O Soli Mio: Exploring Millimeter Wave Radar for Musical Interaction

Francisco Bernardo
Goldsmiths University of London
Department of Computing
f.bernardo@gold.ac.uk

Nicholas Arner
Embodied Media Systems
nicholas.arner@embodiedmediasystems.com

Paul Batchelor
Stanford University
Center for Computer Research in Music and Acoustics
pbatch@ccrma.stanford.edu

ABSTRACT

This paper describes an exploratory study of the potential for musical interaction of Soli, a new radar-based sensing technology developed by Google's Advanced Technology and Projects Group (ATAP). We report on our hands-on experience and outcomes within the Soli Alpha Developers program. We present early experiments demonstrating the use of Soli for creativity in musical contexts. We discuss the tools, workflow, the affordances of the prototypes for music making, and the potential for design of future NIME projects that may integrate Soli.

Author Keywords

Millimeter wave radar, Rapid prototyping, Gestural interaction, New Musical Interfaces

ACM Classification

H.5.2 [Information Interfaces and Presentation, HCI] User Interfaces — Input devices and strategies, Interaction styles, Prototyping, Graphical user interfaces (GUI), H.5.5 [Information Interfaces and Presentation] Sound and Music Computing.

1. INTRODUCTION

Google ATAP's Project Soli [6] is a new radar-based gesture sensing technology that presents an unique potential for the development of new musical interfaces (NMI), digital musical instruments or musical controllers. Soli seems to provide a good solution space for creating musical interfaces controlled by fine-grained micro gestures at high speeds, combining very good portability, small form factor, high temporal resolution and data throughput, and several control dimensions based on digital signal processing (DSP) features and gesture recognition capabilities that are exposed by the Soli SDK.

To the best of our knowledge, no previous work exists that uses radar-based sensing for developing new musical interfaces. In this paper we first provide a brief overview of the technology, highlighting some features of interest for gestural control of music. We also review prior work that addresses fundamental aspects of sensing technology for musical interaction. We depict some initial experiments and early advances in building prototypes for musical interaction in the context of the Soli Alpha developers program and at an early stage of Soli development. We discuss the potential of Soli for musical interaction, based on the rapid prototyping workflow that we adopted and on our hands-on experience. We argue that Soli may be very well aligned with new use cases of wearable NMIs and that it provides interesting possibilities for creative exploration by the NIME community.

1.1 The Soli Alpha Developers program

The authors are participating on the Project Soli Alpha DevKit Early Access Program, which began in October 2015. They have been selected to be members of the Soli Alpha Developers, a closed community established by Google ATAP, and have received a Soli development board and software development kit (SDK). Soli Alpha developers contributed to the development of Soli by providing initial testing, bug reports, fixes, tips and tricks, and by using the Soli Alpha DevKit to build new concepts and applications with real-world use cases that leveraged on the characteristics of millimeter wave radar.

With the progressive maturation of the Soli SDK, Google ATAP challenged the Alpha Developers community to submit their applications for participation in the Soli Developers Workshop. Ten submissions, including the authors' work, were selected based on their novelty, technical level, polish and "wow" factor. They were invited to present their demos and attend a workshop in early March 2016, held by Google ATAP in Mountain View, California, where they delivered hands-on project demos of their work and received feedback from members of the Project Soli team and other alpha developers. The authors' projects were also selected to feature in a video that was showcased at the Google I/O developers' conference in May, 2016.

2. BACKGROUND

Research around gestural control and expressivity in musical contexts is broad, interdisciplinary and has a long history [8]. As new sensing technologies emerge, they can extend the possibilities for acquiring new aspects and nuances of human gesture for musical interaction. However, gestural interfaces, which are usually characterized as more 'natural' due their to richer and more direct use of the human sensorimotor skills, have fundamental problems that may undermine their use [14]. Some NMI and musical controllers adapt gestural vocabularies from pre-existing instruments, which leverage on previously developed motor control abilities [10]. For alternate controllers (i.e., not based on acoustic instruments or the keyboard paradigm), designers usually define idiosyncratic gestures, which poses strong limitations to their general adoption.

NIME'17, May 15-19, 2017, Aalborg University Copenhagen, Denmark.

Besides these fundamental problems, gestural interfaces can be limited by their constituent sensing technologies. Some of the previous design approaches in NIME incorporate RGB cameras [13] or more recent RGB+D cameras, such as Kinect [2] or Leap Motion [7], which leverage computer-vision techniques for real-time pose estimation. While these technologies provide reasonable accuracy, resolution and latency for supporting musical interaction, they also impose design constraints, such as a stationary interaction space, occlusion effects, specific environmental lighting conditions and direct line-of-sight with the sensors. Other approaches make use of wearable technologies, providing the performer with more freedom of action. They may assume different forms, such as data gloves [17], armbands [15] or personal area networks [5], which are worn or positioned in the performer's body, and combine different types of sensors [11] and data fusion techniques [12]. However, there have been reports of issues concerning maintainability and reliability [4], obtrusiveness, and strong co-occurrence of the same types of sensors and same simple instrumentation solutions [11].

2.1 Google ATAP, Project Soli

Soli's technology stack and design principles, covering radar fundamentals, system architecture, hardware abstraction layer, software processing pipeline for gesture recognition and interaction design principles are thoroughly reviewed in [9]. We build upon this work and provide a brief technical overview of Soli to highlight features that motivated us as Soli Alpha developers and that we consider of interest to NMI designers. Soli is a millimeter-wave radar sensor that has been miniaturized into a single solid-state chip. This form factor gives Soli several advantages in terms of design considerations, such as low-power consumption, ease of integration with additional hardware, and a potential low cost that may result from large scale manufacturing. Soli uses a single antenna that emits a radar beam of 150 degrees, which illuminates the whole hand with modulated pulses emitted at very high frequency, between 1-10 kHz. The nature of the signal offers high temporal resolution and throughput, the ability to work through certain materials (e.g., cloth, plastic, etc.) and to perform independently of environmental lighting conditions.

Soli's radar architecture has been designed and optimized for mobile gesture recognition and for tracking patterns of motion, range and velocity components of complex, dynamic, deforming hand shapes and fine motions at close range and high speeds. By leveraging on the characteristics of Soli's signal and processing pipeline, which prioritizes motion over spatial or static pose signatures, Soli's gesture space encompasses free-air, touch-less and micro gestures—subtle, fast, precise, unobtrusive and low-effort gestures, which mainly involve small muscle groups of the hand, wrist and fingers. Soli offers a gesture language that applies a metaphor of virtual tools to gestures (e.g., button, slider, dial, etc.), aiming for common enactment qualities such as familiarity to users, higher comfort, lesser fatigue, haptic feedback (by means of the hand acting on itself when making a virtual tool gesture) and sense of proprioception, (i.e. "...the sense of one's own body's relative position in space and parts of the body relative to one another" [9]. Radar affords spatial positioning within the range of the sensor to provide different zones of interaction, which can act as action multipliers for different mappings with the same gesture.

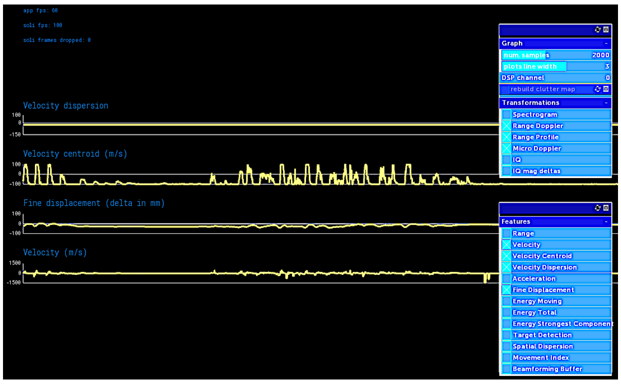

Figure 1: SoliDSPFeatures2OSC GUI

3. EXPLORING SOLI FOR MUSICAL INTERACTION

Our experiments explore how Soli can be used in creative audio-visual and musical contexts and include building a tool for rapid prototyping of creative applications, real-time parametric control of audio synthesis, interactive machine learning, and a mobile music application. The video documenting these explorations can be found in `https://youtu.be/WGlVzIlJvno`

3.1 SoliDSPFeatures2OSC wrapper

SoliDSPFeatures2OSC is an Open Sound Control (OSC) [21] wrapper with a graphical user interface (GUI), implemented as a standalone application using the Soli SDK library for C++ and the OpenFrameworks addon that accompanies it. This application can be used as a streamlined data source component of a modular, application-based pipeline for rapid prototyping and exploration of Soli DSP transforms and high level features. The features are sent via OSC messages to other processes for processing and rendering. This pipeline can be easily customizable and may include other components such as Wekinator for machine learning, Max/Msp, or any other applications that support OSC.

The GUI of *SoliDSPFeatures2OSC* (Fig. 1) enables the individual selection and activation of Soli DSP transforms and core features, plotting real-time features graphs, and logging of the overall amount of selected features that are pushed through the pipeline. All the available data transforms and core features in Soli C++ API have been exposed in the GUI. We made the code available on private Github repo for the Alpha Developers community for convenience in experimenting and rapid prototyping with Soli and other creative tools. For the time being and because of nondisclosure constraints, it is unavailable to the general public.

3.2 Soli for Real-time Parametric Control of Audio Synthesis

We have conducted experiments focused on the use of Soli for real-time parametric control of audio synthesizers. Using the classification modules of the Gesture Recognition Toolkit (GRT) [3], we implemented a machine learning pipeline to detect swipe gestures and map them as generic triggers that activate musical events. We built a set of interactive musical patches written in Sporth [1], a stack-based domain-specific language for audio synthesis, that use Soli's range parameter and the swipe-gesture detection events to explore how gestural control and different kinds of mappings can be used to create a compelling instrument in combination with physical modeling synthesis. They are described as follows:

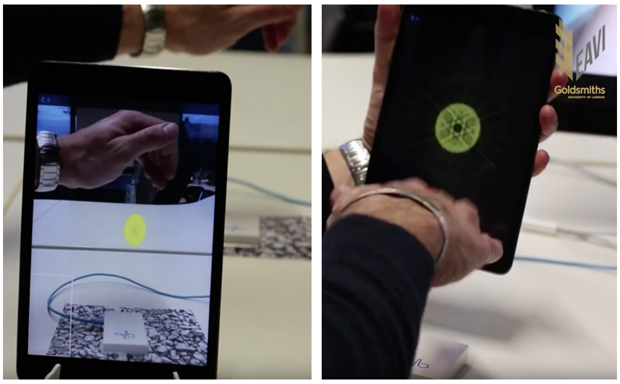

Figure 2: a) spatial UI view for mobile augmented reality and b) skeumorphic UI view for mixed multi-touch and micro-gestural control

- *Wub* - Soli's range parameter is used for enabling the user's varying hand height to control the speed of a low-frequency oscillator (LFO) and cutoff frequency of a band-limited sawtooth oscillator, to produce the *wub*-bass sound, popular in dubstep.
- *Warmth* - Hand presence triggers three frequency modulation voices to randomly play one of six notes for a finite duration, and the range parameter is used to control the amount of applied reverberation.
- *Flurry* - an arpeggiated, band-limited square wave oscillator, where the speed of the oscillator's arpeggiation is controlled by the range parameter. When a swipe is detected, an envelope filter is applied to the oscillator output, altering the timbre.
- *FeedPluck* - patch where swipe gestures act as excitation signals for injecting energy in a Karplus-Strong physical model of a plucked string; the changing height of the hand above the sensor controls the synthetic reverberation and feedback delay line applied to the model's output.

3.3 Soli for Mobile Musical Interaction

We have developed a prototype to explore musical interaction using the Soli sensor in mobile scenarios. This prototype consists of an iOS application that receives a network stream of real-time OSC messages from a computer hosting Soli and running *SoliDSPFeatures2OSC*. The application loads five audio samples into memory and two different application views that use the Maximilian library [18] to render a real-time abstract visual representation of the Fast Fourier transform (FFT) analysis of the playback and time stretching parameterization of each sample. Each view supports the task of changing the current sample in playback and the control of a time-stretching effect for prototyping interaction in two mobile scenarios; a) Soli was attached to an iPad, where the user can interact using both air gestures via Soli and/or the multi-touch screen; b) a mobile augmented reality experience, where Soli is decoupled from the iPad to support situated interaction with an object or surface or, used as a wearable (e.g., kept inside the pocket) as part of a body-area-network.

In both scenarios, we used the same set of Soli features (acceleration, fine displacement and energy total) as parameter in a direct mapping strategy with configurable thresholds so that sudden air gestures within the range and in the direction towards Soli would trigger sample change, while more fluid and continuous gestures would control the time stretching of the current selected sample.

4. DISCUSSION

Our study consists of a preliminary assessment where we explored the following set of musical tasks with Soli for musical interaction as suggested by [19]:

1. Continuous feature modulation (amplitude, pitch, timbre and frequency of LFO).
2. Musical phrases (articulation, speed of arpeggiator).
3. Combining pre-recorded material (triggering samples).
4. Combining elementary tasks (modulation of and playback of samples, setting audio-visual modes).
5. Composing a simple audiovisual narrative.

We find that this set of musical tasks based on elementary gestures and mappings, confirms the potential of Soli to enable different kinds of musical interaction. Our perspective builds upon a workflow and set of tools that converge with community practices around appropriation of sensing technologies for rapid prototyping of NMIs. By providing an OSC wrapper for the processing pipeline exposed by the Soli SDK, our intention was to support a more flexible, easy-to-use applicational pipeline for rapid prototyping. Notwithstanding, Soli SDK supports a straightforward and developer-friendly approach; we were able to easily access configuration parameters, DSP transforms and features, and to expose them for rapid prototyping of musical interaction and expressive control. It was easy to get the “low hanging fruit“ by designing explicit mappings for our exploratory musical tasks that leveraged on Soli features such as range, fine displacement, velocity, acceleration, and discreet recognition of swipe gestures. The set of different musical tasks that we have explored with Soli demonstrates its multifunctional and configurability capabilities for different applications. Designers will have the option to make use of the unique affordances that the sensor offers either in isolation, in redundancy, or in combination with other kinds of sensors traditionally used in NMI design practice. Soli's characteristics conform with many of the attributes identified for successful wearables, placed in different locations of the body as constituents of personal or body area networks. It's compact design can be used to adapt to, or even to disappear into, lightweight, aesthetically pleasing, variably shaped NMIs, designed to suit wearers, without being obtrusive or by minimizing the impact on their playability. Furthermore, single solid-state chips have less maintainability and reliability issues than composite solutions, which added to the ease of integration with embedded hardware platforms such as Raspberry Pi3, constitute clear advantages for the NIME community.

As far as discrete recognition of other gestures beyond swipe, it was difficult to obtain good results in a straightforward manner. To take advantage of the fine temporal resolution and high-throughput of Soli, conventional machine learning techniques that use light training and fast workflows appear to be insufficient. In the course of development the need for more sophisticated techniques such as deep learning (DL) became evident. This is corroborated by the difference in the number of gestures recognized and success rates between using standard machine learning techniques that use SVM or decision trees [9] and DL [20]. On one hand, DL seems to provide great potential for new discoveries in terms of gesture recognition. On the other hand, it may restrict the scope of deploying Soli to computationally more powerful devices. Hence, there is a great margin for exploration of how micro-gestures and the virtual tool language could be useful in musical contexts. We envision them as particularly useful for speeding up the activation times of triggering musical events, mode-setting, and for fine-grained audio parameter manipulation and performance with micro-tonal music systems. We believe that, just as with NMI designs that adapt gestural vocabularies from pre-existing instruments and that leverage on previously developed motor control abilities, the virtual tool gesture language can be appropriated and used to foster general adoption.

While Soli is still in its early stages of development, it offers great potential to fulfill an interesting set of use cases for musical expression, particularly for the category of alternate controllers and for wearable instruments; it may allow more freedom of action, and more intimate, nuanced control over a musical composition, performance or digital instrument than other existing sensors. By providing the user with haptic feedback and proprioception through their own hand motion when enacting micro and virtual-tool gestures, it may improve a musician's ability to control a digital musical instrument, as well as helping the user to learn the feel of the instrument more quickly, as argued by [16].

5. CONCLUSION AND FUTURE WORK

In this paper, we discuss the potential of Soli for musical interaction particularly for building alternate controllers and wearable interfaces for musical expression. We also reviewed prior work with sensing technologies in NMI that addresses fundamental aspects of sensing technology for musical interaction in order to contextualize our approach. We provided a brief overview of Soli's technology stack, contextualized our initial efforts at an early stage of Soli development and within the Soli Alpha developers program. We presented our explorations based on musical tasks as means for a preliminary assessment of the potential of Soli for musical interaction and building NMIs.

5.1 Future work

We look forward to apply learning transfer techniques for pre-trained models resulting from DL techniques such as explored in [20]. This will enable the exploration of how the virtual tools language and micro-gestures can be useful in different musical tasks and contexts. We hope to further this work by creating wearable instruments using Soli in combination with embedded hardware and apply user-centric methodologies for their evaluation.

6. ACKNOWLEDGMENTS

We would like to thank Google ATAP for participating in the Soli Alpha Developers program, specially to Fumi Yamazaki and Nicholas Gillian for all support that led to this publication. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N°644862.

7. REFERENCES

[1] Batchelor, P. *Sporth*, December 2016. https://paulbatchelor.github.io/proj/sporth.

[2] N. Gillian and S. Nicolls. A gesturally controlled improvisation system for piano. In *1st International Conference on Live Interfaces: Performance, Art, Music*, number 3, 2012.

[3] N. E. Gillian and J. A. Paradiso. The gesture recognition toolkit. *Journal of Machine Learning Research*, 15(1):3483–3487, 2014.

[4] K. Gniotek and I. Krucinska. The basic problems of textronics. *Fibres and Textiles in Eastern Europe*, 12(1):13–16, 2004.

[5] A. B. Godbehere and N. J. Ward. Wearable interfaces for cyberphysical musical expression. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 237–240, 2008.

[6] Google ATAP. *Project Soli*, April 2017. https://www.google.com/atap/project-soli/.

[7] J. Han and N. Gold. Lessons learned in exploring the leap motion sensor for gesture-based instrument design. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2014.

[8] A. R. Jensenius, M. M. Wanderley, R. I. Godøy, and M. Leman. Musical gestures: Concepts and methods in research. In R. I. Godøy and M. Leman, editors, *Musical gestures: Sound, movement, and meaning*, chapter 2, pages 12–35. Routledge, New York, 2010.

[9] J. Lien, N. Gillian, M. E. Karagozler, P. Amihood, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev. Soli: Ubiquitous gesture sensing with millimeter wave radar. *ACM Transactions on Graphics (TOG)*, 35(4):142, 2016.

[10] J. Malloch, D. Birnbaum, E. Sinyor, and M. M. Wanderley. Towards a new conceptual framework for digital musical instruments. In *Proceedings of the 9th International Conference on Digital Audio Effects*, pages 49–52, 2006.

[11] C. B. Medeiros and M. M. Wanderley. A comprehensive review of sensors and instrumentation methods in devices for musical expression. *IEEE Sensors Journal*, 14(8):13556–13591, 2014.

[12] C. B. Medeiros and M. M. Wanderley. Multiple-model linear kalman filter framework for unpredictable signals. *IEEE Sensors Journal*, 14(4):979–991, 2014.

[13] O. Nieto and D. Shasha. Hand gesture recognition in mobile devices: Enhancing the musical experience. In *Proc. of the 10th International Symposium on Computer Music Multidisciplinary Research*, 2013.

[14] D. A. Norman. Natural user interfaces are not natural. *Interactions*, 17(3):6–10, 2010.

[15] K. Nymoen, M. R. Haugen, and A. R. Jensenius. Mumyo – evaluating and exploring the myo armband for musical interaction. In *Proceedings of the International Conference on New Interfaces for Musical Expression*. Louisiana State University, 2015.

[16] M. S. O'Modhrain. *Playing by feel: incorporating haptic feedback into computer-based musical instruments*. PhD thesis, Stanford University, 2001.

[17] S. Serafin, S. Trento, F. Grani, H. Perner-Wilson, S. Madgwick, and T. J. Mitchell. Controlling physically based virtual musical instruments using the gloves. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2014.

[18] StrangeLoop. *Maximilian*, April 2017. https://github.com/micknoise/Maximilian.

[19] M. M. Wanderley and N. Orio. Evaluation of input devices for musical expression: Borrowing tools from hci. *Computer Music Journal*, 26(3):62–76, 2002.

[20] S. Wang, J. Song, J. Lien, I. Poupyrev, and O. Hilliges. Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pages 851–860. ACM, 2016.

[21] M. Wright and A. Freed. Open sound control: A new protocol for communicating with sound synthesizers. In *Proceedings of the 1997 International Computer Music Conference*, pages 101–104, 1997.