Human Arm Movement Data Acquisition and Processing

Myron Velton Rodrigues  
UG Student  
Department of Electronics and Telecommunication  
Don Bosco College of Engineering

Abdul Rakib M. R. Sarmadi  
UG Student  
Department of Electronics and Telecommunication  
Don Bosco College of Engineering

Sunny Rudresh Sinari  
UG Student  
Department of Electronics and Telecommunication  
Don Bosco College of Engineering

Veliston Joao Rocha  
UG Student  
Department of Electronics and Telecommunication  
Don Bosco College of Engineering

Anisha Maria Cotta  
Assistant Professor  
Department of Electronics and Telecommunication  
Don Bosco College of Engineering

Abstract

In recent times there is an increasing need for robots for operating in different inhumane situations. They may involve collecting data from harsh environments, military use, to carrying out maintenance work on space stations. The use of such robots requires skilled personnel as the controllers used are not human friendly. Here we propose to build a human arm interface which can read and process data to be further used by the controlling robots through the use of linear carbon potentiometers. For proper control and to reduce the noise coming from the potentiometers we use an averaging algorithm to smoothen out the output. This data can then be used to control various robots by further mapping the inputs and outputs. Finally, this interface is expected to reduce human efforts in controlling robots.

Keywords: Linear carbon potentiometers, Averaging algorithm, Arduino

I. INTRODUCTION

Nowadays, robots are increasingly being integrated in working tasks to replace humans. Robots are increasingly being used in different fields of operation including production lines in industries, military, hospitals and space exploration. Some robots have a built in feedback systems which enables them to self-control their movements while others use an open loop system where human control is required [1-2]. It becomes a tedious job to train personnel for controlling these robots and many a times the operating conditions are harsh for humans. Therefore a simple human arm interface which records natural human arm movements and digitizes this data can prove handy for controlling robots.

Currently there are various sensors available to record human movements such as pressure sensors, flex sensors, biomedical sensors which are quite expensive hence we use simple linear carbon potentiometers placed at every joint on the human arm to record motion [3-4]. The inbuilt analog to digital convertor of the Arduino UNO development board which uses Atmega328 Microcontroller is used to digitize the analog data from potentiometers. The potentiometers can prove noisy hence the Moving averages algorithm is used to smoothen out the data [5]. This data can then be either stored for programming robots in industries to perform a particular task or it can be live streamed to the robot for real time operation.

II. BLOCK DIAGRAM

The block diagram in Fig.1. Shows that the linear carbon potentiometers are connected to the microcontroller which converts the analog input data to digital data via the inbuilt analog to digital converter and then this output is processed by the microcontroller processing unit to remove any noise. The output from the processing unit is finally stored to be used by robots or is live streamed to a robot to control it in real time.
III. DESIGN AND DESCRIPTION

A. Hardware Design
As shown in Fig. 2. The hardware required is linear carbon potentiometers to be placed at the joints of fingers (index and thumb), Wrist, Elbow and Shoulder. The potentiometers are then connected to the microcontroller.

B. Software Design
The software part involves acquiring the data from the potentiometers via the Analog to Digital converter and processing it to smoothen out the output using the Moving averages algorithm this reduces the noise that might result in sudden spikes in the output data.

IV. IMPLEMENTATION

A. Hardware implementation
Fig. 3. Shows the interface connected to a human arm. The static terminals of the potentiometers are shorted and connected to Supply of 5 volts and common ground. The slider terminal of each potentiometer is connected to a separate Analog to Digital converter channel on the Arduino UNO. Acrylic material is used to make the moving and static shaft connected to the knob and body respectively of the potentiometer. The acrylic shafts are fastened to human arm with Velcro as shown in Fig. 3. Human arm movements cause the acrylic shafts to move and in turn rotate the potentiometer slider which in turn changes the voltage output at the slider terminal.

The Arduino UNO board has a 10 bit Analog to Digital convertor hence the values for full swing voltage range from 0 (potentiometer slider to minimum position) to 1023 (potentiometer slider to maximum position).
B. Software implementation

The code to acquire data from the potentiometers is developed using the Arduino IDE which uses C++ language. The Smoothing algorithm of Moving averages is implemented on Arduino itself. The first element of the moving average is obtained by averaging a fixed number of initial elements in the given set of readings. The subsequent averages are obtained by eliminating the first element and including the next element after the last element of previous subset. This process is repeated over the entire data series. The real time plot of the original reading is obtained using an inbuilt graph plotter in the Arduino IDE software and is shown in Fig. 4. Also the smoothened output is shown in Fig. 5. and Fig. 6. It is seen that the output signal is smoothened more as the window size of the moving average algorithm is increased.

Fig. 4: Graph of the original reading taken from one of the sensors

Fig. 5: Graph showing output of the moving average algorithm with window size 5

Fig. 6: Graph showing output of moving average algorithm with window size 10
V. CONCLUSION

The human arm interface can be used to program industrial robots and various other robots which use robotic arms for their operation. It significantly reduces the workload of programming robots and controlling them with joystick controllers. The processing done on the input movement data helps in reducing noise spikes ensuring smooth movement of interfaced robotic arms.

REFERENCES