A Surface Acoustic Wave Touchscreen-Type Device Using Two Transducers

Mehrdad Ghaziasgar, James Connan
Department of Computer Science
University of the Western Cape, Private Bag X17 Bellville, 7535, South Africa
Telephone: +(27) 21 959-3010, Fax: +(27) 21 959-3006
Email: 2438188@uwc.ac.za, jconnan@uwc.ac.za

***Abstract*—Current wireless human-computer interaction devices such as wireless mice and touchscreens, by-and-large, incorporate a sophisticated electronic architecture. The sophistication achieves wireless capabilities but carries over a cost overhead. In this paper we lay the foundation for developing a novel human-computer interaction device with reduced hardware sophistication. We developed a surface acoustic wave touchscreen-type device using only two transducers, as opposed to, typically, three or more transducers in conventional surface acoustic wave touchscreens. The transducers are mounted on a glass surface and connected into the line-in of a stereo sound card. User-initiated taps are detected, analysed and located on the surface, and the mouse cursor is moved to the computed screen location.**

***Index Terms*—Surface Acoustic Wave, Touchscreen, Transducer**

I. INTRODUCTION

HUMAN-computer interaction has developed into a wide and sophisticated field. In order to tap into the vast power that computers provide, humans have devised various devices, methods and means to pass in commands to computers in order to direct the computing process in the direction of a particular task.

Historically, in directing the computing process, there was a focus on user efficiency and flexibility [1]. The flexibility mentioned mostly applied to the variety of actions that the user was capable of initiating and the efficiency, to the degree of ease in doing so [1]. Thus, using a keyboard and manually rewiring the computer's electric-circuit may be comparable in their flexibility but the efficiency of the former far outweighs that of the latter. The focus group has therefore been programmers, and the goal, increasing their efficiency and flexibility [1].

But as noted in [1], the focus group has greatly shifted to the common user and device design has adjusted itself to satisfy this group. Flexibility and efficiency now aim towards common users as well [1]. Today, devices such as wireless mice and keyboards provide users with more mobility and physical freedom, given there are no tightly constraining wire attachments to these devices. Other devices such as touchscreens allow the user to interact with the computer directly, initiating clicks and other actions by directly touching the computer screen. This, as noted in [2] and [3], provides a sense of personal involvement and immersion, which enhances the user experience. In addition, it is an ideal device for users who are not very familiar with computers, such as the disabled and elderly people [2].

Unfortunately, the advantages that these technologies provide come at the expense of increased hardware complexity and sophistication which, in turn, leads to an increased device cost. In some cases, this cost may become exorbitant to some users. The need, therefore, exists for a device that provides at least some of the efficiency and flexibility of these technologies without carrying over the hardware sophistication or cost. Our work attempts to lay the foundation for such a device.

We proposed the creation of a surface acoustic wave (SAW) touchscreen-type device with reduced complexity. Whereas conventional SAW touchscreens use a multitude of transducers [4] [5], we reduced this number to only two transducers. Additionally, the surface that we used was not modified or specialised in any way, except in its dimensions. Where possible, we shifted the intelligence of the device from the hardware to the software so that we relied more on post-processing intelligence than innovative hardware design.

The two transducers are mounted on a glass surface in a predetermined configuration and connected into the line-in of a stereo sound card. The configuration shall be explained in subsequent sections. As an object taps on the glass surface, geometrical and physical identities as well as the sound information collected on each of the transducers is used to determine the position of the object. This position is then mapped onto a corresponding location on the computer monitor. Subsequently, the mouse cursor is moved to that location.

II. CONVENTIONAL SURFACE ACOUSTIC WAVE TOUCHSCREENS

In this section, we give a brief description of surface acoustic wave touchscreens but we refer the reader to [4] and [5] for further reading. Conventional surface acoustic wave touchscreens are interference-based devices. Refer to Fig. 1, the surface overlay is flooded with acoustic waves of a particular amplitude and frequency in the X and Y directions. To achieve this flooding, one output transducer is mounted for each direction, X and Y, which produces an acoustic wave.

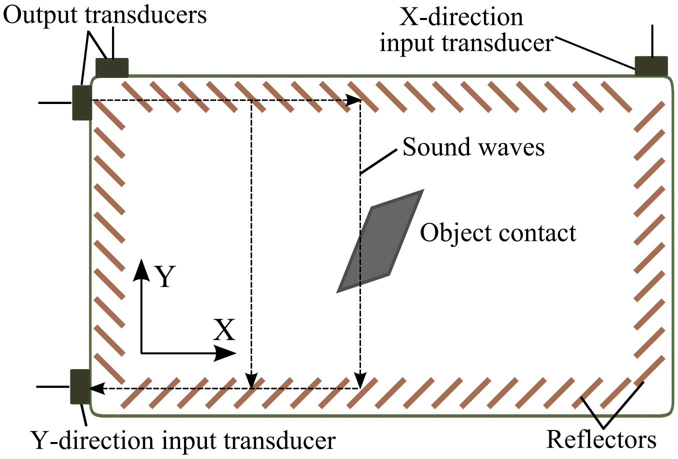

Fig. 1. Schematic of a conventional surface acoustic wave device.

This wave is then reflected off a series of reflectors in its path which redirect the wave in a perpendicular direction. In this way, the original wave is split and flooded onto the overlay. Fig. 1 illustrates the production, reflection off two reflectors and collection of one such wave for the Y direction. Note that the splitting is also done in time so that the first reflector will receive the wave before the second reflector. The second reflector will, in turn, receive the wave before the third, and so on. This is done for both output transducers and both directions.

The redirected waves, in turn, meet with reflectors lined along the bottom and right edges of the panel which again redirect the wave in the perpendicular direction which is then collected by the input transducers, with the X and Y input transducer positioned on the top right and bottom left of the panel respectively. Note, again, that the wave reflected off the first reflector will arrive first, and that reflected off the second reflector, second, and so on. Thus, for example, the first wave that arrives at the Y-direction input transducer corresponds to the horizontal line joining the first top and bottom reflector. If an object was in contact with the surface overlay while the split waves were propagated over the latter, the former absorbs some of the energy of those waves that made contact with it. Other waves remain undistorted. The sound waves received at the input transducers are then analysed to observe any interference that may have occurred, indicating that an object has touched the panel. The position of the touch can then be located in both directions by noting the time when the distorted wave arrived, indicating the x and y coordinates of the touch.

In order to sychronize the transducers and control the operation of the hardware, some commercial versions of the device incorporate a separate external electronic controller [6]. This extra hardware, in addition to the specialized glass used by the device, and the number of transducers typically used, adds extra complexity and cost to the device.

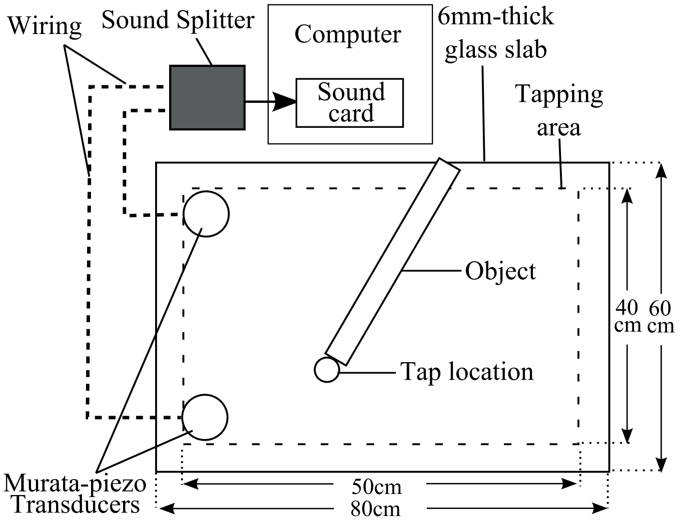

Fig. 2. Illustration of the hardware setup.

III. THE DEVICE

We now describe the setup and design of our device. The device has both a hardware and software component. Data collection is done in the hardware and data processing and subsequent reactions to processed data are done in the software. We describe each component in the subsections below.

A. The hardware component

Referring to Fig. 2, the device consists of a slab of 6mm thick glass, of width and height 80cm and 60cm respectively, onto which is mounted two murata-piezo transducers. No sophistication is employed in the mounting of the transducers, so that they are attached to the glass by means of an adhesive putty, but their positions need to be exactly placed. The top and bottom transducers are placed 15cm from the left edge of the glass slab and 10cm from the top and bottom edges of the slab, respectively. The margin between the transducers and slab edges reduces sound reflections off the slab edges, which cause distortions. The transducers are plugged into the line-in of a stereo sound card, one into the right channel and one into the left by means of a sound splitter. Based on the principle we used, our device is only able to detect taps, and is not able to determine, after the tap, whether or not the object is still in contact with the surface. Thus, the user interacts only by means of taps and not drags or contact-dependent actions. Further, for the scope of this project, only single - not double - taps are considered.

The configuration above therefore creates a system in which the transducer that is plugged into the right and left channel of the sound card, respectively, mark the top-left and bottom-left corner of a rectangular area of width 50cm and height 40cm. This is the area in which the user will be expected to tap - the tapping area. Again, a 15cm bracket has been left between the right edge of the tapping area and right edge of the glass slab.

```
While the system is in the running state
    Read the current sample window from the sound card
    Convert the data of each channel from bytes to samples
    Search for peaks exceeding the noise threshold on each channel
    If such peaks are detected or if there has been an energy carry over on either channel
        If there is no energy carry over on either channel
            Compute the critical breakpoints
        Compute the total energy under the peak of either channel
        If there is no more energy carry over on either channel
            Compute the time difference
            Compute distances
            Compute the location of the object
            Move the cursor at the correct location on the screen
```

Fig. 3. The generic algorithm of the software component.

B. The software component

The software component of the device is the equivalent of the electronic computer board of some surface acoustic wave touchscreens such as the one in [6]. It is, however, a less sophisticated and more flexible alternative to the same, at lower cost. Fig. 3 summarizes the general algorithm design of the software component.

The steps in Fig. 3 can be generalized into five steps. Specifically, these steps are capturing sound data from the sound card, processing the sound data to extract information relevant to determining whether a tap occurred, using the information extracted to determine the location of the tap and moving the mouse cursor to the correct corresponding location on the screen. Moving the mouse to the correct corresponding location on the screen is a simple proportionality problem between the surface and screen-resolution dimensions, and, for reasons of brevity, shall not be explained. We explain other steps in the subsections below.

1) Capturing sound data: The user initiates taps which generate sound waves in the glass surface. These are captured by the transducers which then send this information into the sound card. In the software, the sound information is retrieved by periodically probing the sound card to retrieve sound samples, a process known as sampling. The rate at which the sound card is probed is known as the sampling rate, and for the purpose of this project, we used a sampling rate of 44100 samples per second. Note that the sound card allows us to read in these 44100 samples in parts so that, for example, we may read in a half of them - 22050 - in one iteration of our program and the other half in the next. This technique is used to provide a high sensitivity level to the user. As long as the time to make the iteration is less than $\frac{1}{44100}$ seconds, there will be no information loss. The effects of this information loss shall be explained in subsequent sections. For the purpose of this project, we read the 44100 samples in 5 parts, 8820 samples at a time, in order to provide a high level of sensitivity to the user.

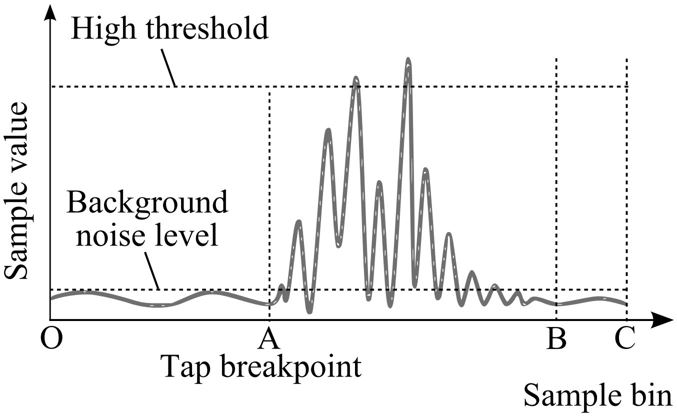

Fig. 4. Graphical representation of a typical tap on one of the transducers.

Also, whilst the sound information arriving from the transducer is of an analog nature with a precise voltage value, when this is converted to digital information for use with software, a limited level of precision can be achieved. This precision is specified by the number of bits that shall be used to represent each converted sample. For the purpose of this project, we used 16 bits - 2 bytes - to represent a single sample. We found that this level of precision was high enough to provide sufficient information and, at the same time, low enough to ensure fast processing times. [7], [8] and [9] will provide further information on the sample rate and bits per sample variables.

2) Extracting information to detect a tap: Fig. 4 is a graphical representation of a typical tap, as recorded by one of the transducers. The values of the 44100 samples on the horizontal scale - each of which shall be referred to, hereonforth, as a sample bin - are plotted against the sample value on the vertical scale. Note that the time between sample number 1 - point O on the graph - and sample number 44100 - point C on the graph - is 1 second.

In order to determine whether a tap has occurred, a threshold sample value, which we shall hereonforth refer to as the high threshold, is set, the exceeding of which will indicate that a tap has occurred. Thus, if the sample set is looped through and a sample value is found which exceeds the high threshold, the system will conclude that a tap has occurred on that transducer.

However, in order to filter out noise and events such as someone hitting one of the transducers by mistake, we further constrain the criterion for a valid tap to occurrences when the sample set of both transducers exceed the high threshold. This has been found to be highly effective for noise filtering, making the device highly robust to noise.

3) Extracting information to determine tap location: Two sources of information within the sound data were used for this project: the total energy of the tap peak at each transducer and the time difference between the times when the tap peak reached each of the transducers. We shall explain how each of these were extracted in this subsection and how they are used to determine the location of the tap in subsequent subsections.

We have divided the horizontal scale of Fig. 4 into three distinct regions. Regions OA and BC are background noise picked up by the transducer before and after the tap peak occurred. The point A is the exact point at which the tap sound waves reached the transducer and a significant disturbance was observed by the transducer in the region AB. Region AB is therefore the tap peak.

The total energy of the tap peak is equal to the area under the graph in region AB. The point A is the exact time at which the tap peak arrived at the transducer. We developed an algorithm that is able to extract both the total energy and the arrival time at the transducer. Note that we shall label the left- and right-channel transducers, respectively, as 1 and 2 from hereonforth. The algorithm is comprised of the following computational steps:

1) Compute the derivative of the sample set by means of a finite difference approximation [10]. For any two sample values in the sample set S_i and S_{i+1}, the derivative, D_i of these values is given by:
$$D_i = S_{i+1} - S_i \tag{1}$$
2) Loop through the derivative set and find the maximum value. We shall hereonforth denote this by the symbol D_m.
3) Loop through the derivative set from bin 1 upto bin $\frac{1}{3}D_m$ and determine the value whose absolute value is highest in this region. This was done in order to determine the highest background-noise value, which we shall hereonforth refer to as the noise threshold.
4) Loop through the derivative set from bin $\frac{1}{3}D_m + 1$ onwards and find the first bin that just exceeds the noise threshold. This is the bin and exact time at which the tap peak has just reached the transducer, equivalent to point A in Fig. 2. We shall refer to this point, hereonforth, as the tap breakpoint and denote it by the time symbol t_x where the subscript x refers to the transducer on which the observation was made.
5) Compute the cumulative sum of the sample values of the sample set from bin $\frac{1}{3}D_m + 1$ and spanning 1200 bins which is taken to be the total energy of the wave. Whilst there was no exact way of determining where the tap peak ended, we found that taking an exact bin length of 1200 bins sufficed to give a reasonable approximation of the same.

   Note that in some cases the tap peak may extend over two sample sets - a situation which we shall hereonforth refer to as an energy carry over - such as in cases where the tap breakpoint occurs in the latter bins of a sample set. We accomodated for such cases by testing for occurrence of these energy carry overs and accumulating the energy over both sample sets if necessary.

The above process is repeated for both transducers simultaneously in real-time. At this point we have the total energies E_1 and E_2 of the transducers as well as the times t_1 and t_2 at which they occurred. We, however, are only interested in the time difference, as has already been explained, which is obtainable by taking the difference of times t_1 and t_2. We shall denote the time difference by the symbol Δt.

4) Computing the tap location: We required a physical model that would relate the sample values in region AB to the location of the object. We shall explain the model that we developed and used.

The model that we made use of makes use of the fact that the total energy E of a sound wave travelling through a solid surface is inversely proportional to the distance d which the wave has travelled, with the constant of proportionality k. This is expressed mathematically as:

$$\begin{aligned} E &\propto \frac{1}{d} \\ E &= \frac{k}{d} \end{aligned} \tag{2}$$

Thus, if the energy of the sound wave at a point is measured, the distance that the wave has travelled can be determined. The constant k needs to be determined experimentally for each transducer. This is done by plotting the graph of total energies versus the reciprocal of their corresponding distances for a set of distances and determining the slope of the line fitted to the graph by means of a linear least squares regression technique.

Applying equation 2 to each of the transducers we arrive at an energy-distance relation for the transducers:

$$\begin{aligned} E_1 &= \frac{k_1}{d_1} \\ E_2 &= \frac{k_2}{d_2} \end{aligned} \tag{3}$$

Taking the ratio of the energy-distance relations for the transducer 1 to transducer 2 yields:

$$\begin{aligned} \frac{E_1}{E_2} &= \frac{k_1}{k_2}\frac{d_2}{k_1} \\ d_2 &= \frac{k_2}{k_1}\frac{E_1}{E_2}d_1 \end{aligned} \tag{4}$$

Equation 4 is an equation of two unknowns d_1 and d_2. We shall use the time difference to develop another equation in terms of d_1 and d_2 in order to be able to solve it. The time taken for a sound wave of speed S to travel a distance d through a solid is given by:

$$t = \frac{d}{S} \tag{5}$$

Applying equation 5 to each of the transducers yields:

$$\begin{aligned} t_1 &= \frac{d_1}{S} \\ t_2 &= \frac{d_2}{S} \end{aligned} \tag{6}$$

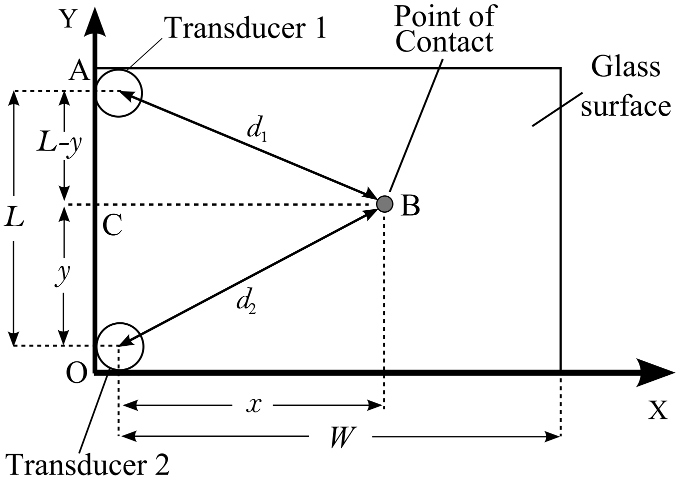

Fig. 5. The geometrical setup and coordinate system used.

Taking the difference between t_1 and t_2 yields the time-difference equation, which we then rearrange to obtain an equation in terms of the time difference and d_1 and d_2:

$$\begin{aligned} t_2 - t_1 = \Delta t &= \frac{d_2 - d_1}{S} \\ S\Delta t &= d_2 - d_1 \\ d_2 &= S\Delta t + d_1 \end{aligned} \tag{7}$$

Solving equations 4 and 7 simultaneously and eliminating d_2 yields the following solution for d_1:

$$d_1 = \frac{S\Delta t}{\frac{k_2}{k_1}\frac{E_1}{E_2} - 1} \tag{8}$$

We can solve for d_1 in this equation and use it in equation 7 to solve for d_2. We now have the distance of the location where the tap occurred from each transducer. The task remains to use the distances d_1 and d_2 to find the coordinates of the tap location on the surface, x and y. We represent the geometrical setup and our choice of coordinate system in Fig. 4.

We have introduced two variables in Fig. 5. L denotes the distance between the transducers which is also the height of the surface and W denotes the width of the surface. It can be seen in Fig. 5 that two right-angled triangles ABC and OBC, pertaining to transducers 1 and 2 respectively, are formed. Applying Pythagoras' theorem to each of these triangles yields two equations containing the required x- and y-coordinates, which can be solved simultaneously to yield a solution for the y-coorindate of the tap location as follows:

$$\begin{aligned} {d_1}^2 &= x^2 + (L - y)^2 \\ {d_2}^2 &= x^2 + y^2 \\ y &= \frac{{d_2}^2 + L^2 - {d_1}^2}{2L} \end{aligned} \tag{9}$$

Substituting the solution for y in one of the other equations in equation set 9 then yields a solution for the x-coordinate as well. The tap location has been determined.

IV. DEVICE TESTING

The device testing aimed to evaluate the usability of the device. Testing was therefore done from a user perspective and focused on the user's opinion on how well the device was performing [11]. Five usability requirements were evaluated. The usability requirements and their evaluation were as follows:

1) Test 1: System response rate. The system was expected to respond to every tap the user initiated. To test this, the user initiated sets of 100 taps and noted down the number of taps the system did not respond to.
2) Test 2: System response time. The system was expected to respond to taps in a minimal amount of time. To test this, the user initiated sets of 100 taps and noted down, for each tap, whether or not a lag had occurred. Taps which received no response at all were not recorded as part of this test, and were part of test 1.
3) Test 3: System sensitivity. The system was expected to respond to both average and lower intensity taps so that even light taps receive a response. To test this, the user initiated sets of 100 taps of low intensity and sets of 100 taps of average intensity and noted down the number of taps of each category that registered no system response.
4) Test 4: Noise robustness. The system was expected to be robust to background noise and only respond to valid user taps. To test this, the device was placed idly, for two hours at a time, in a high noise environment - a busy office - and a low noise environment - a quiet computer lab - and was configured to record the number of times it registered what it deemed to be a valid tap.
5) Test 5: Placement accuracy. The device was expected to detect the location of the tap with a high accuracy. Having noted the limitations of the system, it was decided that if the computed screen location fell within a 30x30 pixel area of the actual location, this would be deemed a high enough accuracy. This choice was derived from the fact that a 30x30 pixel area maps well onto an icon and in order to use the device, a user should at least be able to click an icon. To test this, the user initiated sets of 100 taps and noted down the number of taps that fell within the high-accuracy region of that tap.

V. TEST RESULTS

TABLE I
RESULTS FOR TEST 1 - SYSTEM RESPONSE RATE

Observation type	Percentage of taps (%)
Response observed	90
No response observed	10

TABLE II
RESULTS FOR TEST 2 - SYSTEM RESPONSE TIME

Observation type	Percentage of taps (%)
Noticeable lag	3
No noticeable lag	97

TABLE III
RESULTS FOR TEST 3 - SYSTEM SENSITIVITY

Observation type	Percentage of taps (%)	
	Average intensity	Low intensity
Response observed	93	48
No response observed	7	42

TABLE IV
RESULTS FOR TEST 4 - NOISE ROBUSTNESS

Noise category	Number of taps detected
Low	0
High	3

TABLE V
RESULTS FOR TEST 5 - PLACEMENT ACCURACY

Accuracy	Percentage of taps (%)
Low	64
High	36

VI. DISCUSSION

1) Test 1: It was found that the device provided a high response rate and only 10% of taps were completely ignored by the system.
2) Test 2: It was found that the device provided a fast response time, with only 3% of taps displaying any noticeable lag to the user.
3) Test 3: The system was also found to have a very high sensitivity for average intensity taps, responding to all but 7% of these taps, but only responded to 48% of the low intensity taps.
4) Test 4: It is clear from the results that noise robustness is a great strength of the device. Under low noise conditions, the device did not pick up any noise at all and under high noise conditions, it only picked up noise three times.
5) Test 5: It was found that the system is able to achieve a high accuracy for 36% of taps. This accuracy, while not ideal, is acceptable for the time being given the time constraints that were in place for the project. We shall discuss plans to increase this accuracy in the next section.

VII. FUTURE WORK

The device is not, at this stage, able to perform to a usable level of accuracy. The main reason for the lack of accuracy is the simplicity of the physical model employed at this stage. This project, being a feasibility and exploratory study, employed the simplest physical model to compute the position of the tap. The model can, however, easily be improved, improving the accuracy of the device. Improvements will include better sources of information extracted from the wave and accounting for wave reflections off the edges of the slab.

A self-calibration feature will further enhance the performance and application of the device. It is intended to present the user with a set of tasks which will systematically calibrate the device to varied tapping area dimensions, slab materials and slab thicknesses. Tasks will involve tapping a set number of times at particular locations, such as at the midpoint between or the right next to the transducers, giving an indication of the distance between the transducers. Such a mechanism will allow the device to, among other things, transform an ordinary monitor screen into a touchscreen-type device. Many other applications exist.

Other areas worth exploring include using sound cards with more channels or using two stereo sound cards in a single computer. This, however, will be subject to cost evaluations in order to maintain the reduced cost and complexity rationale of the project.

VIII. CONCLUSION

In this paper we have described how we set the foundations for the construction of a surface acoustic wave-type device using only two transducers and a non-specific slab of glass. We described our hardware setup which collects sound data, as well as the software component which processes the sound data and positions the mouse cursor at the computed location on the screen. We described the testing done on the device which revealed that, currently, the device cannot achieve a commercially applicable accuracy. We have, however, also described how we intend to increase the accuracy of the device in our future work, as well as making other improvements that focus on device flexibility and portability. The foundations have been set for a powerful, low-cost touchscreen-type device.

REFERENCES

[1] J. Grudin, ”The Computer Reaches Out: The Historical Continuity of Interface Design”, in Proc. CHI ’90, Seattle, Washington, United States, Apr. 1990, pp. 261-268.
[2] A. Holzinger, ”Finger Instead of Mouse: Touch Screens as a Means of Enhancing Universal Access”, in Proc. 7th ERCIM Workshop on User Interfaces for All, Paris, France, 2002.
[3] M. A. Srinivasan and C. Basdogan, ”Haptics in Virtual Environments: Taxonomy, Research Status, and Challenges”, Computers & Graphics, vol. 21, no. 4, pp. 393-404, 1997.
[4] R. A. Quinnell, ”Touchscreen Technology”, EDN, Nov. 1995.
[5] B. G. Liptak, ”Instrument Engineers’ Handbook: Process Control and Optimization”, CRC Press, 2005, pp. 845-848.
[6] (2008, May). Acoustic Pulse Recognition (APR) [Online]. Available: http://media.elotouch.com/pdfs/marcom/apr_wp.pdf
[7] A. Holzinger, ”Multimedia Basics-Technology: V.1”, Firewall Media, 2005.
[8] A. Davidson, ”Killer Game Programming in Java”, O’Reilly, 2005, pp. 249-254.
[9] A. Marks, ”The Complete Guide to Game Audio: For Composers, Musicians, Sound Designers, and Game Developers”, Focal Press, 2001, pp. 175-179.
[10] D. Hughes-Hallett, A. M. Gleason and D. Flath, ”Calculus”, Wiley, 1997, pp. 474.
[11] M. van Welie, G. C. van der Veer and A. Eliens, ”Breaking Down Usability”, in Proc. Interact ’99, Edinburgh, Scotland, 1999.

Mehrdad Ghaziasgar is currently an M.Sc student at the University of the Western Cape. He is currently doing research on sign language synthesis and novel communication applications on mobile interfaces for the Deaf and hard of hearing.

James Connan heads up the South African Sign Language (SASL) research group. He has a wide range of interests that include databases, computer Vision and machine learning.