Relative Position Tracking for User Interface Controls Using Doppler Shifted Ultrasound

Justin Bradfield and Alex Mudd

Cornell College Phy312, November 2012, Professor Derin Sherman

Introduction

As computers and technology become a more integral part of our life it becomes advantageous to develop more intuitive ways of interacting with them. There are many technologies in the field of computer-human interaction. They range from mice and keyboards, to touchscreens and all the way to voice control. We took inspiration for our project from a recent Microsoft research paper that uses ultrasound and Doppler shift to add some touch-less gestures for application control [1]. In the project done by Microsoft, users had one dimension of control: movement directly toward the computer, for application gestures. Our goal was to create a proof of concept for a three dimensional control system that uses Doppler shift to track motion with minimal transmitters and receivers. The overall goal for our gesture control systems is for it to be implemented on a laptop or desktop computer with minimal hardware addition.

Inspiration for Ultrasound Gesture control

In this section we'll focus on technologies that use ultrasound and the Doppler shift for gesture control. These include technologies like the product created by Elliptic Labs that created a software gesture suite using many ultrasound transmitters and receivers [2], and technologies such as the research done by Microsoft which use one emitter and receiver [1].

According to the Microsoft research paper normal computer speakers outputting anywhere from 18 to 22 KHz as well as a microphone can be used to detect sound reflected from the emitters. By looking at Doppler shifted sound motion perpendicular to the screen of the computer (see Methods). They recognize gestures based on this information by examining broadening in the FFT (see Methods). They also analyze the length of time these broadenings occur to filter out movements that are too long or short to be gestures. The benefits of this approach are that they can recognize accurately various gestures and require no additional hardware changes to a computer to run. The drawbacks include only measure gestures in one dimension, some people may hear the sound, and some hardware receivers cannot collect data on that high of frequencies [1]. This suggests that adding additional hardware may be a more effective approach for creating a robust gesture system.

EllipticLabs is a company that has recently produced a software suite for touchless gesture controls using Doppler shift. The software currently runs on Windows 8 operating systems and provides very robust gesture controls. Although they don't cover their methods in detail it appears that they use a similar method to that described in the Microsoft paper while using time of flight for ranging and leading edge detection. The reason that their control system has more features and gesture recognition comes from hardware implementation. According to their website the system uses multiple emitters, and 6-8 receivers. Even with all of this hardware overhead they claim to use far less energy than similar camera based gesture recognition systems [2].

Other Technologies for Gesture Recognition

You may have seen movies such as *Minority Report* that show touchless gestures and touchless computer control. Some of these movies were made ten years ago and displayed this technology as cutting edge and futuristic. Today there are many technologies that can be used to capture touchless gestures. Some of these systems use infrared cameras and triangulation such as the Xbox Kinect by Microsoft; others like the MGC3130 chip can detect motion based on disturbances in the electric field created by the screen of a computer [3].

Range finding cameras like that in the Xbox 360 generally use infrared emitters and detectors to map all ranges in target area. There are many techniques used to find depth, but the most common one is triangulation. This is accomplished using one sensor emitting infrared light. This light then bounces off of an object and into a sensor which is placed a known distance apart from the emitter. Analyzing where the light hits the sensor gives an angle of reflection, from this the sensor can deduce the distance away the object is. This method is great for generating depth maps of a large area, and can be used to get gestures; however, it provides far more data than needed, and can be expensive. The extra data created by these technologies can be used in various ways and are susceptible to privacy invasion. According to NewsScientist there are patents out for Kinect software that monitor facial expressions to find how people react to certain advertisements [4].

The use of electric fields to track motion and provide gesture control is a very new and exciting technology. A new e-field detector microchip was unveiled November 2012, as we were beginning implementation of our project. This detector consists of 5 electrodes. One large central electrode is placed in the middle of the other four electrodes, which gather data about disturbances in the electric field. Close analysis of this data lets them separate different movements; such as, circular finger motions, side swipes, up and down swipes, flicks and more. This technology is relatively cheap, and the data is not susceptible to privacy invasion. The only drawback is that the gestures must be made very close to the sensor [3].

Theory

Ultrasonic sound, sound over 20Khz, as a distance measuring tool is by no means novel; however, new uses for ultra sound detection continue to appear all the time and include such things as detecting fish and creating 3-D maps of buildings using robots. The basic principle behind ultra sound mapping is the Doppler Effect. This phenomenon, explained by Christian Doppler in 1842 [5], states that frequency of a wave is shifted based on the difference in velocity between the observer and emitter. This shift in frequency follows the following relationship:

where c is the speed of sound.

While Doppler shift of ultrasound is the backbone of our experiment, we still need a way to measure the shift. In order to do this we used FFTs, or Fast Fourier Transforms. A Fourier Transform is a way of breaking down all waves into their constituent sine waves. All waveforms can be represented by the linear combination of the set of all sine waves. We used LabView's built in FFT library and therefore will not be going over the specifics of this algorithm. It suffices to say that using the built in functions we could depict all frequencies the receiver detected and their amplitudes.

In our project we used aliasing to shift our received signal down to a more manageable range. This decreased the amount of extra data we had to sort through, and allowed our computer to sample at slower rates. To explain aliasing I'll start at explaining on some basics of sampling data.

When a computer or device reads in data, when it goes through an analog to digital converter, it has to be sampled at a certain rate. Sample is the simple act of detecting the voltage at a certain time. How fast a signal is sampled to view it accurately is called its Nyquist frequency. Nyquist states that the frequency that can be observed accurately = sample rate / 2. If a faster frequency trying to be detected it will fold back into the range of sample rate / 2. This is called aliasing. In our project we use aliasing to fold the ultrasound frequency into a range from 0 to 3000Hz.

Method

Our experiment breaks down into four main steps. They are as follows: accurate signal generation and amplification, the emission and detection of sound waves, the amplification and filtration of the detected sound waves, and the processing of the data gathered by the computer.

1. Accurate signal generation

Outside of the LabView program designed to process the data, generating the signals to be emitted was the most complex part of the circuit. This portion of the experiment takes the input from a quartz oscillator and sends it through three different counters to divide the signal down to usable values, which were then both emitted through the transducers in the second stage of the experiment, and sent to the Data Acquisition Board (DAQ Board) to be used as a reference.

The transducers we used in this experiment were only able to emit and detect frequencies of about 25 ± 1 kHz, leaving us with a small window of frequencies to use for multiple dimensions. We had the option of using speakers and microphones that worked at a higher range around 40 kHz, but we decided to go with the lower range to meet our goal of computer implementation with minimal additional hardware, as this would be near the range in which computer standard computer hardware can emit and detect.

Based on the equation for the Doppler effect (explained in the theory section), we were able to calculate values for frequency that would allow us to accurately detect distinct Doppler shifts from speeds up to roughly 3 m/s (creating a Doppler shift of about 227 Hz). Since the sound waves were bounced off an object moving relative to both the emitter and receiver (which were stationary), the shift from the reference frequency would be detected at double the frequency as it would as dictated by the equation above. A hand velocity of 1.5 m/s, which is an acceptable speed for fine control of a system, would create a Doppler shift of roughly 225 Hz, meaning that frequencies that were separated by about 450 Hz would create detectable Doppler shifts that did not overlap. Conveniently, the quartz oscillator (frequency 1.2288 MHz) we used could be divided down to frequencies of 24.5 kHz (divided by 50), 25.08 kHz (divide by 49), and 25.6 kHz (divide by 48). The circuits designed to output the desired frequencies are shown in Figures 1 through 3.

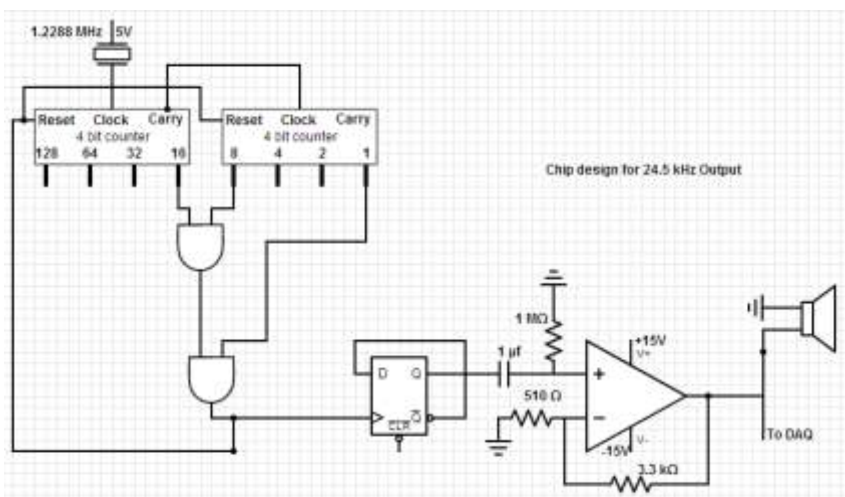

Figure 1: Schematic of the circuit designed to output the low frequency, 24.5 kHz.

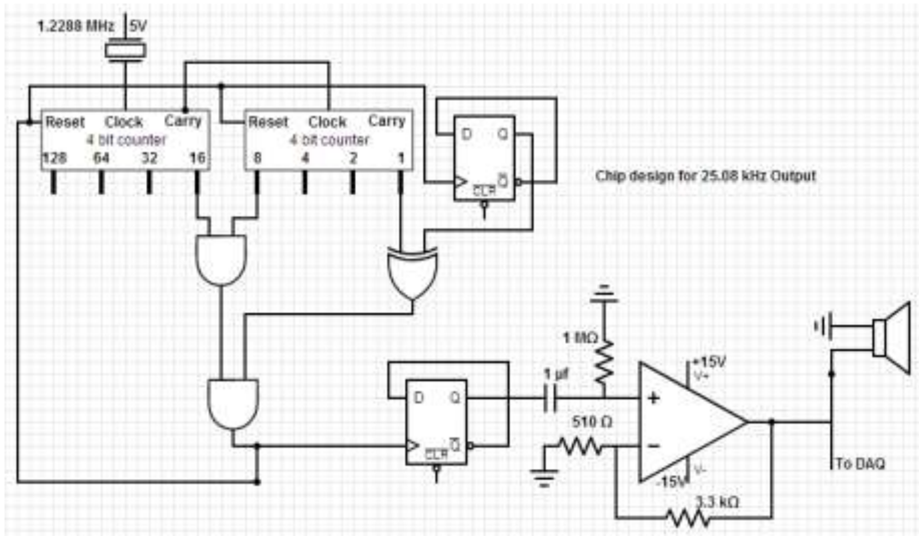

Figure 2: Schematic of the circuit designed to output the middle frequency, 25.08 kHz.

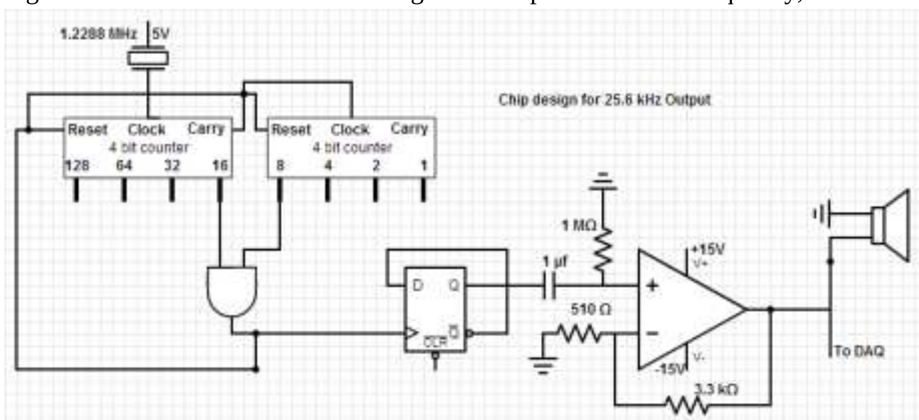

Figure 3: Schematic of the circuit designed to output the high frequency, 25.6 kHz.

2. Emission and detection of sound waves

The layout we decided to use for our transducers is shown in Figure 4. The outer three transducers are used as emitters, with the center transducer used as a receiver. Signal reflection from surfaces is discernible from background noise at around 1.5 meters, with the optimal range of the system between 1.25 and .5 meters. This set up was chosen as it would work well for two-dimension gesture recognition, as it would be easy to determine what relative direction from the receiver the hand was moving, as the signals from the emitters nearest the hand would not only be detected stronger than those further away, but also register a stronger Doppler shift. Movement towards and away from the set up could be detected based on the Doppler shift of all three signals.

However, this set up would not work well for accurate three dimensional mapping. While we'd be able to detect motion, it would be difficult to determine how fast the object was moving. The best organization of transducers for our set up to accurately track three dimensional movement would be to have the receiver at the origin and one emitter on each standard axis.

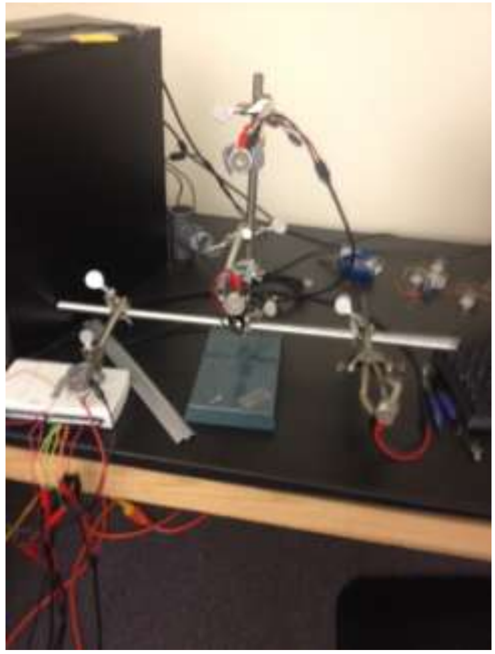

Figure 4: The transducer set up used for emitting and detecting sound waves. The receiver is in the center.

3. **Amplification and filtration of detected sound waves**

After the transducer receives the signal, it is amplified ten times using an instrumentation amp, and then sent through a high pass filter designed to attenuate frequencies below 10 kHz. This reduces a large amount of the extraneous noise being picked up by the transducer. After being filtered, the signal is then sent through a non-inverting op amp with a gain of 20. Most signals picked up reflected from distances of less than a quarter meter end up hitting the rails, being amplified over 15V, after this amplifier, thus the reason for the minimum distance on the optimal range. The design of this section is shown in Figure 5.

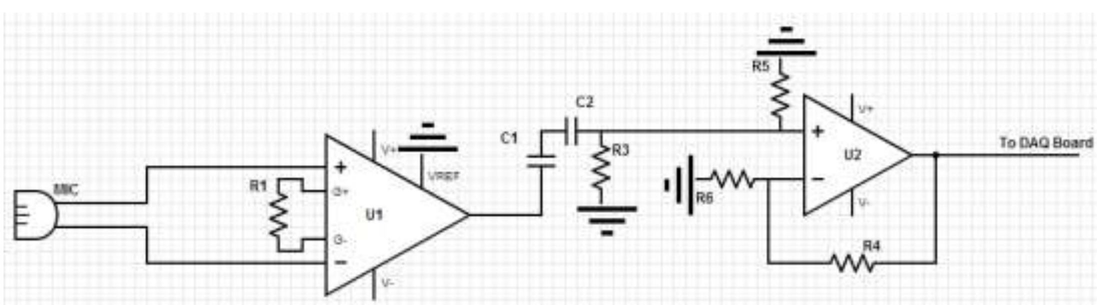

Figure 5: Schematic of the circuit designed to detect, amplify and filter the detected signals.

4. **Processing data by the computer**

In the DAQ board, the signals are aliased down from 24.5 kHz - 25.6 kHz to roughly .56 kHz – 1.59 kHz by reducing the sample rate to 6 kHz. This reduces the amount of data the computer has to process. The reference signals and the received signal are sent through Fourier

Transformations and then peak detectors, returning the frequencies of each reference and each detected peak in the received signal

The peak values of the received signal are then sorted and compared to the reference frequencies to determine which frequency each is most likely Doppler shifted away from. If there is at least one peak in the received signal for each reference frequency, then they are all processed to determine the velocity causing the detected shift. If that velocity is determined to be valid by manually set limits, it is then passed on with the loop time to be integrated to determine the position of the detected object relative to its starting point. As the project currently stands, relative position is only calculated perpendicular to the transducer array.

Results

In this experiment we found that we can accurately measure position in one dimension using the Doppler shift. In various trails we were able to detect objects from 1 meter to .5 meters at a low and somewhat consistent rate. These results were accurate to about 5 centimeters. Since we only needed relative position we accepted this data as "accurate enough".

In two dimensions we set up the receivers and emitters a few inches apart, one on the left, and one on the right. We saw promising Doppler shifts when we moved our hands horizontal to the receiver, and shifts that were similar in when we moved objects straight towards the receiver. Without changing our position algorithm we were able to detect which receiver our hand had moved toward or away from. At this point in time we were able to accurately detect left and right movements, but we did not attempt to test for accurate two dimensional position tracking.

Toward the end of our project we were able to create a set up for three dimensions that used three sensors. We were able to split up our emitters into 3 frequencies that were space approximately 500Hz apart. We were able to separate these emitter frequencies, and detect Doppler shifts up to 250Hz away from each frequency. This data was mostly qualitative and found by analyzing FFT graphs.

Improvements and Long Term Goals

There are many ways this project could be taken further. For example, we could improve the accuracy of the measurements with a sharp band pass filter. One of our goals was to use primarily open source software, and, in order to meet this we could move data acquisition and processing out of LabView and instead have the data be read in by an Arduino chip and processed via code written in ideally either Processing or C++. This would open up a wide range of possible goals, such as optimizing the circuit to run off of 5V so it could be powered by a USB port, as well as the possibility of having the computer output the signals instead of dividing them down from a quartz oscillator on an exterior protoboard. Additionally, we could add some software integration for the project in order to utilize the gesture detection.

Conclusion

The goal of this project was to create a proof of concept for a three dimensional gesture mapping system that used minimal hardware. Although our hardware was far from minimal, we showed that we could use ultrasound receivers and detectors with sharp resonance to send multiple channels of signal, and detect Doppler shifts of 250Hz from three separate emitters. It would seem that we have a fair ways to go to both with software and hardware implementation to meet our goals. We do, however, believe we have come up with a final product that has minimal transmitters and receivers and is capable of robust three dimensional gesture controls.

References

1 Sidhant Gupta, Dan Morris, Shwetak N Patel, Desney Tan, *SoundWave: Using the Doppler Effect to Sense Gesutres*, Microsoft Research, University of Washington, 2012

2 http://www.ellipticlabs.com/, Elliptic Labs , 11/2012

3 Stephen Evanczuk, “Microchip elevates HMI design with world’s first e-field-based 3D gesture-recognition chip” , http://www.edn.com/electronics-products/electronic-product-reviews/other/4401197/Microchip-elevates-HMI-design-with-world-s-first-e-field-based-3D-gesture-recognition-chip?cid=EDNToday, 2012

4 Jacob Aron, Kinect to watch your emotions and serve up ads, http://www.newscientist.com/blogs/onepercent/2012/06/kinect-to-watch-your-emotions.html, NewScientist.com,11/6/2012

5 Doppler Effect, http://en.wikipedia.org/wiki/Doppler_effect, Wikipedia, 17/11/2012