IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 12, Issue 4, Ver. IV (Jul.-Aug. 2017), PP 29-35
www.iosrjournals.org

A Prototype for Human Computer Interactraction Using Myoelectric Interface

*M. Naga Malleswari[1], Mr. V.Purnachandra Reddy[2]
1 Project Student, Department of E.C.E, Vasireddy Venkatadri Institute of Technology, Nambur, AP,India
2 Asst. Professor, Department of E.C.E, Vasireddy Venkatadri Institute of Technology, Nambur, AP, India
Corresponding Author: M. Naga Malleswari

Abstract: *To achieve highly dexterous manoeuvre with many degrees of freedom is of significant importance to perform normal hand-like movements. We present a powerful tool, Myoelectric Interface for Computer and Human Interaction (MIRCHI), which can be used in prosthesis, physiotherapy and Gaming technology. MIRCHI is useful for people undergoing upper limb prosthesis. Unlike most of the exiting prosthesis tools which are dynamic subject- or task-specific, the proposed system takes input from the muscle of the subject using surface electromyography sensors and generates corresponding movements with the aid of some predefined mapping functions. We have tested different movements from various subjects. Results show that the subject is able to make custom movements with lot of ease. The entire system is developed in Ubuntu 12.04 LTS and OpenGL is used for GUI.*

Keywords: *Degree-of-Freedom (DoF), Human Computer Interaction, OpenGL., Prosthesis, Ubuntu 12.04 LTS.*

Date of Submission: 31-07-2017 Date of acceptance: 17-08-2017

I. Introduction

Myoelectric refers to the electrical properties of muscle tissue from which impulses may be amplified, used especially in the control or operation of prosthetic devices. Muscle contractions can be classified using pattern recognition of myoelctric signals and this work is being carried out since 1960's [1]. There is a great mismatching between the results obtained from myoelectric control and clinical practice. This is due to the fact that most of the EMG systems in the clinics have single or dual myoelectric signal amplitude-based control strategies [2]. These systems have limitations and in order to achieve many *degrees-of-freedom* (DoF), we need advanced control schemes. Pattern recognition has repeatedly shown the capacity to discriminate between many classes of motion [1] and, as such, is a logical candidate. The challenge becomes a matter of bridging the gap between theoretical, offline research and a real-time, clinically viable solution. This not only indicates a need for a real-time implementation of pattern recognition based myoelectric control, but also a presentation style which can be accepted by, and mutually beneficial to, both the research and clinical communities. The analysis of myoelectric signals and processing is still a good research field. MES processing techniques have been widely analyzed and they continue under research. The classical analog processing systems have been very reliable in psychoanalysis robotics [3]. However, since some years ago, improvements in signal processors rely on digital processing [4-6]. Recently, high-end digital processors are used in myoelectric processing and analysis. The present work focuses on developing a simple yet powerful interface to provide interaction between amputees and a robotic arm. The rest of the paper is organized as follows. In section II, we discuss on few issues related to the present work in II, and we then discuss about the proposed work along with the details of both hardware and software in II. This is followed by discussions and results.

II. Related Work

Since 1970s, many algorithms have been developed to train a decoder for a given user. Machine learning techniques are used to develop a specific decoder based on a set of training data. Classification techniques such as neural networks [7], [8], support vector machines [9], [10], [11] and random forests [12], among others, are commonly used for a discrete set of commands. For continuous commands, black-box modeling [13] and regression methods [14] are most common. However, studies by Ajiboye et al. have suggested that only a sparse set of natural muscle synergies are user-independent and form a low-level basis for muscle control [15]. Moreover, the system is highly nonlinear and it is very difficult to achieve good decoding accuracy even when it is trained on a single user [16]. As a result, these machine learning techniques currently result in decoders that are highly user-specific, training intensive, and limited in accuracy, all of which hinder the general performance of myoelectric controlled interfaces. In more closely related work, closed-loop myoelectric controlled interfaces are further investigated by Radha Krishnan et al. in [17] to understand human motor learning. As in [18], two decoders, classified as intuitive and non-intuitive, decode EMG signal amplitude from eight muscles to generate a 2-D cursor position. The intuitive decoder maps six of the eight muscles to a vector along the 2-D plane that is most consistent with the action on the limb when the muscle contracts. The non-intuitive decoder maps six of eight muscles randomly along equally spaced vectors in the 2-D plane. Subjects are able to learn the decoders in both experiments, with performance trends best fit by exponential decay. Additionally, the results show that the intuitive decoder helps subjects achieve better performance initially, but the non-intuitive decoder has a steeper learning rate that made performance for both decoders almost equal after 192 trials.

Pistohl et al. [19] compare subject performance for two different myoelectrically controlled tasks. The first task is a standard cursor control task, similar to [17]. The second uses a similar mapping function, but with removed redundancies such that each muscle operates individual fingers of a robotic hand. The mapping function is intentionally made nonintuitive to users in order to emphasize a steeper learning curve. The results show similar performance trends when given visual feedback for both cursor control and hand control, indicating that subjects are able to learn separate models to effectively reach the goal in both tasks. This previous research has established that humans are capable of forming inverse models for various decoders when presented with closed loop feedback. This paper uses myoelectric interface augmented with the control variable mapping to provide a closed-loop feedback control for subject-learning process.

III. Proposed Work

A. Setup

The study comprises of experiments performed on various subjects which are designed to identify whether each subject is able to do the given control tasks. We have used Muscle sensor Kit V3 from Advancer Technologies [20]. It has two-channels and is designed to be used directly with a microcontroller. These sensors do not output a raw EMG signal but rather an amplified, rectified, and smoothed signal that will work well with a microcontroller's analog-to-digital converter (ADC).

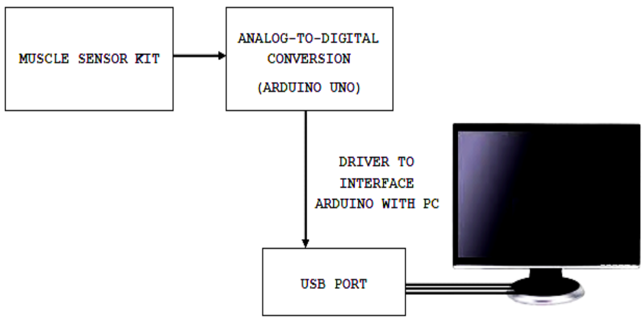

Fig. 1 Block Diagram

The experiments are carried out by changing the positions of electrodes for each movement. The block diagram of the entire setup is shown in Fig. 1. The electrodes of the muscle sensor kit are connected to the muscles of the subject. The analog signal generated by the electrodes because of muscle actions, which is sampled at 1 KHz, rectified, and low-pass filtered, is given to the ADC. The ADC is designed using Arduino UNO [21]. The digital output is then fed to the serial port of PC and a C program is written to interpret the data. The control tasks are made visible to the subject by providing a Graphical User Interface (GUI) which acts as a closed-loop feedback system for the subject. This GUI is developed using OpenGL [22]. This entire environment is created in Ubuntu 12.04 LTS Operating System.

B. Tasks and Mapping Functions

We have designed the system to provide a total of *eight* different movements. A computer screen infront the subject displays a ball at its center. The task of the subject is to move the ball in the desired direction. Myoelectric signals are obtained from four different muscles of the arm which are namely i) Biceps Brachii (BB), ii) Triceps Brachii (TB), iii) Flexor Carpi Radialis (FCR), iv) Extensor Carpi Ulnaris (ECU). The mapping functions transform the EMG amplitudes to control signals in required proportions.

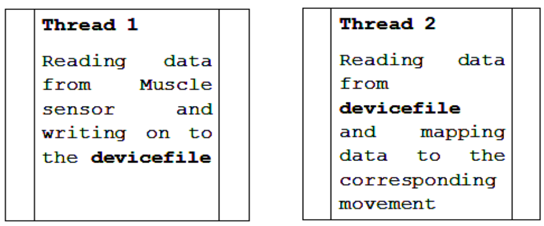

Fig. 2 Working Model of MIRCHI

The *working model* of the system is shown in Fig. 2. The entire process is divided into two threads, *Thread 1* and *Thread 2*. In Thread 1, the analog data from the muscle sensor is converted into digital by the ADC and this digital data is made available on the device's serial port. Thread 2 read data from the device's port and process this data, then applies mapping function on the data to generate the corresponding movement. The POSIX standard threads [23] are used. A driver has been development to for communication between Arduino and PC Serial port. The flowchart of the system in Fig. 3 gives a complete picture of the entire system process. The hardware model of the muscle sensor kit powered with a battery is shown in Fig. 4. Fig. 5 shows the interfacing of the muscle sensor kit with the Arduino. RS-232 cable is used to connect Arduino to serial port of the PC.

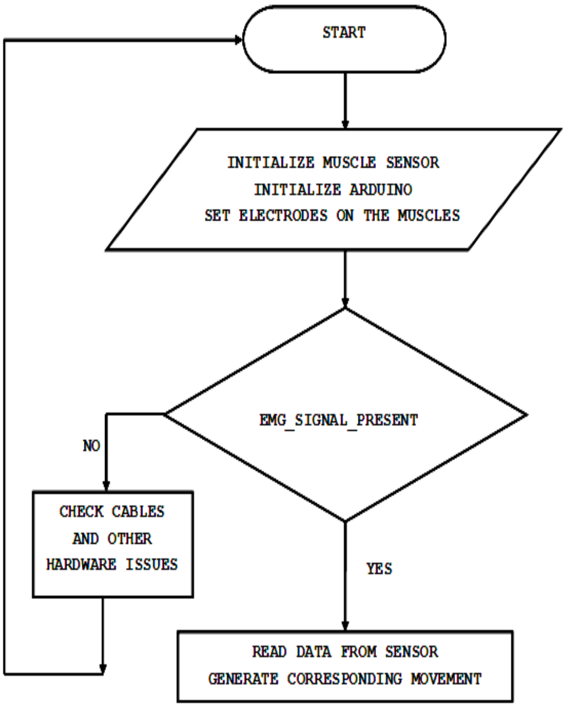

Fig. 3 MIRCHI Flow Chart

C. OpenGL

OpenGL is a platform-independent API (application programming interface) for rendering 3-D graphics. A big advantage of using OpenGL is that it is a widely supported industry standard. OpenGL defines a set of functions for doing computer graphics. OpenGL was specifically designed to be platform-independent, so it would work across a whole range of computer hardware – not just Silicon Graphics machines. The combination of OpenGL's power and portability led to its rapid acceptance as a standard for computer graphics programming. It provides 3D geometric objects, such as lines, polygons, triangle meshes, spheres, cubes, quadric surfaces, NURBS curves and surfaces. It supports the manipulation of images as pixels, enabling frame-buffer effects such as antialiasing, motion blur, depth of field and soft shadows.

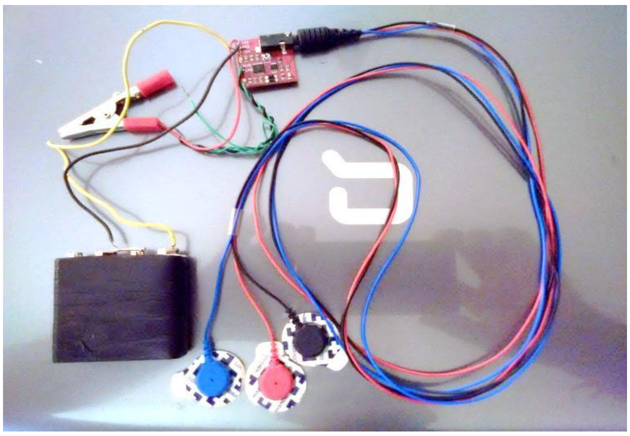

Fig. 4 Powering the Muscle Sensor Kit

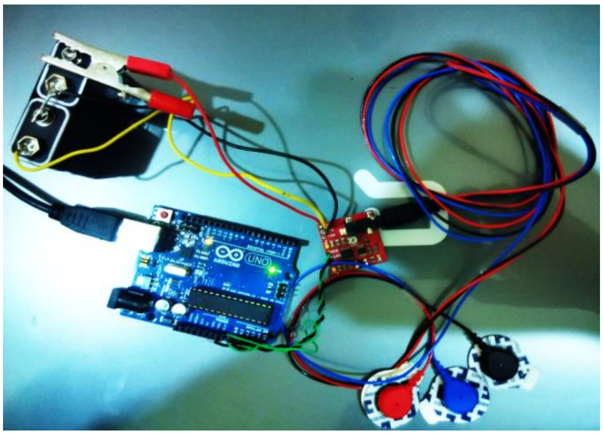

Fig. 5 Interfacing the Muscle Sensor Kit with Arduino

Fig. 6 Movement Diagonally Left-up

Fig. 7 Movement Diagonally Left-down

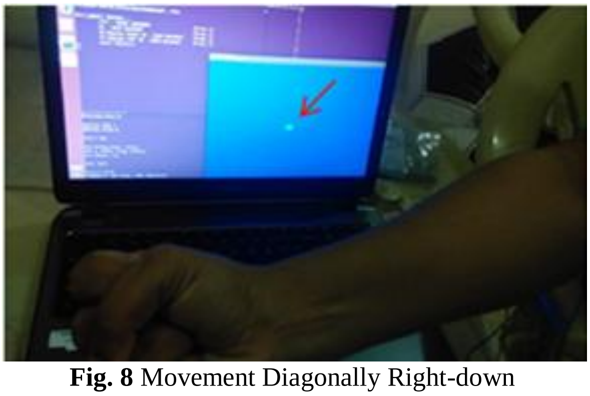

Fig. 8 Movement Diagonally Right-down

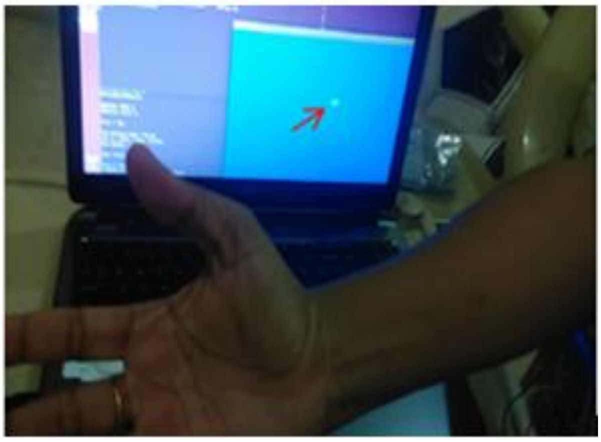

Fig. 9 Movement Diagonally Right- up

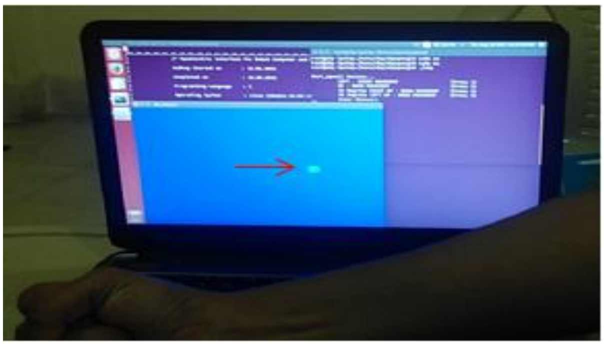

Fig. 10 Movement Horizontal Right

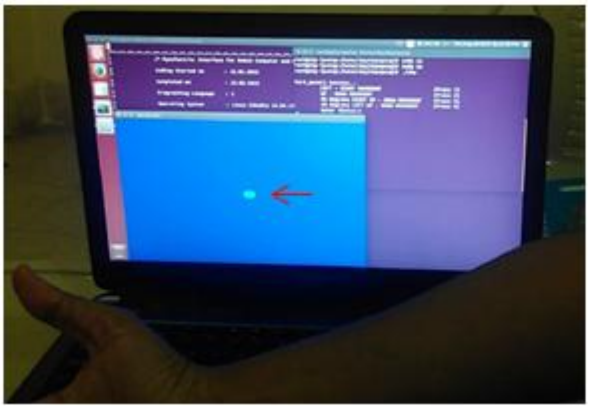

Fig. 11 Movement Horizontal Right

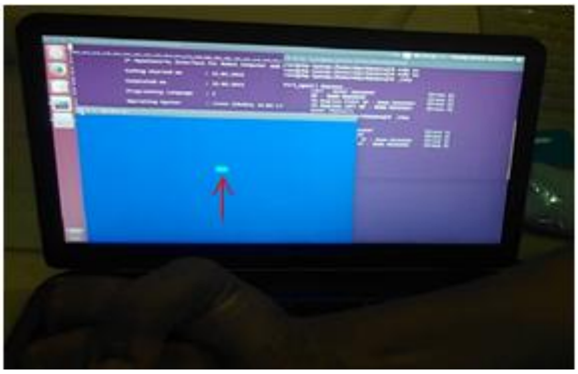

Fig. 12 Movement Vertical up

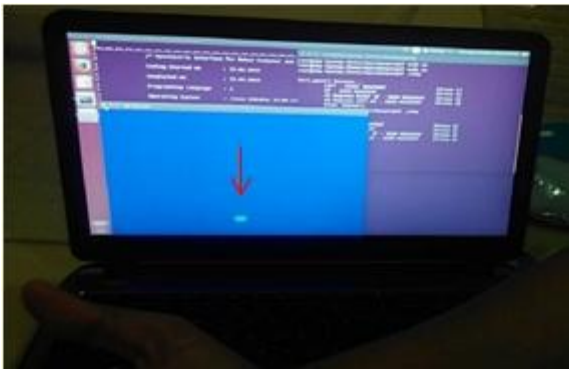

Fig. 13 Movement Vertical down

IV. Results

The results of the various movements are shown in Fig. 6 to Fig. 13. Fig. 6 shows the movement in the diagonal left upward direction. Fig. 7 shows diagonal left downward movement. Similarly, Fig. 8 shows movement in the diagonal right downward direction and Fig. 9 shows movement in the diagonal right upward direction. Fig. 10, Fig. 11, Fig. 12 and Fig. 13 shows movements in horizontal right, horizontal left, vertical up and vertical downward movements respectively.

V. Conclusion

We have proposed a myoelectric interface which can be used in prosthesis and robotic applications. The use of mapping functions makes the system independent of subject specific and task specific prosthesis systems. More importantly, it is evident that there is no need for user-specific decoders as long as the user has the knowledge about the mapping function exiting between neural activity and control task command. The system uses simple electronic components which are readily available and is cost-effective. Efficiency of the system can be improved by increasing more number of channels which increases number of simultaneous movements. The system can further be used in various fields such as Physiotherapy, Gaming and Entertainment.

References

[1] Parker, P., Englehart, K. & Hudgins, B., “Myoelectric Signal Processing for Control of Powered Limb Prostheses”, Journal of Electromyography and Kinesiology, Vol. 16, Issue 6, 541-548, December 2006.
[2] Merletti, R. & Parker, P., “Electromyography: Physiology, Engineering and Noninvasive Applications”, IEEE Press Series on Biomedical Engineering, 460-464, 2004
[3] Kampas,P. Myoelektroden—Optimal eingesetz Medizinisch — Ortohopäsdische Technik 2001, 1, 21-27.
[4] Zardosthi-Kermani, M.; Wheeler, B.C.; Badie, K.; Hashemi, R.M. EMG feature evaluation for movement control of upper extremity prostheses. IEEE Trans. Rehab. Eng. 1995, 3, 324-332.
[5] Englehart, K.; Hudgins, B.; Parker, P.A. A wavelet-based continuous classification scheme for multifunction myoelectric control. IEEE Trans. Biomed. Eng. 2001, 48, 302-311.
[6] Huang, Y.; Englehart, K.B.; Hudgins, B.; Chan, A.D.C. A Gaussian mixture model based classification scheme for myoelectric control of powered upper limb prostheses. IEEE Trans. Biomed. Eng. 2001, 52, 1801-1811.
[7] S. Bitzer and P. van der Smagt, “Learning EMG control of a robotic hand: Towards active prostheses,” in Proc. IEEE Int. Conf. Robot. And Automat. ICRA, May 2006, pp. 2819–2823.
[8] D. Nishikawa, W. Yu, H. Yokoi, and Y. Kakazu, “EMG prosthetic hand controller using real-time learning method,” in Proc. IEEE Int. Conf. Syst., Man, Cybern., 1999, vol. 1, pp. 153–158.
[9] M. S. Erkilinc and F. Sahin, “Camera control with EMG signals using principal component analysis and support vector machines,” in Proc. IEEE Int. Syst. Conf. (SysCon), Apr. 2011, pp. 417–421.
[10] G. Naik, D. Kumar, and Jayadeva, “Twin SVM for gesture classification using the surface electromyogram,” IEEE Trans. Inform. Technol. Biomedicine, vol. 14, no. 2, pp. 301–308, Mar. 2010.
[11] F. Orabona, C. Castellini, B. Caputo, A. Fiorilla, and G. Sandini, “Model adaptation with least-squares SVM for adaptive hand prosthetics,” in Proc. IEEE Int. Conf. Robot. Automat., May 2009, pp. 2897–2903.
[12] M. V. Liarokapis, P. K. Artemiadis, P. T. Katsiaris, K. J. Kyriakopoulos, and E. S. Manolakos, “Learning human reach-to-grasp strategies: Towards EMG-based control of robotic arm-hand systems,” in Proc. IEEE Int. Conf. Robot. Automat. (ICRA), May 2012, pp. 2287–2292.
[13] P. K. Artemiadis and K. J. Kyriakopoulos, “EMG-based teleoperation of a robot arm in planar catching movements using ARMAX model and trajectory monitoring techniques,” in Proc. IEEE Int. Conf. Robotics Automat., 2006, pp. 3244–3249.
[14] P. K. Artemiadis and K. J. Kyriakopoulos, “An EMG-based robot control scheme robust to time-varying EMG signal features,” IEEE Trans. Inform. Technol. Biomedicine, vol. 14, no. 3, pp. 582–588, May 2010.
[15] A. B. Ajiboye and R. F.Weir, “Muscle synergies as a predictive framework for the EMG patterns of new hand postures,” J. Neural Eng., vol. 6, no. 3, p. 036004, Jun. 2009.
[16] F. E. Zajac, “Muscle and tendon: Properties, models, scaling, and application to biomechanics and motor control,” Stanford Univ., vol. 17, p. 52, 1989.
[17] S. M. Radhakrishnan, S. N. Baker, and A. Jackson, “Learning a novel myoelectric-controlled interface task,” J. Neurophysiol., vol. 1, p. 47, 2008.

[18] S. M. Chase, A. B. Schwartz, and R. E. Kass, "Bias, optimal linear estimation, and the differences between open-loop simulation and closed-loop performance of spiking-based brain-computer interface algorithms," Neural Networks, vol. 22, no. 9, pp. 1203–1213, Nov.2009.
[19] T. Pistohl, C. Cipriani, A. Jackson, and K. Nazarpour, "Abstract and proportional myoelectric control for multi-fingered hand prostheses," Anna. Biomed. Eng., vol. 41, no. 12, pp. 2687–2698, Dec. 2013.
[20] Advancer Technologies Muscle Sensor Kit V3 manual https://github.com/AdvancerTechnologies/Muscle-Sensor-v3/blob/master/Documents/Muscle%20Sensor%20v3%20Users%20Manual.pdf?raw=true
[21] Arduino UNO https://www.arduino.cc/en/Main/ArduinoBoardUno
[22] OpenGL Tutorial http://www.opengl-tutorial.org/

IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) is UGC approved Journal with Sl. No. 5016, Journal no. 49082.

M. Naga Malleswari. "A Prototype for Human Computer Interactraction Using Myoelectric Interface." IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), vol. 12, no. 4, 2017, pp. 29–35.