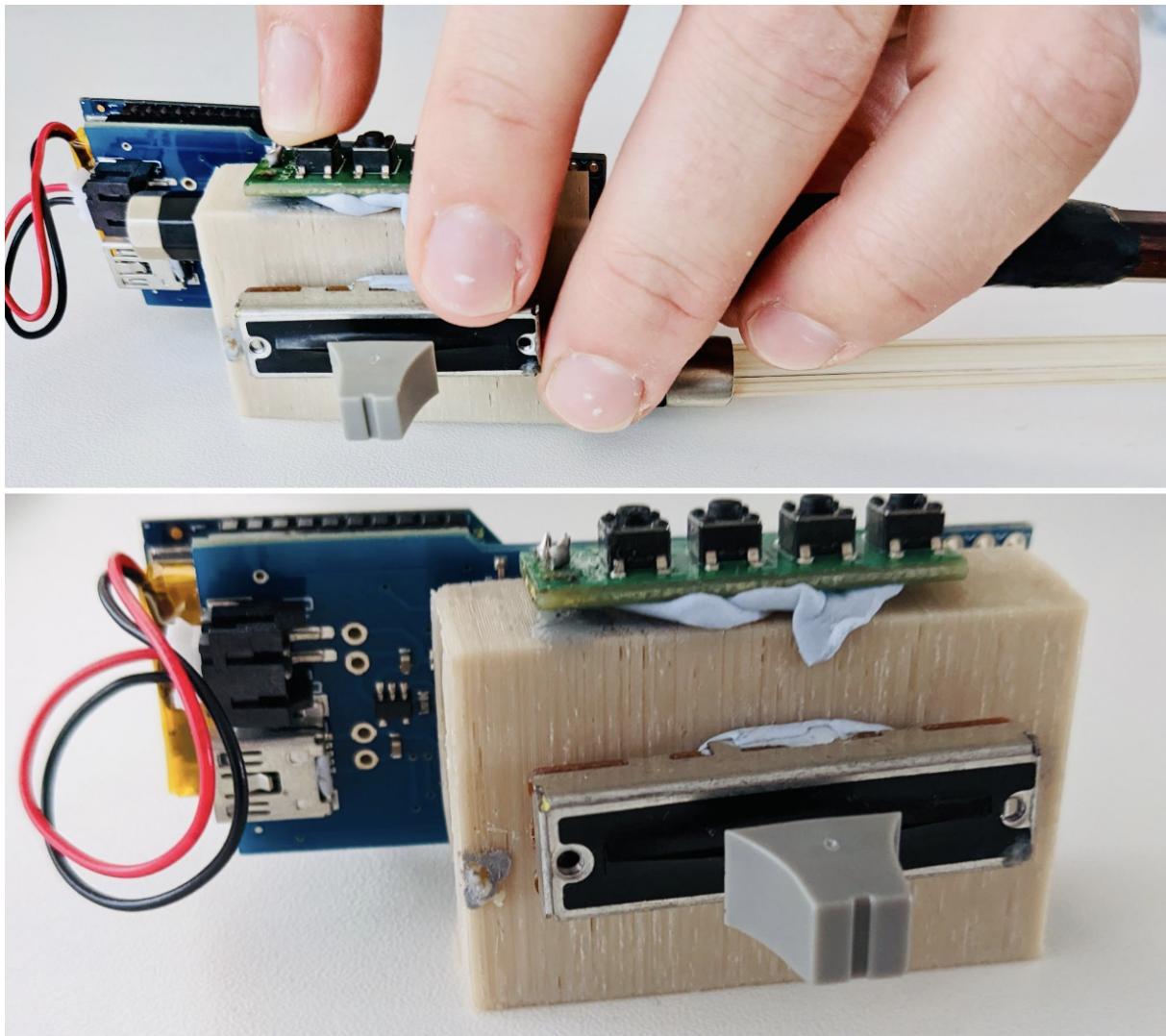


Figure 1: Wireless Audio-Visual Violin Bow Interface

2 RELATED WORK

Miranda and Wanderley categorise different types of musical interfaces, with augmented, hyper or hybrid instruments being those that extend the possibilities of an acoustic instrument through the addition of multiple sensors, whilst retaining the instruments original sound and performance gestures (Miranda & Wanderley, 2006). Significant developments include Diana Young's *Hyperbow Controller* (Young, 2002, 2008), Machover and Chung's *Hypercello* (Levenson, 1994; Paradiso & Gershenfeld, 1997), *vBow* developed by Nichols (Nichols II, 2003) and Bevilacqua's *Augmented Violin* (Bevilacqua et al., 2006). With the sensors integrated directly into the acoustic instrument, these new interfaces are designed for detailed analysis of intricate real-time playing gestures (Bevilacqua et al., 2006), augmenting the possibilities of the instrument, whilst preserving the performer's natural relationship with the instrument.

Building on these concepts but with the design consideration of minimal invasion to the acoustic instrument are Kimura's *Augmented Violin Glove* (Kimura et al., 2012) and Reid's *MIGSI* (Reid et al., 2016) which hold the sensor technology inside a custom designed housing that is made to be both easily integrated and removed. Bevilacqua discusses another approach to augmenting an acoustic instrument which incorporates interactive sensors such as FSRs, push buttons and sliders (Bevilacqua et al., 2006).

Although both approaches aim at enhancing and expanding the possibility of an acoustic instrument (Machover, 1994), the latter adds new gestures for the performer (Bevilacqua et al., 2006; Overholt, 2014). Ko and Oehlberg's *TRAVIS II* (Ko & Oehlberg, 2020), allows the performer to trigger a selection of audio and visual presets via four FSRs. The interface technology, designed to augment the fingerboard of the violin, with the exception of the custom made sensing fingerboard itself, by using 3D printed clamps allows for complete removal of all other sensor technology from the body of the violin. Overholt's *The Overtone Fiddle* (Overholt, 2011) combines the retention of traditional technique as well as introducing new extended performance gestures. Though the Overtone Fiddle allows performer control through traditional violin technique, it is an entirely new custom designed instrument with extensive augmentation capabilities. Despite the variations in physical design considerations, these interfaces all communicate sensor data to appropriate musical software environments which is then used to transform the traditional acoustic capabilities of the instrument into new musical outcomes (Machover, 1989). These two design components are therefore of equal significance within the development of a new musical interface. In the case of concurrent dynamic audio and visual software mappings, various approaches have been explored, including Ali Momeni and Cyrille Henry's introduction of a *Dynamic Independent Visual Mapping Layer* (Momeni & Henry, 2006), Jaroslaw Kapuscinski's *Intermedia* (Kapuscinski, 1997; A. Savery, 2016) Bert Bongers and Yolande Harris' *Video-Organ* (Bongers & Harris, 2002) as well as the more recent *AirSticks* (Ilsar, 2018) by Alon Ilsar with real-time interactive visuals designed by Andrew Bluff, Matthew Hughes and others. Our work-in-progress ambitiously aims at incorporating elements from all of these design approaches into one small yet robust musical interface.

3 Design Motivation

Coming from a background in classical and jazz violin as well as composition and music technology, the first author spent a considerable amount of time experimenting with various hardware technologies and software environments. These included electric instruments, multi effects pedals and various pickups. Throughout their academic and professional performing arts career, the first author shifted towards an intermedia creative practice, which combined visuals and audio and branching away from commercially available hardware augmentations. The desire to create an audio-visual, wireless interface that would fit on the frog of the violin bow came from wanting to move away from the visual aesthetic of long cables, bulky pedal rigs and other visually and physically restrictive technologies.

The goal was to create a custom built controller with real-time audio and visual processing capabilities that could be manipulated with just the right hand fingers of an expert violinist, creating a minimally invasive, cost effective, mobile version which effectively combined already existing tech-

nologies.

This design would also allow for easier interdisciplinary collaborations, where the violinist could effortlessly move around the stage and interact with dancers, actors or any digital set component such as a projection surface or lighting design. As a small and mobile device, the interface would retain a sense of mystery and magic within a live performance, where the audience would not be distracted by the interface and the process of audio and visual manipulation. An aesthetic that is otherwise very present when using an electric instrument or a foot pedal. As a custom built controller, the interface would be designed around the physical and technical capabilities of the violinist, with optimal sensor placement and a software mapping system.

4 Implementation

We began to design the physical interface by testing a variety of different sensors with the Arduino and Max/MSP software. Short snippets of live violin audio were recorded and then manipulated by the sensors which were either attached to a breadboard or simply laid out on a desk. This physical and software prototyping process lasted about three months, involving a cyclical practice based research approach of practice, theory and evaluation (Candy, 2006; Candy & Edmonds, 2021), where an idea for an audio-visual work would be realised and tested, accumulating in a series of Max abstractions and an Arduino sketch that could then be used in a larger, more cohesive composition. As the music became predominantly loop based, we decided to shift to Max for Live, and control its parameters via the live object model¹.

4.1 Physical Design

The goal of creating this interface was to make a real-time controller capable of audio and visual processing through direct manipulation of right hand fingers. As a starting point of the design process, we began working with just two controller sensors. These were a potentiometer slider², and a four button keypad³. These were chosen for ease of both physical interaction and initial software mappings. In order to facilitate the wireless design consideration, we chose to work with the XBee ZB Zigbee wireless modules⁴, mounting the radio module directly onto an Arduino Fio board.

Another design consideration was making sure the interface was easily removable from the frog of the bow. This was important as the interface is intended for use by an expert practitioner and their professional instrument setup. This was enabled through the design of a series of 3D printed casings, that slid on and off the bow. Multiple iterations were needed, as the rigidity of the casings did not allow for slight variations in bow frog dimensions. The sensors were then mounted onto the chosen casing using various adhesives. As an initial physical prototyping model, this approach worked well, allowing for rapidly exploring different sensor placement positions in order to find an optimal one. However, in a live performance setting, these adhesives were not secure enough and both the casing and the sensors shifted around too much.

4.2 Software Design

After successful data communication between the sensors, Arduino and Max software was established, we focused on deciding how to best approach software mappings in order to achieve robust and creatively meaningful results. We decided to approach this problem using a practice led trajectory, where a live audio-visual work for solo violin and the interface was used as a means to develop a series of criteria for software mappings and compositional approaches (Candy & Edmonds, 2021).

¹https://docs.cycling74.com/max8/vignettes/live_object_model

²Adafruit Slider Trinkey - USB NeoPixel Slide Potentiometer

³Vanki Arduino keypad 4 Button Key Module Switch Keyboard for UNO MEGA2560

⁴[https://www.digi.com/resources/examples-guides/basic-xbee-zb-zigbee-\(series-2\)-chat](https://www.digi.com/resources/examples-guides/basic-xbee-zb-zigbee-(series-2)-chat)

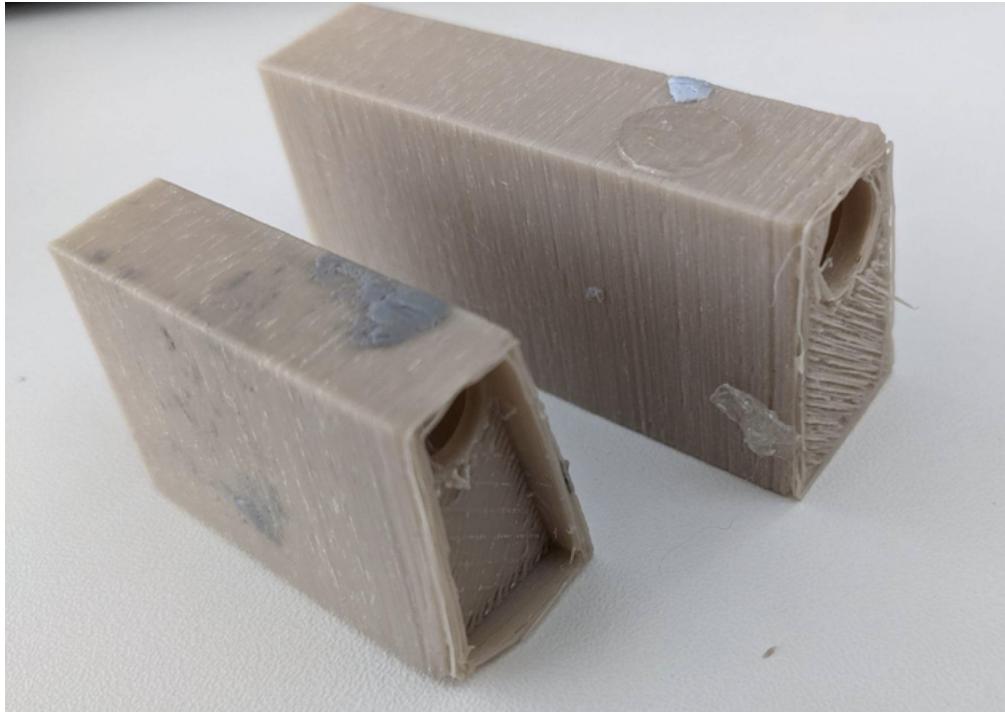


Figure 2: Examples of 3D printed casings

4.2.1 Audio Mapping

The four buttons functioned as toggles, with the data mapped to trigger on and off signals. This data was used to control various parameters in Ableton using the live.object and live.path objects. These included triggering clip recordings, looping on and off and deleting recorded clips in real-time. The potentiometer slider data was scaled in a variety of ways, depending on the current section of the composition.

At times, the data was scaled to receive floats between 0. and 100., where the range of those numbers would control sliders of in-built Max for Live objects, such as the Dry/Wet signals of the Insinkorator or Dual Harmonizer. In other instances, the slider data would be scaled to a range between 0 and 3, acting as a secondary series of buttons, with each integer linked to a direct parameter, such as opening and closing of a gate object or starting and stopping playback of a buffer.

4.2.2 Visual Mapping

All visual elements for this initial prototype were processed in Jitter and consisted of video footage. A variety of processing techniques were explored, such as cross-fading between multiple video streams and brightness and controlling contrast and saturation using the jit.brcosa object. These techniques were well suited to being controlled by the potentiometer slider with its analog data stream being easily scaled to range from 0. to 1.

The biggest mapping challenge was switching or moving between parameters, much like any transition within a piece. Attempting to control multiple parameters simultaneously proved to be somewhat chaotic and less meaningful. One solution was to outline distinct sections within each piece, marked by a counter object in Max. At specific moments throughout a piece, the counter would trigger an opening or closing of a gate object, that would transition from current section to the next. This approach eased the load off the two existing sensors and allowed for more elaborate mappings within each section. However, this also created a very rigid structure, leaving little room for spontaneity.

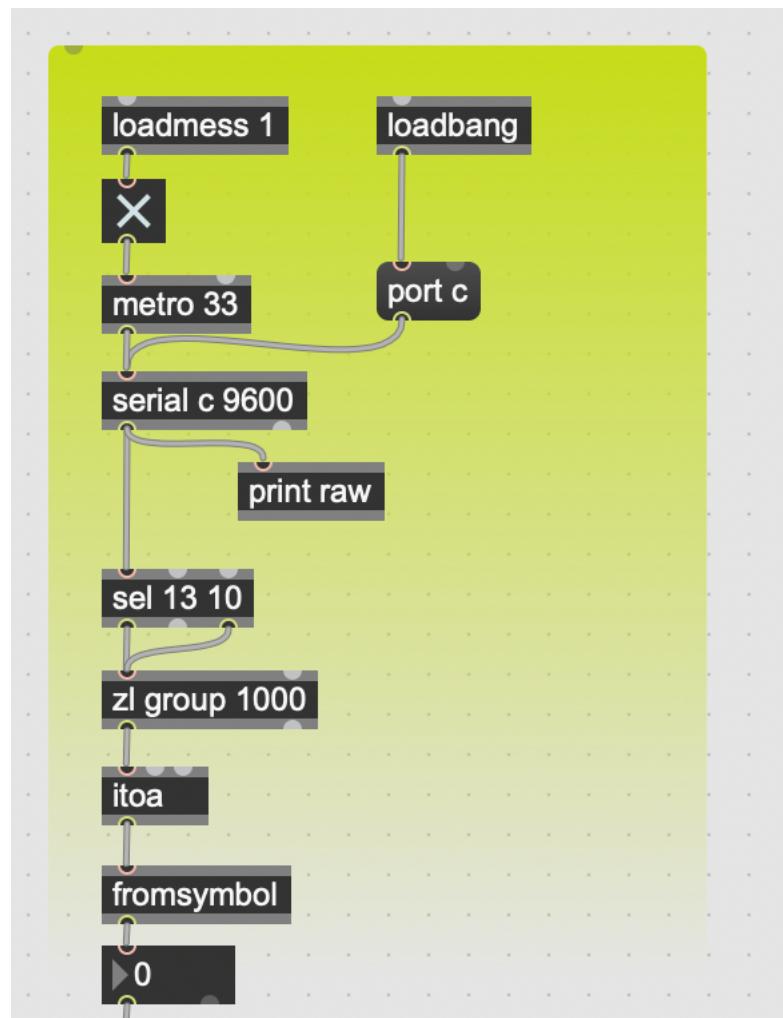


Figure 3: An abstraction for receiving sensor data from Arduino software

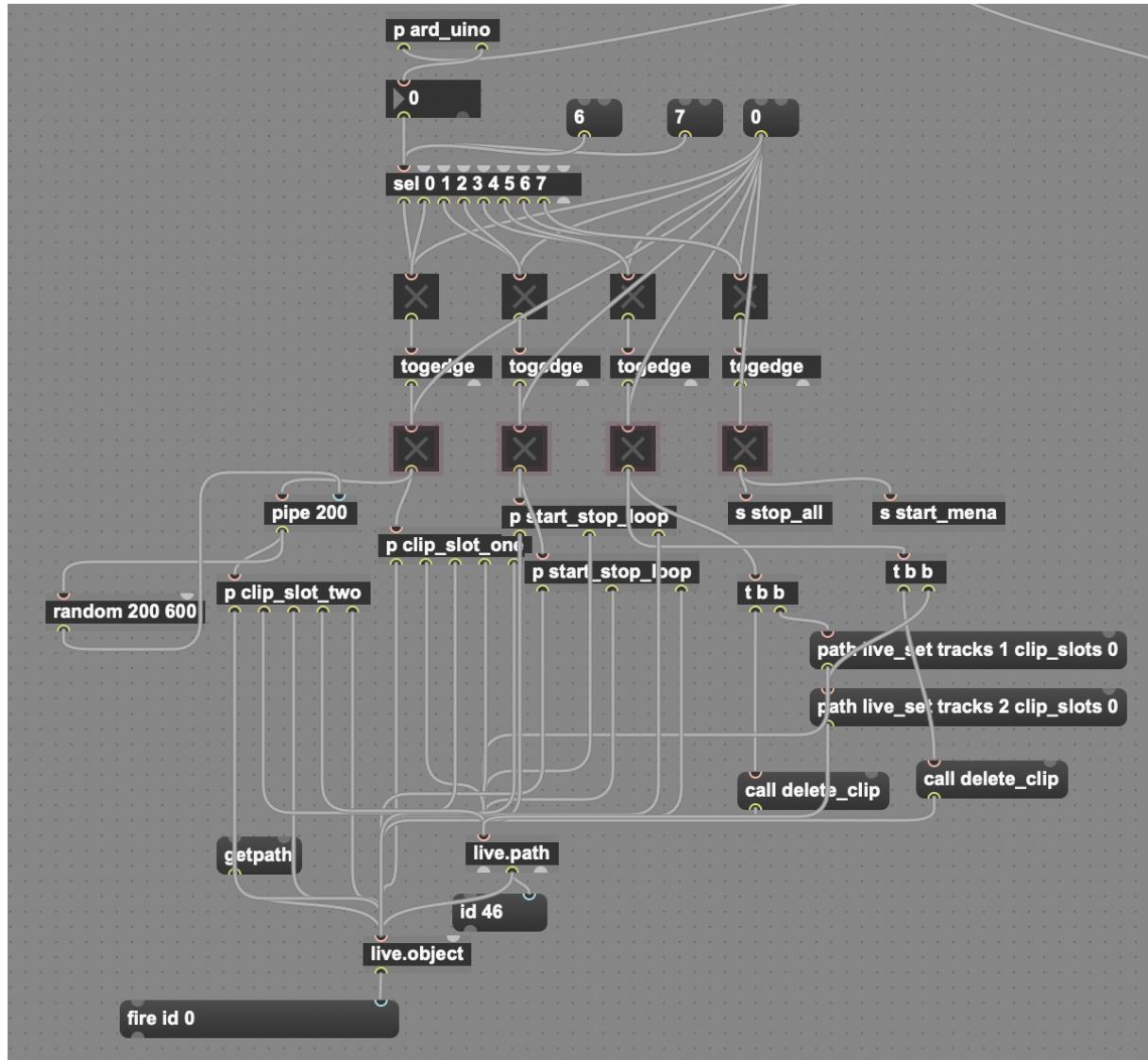


Figure 4: Max for Live patch with audio mappings

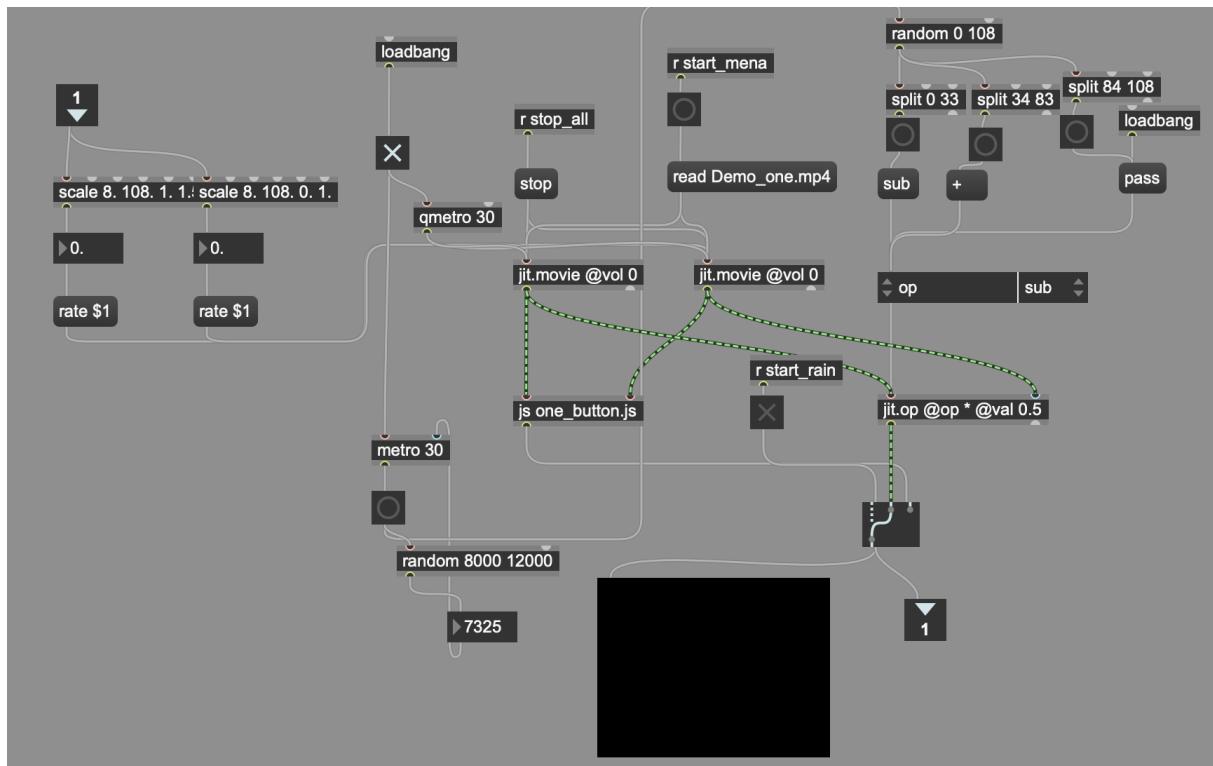


Figure 5: Max for Live patch with video mappings using Jitter

5 Creative Outputs

We developed and performed two works using the initial working prototype. The first performance situation the interface was used for was a demonstration video that was filmed in June of 2022. The piece was a combination of a free solo violin improvisation as well as some predetermined structural constraints and a prerecorded audio sample played on viola, which ultimately dictated the harmonic structure and mood of the work. The second performance situation was a cross university collaboration between University of Technology Sydney and Macquarie University. A robotic arm was used to visualise the first author's playing in real-time using paint on canvas. This performance situation took place over one day with multiple iterations of the piece and visualisations.

5.1 Demonstration Recording

In June of 2022A, we filmed a live recording of a demonstration video at the University of Technology Sydney . The composition for solo violin and interface was approximately four minutes in length, and consisted of prerecorded viola samples, live audio sampling and processing and live video processing. The composition was a culmination of the theoretical frameworks developed throughout the period of January 2022 up to June of that year. Musically, the composition was a combination of an improvisation and structured sections, dictated by a Max for Live patch. There were loosely three sections;

A free improvisation with live audio sampling and processing using the Instinktorator audio effect. The recording, playback and looping all triggered by push buttons on the interface. The video footage was initiated with the same push button used to process audio. Brightness and cross fading was controlled by the potentiometer slider.

The second section was triggered by a different push button, which triggered the playback of a prerecorded viola melody. Real-time improvised violin melodies were layered on top of this sample, with further layers created by short real-time sample playback. Each sample being between 10 and 30 seconds in length, with various effects, such as reverse playback, changed playback rate and various

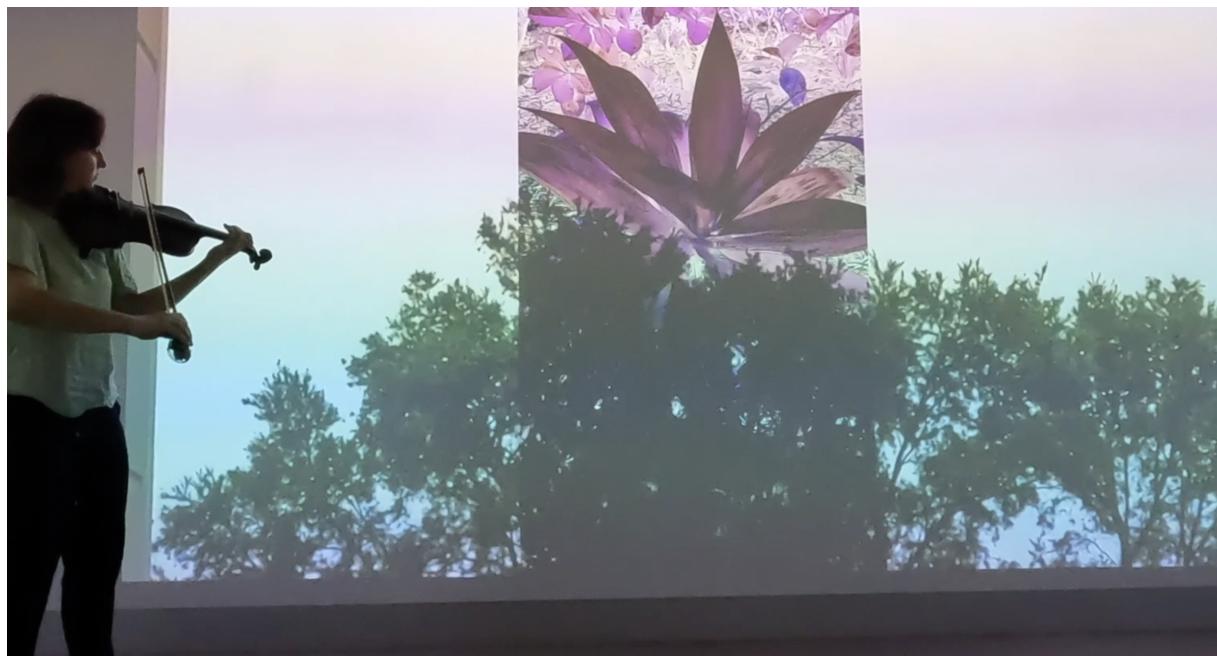


Figure 6: Demonstration video recording

harmonizers. For the video footage, this section switched to saturation, but cross fading remained.

The last section deleted the short audio samples one by one, thinning out the layers and creating a natural decrease in volume, descending into an ending. The video footage played out to the end and came to a natural pause as the patch stopped.

[Link to video here](#)

5.2 Reflection

With most of the design focus being centered on the physical implementation and software mappings, the musical development was unfortunately a secondary priority. This became very evident when the piece became predominantly a free improvisation, with harmonic and rhythmic elements consisting solely of the prerecorded viola sample. Manipulating the sensors with right hand fingers whilst playing also proved to be surprisingly challenging, partially due to the physical design of the interface, where the sensors were not optimally placed for efficient manipulation, but also due to the mental load of performing and incorporating new gestures into that performance required to manipulate the interface sensors.

5.3 Performance Visualisation

Similar to the demonstration video, this was mostly a free improvisation, with a structure revolving around a time constraint of approximately 8 minutes. This is approximately how long it took the robotic arm to fill the canvas using all four available paint colours. Audio effects were worked out to align with paint color changes and the corresponding layering and mixing of paint on canvas. Hence, the piece always began with solo violin and black paint.

When the robotic arm was ready to add a blue color to the canvas, a push button on the audio-visual interface was used to trigger an addition of the Max for Live dual harmonizer audio effect. The potentiometer slider was used alternate between two octaves below and two octaves above the real-time audio from the violin. This thickened out the sonic texture, aligning with the painted canvas.

From the onset of the piece, short live audio samples of approximately 10-20 seconds in length were being recorded. Each with a separate audio effect. Using a push button again, the dual harmonizer was gated off and the recorded audio samples would start looping one by one, thickening out the sonic



Figure 7: Rehearsing with the robot painting arm

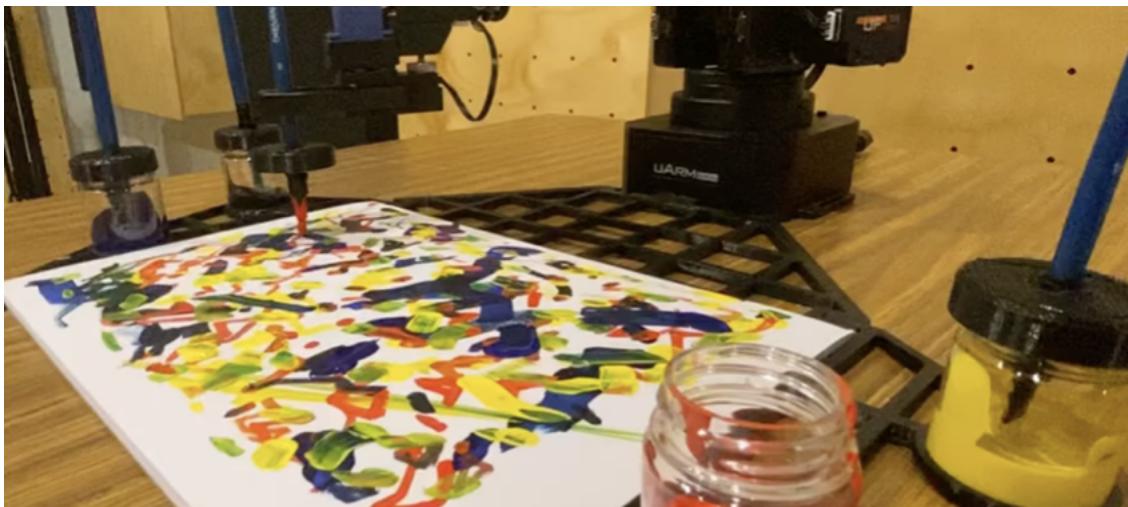


Figure 8: Completed canvas

texture more and more, as the robotic arm added yellow and then red. The piece finished on a large crescendo as the canvas filled with vibrant colors and a variety of short and long strokes (R. Savery et al., 2022).

[Link to video here](#)

5.4 Reflection

The absence of a visual component from the interface in this work allowed for a deeper investment into the audio mappings. Since this collaboration followed the demonstration video recording, many of the physical and virtual parameters were already flushed out and familiar. This allowed the first author to focus more on the musical aspects of the performance whilst becoming accustomed to responding to a robotic arm collaborator.

The physical manipulation of the interface became somewhat easier after re-evaluating the demonstration video recording and relocating the sensors to different parts of the casing. We moved the push button panel was moved slightly higher up the bow to stop the right hand pinkie from involuntarily

suppressing the outer most button during bow changes and lifts. The potentiometer slider was also moved slightly to the left, away from the bow screw to allow for better reach from the right hand middle finger, which is used to manipulate the sensor.

6 Conclusion and Future Work

The wireless audio-visual violin bow interface described in this paper has laid a strong foundation for the next iteration of this ongoing research project. This prototype has been a robust proof of concept, showing that a wireless sensor enhanced interface can be successfully manipulated in real-time by a violinist's right hand fingers within a variety live performance situations. The work-in-progress prototype has also successfully demonstrated that both audio and visuals can be concurrently developed and processed, opening up a wide potential for future creative output and interactive audio-visual system design.

A more recent version of the interface is using a silicone encasement for the sensors, which can slide on and off the bow frog. This more flexible material allows for discrepancies in bow size whilst retaining the sensors in place securely during a live performance situation. Two more sensors have been added to this version; a force sensor, that is fitted on the outside of the silicone casing bellow the bow frog for manipulation with the right hand thumb and a 3-axis acceleromter, mounted on the side of the Arduino Fio board. The latter, with representation of nuanced gestures through its data flow, will require a considerable amount of practice and performance experience to fully develop and master. Two Flora neopixel LEDs have also been added for visual feedback so as to avoid external help and create a more trusting relationship between the system and the performer (Kimura, 2003).

The design focus for future works is predominantly preoccupied with software mappings. The initial steps are aimed at developing a modular approach to both audio and visual mappings for each sensor, similar to approaches taken by Bonger and Harris in the development of the *Video-Organ* (Bongers & Harris, 2002). This will allow for spontaneity in live performance situations, easier collaborations with other artists and propel robust composition ideas. After this design process, we will focus on developing a more autonomous software system, where the audio and visual elements will have some component of agency and interaction, resembling the directions taken by some of the works incorporating Ilsar's *AirSticks* such as *H2.O* (Ilsar, 2018).

7 Acknowledgments

This research is supported by an Australian Government Research Training Program Scholarship.

References

- Bevilacqua, F., Rasamimanana, N. H., Fléty, E., Lemouton, S., & Baschet, F. (2006). The augmented violin project: Research, composition and performance report. *New Interfaces for Musical Expression*, 402–406.
- Bongers, B., & Harris, Y. (2002). A structured instrument design approach: The video-organ. *New Interfaces for Musical Expression*, 86–91.
- Candy, L. (2006). Practice based research: A guide. *CCS Report*, 1, 1–19.
- Candy, L., & Edmonds, E. (2021). Practice-based research. In C. Vear (Ed.), *The routledge international handbook of practice-based research*. Routledge.
- Ilsar, A. (2018). *The AirSticks: A new instrument for live electronic percussion within an ensemble* [PhD thesis].
- Kapuscinski, J. (1997). *Mudras*. University of California, San Diego.
- Kimura, M. (2003). Creative process and performance practice of interactive computer music: A performer's tale. *Organised Sound*, 8(3), 289–296.

- Kimura, M., Rasamimanana, N., Bevilacqua, F., Zamborlin, B., Schnell, N., & Fléty, E. (2012). Extracting human expression for interactive composition with the augmented violin. *New Interfaces for Musical Expression*.
- Ko, C., & Oehlberg, L. (2020). Touch responsive augmented violin interface system II: Integrating sensors into a 3d printed fingerboard. *New Interfaces for Musical Expression*.
- Levenson, T. (1994). Taming the hypercello. *The Sciences*, 34(4), 15–18.
- Machover, T. (1989). Hyperinstrument: Musically intelligent and interactive performance and creativity systems. *International Conference in Computer Music*, 186–190.
- Machover, T. (1994). Interactive instruments for experts and amateurs. *The Journal of the Acoustical Society of America*, 95(5), 2888–2888.
- Miranda, E. R., & Wanderley, M. M. (2006). *New digital musical instruments: Control and interaction beyond the keyboard* (Vol. 21). AR Editions, Inc.
- Momeni, A., & Henry, C. (2006). Dynamic independent mapping layers for concurrent control of audio and video synthesis. *Computer Music Journal*, 30(1), 49–66.
- Nichols II, C. S. (2003). *The vbow: An expressive musical controller haptic human-computer interface*. Stanford University.
- Overholt, D. (2009). The musical interface technology design space. *Organised Sound*, 14, 217–226.
- Overholt, D. (2011). The overtone fiddle: An actuated acoustic instrument. *New Interfaces for Musical Expression*.
- Overholt, D. (2014). Advancements in violin-related human-computer interaction. *International Journal of Arts and Technology* 2, 7, 185–206.
- Paradiso, J. A., & Gershenfeld, N. (1997). Musical applications of electric field sensing. *Computer Music Journal*, 21(2), 69–89.
- Reid, S., Gaston, R., Honigman, C., & Kapur, A. (2016). Minimally Invasive Gesture Sensing Interface (MIGSI) for Trumpet. *New Interfaces for Musical Expression*, 419–424.
- Reid, S., Sithi-Amnuai, S., & Kapur, A. (2018). Women who build things: Gestural controllers, augmented instruments, and musical mechatronics. *New Interfaces for Musical Expression*.
- Rose, J. (n.d.). *The interactive violin bow: its history, role, and potential in improvised music—a personal perspective*.
- Savery, A. (2016). *Intermedia storytelling* [Master's thesis]. UC Irvine.
- Savery, R., Savery, A., & Baird, J. (2022). Robotic arm generative painting through real-time analysis of music performance. *Proceedings of the 10th International Conference on Human-Agent Interaction*.
- Van Der Linden, J., Schoonderwaldt, E., Bird, J., & Johnson, R. (2010). Musicjacket—combining motion capture and vibrotactile feedback to teach violin bowing. *IEEE Transactions on Instrumentation and Measurement*, 60(1), 104–113.
- Young, D. (2002). The hyperbow controller: Real-time dynamics measurement of violin performance. *New Interfaces for Musical Expression*.
- Young, D. (2008). Classification of common violin bowing techniques using gesture data from a playable measurement system. *New Interfaces for Musical Expression*.