# Wearable FSR based Device for Muscle Activity Monitoring*

Pratap Bhanu Solanki
solankip@msu.edu

Thassyo Da Silva Pinto
thassyo@msu.edu

December 9, 2014

**Abstract**

Muscle activity at the arm are responsible for enabling many human tasks. Motion data related to the muscles not only can help in the rehabilitation progress of injured people, but also can be used for pattern recognition in a computer input device. We developed a wearable device using force-sensing resistors for detecting the inflation of different muscle groups at the forearm and upper arm. The device was fabricated using two elastic straps with a wireless module to allow the performance of mobile activities. By capturing the pressure generated at the muscle belly, the sensing straps were used as a human interface device for controlling the computer keyboard and mouse cursor. In addition, we implemented a neural network for recognizing the patterns produced by multiple arm and hand gestures. Finally, force-sensing resistor shows promising application since they are unobtrusive, lightweight and has fast response.

## 1 Introduction

Most human daily activities are associated with arm movement. From grabbing and turning the car key to texting a message on a cellphone, a variety of actions involving fingers and wrist motion are performed instantaneously to our eyes. Different group muscles at the arm are activated through the nerve-muscle connection when command signals from the brain are transmitted in order to execute a certain task. Commercially available devices can capture these bio-signals by using a technique called electromyography (EMG). In this procedure, electrodes are attached to the skin surface for measuring the electric potential (voltage) generated by the muscle cells.

Physiological data acquisition is an important feature for monitoring the progress and body performance of people in rehabilitation process. Unobtrusive sensing mechanisms contribute to no contamination while providing a fast reuse. A glove for sensing finger motion (MusicGlove, Flint [1]) can help patients with neurological conditions such as stroke and cerebral palsy, to improve their hand function while playing a therapy-oriented music game. Researchers have developed a force myography (FMG) device for extracting signals of the upper-extremities and classify the movements related to a drinking task[2] . Other researchers also investigated the use o force-sensing resistors (FSRs) for capturing the pressure applied by the muscles at the forearm [4] [5], and at the leg for cycling activity [6]. However, those system do not provide any type of wireless communication, preventing it to be implemented for activities that requires mobility.

The interaction between human and machine is becoming, by each generation, more and more close. All this progress occurs due to the several needs that emerges when technology evolves, which can be considered but are not limited to: fast response, mobility, compactness, accessibility. New devices are being developed in order to provide advanced interface methods. An armband device (Myo, Thalmic Labs [3]) introduce a new concept of which could replace the usual method of controlling computers. By using EMG sensors and highly sensitive nine-axis IMU, the device is able to capture hand gestures and communicate with computers and mobiles via Bluetooth. Another work investigates the use of an electroencephalography (EEG) device for controlling the mouse cursor through facial expressions, allowing paralyzed people to have easy control of a computer [7].

*This work is done under Prof Xiaobo Tan as a course project of ECE819 (Smart Materials Sensors and Actuators) at Michigan State University

Force sensing resistor (FSR) is a material whose resistance changes when a force or pressure is applied. It is also known as Force-sensitive-resistor. In this present work, we investigate the use of FSRs in a wearable and flexible device for sensing muscle pressure at the forearm and upper arm, enabling the recognition of gestures related to arm, hand and wrist movement. We implemented it as human interface device for computer applications, allowing the control of the keyboard input and mouse cursor. Furthermore, by using a neural network platform, it was possible to identify the muscles associated with multiple activities.

## 2 Selection of Sensors

Initially we wanted to make a device, which can capture immediate muscle movements. Initially we thought of using standard piezo-extension-sensors from piezo.com. We thought of measuring the contraction and relaxation muscles by attaching the PZT strip with a band. This way if the muscle expands, the overall circumference would increase and hence the length of PZT strip will increase. However the cost of cheapest PZT strip was more than 100$. We thought of looking for other cheap smart materials sensors. We found two sensors: Force Sensitive Resistor (FSR) and PVDF (Polyvinylidene fluoride) vibration sensor, both of which costs less then 10$.

A PVDF vibration sensor (figure 1) has a flexible piezo polymer film made by Measurement Specialities$^{TM}$, which works on the bending principle of piezoelectric film. It has an inertial mass attach to the tip of it, which helps in capturing vibration. So when there is a vibration, it bends up and down and hence it creates an oscillatory voltage output.

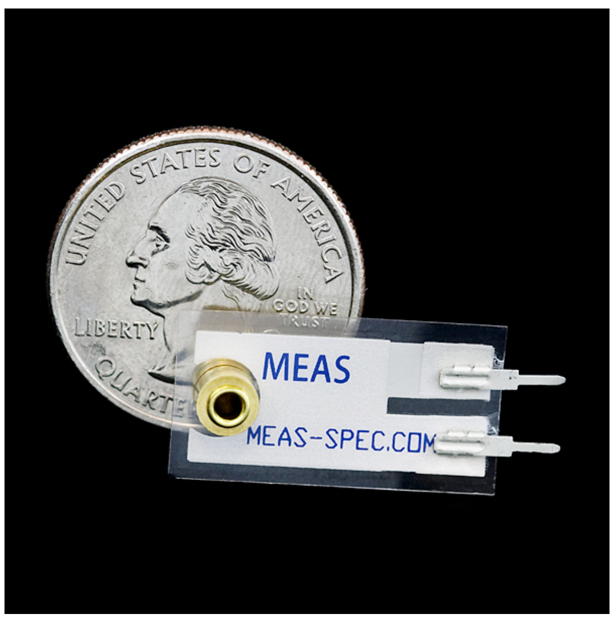


Figure 1: PVDF vibration sensor [12].

In our application, we were trying to use it to capture forearm muscle vibration signals, similar to Electromyography (EMG)[9]. However since this sensor works on the principle of bending, it did not give any output when it was fastened tightly to hand muscles. So we decided to use our other option which is Force Sensitive Resistor, which is discussed in more detail in next section.

## 3 Force Sensitive Resistor

A force-sensing resistor (FSR) is a conductive polymer that has a decrease in its resistance when there is an increase on the amount of force applied at its surface. The FSR is a patented technology from Interlink Electronics [10] that was first introduced as sensory device in musical instruments. In our work we are using it as a pressure sensor to sense the actuation of hand muscles. These FSR sensors have a force sensitivity range from 0.1 to 10 N, and an active area (diameter) of 5.08$mm$. By using a voltage divider configuration (Figure 2) it is possible to measure the change in resistance of each FSR. The output is described by equation 1.

$$V_{out} = V_{in}(\frac{R_2}{R_2 + R_{FSR}}) \tag{1}$$

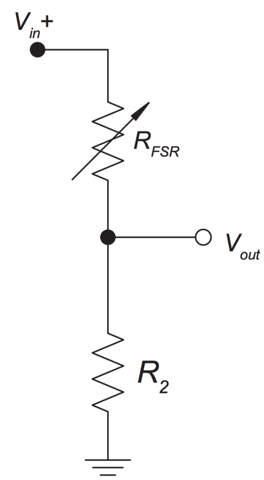


Figure 2: FSR's voltage divider circuit [8].

## 3.1 Basic Construction of FSR

Force sensitive Resistor consists of two membranes separated by thin air gap (figure 3).The air gap is maintained by a spacer adhesive around the perimeter of the two membranes. The space has thickness about 0.03$mm$ to 0.1$mm$. It separates the two substrates and holds the sensor together. The $1^{st}$ membrane is actually the force sensitive resistor layer, which is printed with carbon based ink. The $2^{nd}$ membrane consists of two sets of interdigitated fingers that are electrically distinct. Where each set is connected to one of the output terminal of the sensor. So when the two substrates are pressed together the microscopic protrusions from the FSR membrane shorts the interdigitated fingers. Now this is not just like on and off situation, here at low forces there are only few of the protrusions (figure 4), which are tall, make contact with the interdigitated fingers and as the force increases more and more points make contacts. As a result the resistence between the conducting fingers is inversely proportional to the applied force.

## 3.2 Mathematical Modeling of FSR

The simple mathematical model of FSR can be viewed as a linear mass-spring-damper model(Figure 5). The model consist of a mass block which is connected to the base via a spring and a damper. The differential equation of motion of mass would be:

$$F = m_{ac}\ddot{y} + c\dot{y} + ky \tag{2}$$

Where $F$ is the force applied at the FSR's surface, $y$ is FSR's displacement, $m_{ac}$ is the mass of the active area of the sensor, $c$ is the damping constant and $k$ is the spring constant. It is assumed that the voltage across the terminals of FSR is proportional to displacement.

$$V \propto y \tag{3}$$

So the force voltage relation can be written as in equation

$$F = A(m_{ac}\ddot{V} + c\dot{V} + kV) \tag{4}$$

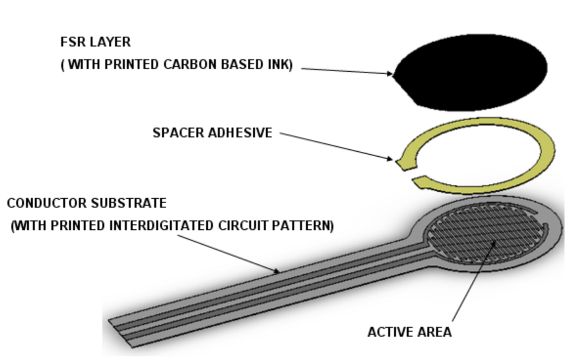


Figure 3: Basic FSR construction [10]

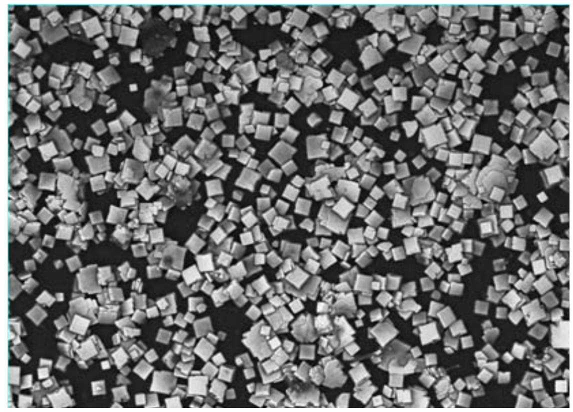

Figure 4: FSR ink micrograph [10]

Where A is a constant of proportionality. Value of A calculated by experiments is $2.25 \times 10^{-5} V/m$. This model is an approximate linear model whose perfomarmance can be quantified by a metric called fitness performance. This linear model has a fitness performance of around 60-70%. More accurate non linear models are available [8]: Hammerstein Model, Weiner Model and Hammerstein-Weiner model. These models include non-linearities at input or output and improve the fitness performance to more than 90%.

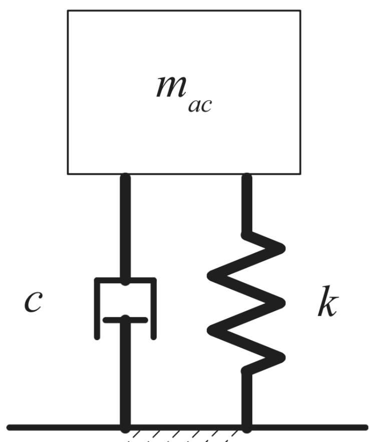


Figure 5: FSR's mass-spring-damper model [8].

## 4 The Wearable Device

We developed a wearable device (figure 6) using fast prototyping available tools in order to achieve a quickly deployment. The device is able to communicate wirelessly with a base station (e.g. desktop computer), giving more freedom for the user when performing some activity. In addition, it was built with an elastic strap to allow the muscle inflation, making it comfortable to wear when attached directly on the skin or either on top of clothes. Its purpose is for sensing muscle contraction/extension at the forearm and upper arm regions.

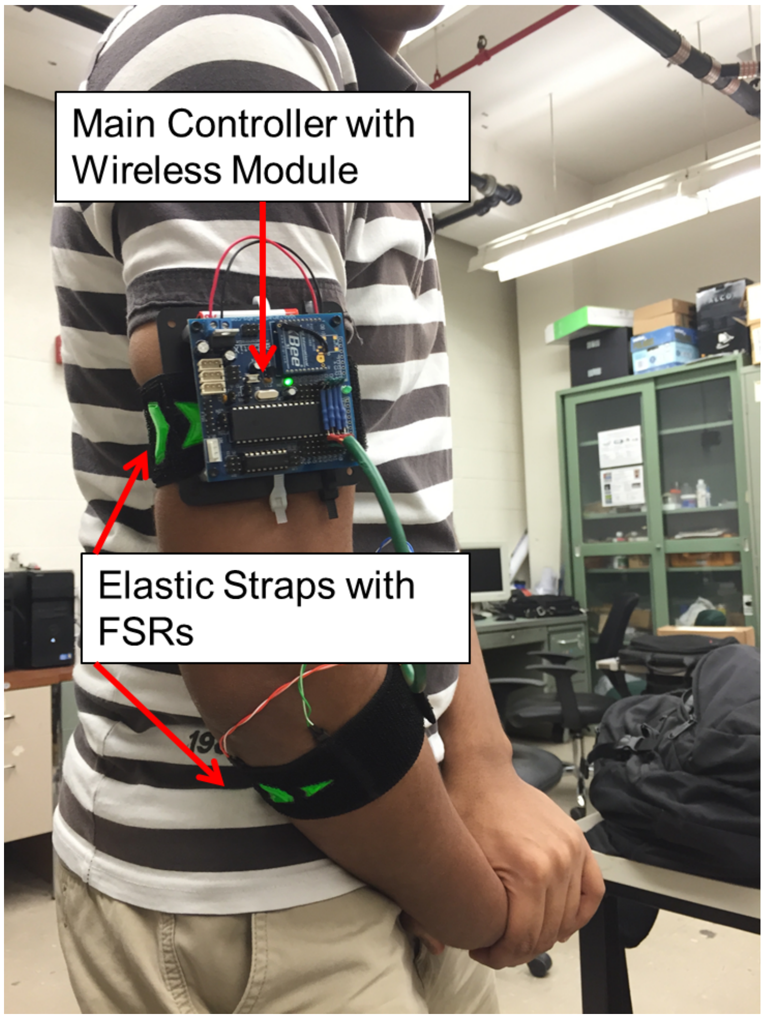


Figure 6: Wearable device.

### 4.1 Main controller

The device has as its main controller a high level development tool (ArbotiX RoboController, Vanadium Labs), which allows the addition of other mounting components such as wireless modules. It is based on an 8-bit AVR microcontroller (ATmega644p, Atmel) [11] running in a clock speed of 16MHz. The controller is responsible to capture the data from the FSRs and send the value corresponding to each sensor to the stationary machine, which will be able to process the data using some software or analysis tool. It reads the sensor signal trough an analog-to-digital converter (ADC) pin, which converts the changing voltage to a number between 0 and 1023. During each iteration, the microcontroller will store all sensors values and transmit them in a message (string) through the serial pins connected to the wireless module. In order to get more accurate data, we have set an average calculation that will send the mean value of each sensor after only a certain number of readings.

### 4.2 Wireless module

As a great advantage in our device, we have incorporated two wireless serial modules for long range communication (XBee Series 1, Digi), completely removing the motion restriction of the user, which on the other hand are present in expensive and sizable wired equipment. This module has a communication range of 10 meters and a maximum data rate of 250kpbs. When the device is powered on, the wireless module starts to send all data related to the FSRs reading. In order to be able to receive the messages sent by the wearable device, a USB to serial base unit (XBee Explorer USB, SparkFun) was connected to one of the modules. This unit can then be hooked up to any laptop or desktop computer, permitting the use of any program for extracting information of the muscle activity.

### 4.3 Use of force-sensing resistors

In this project, we have integrated six force-sensing resistors (FSR 400, Interlink Electronics) for sensing the pressure applied by a chosen group of anterior/posterior muscles at the forearm and upper arm of a human body. With this resistance-to-voltage conversion method we were able to read the amount of voltage at all ADC pins of the microcontroller when some pressure was applied on the sensor. The output voltage increases with increasing force at the FSR surface. From figure 7 we can see the purple curve which corresponds to $10k$ of the value of base resistor ($R_2$), gives highest voltage output range of $2.5V$. Hence in order to maximize the force sensitivity range, a $10k$ ohm resistor was chosen based on the manufacturer data for a standard FSR with various measuring resistors.

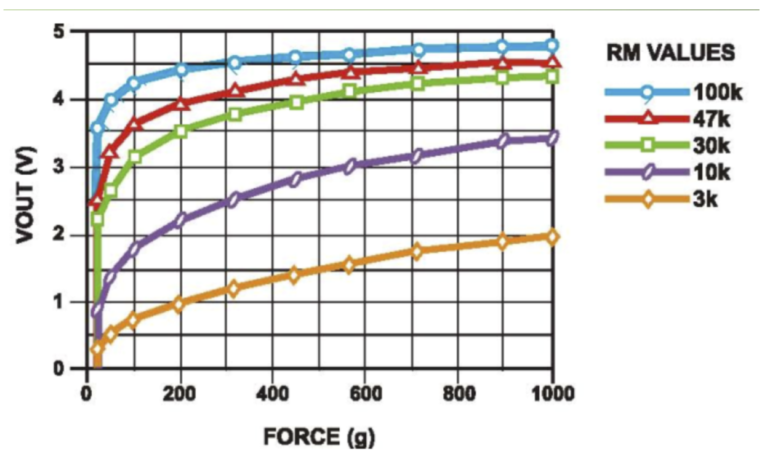


Figure 7: Force vs $V_{out}$ for various values of $R_2$ [10].

### 4.4 Positioning of sensors on muscles

The sensors locations were intentionally chosen to get wide force sensitivity range and to capture the muscle motion associated with the gestures selected for this project. The wearable device is comprised of two flexible straps, one for the upper arm and the other for the forearm. Each sensor is attached to the strap with a double-side tape and positioned directly in the center of the muscle belly, since the deformation in this area is expected to be larger. For the upper arm strap, there are two fixed FSR sensors which are sensing the biceps brachii (anterior muscles) and the triceps brachii (posterior muscles). In this case, the device will be able to detect whenever the user extends or flexes the forearm. Since the forearm contains many muscles groups that are responsible for controlling the hands and wrist actions, we designated four FSRs to cover this sensing area. The muscles at this location are: the brachioradialis, flexor carpi ulnaris, flexor carpi radialis, palmaris longus, extensor digitorum and extensor carpi ulnaris.

## 5 Pattern Recognition using Neural Networks

Earlier our plan was to use the vibration signal from muscle to identify the muscles activity. The vibration signals were supposed to be a long time series signals. Hence we thought of using neural networks to identify and classify pattern in the time series signal output. We were unable to use the vibration sensor and instead of that we used multiple pressure sensors, however we still planned to use neural network to classify hand gesture from the output of sensors.

To identify hand gestures first we need to define them. We have one degree of freedom in elbow and 3 degrees of freedom in wrists. Taking into account these degrees of freedom. We defined following gesture entities:

1. **Wrist yaw:** Yaw motion of wrist, It is to be noted that wrist can do yaw motion in one direction only
2. **Wrist pitch positive:** Accounts for pitch motion of wrist, with motion in direction of palm
3. **Wrist pitch negative:** Also accounts for pitch motion of wrist, with motion in direction opposite to palm
4. **Wrist roll:** Accounts for roll motion of pitch, Here also wrist can do roll in one direction. However one can say that roll can be in two direcitons by keeping the relaxed position in middle of the roll extremes. Hence we can define the relax position to be one of the extreme
5. **Grasp:** It accounts for the grasping some object by hand, this is not associated with degrees of freedom of elbow and wrist
6. **Biceps:** Bending the arm completely such that wrist almost touch the shoulder, giving full expansion to biceps
7. **Triceps:** Stretch arm completely and give stress to tricesps

Here, each gesture entity can have value 1 or 0. In addition to this 'wrist pitch positive' and 'wrist pitch negative' both cannot simultaneously be equal to 1 and also 'biceps' and 'triceps' also cannot be simultaneously equal to 1. Consider an example where user is asked to do this gesture:

$$[wrist\_yaw, wrist\_pitch\_positive, biceps]$$

Then the gesture vector can be would be the vector of binary values with 1's in the place of the entities which are present in the gesture and zero for other entities. So for the given case the gesture vector would be:

$$[1, 1, 0, 0, 0, 1, 0]^T$$

Similar to the above gesture, there would be 72 total gestures. To train the network we need data. The data is collected by a matlab program where the system randomly generates one gesture out of the possible 72 gestures. The user is asked by program to adjust his hands according to the generated gesture. When user become ready then he presses enter and then the program starts collecting 100 samples of the output of the sensors. In the execution of code there is a provision to discard the collected data of the current gesture, incase he feels that he was not able to maintain the gesture for so long and made some error. After successful collection of data user has a choice to either collect more data or to end the program. This way we collected a random sequence of input-output samples of gestures.

To classify the actions we used Pattern Recognition App of Neural Network Toolbox of MATLAB. We chose the network with 1 hidden layer with 50 neurons (Figure 8).

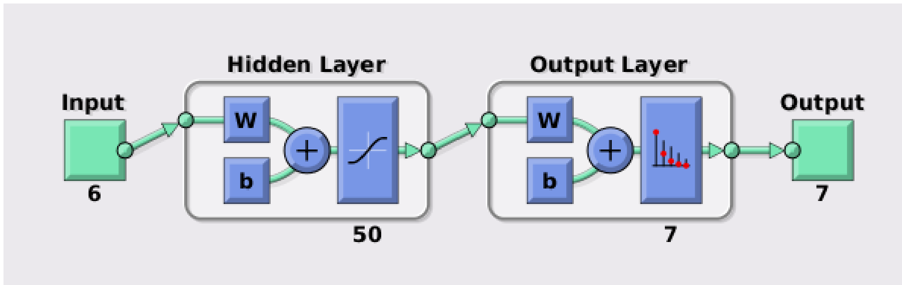


Figure 8: Pictorial depiction of neural network.

One successful system should be able to identify all of the 72 hand gestures distinctively. In our case the sensor values are not so much consistent. The output of sensors are not much correlated with the hand gestures. Figure 9 shows the output of sensors for one random sequence of actions.

From the figure we can see that the sensors value are very noisy and only biceps and triceps muscles sensors are giving prominent and consistent output.

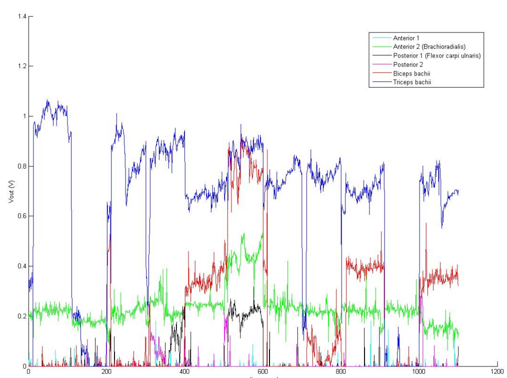


Figure 9: Sensor values samples for a sequence of gesture.

For this case we try to look at a smaller picture where we were considering only biceps and triceps. So we were considering only 2 dimensional input (Sensor from biceps and triceps only) and output vectors (considering only biceps and triceps gestures). From figure 10 we can see even for just two gestures, the network is unable to get train properly. There is just 70% accuracy in classification.

Observing these poor results. We reduce the total number of gestures and active sensors. We chose sensors on anterior1, anterior2, biceps and triceps and remove the two sensors on posterior muscles. Also the output is also simplified and now there are just 5 total active gestures as compared to previous 72 gestures. These gestures are namely:

1. **Pitch:** Only wrist would be on pitch position, everything else would be relaxed, similar to [0,1,0,0,0,0,0] of previous notation and is [1,0,0,0] in new notation

2. **Grasp pitch:** Wrist pitch with a grasp and everything else relaxed, similar to [0,1,0,0,1,0,0] of previous notation and is [0,1,0,0] with new notation

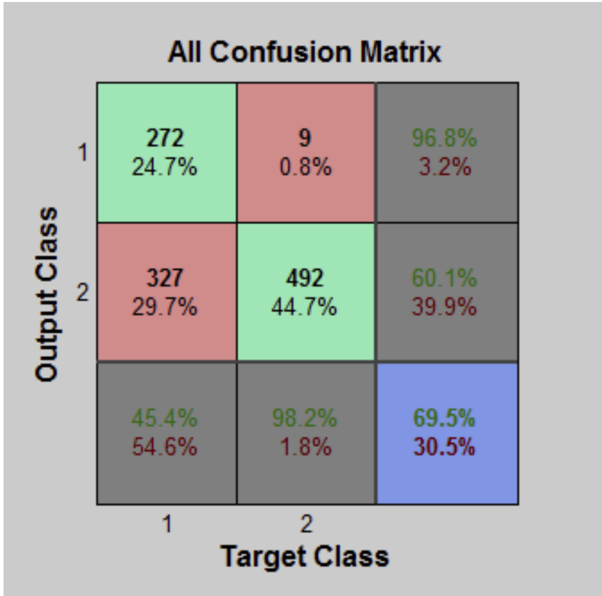


Figure 10: Sensor values samples for a sequence of gesture.

3. **Flexion:** Biceps would be stressed by fully bending the arm from elbow, similar to [0,0,0,0,0,1,0] of previous notation and it is [0,0,1,0] in new notation

4. **Full Extension:** Triceps would be stressed by fully stretching the arm from elbow, similar to [0,0,0,0,0,0,1] of previous notation and it is [0,0,0,1] in new notation

5. **Relax:** Everything would be relaxed, i.e in zero position, similar to [0,0,0,0,0,0,0] of previous notation and it is [0,0,0,0] in new notation

The new neural network has now 4 dimensional input vector and 4 dimensional output vector. Apart from this we also averaged the sensor values over 20 readings, this removes noise from the sensor data and we get more cleaner and consistent values. Figure 11 shows the plot of new sensor values. From the figure we can see the data we get is comparatively clean and each sensor has some prominence across different gestures.

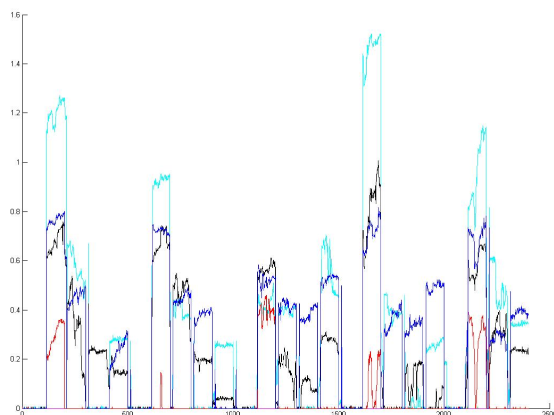

Figure 11: Sensor values samples for sequence of new gestures

We try training the neural network again with the new set of data. From figure 12 we can see that even though we have 4 gestures now the accuracy in all 4 cases is more than 90%. Hence using the neural network on real time and the mapping of gestures with muscles, we can identify the activated muscle at each gesture.

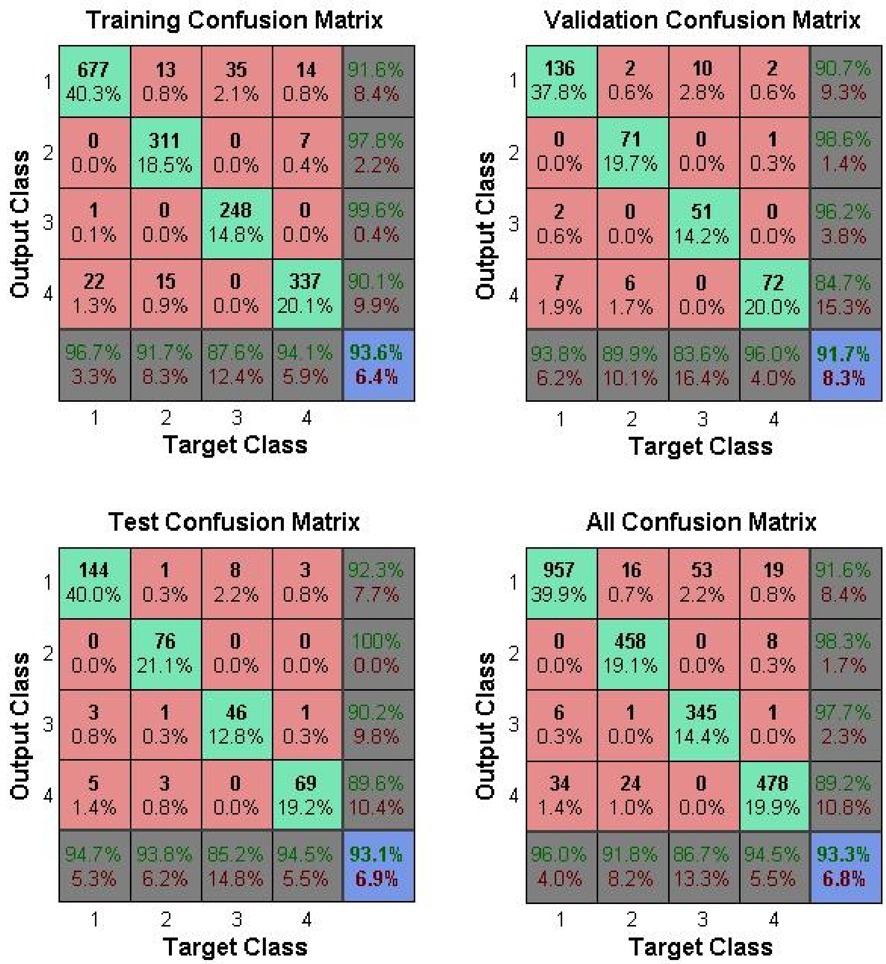


Figure 12: Confusion matrices for training, validation and testing purpose.

# 6 Human Interface Device(HID)

Different types of devices are used as interface between a user and a machine such as keyboards, mice and gamepads. With technology advancement new electronic architectures are being explored as well as new user input methods. Other kinds of user interface can also extend the options for people with disabilities or that suffered limb loss, connecting them to the digital world. By defining patterns of forearm and upper arm motion, we were able to use the wearable sensors as a human interface device (HID) for controlling a computer key press/release as well as the mouse cursor. We developed a program using Processing language to open the serial port where the second wireless module was connected, and to process the data transmitted by the main controller. The four new gestures defined in the previous section are used for this experiment. The upper arm motion had the control of the mouse cursor, which could be moved up or down when extending or flexing the forearm, respectively. Moreover, the wrist bending was able to command two predefined keyboard keys. A bend of the wrist with closed hands (Grasp pitch) was characterized as the Enter (return) key, while a motion with a flat hand (pitch) was controlling the Backspace key. In order to keep each key pressed, the user could just hold the gesture for a certain time, where the same could be applied for moving the cursor up and down.

# 7 Conclusion and Future works

In our work we demonstrate that the FSR has potential to work as a pressure sensor for upper arm and forearm muscle pressure sensing. We made a wireless-body-worn device which we are able to transmit muscle actuation data to the nearby pc in real time. Using this real time data we are able to demonstrate the control of mouse and some key-press events. Such kind of visual feedback increases the efficciency of rehabilitation and motivates the patient to do the exercises more, as a result the rehabilitation time decreases significantly. We also show that neural networks can be used in learning and recognition of gestures. In our present work we reduced the gestures to very low number so for identification we used 'if-else' conditions to identify the gestures. Future work involves the research on identifying the exhaustive set of gestures. In that casae the use of neural-networks will play significant role as then the dimension of data will increase.

# 8 Individual Roles

In our work we did most of the things together so with different amount of effort. Following table describes the distribution of work between the team members.

| Work | Pratap's Work (%) | Thassyo's work (%) |
|---|---|---|
| Study and testing of PVDF Vibration sensor | 100 | 0 |
| Study and testing of FSR sensor | 0 | 100 |
| Design and development of physical device | 20 | 80 |
| Enabling wireless communication interface | 0 | 100 |
| Data collection, Neural Network Training and Testing | 100 | 0 |
| Design of cursor and keypress control | 20 | 80 |

# References


[1] https://www.flintrehabilitation.com/

[2] Zhen G Xiao and Carlo Menon, "Towards the development of a wearable feedback system for monitoring the activities of the upper-extremities". *Journal of NeuroEngineering and Rehabilitation* 2014, 11:2

[3] https://www.thalmic.com/en/myo/

[4] Matthias Kreil, Georg Ogris, Paul Lukowicz, "Muscle Activity Evaluation using Force Sensitive Resistors". *Proceedings of the 5th International Workshop on Wearable and Implantable Body Sensor Networks* June 2008, 107:110

[5] Oliver Amft1, Holger Junker, Paul Lukowicz, Gerhard Tro ster1, Corina Schuster, "Sensing Muscle Activities with Body-Worn Sensors". *International Workshop on Wearable and Implantable Body Sensor Networks* April 2006

[6] Georg Ogris, Matthias Kreil, Paul Lukowicz, "Using FSR based muscule activity monitoring to recognize manipulative arm gestures". *11th IEEE International Symposium on Wearable Computers* Oct 2007

[7] Nayel Al-Zubi, "Detecting Facial Expressions from EEG Signals and Head Movement for Controlling Mouse Curser". *Third International Conference on Innovative Computing Technology (INTECH)* Aug 2013, 551 - 554

[8] Mohammad Y Saadeh and Mohamed B Trabia, "Identification of a force-sensing resistor for tactile applications". *Journal of Intelligent Material Systems and Structures* 2013 24: 813

[9] http://medusa.sdsu.edu/Robotics/Neuromuscular/Theses/Miller/MillerThesis.htm

[10] Interlink Electronics, FSR$^{TM}$ Force Sensing Resistors$^{TM}$ Integration Guide REV.A 09/23/10

[11] Atmel, 8-bit Microcontroller with 16K/32K/64K Bytes In-System Programmable Flash ATmega644P Datasheet

[12] Measurement specialities, LDT with Crimps Vibration Sensor/Switch, http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/ForceFlex/LDT_Series.pdf