CONTROLLING PC USING ULTRASONIC SENSORS

Srirama Dheeraj[1], R.Chandu[2], CH.V.N.Anirudh[3], M.Pavan[4], Narendran M[5]

[1] *Student, Computer Science, SRM Institute of Science and Technology, Tamil Nadu, India*
[2] *Student, Computer Science, SRM Institute of Science and Technology, Tamil Nadu, India*
[3] *Student, Computer Science, SRM Institute of Science and Technology, Tamil Nadu, India*
[4] *Student, Computer Science, SRM Institute of Science and Technology, Tamil Nadu, India*
[5]*Assistant Professor, Computer Science, SRM Institute of Science and Technology, Tamil Nadu, India*

Abstract

The aim of this project is to build a machine which can control pc using ultrasonic sensors. In this we are using Arduino to interact between ultrasonic senor and PC. The Arduino can be connected to the PC/Laptop for powering the module and also for Serial communication. Once the connections are done place them on your monitor as shown below. The concept behind the project is very simple. We will place two Ultrasonic (US) sensors on top of our monitor and will read the distance between the monitor and our hand using Arduino, based on this value of distance we will perform certain actions.

1. INTRODUCTION

In order to control pc using ultra sonic sensors, this technique is called Leap motion which enables us to control certain functions on our computer/Laptop by simply waving our hand in front of it. It is very cool and fun to do it, but these laptops are really priced very high. So in this project let us try building our own Gesture Control Laptop/Computer by combining the Power of Arduino and Python. We will use two Ultrasonic sensors to determine the position of our hand and control a media player (VLC) based on the position. I have used this for demonstration, but once you have understood the project, you can do anything by just changing few lines of code and control your favourite application in your favourite way.

The concept behind the project is very simple. We will place two Ultrasonic sensors on top of our monitor and will read the distance between the monitor and our hand using Arduino, based on this value of distance we will perform certain actions. To perform actions on our computer we use Python Pyautogui library. The commands from Arduino are sent to the computer through serial port. This data will be then read by python which is running on the computer and based on the read data an action will be performed.

The incoming time-domain signal are buffered, and Fourier transform is applied on them. The Arduino can be connected to the PC/Laptop for powering the module and also for Serial communication. The result of this operation is magnitude vectors that are spread equally over the spectral width. After each FFT vector is computed, it is further processed to determine the bandwidth of the signals, speed of gestures and motion detection. The detected motions are then converted to pc commands.

2. CIRCUIT DIAGRAM

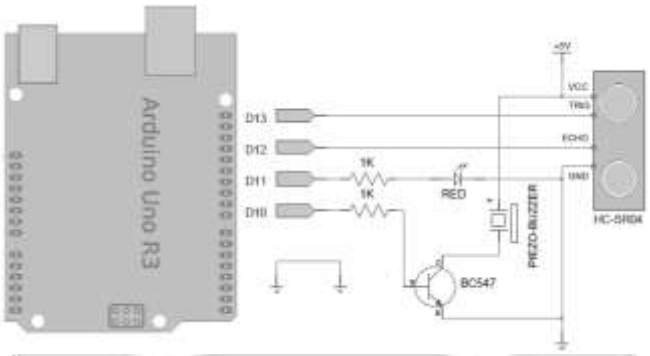

Fig 1: Components in the hardware setup

In order for the ultrasonic sensors to produce differentiable delay times and make triangulation calculations possible, they need to be at some distance apart from each other. Our original idea was to mount the sensors on a frame that would fit onto a computer screen. However, that proved to be time consuming to set up, and would require a very flexible and complex frame in order for the project to work on screens of different sizes. We decided to mount the sensors on a frame instead. A square board was cut out from a poster board, and the sensors were hot-glued into wedges at the top-left, top-right, and bottom left corners of the board. To make the frame stand upright, we cut out a piece of the bottom centre and wedged a rectangular piece of foam in it, forming a T shape at the base so the frame would not fall. The sensor legs are plugged into pin headers, which are soldered onto wires that lead to the rest of the circuit. This way in case of a failure of the sensors, we can easily swap them out.

The following are the gestures and corresponding functions that are executed in the current project:

Gesture Motion	**Function**
Placing both hands	Play/Pause
Right hand Movement	Fast Forward
Left hand Towards Movement	Volume increase/ Decrease
Left hand Movement	Rewind

Table 1: Gesture and function chart

Initial state is the Fill State. The peaks (local highs above a predefined threshold) in the ADC samples fill the buffer until full. Check state is where the local highs are analysed to determine if the motion is one of the six gestures predefined in the database. The total number of peaks that dropped from HIGH to LOW, or number of peaks that rose from LOW to HIGH are counted. Wait Push is transited to from Check state if there is any gesture in the sample data collected. Wait None is transited to from Check State if there was no gesture. Find Peak fills a buffer with one peak, as against the Fill state where the buffer is completely filled with local HIGHs. A square board was cut out from a poster board, and the sensors were hot-glued into wedges at the top-left, top-right, and bottom left corners of the board. To make the frame stand upright, we cut out a piece of the bottom

centre and wedged a rectangular piece of foam in it, forming a T shape at the base so the frame would not fall This separate state is required because the buffer will not need to be completely refilled even if no gesture was detected. The gestures determined and the corresponding pc actions are displayed on the Liquid Crystal Display, simultaneously as the commands are given to the PC.

3. HARDWARE CONSIDERATION

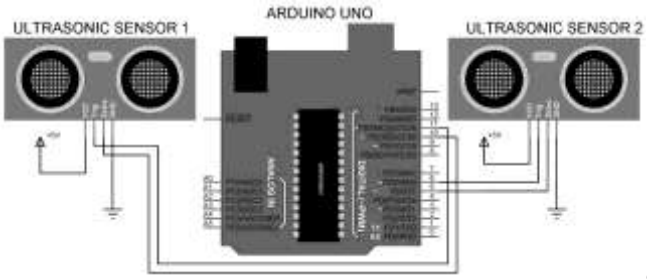

Fig 2: Schematic diagram for hand gesture controlled pc movement

The above circuit comprises of ATMEGA128 microcontroller, HC-SR04 Ultrasonic sensor, two motor driver ICs L293, and four DC motors. The sensor has on-board transmitter and receiver. The sensor's TRIG and ECHO pins are connected to the microcontroller's ADC port. The motor driver is connected to the other input/output port.

The microcontroller gives a logic HIGH at the pin connected to TRIG of sensor at regular intervals. The ultrasonic waves are emitted and if there is any object obstructing these signals, then the ultrasonic waves are reflected back. The receiver in the ultrasonic module receives the frequency shifted signal at the pin connected to ECHO of sensor. These received waves are buffered and stored in the microcontroller. The microcontroller analyses the signal for a pattern. If the pattern matches a gesture in the database, then it further determines the command associated with the gesture. The command is given in the form of logic HIGH or LOW to the pins of the motor driver L293D. These commands correspond to the movements.

To interact with ultrasonic sensors and Arduino we will be using python and modules. Depending on the movement of hand either from left to right or right to left, the sensors are triggered and returns the true or false values in the loop with respect to each sensor at its range of 20cm. Two gesture recognition functions are defined for the forward and backward movement of the images from the directory. If the user moves hand starting from left sensor toward right sensor (in the backward direction), the Arduino sends a 'B' slide command to the serial port. Similarly, if gesture is read by the sensors from right to left (in the forward direction), the Arduino sends an 'F' slide command to the serial port.

4. APPLICATION AREAS

Gesture recognition is useful in processing information from human beings that is not conveyed through speech or other methods. This technology is useful in following areas:

a. Immersive gaming technology: Gestures may be used to control interactions with the gaming console and give a more interactive and immersive experience. Sign Language interpretation: Gesture recognition can be used to transcribe signs into text, just like speech recognition. This would be greatly helpful for the speech impaired.

b. Control through facial gestures: This technology can be used for applications with even more precision like recognizing face gestures. This will be helpful in situations when users cannot use other input interfaces like mouse or keyboard or even hand gestures. This would be additionally helpful in applications like mood sensing.

c. Alternative computer interfaces: Strong gesture recognition can be used to accomplish common tasks performed traditionally with the current input devices such as mouse or keyboard. Gestures, along with

other methodologies like speech recognition can be made to control the electronic appliances and gadgets completely or with little need to type or touch. [4] [5] [6]

d. Remote control: By using gesture recognition, it is possible to use hand alone as a remote control for various devices. The signal must not only indicate the desired response, but also which device to be controlled. [9] [10] [11]
e. Home Appliances control: It is possible to extend the gesture recognition technology to control the household appliances. [10]

f. Remote control: By using gesture recognition, it is possible to use hand alone as a remote control for various devices. The signal must not only indicate the desired response, but also which device to be controlled. [9] [10] [11]
g. Home Appliances control: It is possible to extend the gesture recognition technology to control the household appliances. [10]

5. ADVANTAGES & LIMITATIONS

Gesture recognition is very useful for automation. Gestures, a natural language of humans, provide an intuitive and effortless interface for communication with the computers. They will reduce our need for devices like mouse, keys, remote control or keys for interaction with the electronic devices. When combined with other advanced user interface technologies such as voice commands and face recognition, gestures can create a richer user experience that strives to understand the human "language," thereby fueling the next wave of electronic innovation.

This technology is limited in the sense that all whole of human signs or gestures are not recognizable using this technology. The ultrasonic waves spread out and cannot be used to detect gestures like victory sign, where the gesture is made by two fingers. The control of GUI with hand gestures is really appealing. But the excitement comes with the challenge of limiting unintended hand motions. This step includes writing and testing the system made so far with a program whose supposed function is making the last sensor determine the direction of user's hand. This particular program identifies the direction of hand using the variable declared in the program. With the help of sensor, Arduino informs the serial port whether user has waved hand left to right (B) or from right to left (F). Visually, we can identify the activity of particular sensor with the help of LEDs, meaning that LED will blink as user moves hand in front of sensor connected with it.

6. RESULTS

The gesture recognition using ultrasonic waves is found to be accurate and reliable. The methodology for testing comprised of movement of single hand or multiple hands. Single hand movement was detected accurately. When there are multiple hands, the movement is not detected accurately. The detection did not take into account the background area. The noise in human audible range did not affect the detection.

7. CONCLUSIONS & FUTURE SCOPE

Additional gesture recognition opportunities exist in medical applications where, for health and safety reasons, a nurse or doctor may not be able to touch a display or track-pad but still needs to control a system. In other cases, the medical professional may not be within reach of the display yet still needs to manipulate the content being shown on the display. Appropriate gestures, such as hand swipes or using a finger as a virtual mouse, are a safer and faster way to control the device.

REFERENCES

[1]. Nidhi Gupta, Ramandeep Singh , Sidharth Bhatia. Hand gesture recognition using ultrasonic sensor and atmega128 microcontroller; June 2014 [online]
http://esatjournals.net/ijret/2014v03/i06/IJRET20140306107.pdf
[2]. Dong-Ik Go, Gaurav Agarwal. Gesture Recognition: Enabling natural inter-actions with electronics; April 2012 [online]
URL: http://www.ti.com/lit/wp/spry199/spry199.pdf

[3]. Huang, Pavlovic, Sharma. Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review; July 1997 [online] http://www.cs.rutgers.edu/~vladimir/pub/pavlovic97pami.pdf
[4]. Kimmo Karvinen, Tero Karvinen. Make: Arduino Bots and Gadgets First Edi-tion. United States of America: O'Reilly Media, Inc.; 2011
[5]. Lentin Joseph. Learning Robotics Using Python. Birmingham UK: Packt Publishing Ltd; May 2015
[6]. Fernandez, Mahtani, Crespo, Martinez. Learning ROS for Robotics

[7]. M Porta "Vision-based user interfaces: methods and applications", International Journal of Human-Computer Studies, 2002.
[8]. Henriksen, K. Sporring, J. Hornbaek, K. " Virtual trackballs revisited", IEEE Transactions on Visualization and Computer Graphics, 2004
[9]. Do Jun-Hyeong, Jung Jin-Woo, Sung hoon Jung, Jang Hyoyoung, Bien Zeungnam, Advanced soft remote control system using hand gesture, Mexican International Conference on Artificial Intelligence, 2006
[10]. K. Ouchi, N. Esaka, Y. Tamura, M. Hirahara, M. Doi, Magic Wand: an intuitive gesture remote control for home appliances, International Conference on Active Media Technology, AMT 2005
[11]. Lars Bretzner, Ivan Laptev, Tony Lindeberg, Sören Lenman, Yngve Sundblad "A Prototype System for Computer Vision Based Human Computer Interaction", 2001.