A Force-Sensitive Surface for Intimate Control

Randy Jones [*], **Peter Driessen** [†], **Andrew Schloss** [†], **George Tzanetakis** [†]

[*] Madrona Labs
2045 13th Ave. W, Seattle, WA 98119

[†] University of Victoria, Canada
Music Intelligence Sound Technology Interdisciplinary Centre (MISTIC)

randy@madronalabs.com, peter@ece.uvic.ca, aschloss@finearts.uvic.ca, gtzan@uvic.ca

Abstract

This paper presents a new force-sensitive surface designed for playing music. A prototype system has been implemented using a passive capacitive sensor, a commodity multichannel audio interface, and decoding software running on a laptop computer. This setup has been a successful, low-cost route to a number of experiments in intimate musical control.

Keywords: Multitouch, sensors, tactile, capacitive, percussion controllers.

1. Introduction

This paper presents a device built to scratch a specific itch. For a number of years, we have been interested in the musical possibilities presented by force-sensing surfaces. A tantalizing glimpse of these possibilities was provided by the MTC Express, a device introduced commercially as a musical controller in the year 2000 by Tactex Controls, Inc. We used the MTC Express in a number of pieces from 2001 to 2005, including “Uni” by Andrew Schloss and Randy Jones, and a performance by Randy Jones at NIME 2005 in Vancouver.

One of the applications of the MTC Express that we found most exciting was in playing a physically modeled membrane. In this application, force data from the touch surface was used to both excite and damp a waveguide mesh algorithm [1]. Instead of treating the controller data as multiple discrete touchpoints, the raw force image was applied directly to the physical model. The compatibility of the dynamic 2D force data with the 2D mesh made this a very intuitive use for the controller, and the results were viscerally appealing: some of the salient qualities associated with hand drumming were reproduced as emergent behavior of our combination of software and hardware.

The MTC Express was a gateway to new performance ideas for us, particularly in offering data from a homogeneous, two-dimensional surface. While we explored the uses of this data, the limitations of the device prompted us to some questions. If we could make a force sensing surface with any dimensions we wanted, what would they be? What spatial and temporal resolutions would be sufficient for making an instrument we would still be happy with ten years from now? And by what musical mappings or metaphors could all of this data be applied to expressive music making?

Pursuing answers to these questions has led to a prototype sensor system, previously presented by Randy Jones in his Masters thesis entitled “Intimate Control for Physical Modeling Synthesis.”[2] In this paper we will discuss the construction of the system, and present hardware and software details that we hope will be of use to the NIME community.

1.1. Design Goals

What factors in instrument design contribute to expressivity? How can we make computer-mediated instruments that equal or exceed acoustic instruments in expressive potential? We consider these fruitful questions for long-term research. If we had at our disposal sensor and software systems capable of reproducing the sounding capabilities of acoustic instruments, a wide variety of experiments comparing the acoustic world with the synthetic would be possible. Currently, sensors that can support this research tend to be expensive, one-of-a-kind prototypes. So our first design goal is accessibility: we would like to make a sensor that is reliably and cheaply reproducible.

F. R. Moore introduced the concept of *control intimacy* in his article “The Dysfunctions of MIDI.” [3] Control intimacy seems to a useful criterion in instrument design, in part because it invites both qualitative and quantitative observations. Qualitatively, we can point to examples of instruments that afford high intimacy of control such as the violin and hand drums. These instruments offer tactile feedback from the sounding apparatus, multiple mechanisms for

NIME09, June 3-6, 2009, Pittsburgh, PA

creating and modifying sounds, and have a physical scale that accommodates a wide variety of performance gestures.

Quantitatively, many issues depend on musical context. Focusing on the development of our surface force sensor, we can ask: what sensor capabilities would be needed to make a physically modeled hand drum instrument with control intimacy equal to the real thing? Wessel and Wright [4] give an upper bound on the acceptable latency of sonic response to gesture at around 10 msec, but note that event onsets can be both controlled and perceived with a 1 msec precision in a percussive context. Our experience confirms these numbers, and meeting or exceeding them is another goal in our controller design. Though they imply a lower bound for the sampling rate of 1kHz, it also makes sense to ask what the maximum useful sampling rate would be. When we have gestural signals rather than discrete events as inputs to our synthesis models, arguments based on signal bandwidth seem to make sense. A finger scraping across the drum's surface can make sounds with energy well above 1kHz. This frictional sound is part of the interaction with the drum, and could in theory be made by a sufficiently detailed drum physical model. But creating friction sounds synthetically points to removing them from the actual physical interaction, and real world friction as well as the sounds it produces might be desirable. In general, there is a grey area here worth exploring; such exploration requires high-bandwidth inputs, up to the audio sampling rate, if possible.

Another aspect of an interface that can be quantified is the spatial sampling frequency, or resolution. In the context of playing a physically modeled drum, what is the maximum useful resolution? Considering the spatial distribution of modes on the drum head, as shown in Chladni patterns, we can hypothesize that a resolution on the order of the size of the highest frequency mode is necessary to capture the effects of applied forces on sound. The transverse wave velocity on a membrane is $c = \sqrt{T/\sigma}$, where T is the tension on the membrane in Newtons per meter, and σ is the membrane density in kg/m^2. For a typical small drum, this is approximately 100 m/s according to Fletcher and Rossing [5]. At this speed, a 10 kHz transverse wave is 1cm long. So if the above hypothesis is correct, a spatial resolution on the order of a centimeter is sufficient to characterize the relationship between sensor and sound up to 10 kHz. Other contexts lead to different requirements. In multitouch sensing, for example, distinguishing one finger from two closely spaced fingers is a reasonable goal that requires a resolution on the order of a millimeter.

One of Moore's criteria for intimacy is the match between the range of sounds produced and "the psychophysiological capabilities of a practiced performer"[3]. In general, greater control intimacy is obtained when the range of intentional control over a gestural input controlling a parameter maps more fully to the range of perceptible variation in output caused by that parameter. In this aspect, we can actually improve on acoustic instruments. It doesn't take any special training to touch a drum head so softly that we feel the touch but do not hear a sound. A sensor can easily detect a lighter touch than our fingers are able to, which would allow our entire range of touch to be applied to expressive control.

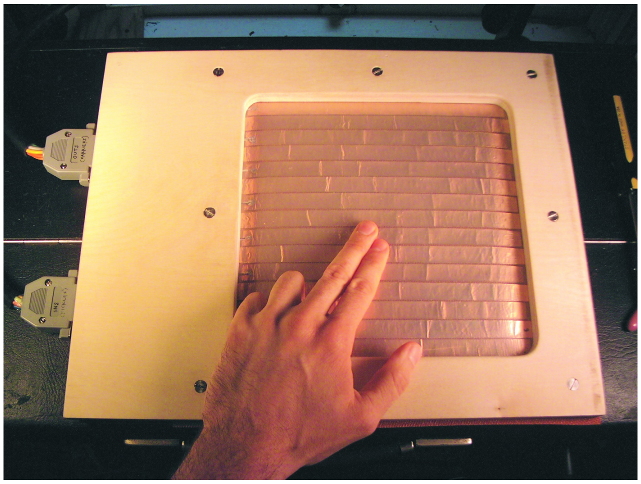

Figure 1. A Passive, Force-Sensitive Surface.

To round out the list of design goals, we can return to size. A 50 by 20 cm surface would be enough to support a wide variety of two-handed gestures. Combining this size with the rest of our modest specifications, we can calculate the amount of data that our ideal controller would generate. At a 1×1 cm spatial resolution and a 40 kHz sampling rate, with 16 bits per force sample, our 50×20 cm surface would require a data channel of 80 megabytes/sec to read. This is at the high end of what today's commodity data transmission protocols allow.

2. Hardware

Using capacitive sensing, we have implemented a prototype surface force sensor (Figure 1) that meets some of the above design goals. The active area of the touch surface is a 20 by 20 cm rectangle with rounded corners. Two copper antenna layers, with a rubber dielectric in between, form a matrix of variable capacitors. Line level AC signals from an audio interface are applied to eight parallel row antennas, capacitively coupled to eight column antennas on the other side of the dielectric. Applied force compresses the dielectric, increasing the capacitive coupling between the two antenna layers. These changes in capacitance are measured for each row/column pair by demultiplexing the received carriers in the frequency domain, a calculation done in software approximately 1000 times per second. Exclusive of the audio interface, the cost of the sensor stack was approximately $50 in materials. Compatible audio interfaces currently start at around $500 in cost. The entire sensor including the interface is small enough to be easily portable, another important consideration in a performance instrument.

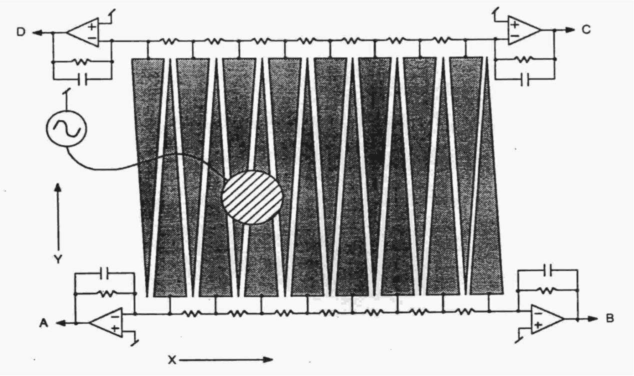

Figure 2. Layout of the Radio Drum backgammon sensor.

2.1. Sensing

Why capacitive sensing? Though the technique has definite drawbacks in its susceptibility to electrical noise and minor variations of sensor geometry, it was the only way we found to meet our design goals. The primary goal of low cost puts most single-point sensors out of reach. Though our prototype has only an 8×8 resolution, we would like future versions to scale up to the 30×30 resolution that our goals point to as acceptable for a small instrument. If we define an affordable instrument somewhat arbitrarily as under $1000, this gives us a budget of about a dollar per grid point, ruling out pressure sensors, FSRs (force sensing resistors), and other commodity technologies for single-point sensors, which are generally an order of magnitude higher in price.

One very interesting sensing possibility lies in the work of Koehly et al. [6]. They have made FSRs using conductive ink and a variety of substrates including paper. This technology could be used to print rather than build a sensor at very low cost, with technology in reach of the individual experimenter. The major stumbling block with this route, for the moment, is the response time of the sensor. Data from the paper cited shows that after a 3 kg load was applied, the experimental FSR took 75 msec to recover to its rest state. For now, this probably makes it unsuitable for our percussive application.

The Tactex MTC Express uses LEDs and photocells to sense the deformation of a web of fiber optic cables. Multiplexing the LEDs allows n LEDs and m detectors to form an $n \times m$ surface, so this approach is feasible in cost. Unfortunately, the response time of photodetectors is also too slow for percussive sensing. Commercial devices have reported bandwidths on the order of 100 Hz.

The idea for our sensing approach came from thinking about how to extend the capabilities of the Radio Drum. The Radio Drum, a three-dimensional gesture sensor using capacitive moments, was originally developed at Bell Labs by Robert Boie and Max Mathews in the late 1980's [7]. The prototype hardware was described as a “backgammon” sensor due to the layout of the sensing antennas, as can be seen in Figure 2. Examples of this hardware are still in use; there have been several versions created over the years by Max Mathews and Tom Oberheim. When Mathews left Bell Labs and arrived at CCRMA, he continued to develop and experiment, making several versions including what he now calls the Radio Baton. These instruments are all identical in basic sensing strategies, but differ in various details. For example, the latest Radio Baton has wireless sticks.

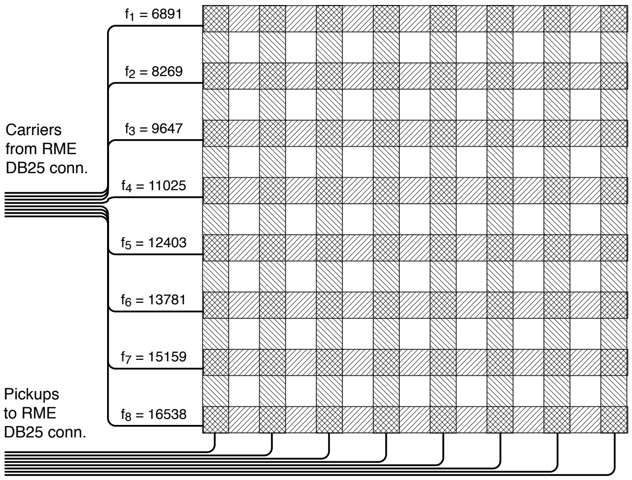

Figure 3. Block diagram of passive sensor hardware.

From the corner amplifier voltages V_A, V_B, V_C and V_D in Figure 2, the positions x, y and z of multiple transmitter coils in a small volume above the pad can be uniquely calculated. Recently, Ben Nevile and others at the University of Victoria have worked to improve the reliability, latency and precision of the Radio Drum [8], [9]. Where the original drum required a custom box of control electronics, the new approach relies on a commodity audio interface to sample the signals received by the antennas.

Because the voltage received by the Radio Drum's antenna is inversely proportional to the coil distance, its spatial resolution near the plane of the antenna is very good. If, instead of freely moving coils, the Radio Drum had a grid of carriers at fixed x and y positions above the surface, it could in principle sense the z position of at each carrier simultaneously. Using rows of carriers overlapping with columns of pickup antennas, this technique could also be used in an $n \times m$ configuration. This leads to the design of our new sensor, shown in Figure 3.

The horizontal carrier antennas each transmit a different frequency of sine wave from the audio interface. Surrounding them and running in between each pair of horizontal strips (omitted in this diagram for clarity) is a grounded guard area. On the other side of the rubber dielectric are the vertical pickup antennas, each of which is connected directly to a line input channel of the audio interface.

2.2. Materials and Construction

Our surface force sensor consists of ten layers of commonly available materials. These are listed from top to bottom in Table 1.

Table 1. Physical Layers of the Sensor

Layer	Thickness	Material
top	12 mm	birch plywood
ground	.01mm	aluminum foil
surface	1mm	polyethylene
carriers	.1mm	copper tape on polyester
bump	3mm	plywood
dielectric	3mm	silicone rubber
pickups	.1mm	copper tape on polyester
spacer	3mm	plywood
ground	.01mm	aluminum foil
bottom	12 mm	birch plywood

A rectangular hole is cut in the top two layers for access to the polyethylene touch surface, a material that comes textured on one side and smooth on the other. We tried using each side as a playing surface, and found the textured one to be very playable. The smooth side doesn't offer us the same feel of connection with the surface, because sideways movements with the fingertips cannot be felt as well. The entire stack of materials has been drilled through and is held together with nuts and bolts. The carrier and pickup layers are made using adhesive-backed copper tape, made by 3M, on a thin 3 mil polyester film. The adhesive on the tape has held up through many cycles of removing and replacing, a handy capability for experimenting with different antenna geometries.

The silicone rubber is a firm, red-orange material chosen for maximum resilience. We would like our sensor to return to its rest state very quickly when a force is removed. Our current prototype recovers from a firm touch in about 10 msec. A thin, gently curved plywood bump between carriers and dielectric helps us get this fast response, by keeping the rubber in a small amount of compression even when the controller is at rest.

The layout of the carrier and pickup layers is a crucial part of the geometry of the device. If the gaps between electrodes are too small, then the adjacent traces on the board are more capacitively coupled, creating crosstalk. If the gaps are too large, there is less area for capacitive sensing. Equations and rules of thumb concerning these tradeoffs are discussed by Baxter [10]. Another issue in our particular device is that the carrier antenna serves as electromagnetic shielding for the pickup antenna underneath. Figure 4 is a scale drawing of our carrier antenna plane. Each carrier strip is connected to a pin of the DB-25 connector at the left edge of the device. The pickup layer has the same geometry, but turned 90 degrees to create vertical strips. Matching DB-25 breakout cables, 1m in length, finish the connections to the audio interface.

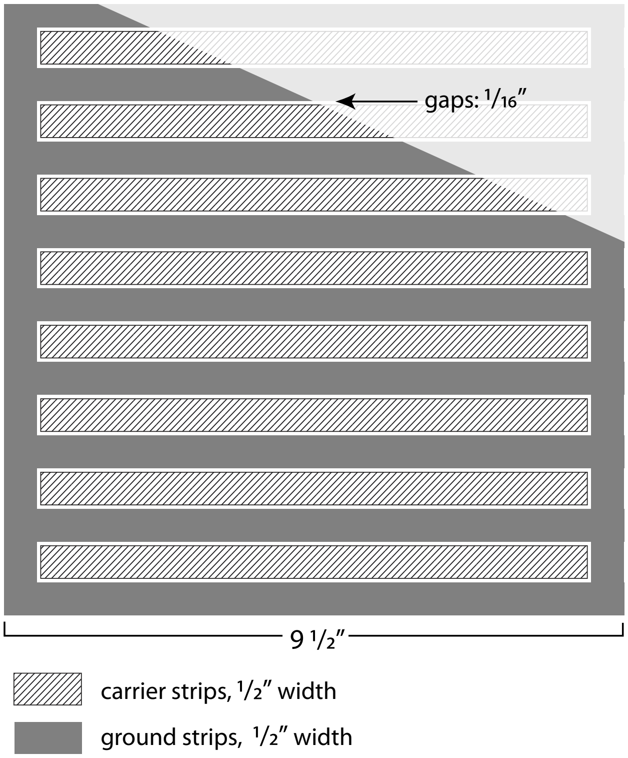

Figure 4. Scale drawing of carrier antenna layout.

2.3. Electronics

We have tested the sensor with two consumer audio interfaces at this point: the RME Fireface 800 and the MOTU Ultralite Mk. 3. We use the RME with its output set to high gain and its input level set to -10dBV, to create maximum gain through the sensor. The MOTU has a digitally controlled analog gain adjustment; we set this to its maximum. With these settings the choice of one interface or another makes little if any difference in the behavior of the system. Using each interface we measured the level of a single pickup, both with a single carrier being transmitted, and with no carriers transmitted in order to measure the noise. The ratio of these measurements puts the SNR of the carrier/pickup pair in the range of 36–38dB. For ideal readings from our passive sensor, specialized active electronics such as charge amplifiers should be used. By using a line level audio interface instead, we are degrading the potential performance of the sensor significantly. Despite this, we have found benefits to our "good enough" approach. The main one is that, to the extent we have made a usefully expressive controller, others who have an extra audio interface on hand can build similar passive devices and start making music without building amplifier circuits. Another is that we developed more capable calibration software than we might have with a better sensor. We have seen a variety of situations in live performance temporarily turn good sensors into bad ones. So making software that can handle a worst-case scenario might come in handy.

Future versions of the sensor will make it more roadworthy and reduce the existing noise. Our cables from the sensor to the interface increase the noise floor by approximately 20dB now. Moving the audio interface or its equivalent into a shielded case with the passive sensor, and making an amplifier circuit that matches the application, will result in a very sensitive, durable, reproducible instrument.

3. Software

A Max/MSP patch, along with custom externals written in C, processes the real-time data from the sensor. The inputs to the patch are the signals from the pickups, each of which covers one entire column of the grid. The amplitude of each carrier received by a pickup is inversely proportional to the distance between the two antennas. Hence, by analyzing the spectrum of the pickup signal we can find the distance across each pickup / carrier junction.

3.1. Demultiplexing

Because we have the luxury of choosing our carrier frequencies, the FFT is a particularly useful algorithm for this application. When all carrier wavelengths are integer divisors of the FFT window length, the separation between bands is complete. Each band of the FFT acts as a filter with zeroes on each of the unwanted carrier frequencies. Also, rectangular windowing can be used, which has two benefits: a constant amplitude response and increased efficiency. We use a 32 sample FFT window at a 44.1kHz sampling rate with no overlap.

After the FFT is calculated for each column, the magnitudes of the complex results are taken. Keeping only the bands where carriers are present results in 64 real signals at 1/32 of the system sampling rate or 1,378Hz. Each signal is the magnitude of one carrier/pickup pair, which depends on the force over a small area of the sensor. Since the 64 signals have coherence in space as well as time, we are justified in calling this whole bundle one *2D signal* from our surface sensor.

3.2. Interpolation

To apply the 2D signal to a synthesis algorithm such as the waveguide mesh, it must be properly upsampled back to the overall system sampling rate otherwise aliasing will be heard. Typically this is done by zero-padding followed by an FIR filter. Unfortunately, this high-quality method of upsampling is computationally expensive. For an acceptably attenuated stopband, below 80dB or so, on the order of 200 filter taps would be required. Factoring out our 32x upsampling into a series of five 2x upsamplers helps somewhat, but even this approach proved impossible to calculate in real time in our system.

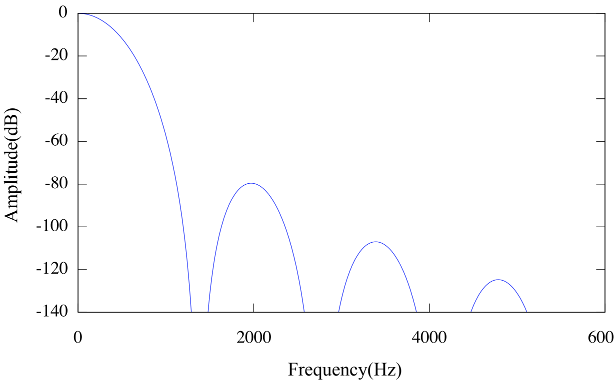

Figure 5. Frequency response of order-5 B-spline interpolator.

Because the bandwidth of our gestural signals is significantly less than the system sampling rate, we can use interpolation filters to do the upsampling without losing much information. Interpolation filters have significant rolloff all through the passband, a characteristic rarely acceptable in audio filters. But in our application they represent an acceptable tradeoff between fidelity and speed. The frequency response of the filter we used is shown in Figure 5. Because the B-spline functions are generated by convolving the unit pulse with itself, then convolving the output with itself, and so on, their frequency responses are powers of the $sinc$ function, $sin(x)/x$. The 5th order B-spline requires 20 multiplies and 22 adds per sample, an order of magnitude less work than the FIR filter. Note that, while the frequency response in the passband is bad, as promised, the first sidelobe is attenuated to -80dB.

3.3. Calibration

Dynamic calibration is used to correct for slow shifts in the sensor's position that result from hysteresis when no force is applied. During operation, the signal from each input taxel is highpass filtered to remove components of motion below a cutoff frequency. This cutoff varies according to the applied force. When no force is applied, components under approximately 50 Hz are filtered out. At a moderate pressure, no filtering occurs. The amplitude defined as a moderate pressure is currently determined in a calibration procedure: the player rubs a hand over the entire device with a constant pressure, and the maximum few values at each taxel are averaged to get that taxel's calibrated pressure value, which is saved to a calibration file.

In addition to the dynamic filtering, a static amplitude threshold, generated based on a histogram of the noise in the system at rest, is used to reject electromagnetic noise. The combination of dynamic and static calibration effec-

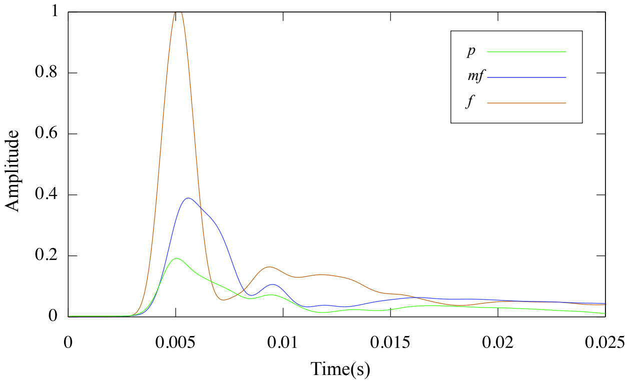

Figure 6. Amplitudes of three hand strikes on the sensor.

tively zeroes the shifting rest values of the sensor, while preserving the dynamics of larger forces. The amplitudes over time of three hand strikes after calibration are shown in Figure 6.

Our calibration method gives a significant improvement in sensitivity over the use of a static threshold alone. Because the cutoff of the highpass filter decreases in frequency with applied force, slow changes at a moderate pressure are preserved. Fast, small changes such as light taps are also preserved. But slow gestures at light pressures are filtered out, a drawback of our current method. By making the sensor itself more mechanically stable, and finding ways to quicken its return to rest from an applied force, we will be able to reduce the effects of the calibration on slow gestures. More details are presented by Jones [2].

Further calibration would be needed to get accurate readings of applied force. However, many acoustic instruments are nonlinear in their responses, so this accuracy may be unnecessary as long as the response is consistent. In a discussion of their own signal-based touch controller, Wessel et al. [11] introduce the useful idea of *haptic regularization,* or linearizing the relationship between intended effort and sensor output.

4. Do It Yourself

A bill of materials for the sensor hardware, a Max/MSP patch that produces calibrated data from the sensor, and all the external objects implemented for this project including source code, are available at http://madronalabs.com/DIY.

5. Further Thoughts

In general, this work can be seen as part of a trend toward smarter computers and dumber sensors, of which the Audio-Input Radio Drum is another example. Just ten years ago, decoding and calibrating our sensor would have required a much more expensive computer. Ten years before that, it would have been impossible to do in real time on commodity hardware. So it makes sense that a great deal of effort has historically gone into the physical design of sensors in order to linearize their responses. Now that significant DSP ability comes at less and less cost, it is becoming possible to generate good data from less perfectly crafted sensors.

Our sensor does not meet all of our design goals, but is a big step toward them from previously available devices, particularly in its sampling rate. Usable in its current form but inviting refinements in both software and hardware, we find our sensor to be a very hackable new tool. We hope that sharing its details here will spur construction, collaboration, and of course, playing music.

6. Acknowledgements

Thanks are due to Adrian Freed, Matt Wright and Lippold Haken for very helpful discussions, and also to Vimeo and Create Digital Music for helping get the word out. See the demo video at http://vimeo.com/2433260. This research supported in part by NSERC and SSHRC Canada.

References

[1] R. Jones and W. Schloss, “Controlling a Physical Model with a 2D Force Matrix,” in *New Interfaces for Musical Expression (NIME)*, pp. 27–30, 2007.

[2] R. Jones, “Intimate Control for Physical Modeling Synthesis,” Master’s thesis, University of Victoria, Canada, 2008.

[3] F. Moore, “The dysfunctions of MIDI,” *Computer Music Journal*, vol. 12, no. 1, pp. 19–28, 1988.

[4] D. Wessel and M. Wright, “Problems and Prospects for Intimate Musical Control of Computers,” *Computer Music Journal*, vol. 26, no. 3, pp. 11–22, 2002.

[5] N. Fletcher and T. Rossing, *The Physics of Musical Instruments*. Springer, 1998.

[6] R. Koehly, D. Curtil, and M. Wanderley, “Paper FSRs and latex/fabric traction sensors: methods for the development of home-made touch sensors,” in *New Interfaces for Musical Expression*, pp. 230–233, 2006.

[7] R. Boie, L. W. Ruedisueli, and E. R. Wagner, “Gesture sensing via capacitive moments,” Tech. Rep. 311401-2099, 311401-2399, AT&T Bell Laboratories, 1989.

[8] B. Nevile, P. Driessen, and W. Schloss, “A new control paradigm: software-based gesture analysis for music,” *Communications, Computers and signal Processing, 2003. PACRIM. 2003 IEEE Pacific Rim Conference on*, vol. 1, 2003.

[9] B. Nevile, “Gesture analysis through a computer’s audio interface: The Audio-Input Drum,” Master’s thesis, University of Victoria, 2007.

[10] L. Baxter, *Capacitive Sensors: Design and Applications*. Institute of Electrical & Electronics Engineers (IEEE), 1997.

[11] D. Wessel, R. Avizienis, A. Freed, and M. Wright, “A Force Sensitive Multi-touch Array Supporting Multiple 2-D Musical Control Structures,” in *New Interfaces for Musical Expression (NIME)*, pp. 41–45, 2007.