Techniques in Swept Frequency Capacitive Sensing: An Open Source Approach

Colin Honigman
[1]California Institute of the Arts
24700 McBean Pkwy
Valencia, California
colinhonigman@alum.calarts.edu

Jordan Hochenbaum
[1]California Institute of the Arts
24700 McBean Pkwy
Valencia, California
jhochenbaum@calarts.eu

Ajay Kapur[1,2]
[2]Victoria University of Wellington
Kelburn
Wellington, New Zealand
akapur@calarts.edu

ABSTRACT
This paper introduces a new technique for creating *Swept Frequency Capacitive Sensing* with open source technology for use in creating richer and more complex musical gestures. This new style of capacitive touch sensing is extremely robust compared to older versions and will allow greater implementation of gesture recognition and touch control in the development of NIMEs. Inspired by the *Touché* project, this paper discusses how to implement this technique using the community standard hardware Arduino instead of custom designed electronics. This sensing technique requires only passive components and can be used to enhance the touch sensitivity of many conductive materials, even biological materials such as plants. This paper also introduces a new Arduino Library, *SweepingCapSense*, which simplifies the coding required to implement this technique. Lastly, this paper will discuss its use in the project known as *Cultivating Frequencies*.

Keywords
Swept Frequency Capacitive Sensing, Capacitive Sensing, Touch, Arduino, Open-Source, Gesture Recognition, Sound Sculpture, Biotechnical Interfaces

1. INTRODUCTION
With the development of electronics, and consequently electronic instruments, the relationship between touch and sound has changed drastically. Touch has not been a requirement of musical expression since Theremin introduced his eponymous instrument in 1920[2]. Although many electronic instruments and interfaces employ touch as the main technique for sound activation, manipulation, and expression, in most cases, the connection between the sound and the gesture is more of an abstraction when compared to acoustic instruments. For example, the same violin in the hands of a novice and a virtuoso will sound immensely different even when the same simple gesture is performed; with many electronic instruments and interfaces, it is not the quality of the sound that changes between the novice and the virtuoso, especially when examining a simple gesture, rather it is the understanding of the instrument and range of sounds and consequent manipulations, and even the crafting of the sound itself that is learned over extended use and practice. The gesture in this situation has been applied, arbitrarily in most cases, to the sound. With acoustic instruments, it was the gesture that physically created the sound, and even had a role in the evolution of the instrument.

In computing, touch dominates the interaction between human and computer. However, touch in this context takes on a slightly different meaning, and differs completely from how one touches a musical instrument. Only recently is touch steadily becoming less of a requirement in the realm of human computer interaction. As computer vision and 3D sensing technologies become more advanced and more accurate, the fundamentals of how we interact with technology are forever changing. Touch, in the computing and NIME communities, has been focused on screens, where a lot of instruments and interactions have been designed around a screen[4, 5, 10], even delving into ways of expanding their sensing abilities[1, 3]. Touch capacitive sensing has not seen many breakthroughs outside of the application to multi-touch screens. At least that was true until Munehiko Sato and his team developed the *Touché* project and their new technique for capacitive sensing called *Swept Frequency Capacitive Sensing*[9]. This new technique opens up a whole dimension of touch sensing capabilities and gesture recognition. For the NIME community, this ability to create unique sensors and combinations of sensors that can take almost any shape, will allow for the development of instruments and interfaces designed for the musical gesture, rather than for the electronic sensor.

In this paper the idea and importance of touch will be explored as it pertains to developing instruments, interfaces, and installations for musical expression, specifically focusing on touch capacitance, and the new technique of swept frequency capacitive sensing, using an Arduino micro-controller, as a means of creating new and non-traditional sensors that allow for greater sensitivity and increased gesture recognition. The older method of touch capacitance will be analyzed and compared to this new method, and the benefits of this new method will be illustrated through various test results. Lastly, this paper will explore this technique through specific use within the context of the project known as *Cultivating Frequencies*, which is a interactive sonic sculpture with a hydroponic garden as its subject, where the plants are given touch sensing abilities through this new technique, allowing the observer to directly interact with the musical system through the plants.

2. History of Touch Capacitance
Touch capacitance is not a new technology nor is it a novelty in the context of NIMEs. It has found its way into many devices and has many unique and industrial applications; from measuring micrometer displacements between silicon cantilevers, to Stud Finders, which detect wooden studs through plaster[6]. However, large-scale use has only found its way to the consumer electronics market in recent years. In the open source community there have been few advancements in the technology driving it, most of which has come in the form of more commercially available parts such as cheaper touch screens and sensors. Because the basic form of a touch capacitance circuit is quite easy to replicate there is no need for industrial grade touch circuitry for most of the DIY and experimental electronic arts community. However, until now, this basic circuit has only allowed the detection of proximity and varying degrees of absolute touch.

This kind of sensor can be implemented as a discrete or continuous sensor depending on the circuit used. This, plus the fact that capacitive sensing can sense conductive objects through other non-conductive materials make it an ideal sensor for industrial and consumer electronics alike. These facts also make capacitive sensing ideal in the development of NIMEs, allowing instrument and interface makers the ability to extract rich touch information while being able to hide the sensor from sight. This is important because often times the development of NIMEs is not only about technology and sensors, but about instrument design, and it can be very hard to negotiate these two objectives when dealing with sensors that have specific spatial and topographic requirements.

While most of the progress in capacitive touch sensing has been seen in the mobile computing field more recently Sato and his team, of the Disney Research *Touché* project, have taken a big step forward for the embeddable capacitive sensor. By exploring ways in which everyday objects can be transformed into rich touch sensors, that not only sense touch but the manner of touch, they were able to implement a unique style of *Swept Frequency Capacitive Sensing* (SFCS). The novelty of this method is found in the "Swept Frequency" portion of the name. While traditional capacitive touch sensors detect conductivity at a fixed frequency, SFCS uses an entire range of frequencies. By sweeping through this range of frequencies the readings from a single sensor take on a two-dimensionality, which, when graphed, makes it easy to actually see the difference between a single-finger touch, a two-finger pinch, and a whole-handed grab.

Below, we will analyze the difference between single and swept frequency techniques and it will become obvious how much more rich the sensor data is with the latter. Before looking at this data, it is important to note that the *Touché* project utilized a unique and purpose built sensor and microcontroller, which is to say, it is not a community standard open-source technology. Intrigued by this work Mads Hobye[1], with the help of Nikolaj Møbius[2], both members of the interactive art studio Illutron[3], figured out how to employ this technique on an Arduino. By acting directly upon the built-in timers of the Arduino, they were able to output a waveform and change its frequency, allowing for SFCS. Details of how they did this will be discussed below.

3. Implementation

Touch capacitance was a novel way of opening a capacitive circuit to allow the introduction of influence from outside conductive objects. Now, touch sensitive electronics have been integrated into the daily fabric of our digital lives. From touch screen phones to track pads, our fingers regularly slide over smooth capacitive surfaces, reaching levels of ubiquity where we no longer are aware of or even impressed by them.

For those who are a part of the open source and NIME communities, touch capacitance has been an area of great interest and inspiration for many years. While multi-touch screens have usurped most of that interest in the past decade, the ability to create your own sensor and hide that sensor within an object may prove to be more important than how we interact with flat screens.

In this section we examine how sensing technology has been achieved in the recent past using open source technologies. This section also introduces a new technique for Touch Sensing using SFCS and a new Arduino library, *SweepingCapSense*.

3.1 Capacitive Sensing

On an Arduino, the standard and recommended way of creating a capacitive touch sensor was by means of the *CapSense*[4] library using the circuit shown below (see Figure 1). This method is simple and involves very few components, at the very least a single resistor and some conductive material, and uses only two digital pins on an Arduino. The coding is very simple as well and requires the setup of the *CapSense* object and then a reading.

[1] http://www.hobye.dk
[2] http://dzlsevilgeniuslair.blogspot.se
[3] http://illutron.dk
[4] http://playground.arduino.cc/Main/CapSense

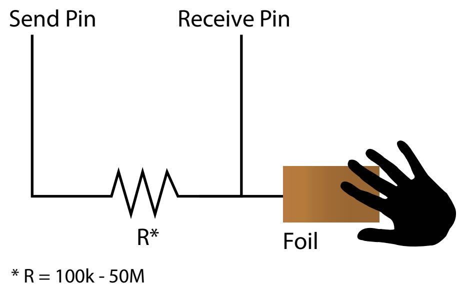


Figure 1: CapSense Diagram

As described on the *CapSense* webpage, the value of the resistor alters the sensitivity of the sensor allowing a range of interaction from absolute touch to proximity of up to 2 feet. The Arduino and *CapSense* method has been ideal due to its simplicity and ease of use. However, beyond proximity and absolute touch, there isn't much more information that can be gleaned from the sensor data.

Later in this section we will test this method against the SFCS method using identical materials and test gestures to visualize the difference in quality of the data returned.

3.2 Swept Frequency Capacitive Sensing

While Sato and his team were the first to implement the SFCS method, it was Nikolaj Møbius who figured out how to do it using an Arduino and passive components. In the *Touché* project they use an AD5932 Wave Generator IC that can produce a sine wave with very high resolution at frequencies between 1kHz and 3.5MHz. While the Arduino can also produce a signal at these frequencies, its resolution is lower and it can only produce a square wave so there are a lot of unwanted harmonics that are produced as well. Møbius figured out that using a simple LC (inductor – capacitor) circuit, also known as a resonant circuit, could help transform the square wave into a sinusoidal wave[7]. Figure 2 shows the components and schematic for Møbius' technique.

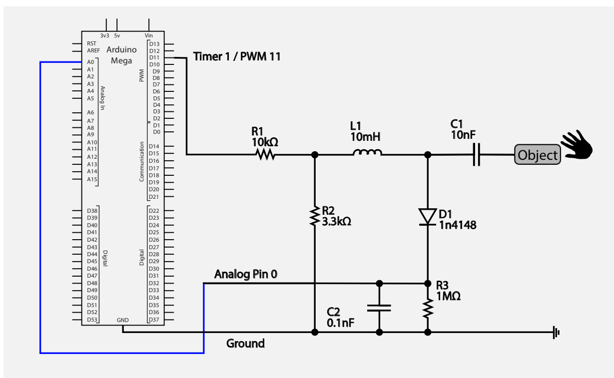


Figure 2: SFCS Circuit for Arduino Mega - Send pins change depending on Arduino model and Timer used

The example code that Møbius gives is a little intimidating for those unfamiliar with using hex code and binary to directly address and alter the hardware timers on the Arduino. To make the code side of this technique as simple as the *CapSense* library, the *SweepingCapSense* library was created. At the time of writing, the library is only for the Arduino Mega but will soon be available on as many versions as possible.

The *SweepingCapSense* library uses two main objects, *SweepingCap* and *Touch*, which control the sweeping frequency signal and store the touch data respectively. When constructing the *SweepingCap* object there are two options, one (seen in Figure 2), to set how many frequencies to sweep (up to 198) and which of the 16 bit timers to use, and two, to use all four 16 bit timers by only passing in the number of frequencies. The *Touch* object constructor requires only one variable to set the Analog input pin, however the *Touch* object is not a requirement to achieve SFCS, rather it is a convenient data structure to store the values returned at each frequency.

To use the *SweepingCap* object, requires only two other steps. One, to call the setup() method inside the Arduino setup function. And two, to call the sweep() method inside of a for loop that counts to the number of frequencies set in the constructor. Passing the iterator into the sweep() function sets the frequency. The *Touch* object has several steps as well. It needs to be reset() at the top of the main loop; inside the sweeping frequency for loop, readPin() can be called to automatically read and store the value at the current frequency; the topValue() and topPoint() can be determined, stored, and retrieved; and the value can be interpolated, with setter and getter methods.

4. Evaluation

This section shows analysis comparing both techniques. Below the sensor readings for two test materials with different touch positions and touch gestures for both the old and new techniques for capacitive sensing. The objects used were a copper electrode (approximately 5cm x 2cm) and a small rosemary plant (approximately 15cm tall) with the electrode placed in soil. The same series of tests were conducted with both objects using the two libraries and circuits, and the results can be seen below in Table 1, and Figures 3 and 4.

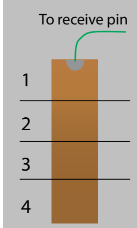
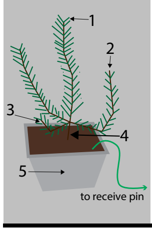
Object	Gesture	Results (by position)				
Electrode	**Position**	**1**	**2**	**3**	**4**	
	Finger Pad	1426	1430	1432	1428	
	Fingertip	1375	1385	1391	1400	
	Two Finger Pinch *(Edges)*	1450	1425	1400	1390	
	Two Finger Pinch *(Flat Sides)*	1450	1440	1435	1483	
	Two Finger Press *(Flat Side)*	1470	1475	1482	1483	
Plant	**Position**	**1**	**2**	**3**	**4**	**5**
	Single Finger Touch	61	59	60	900	67
	Two Finger Pinch	70	57	150	1247	389
	Three Finger Pinch	116	80	240	1040	435
	No touch *(2cm away)*	30	33	60	76	60
	Grab	118	82	163	571	730

Table 1: Results from CapSense Test

Using the *CapSense* library we see in Table 1 that only a single piece of data is given. While some gestures may be recognizable with this single piece of data, overall, it is difficult to clearly understand these results as unique gestures. Table 1 shows that some values appear at different positions and different gestures, for instance, on the electrode a fingertip at position 4 is perceived to be the same as a pinch at position 3. At least the plant's test results show that the dispersion of the signal through the soil and into the plant helps to create a wider range of results. However, these results tend to be very susceptible to outside influences that result in a lowered repeatability over time.

Using Møbius' circuit and the *SweepingCapSense* library, with the same objects and gestures, results in a different kind of data altogether. As seen below in Figures 3 and 4, the data representation of a single gesture is a two-dimensional data set that not only returns a peak value similar to the previous method, but also returns the values at every other frequency. The resulting shape gives greater insights into the object's response to the interaction and allows more opportunity for gesture recognition.

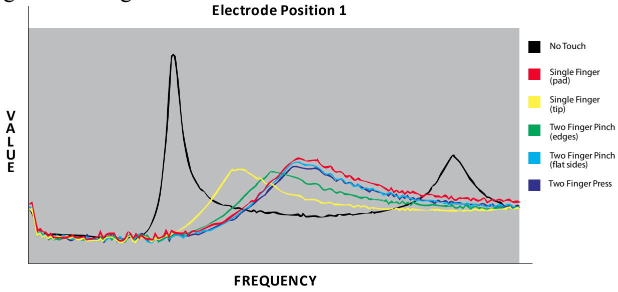


Figure 3: Electrode test results using SFCS method

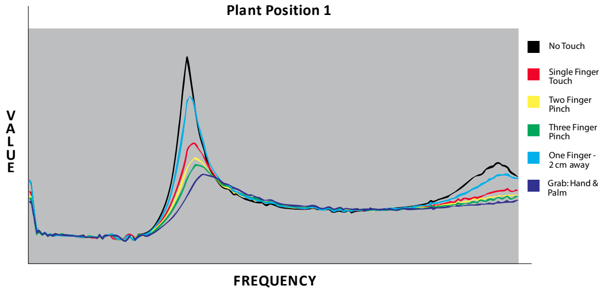


Figure 4: Plant test results using SFCS method

These graphs only show the results of each gesture at one position. It is important to note that this technique is not only sensitive to gesture but to position as well. Figure 5 shows the results of the same gesture, a two-finger pinch, in all the testing positions on the rosemary plant.

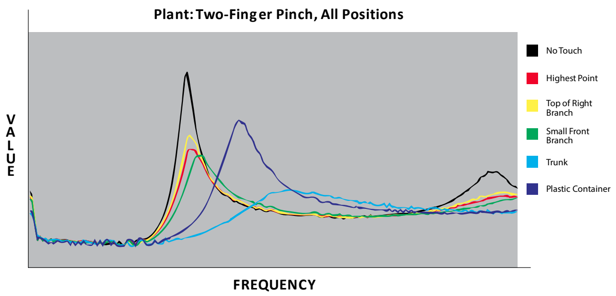


Figure 5: Plant test - one gesture (two-finger pinch), all positions.

The data clearly shows a vast improvement over the single frequency method in all aspects. Even when the resonant frequency is similar there is enough difference throughout the rest of the values to discern gestural subtleties. Also, the repeatability of these shapes, if not the specific peak frequency and value, is very high. This suggests that with enough testing and training, a single sensor could detect many different gestures and interactions.

5. Case Study: Cultivating Frequencies

Cultivating Frequencies is a hydroponic garden and sound installation that uses this technique for SFCS and the *SweepingCapSense* library to turn each of its fifteen plants into touch sensors. The installation also includes several hydroponic sensors to collect data and use it to drive a generative musical system. The touch sensitive plants encourage users to interact with the garden as a living thing and as a sound installation. Musically, the plant sensors give the users note-level and effect-level control over a series of oscillators and audio effects. MAX/MSP was used for the sound design and generative system and an Arduino Mega with a custom designed and printed shield was used for the touch sensors and the ultrasonic sensors (for user detection).

Figure 6: The garden at its permanent home in Aiguablava, Spain

Inspired by the *Botanicus Interacticus* project[8], which applied the new *Touché* technology to plants, this project is perhaps the first to utilize the open source method on such a large and permanent scale. As the installation grows, it seems logical that the data will grow as well. However, just how much the data and interaction is affected in the long run is unknown. Fortunately, another aspect of the project is to make a comprehensive collection of all data for as long as the piece runs, so this change will be monitored and freely displayed for all who are interested.[5]

6. Conclusion

Touch is an integral part of the human experience. It brings with it an intimate and innate knowledge that often times can only be understood through the act itself. Almost all traditional musical instruments require the player to touch them in order for sound to be produced. Indeed it is the physical nature of the instrument that allows a performer to achieve virtuosity through repeated movements and gestures, eventually allowing them to free their minds from the physical movements and focus on higher-level musical ideas such as phrasing, expression, and improvisation.

Touch, as it is used in technology, lacks a certain subtly that is readily found in most tools and instruments. In the development of NIMEs touch has focused so much on multi-touch displays and gesture-restricting sensors that the conversation of what touch in other contexts can achieve has been almost forgotten. Some of the flaws with the current trend of new interface development in music are lack of imagination (i.e. grids, banks of knobs/sliders), an over complication of design, or a reliance on novelty. Striving to find the one controller that "does everything," is an interesting pursuit but it seems to suffer from a lack of permanent cultural relevance. The NIMEs that last tend to be simpler and are approached more like instruments than interfaces.

This new technique for *Swept Frequency Capacitive Sensing* will hopefully reignite an interest in the possibilities for touch based and gestural NIMEs. Further development is needed to create easier methods for achieving gesture recognition and more studies need to be done to test the limits of this technology. In *Cultivating Frequencies*, fifteen plants running off the four 16 bit timers on an Arduino Mega required a lot of processing power and memory. Because of the time required to read each pin at each of the frequencies, the system slows down considerably with each added touch point. Strategies can be implemented so that all sensors are not reading at once but this also limits the interaction possibilities.

7. ACKNOWLEDGMENTS

The author would like to thank the team from Art Center College of Design, including: Kenneth Cameron, James Meraz, Jason Pilarski, Sean Chen, Marc Dubui, and Wen Han. This amazing team of teachers and students helped conceive the project *Cultivating Frequencies*, which lead to the developments discussed in this paper. The author would also like to Ajay Kapur, Jordan Hochenbaum, and Owen Vallis for their mentorship and support.

8. REFERENCES

[1] Gelineck, S. et al. Towards an Interface for Music Mixing based on Smart Tangibles and Multitouch.

[2] Glinsky, A. 2000. *Theremin: Ether Music and Espionage*. University of Illinois Press.

[3] Hochenbaum, J. and Kapur, A. 2011. Adding z-depth and pressure expressivity to tangible tabletop surfaces. *Proceedings of the International Conference on New Interfaces for Musical Expression (Oslo, Norway, 2011), Jensenius AR, Tveit A., Godøy RI, Overholt D.,(Eds.)* (2011), 240–243.

[4] Hochenbaum, J. and Vallis, O. 2009. Bricktable: A musical tangible multi-touch interface. *Proceedings of Berlin Open Conference '09. Berlin, Germany* (2009).

[5] Jordà, S. et al. 2007. The reacTable: exploring the synergy between live music performance and tabletop tangible interfaces. *Proceedings of the 1st international conference on Tangible and embedded interaction* (2007), 139–146.

[6] Larry, B. 1997. *Capacitive sensors design and applications*. New York: IEEE Press.

[7] Nikolaj Møbius 2012. Dzl's Evil Genius Lair: Arduino do the Touché dance.

[8] Poupyrev, I. et al. 2012. Botanicus Interacticus: interactive plants technology. *ACM SIGGRAPH 2012 Emerging Technologies* (2012), 4.

[9] Sato, M. et al. 2012. Touché: enhancing touch interaction on humans, screens, liquids, and everyday objects. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2012), 483–492.

[10] Yu, N.-H. et al. 2011. TUIC: enabling tangible interaction on capacitive multi-touch displays. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2011), 2995–3004.

[5] www.cultivatingfrequencies.com