Reever Control: A Biosignal Controlled Interface

Vinicius Horn Cene, Gabriela Favieiro, Luciana Nedel, and Alexandre Balbinot.

Abstract— **With the development of new technologies, new accessibility solutions have emerged to increase the inclusion of user with disabilities. This paper details the development of *Reever Control*, an interface based on inertial sensors and sEMG signal processing to control a cursor in a virtual environment. The metrics of *time*, *number of false clicks* and *average absolute error* were used to characterize the system and compare it with *Camera Mouse*, an image processing-based interface. The *Reever Control* showed improvements compared to uniaxial control, robustness to artifacts, support of right click and runtime compatible with Camera Mouse.**

I. Introduction

The use of surface electromyography (sEMG) as a control interface has been studied since 40's [1]. One great focus of interest in sEMG signal processing is the signal classification in order to activate prosthetic limbs that mimic the natural gesture performed by an amputee subject. Due to its stochastic nature, the processing of sEMG for movement characterization generally involves a considerable Computational Intelligence (CI) complexity in terms of pre-processing methods [2]–[4], feature selection [3] and classifiers such as Artificial Neural Networks (ANNs) [4], Support Vector Machines [5], Logistic Regression [3], [6] among others. The sEMG signals is frequently used along other sensing techniques such as inertial sensors [7].

Micro-Electro-Mechanical Systems (MEMS) combine computational components with microscopic mechanics such as sensors, valves, cogwheels, mirrors and actuators incorporated into integrated circuits. Accelerometers of capacitive and piezoresistive effect as well as gyrometers make part of the large family of MEMS sensors devices and are widely used on biomechanics sensory systems [8].

Considering this two types of sensors, in this paper we propose an interface which combines those two biological signals designed (but not exclusive) for individuals with severe motion restrictions. The proposed interface that we compared with *Camera Mouse* was named *Reever Control*.

Vinicius Horn Cene and Gabriela Favieiro are with the Federal University of Rio Grande do Sul (UFRGS) at Electrical-Electronic Instrumentation Laboratory (IEE), Porto Alegre, RS Brazil (+55) 51- 3308-4441; e-mail: vinicius.cene@gmail.com).

Luciana Nedel is associate Professor of the Federal University of Rio Grande do Sul (UFRGS), at Human-Computer Interaction Laboratory (Informatics Institute), Porto Alegre, RS Brazil (+55) 51 3308-7037; e-mail: nedel@inf.ufrgs.br).

Balbinot, A. is Titular Professor of the Federal University of Rio Grande do Sul (UFRGS), coordinator of Electrical-Electronic Instrumentation Laboratory (IEE), Porto Alegre, RS Brazil (+55) 51- 3308-4441; e-mail: alexandre.balbinot@gmail.com).

II. Methodology

The *Reever Control* consists of the set of an inertial sensor positioned on the flap of a cap used by the subject for the mouse (cursor) control and two sEMG channels extracted from the left and right cheek in order to perform the left and right click, respectively. The movements which control the cursor and the electrode placement are presented in the Fig. 1.

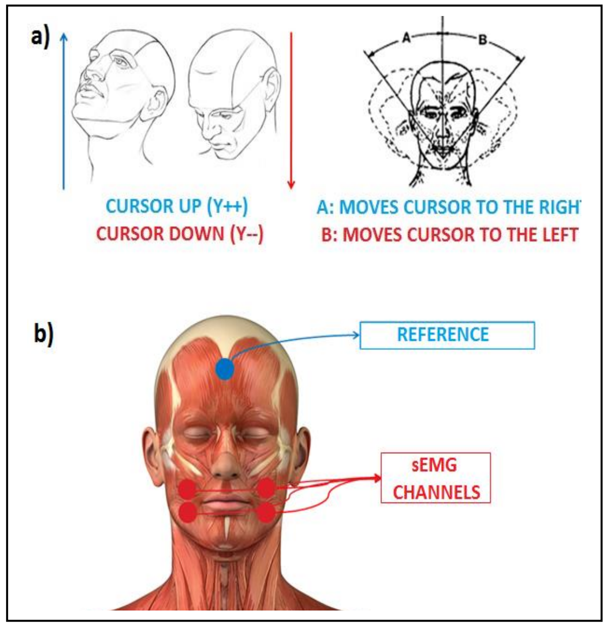

Figure 1. System control: a) cursor movement and b) mouse clicks.

While the vertical cursor movements are performed through affirmation head movements (up and down), the horizontal movements are performed through the head rotation (tilting the head left and right). The mouse clicks are performed through activations obtained by surface electromyography (sEMG) signals. The signals of sEMG derived from the *medial pterygoid muscles* on both sides of the face, as shown in Fig. 1.b. Thus, the mouse clicks could be performed by flexing the muscle in a gesture similar to a smile and the commands for interaction with the computer depend exclusively on signals from head movements and muscle activation. All data processing was embedded on an Arduino DUE platform, which is connected to a computer via an USB port.

A. Data Acquisition

The sEMG data acquisition was performed through disposable surface electrodes and a commercial EMG system (*EMG 830 C da EMG System do Brasil*) which has eight EMG channels with a combined gain of 2000 times and a 4th Order Butterworth bandpass-filter (20-450 Hz). The sample frequency of the sEMG signal was 1 kHz with 10 bits of quantization.

The inertial sensor used to capture the head movements was the *MPU-6050*™ from *Invensense*. The sensibility of de *250 DPS* (*Degrees Per Second*) and ± 8*g* was set in the MPU sensor for the gyroscope and accelerometer, respectively.

B. sEMG Data Processing

The sEMG signal passed through a full-wave rectification and thereafter the moving average of the signal for 100 samples (sliding-windows with no overlap) was extracted directly on the binary value of A-D conversion of the system (0-1024 for 10 bits). The threshold of the system was generated automatically from the first 100 samples (at a rest condition) for each subject.

C. MPU-6050 Data Processing

The head movements were tracked by the Pitch and Roll angles calculated through the coordinate transformations as presented on (1) and (2), respectively.

$$Pitch = arctan\left(\frac{G_y}{\sqrt{{G_x}^2+{G_z}^2}}\right) \quad (1)$$

$$Roll = arctan\left(\frac{-G_x}{G_z}\right) \quad (2)$$

where G_x, G_y and G_z are the coordinates calculated on *x*, *y*, and *z* axis.

Complementary, a digital *low-pass filter*, described on (3), was used to remove artifacts of the inertial sensors.

$$Y_t = \alpha \times X_t - (1 - \alpha) \times Y_{t-1} \quad (3)$$

where $\boldsymbol{Y_t}$ is the output filtered signal, $\boldsymbol{Y_{t-1}}$ is the previous output signal, $\boldsymbol{X_t}$ is the mean accelerometer value and $\boldsymbol{\alpha}$ is the filter coefficient (0.5 in this application).

D. Camera Mouse

Camera Mouse is freeware tool that allows the control of the mouse cursor on a Windows environment using a computer vision approach just by moving the user's head (for the tests, the tip of the nose was used as reference) [9]. The mouse clicks are performed every time the cursor is kept still for a pre-defined amount of time (2 seconds, in this paper). Both the motion sensitivity and the click time can be adjusted. However, the click is restricted only to the left click, and there is no possibility to perform a right click without other auxiliary software. The *Camera Mouse* was the reference interface that against *Reever Control* was compared.

E. Arduino DUE Embedded Software

Considering the four inputs offered by the system (two sEMG channels, *Pitch* and *Roll*), all the signal processing is performed on the Arduino platform, as presented on Fig. 2. With the signal processing, it is possible to identify the mouse movement commands for the *x* and *y* coordinates and the activation of both mouse buttons. The Fig. 2 presents the flowchart of the embedded software on the Arduino board. The Arduino DUE already has s native support to function as a PC peripheral and has a mouse library already developed in the IDE itself.

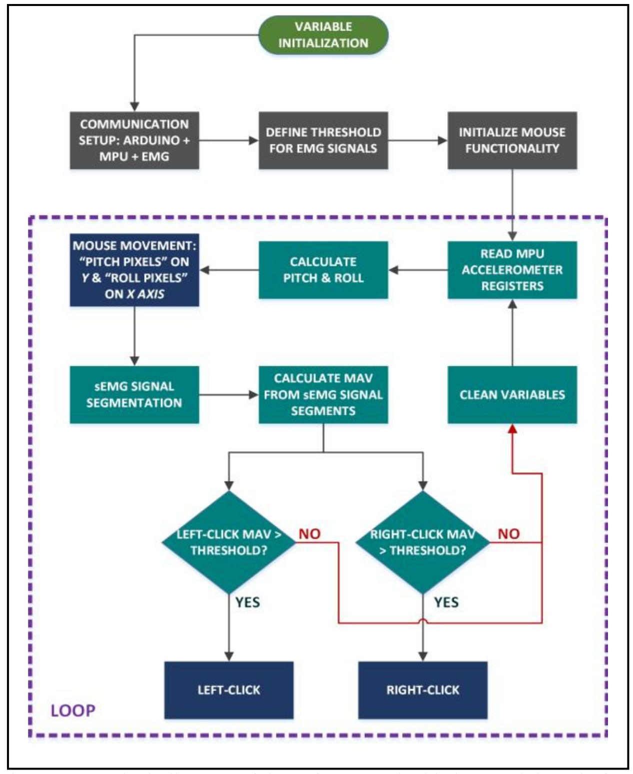

Figure 2. Block diagram of the software embedded on Arduino platform.

F. Assays

In order to familiarize the users with both interfaces, 5 minutes of "freestyle use" of the *MS-Paint Brush* were given to each one of the 10 subjects as a prelaminar interaction. Thereafter, they were directed to a *GUI* interface developed in *Matlab* and asked to perform five random repetitions of the tests involving four different shapes and sizes, as presented on Fig. 3.

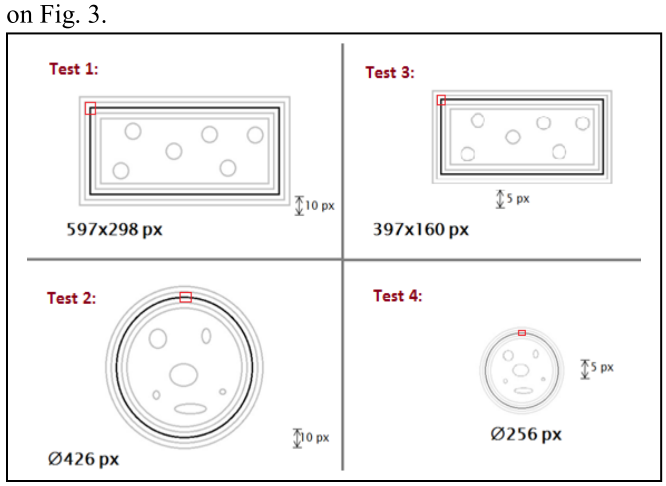

Figure 3. Four different geometries and sizes tested on the assays used to check the response/sensibility of each interface to different scenarios.

For each test, the users were asked to *left-click* inside the red rectangle to start and follow the black bold line as best as possible in a *clock-wise* movement. Once they reached the red rectangle again, they were asked to *right-click* inside it to finish the test. With the *Camera Mouse*, the *right-click* was performed with the *mouse*, since the interface has the restriction of not allow switching of buttons (*left/right click*). Each subject performed five repetitions of each assay to test the influence of training in the control of the device.

G. *Evaluation*

The evaluation of the system was performed based on the response variables: *total time* (s); *number of clicks*; *total absolute error* (pixels) and *mean absolute error* (pixels).

The *total time* variable corresponds to the time between the *left-click* (beginning of the test) and *right-click* (end of the test). The *number of clicks* variable is the total number of the clicks performed during the assay (ideally two), that metric intends to detect "*false-clicks*" occurrence. The *error* variable is the absolute difference from the reference coordinates and those made by the users. For the tests 1 and 3 (rectangles), the error is the difference of the closest x or y axis, as presented on Fig. 4.a and for the tests 2 and 4 (circles) the error is calculated by the equation (4), as presented on Fig. 4.b. The *total error* is the sum of absolute error through the test while the *mean absolute error* is the total error divided by the total pixels of the trajectory.

$$CAE = \left|\sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} - Ref_{Radius}\right| \quad (4)$$

where CAE is the Circular Absolute Error and Ref_{Radius} are the values of reference coordinates considering the circle center (x_0, y_0) and the angle variation with a constant radius.

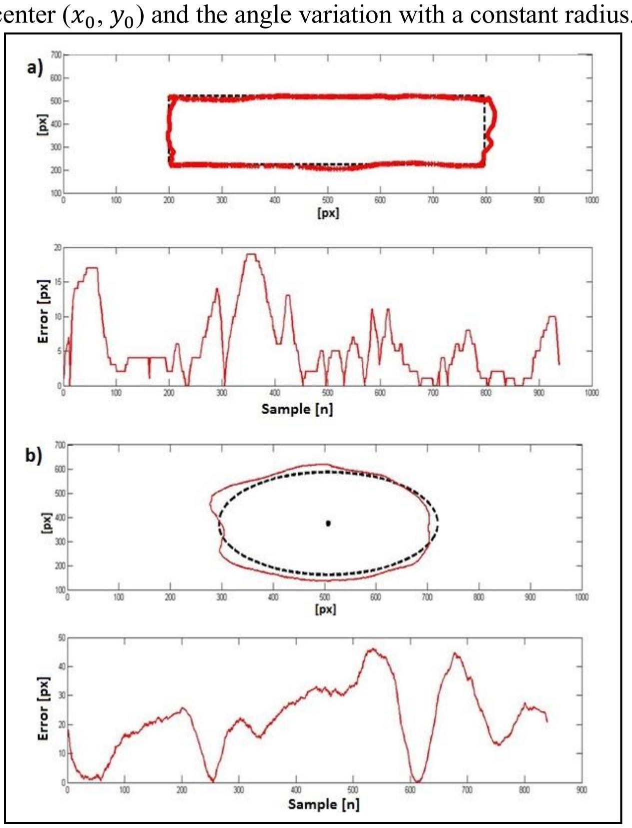

Figure 4. Error measurent on a) Rectangle (test 1) and b) Circle (test 2).

III. Results and Discussion

Different ANOVA statistical tests (*p=0,05*) were performed for each one of the response variables of the system (presented on Table I) in order to determine which parameters among *subject*, *method*, *test* and *repetition* could affect the time, error and click numbers significantly. The ANOVA analysis showed that the four different factors affect significantly the *time* response of the system, the same occurred for the *mean absolute error* response. As expected, the *number of the clicks* is affected by *subject*, *method* and *repetition*, but not for the variation of the *tests* performed. The Table I presents the mean results of the tests performed.

TABLE I. Mean Values of The Tests for The Three Different Response Variables.

Variable	Camera Mouse	Reever Control
Time	**18.02 ± 1.10**	**22.67 ± 0.90**
Number of Clicks	**3.80 ± 0.41**	**2.03 ± 0.03**
Mean Absolute Error	**11.46 ± 0.90**	**15.10 ± 0.70**

Although a small difference is observed, the *Camera Mouse* interface offers greater speed in the tasks completion. However, it is pertinent to note that the *Camera Mouse* does not support *right-clicks*, which were performed using the PC mouse and certainly caused a gain in time performance. The sensitivity of the *Camera Mouse* click was set to 2 seconds, considering that the final click would also take at least another 2 seconds and the standard deviations, the times between the two interfaces begin to be equivalent and it is no longer possible to define a clear advantage in favor of the *Camera Mouse*. Therefore, in relation to time, it is correct to confirm that the two interfaces offer a compatible time.

The *Reever Control* has a greater robustness to the occurrence of false clicks. In fact, this is one of the biggest advantages seen in comparison with *Camera Mouse*. The *Camera Mouse* has a sensitivity adjustment in movement and click time. Finding the balance between the cursor shift sensitivity and the idle time required for the click action is a major challenge. Once the sensitivity is too high, it is impossible to control the cursor. Therefore, for the execution of these tests, one of the lowest sensitivities available for the interface was chosen. However, when the movement is performed carefully, the movement intention is not detected properly and the click action is activated, causing a false click. This problem occurred even when the second largest "clicking-time" (2 seconds) was chosen. The *Reever Control* presents another advantage which is to define the threshold of clicks automatically, and, moreover, enables *left/right* and *double-clicking* at a good artifact rejection rate.

The absolute error measurement aims to describe how precisely the user was able to follow the proposed test path. It is worth to highlight that an accurate tracking of the trajectory is extremely difficult to perform, even by an experienced user using a mouse. Despite the *Camera Mouse* show slightly better performance, those results are composed of the means of the four tests performed on both methods. To better understand the behavior of both methods, the Table II presents the results of both of them on the four different tests from which is perceptible the advantage of

Reever Control for linear tracking and the Camera Mouse advantages on the circle shape object.

TABLE II. MEAN VALUES ERROR FOR THE FOUR DIFFERENT TESTS.

Method	Test 1	Test 2	Test 3	Test 4
Camera Mouse	**9.2 ± 1.1**	**19.3 ± 1.6**	**7.2 ± 0.6**	**10.1 ± 1.1**
Reever Control	**6.1 ± 0.8**	**29.7 ± 2.6**	**6.2 ± 0.7**	**18.4 ± 2.4**

The result presented helps to explain the characteristics of *Reever Control* observed during the tests: a) High stability for uniaxial displacements (only in x or y), which makes it very precise to perform rectangular movements such as those of tests 1 and 3; b) Diagonal movements are very difficult to perform, making it difficult to follow the paths proposed in tests 2 and 4, which can be evidenced by the relatively large error recorded.

At the same time, the *Camera Mouse* performs uniaxial paths relatively well and offers a reasonable control for tracking curved trajectories. Although this feature is not as essential in using the cursor for controlling an operating system, it is something to consider.

IV. CONCLUSION

The *Reever Control*, an interface for people with severe mobility limitations is presented and from tests involving different geometric shapes of different sizes the characterization of the system is performed in comparison to *Camera Mouse* as a reference interface. The evaluations of the interfaces focused on the time taken to perform the tasks, the number of false clicks that occurred during the tests, the average error in relation to the proposed path and the evaluation of the type of form used in the test.

Regarding the average *time* taken to perform the tasks, the interfaces present similar performance, although when the time for the click performed by the Camera Mouse is discarded, it presents a slight advantage.

The *Reever Control* shows a significantly superior performance in relation to *false-clicks*, almost without occurrences. *False-clicks* events are quite frequent in the *Camera Mouse*, even when using the maximum delay for the activation of the click command (2 seconds).

Camera Mouse has a slight advantage the average in relation to cumulative *error* attributed in large part to its performance when following curves, a difficult task to be fulfilled with *Reever Control*. However, *Reever Control* showed very good stability in uniaxial movements (difficult to achieve even by experienced users using a mouse).

Both interfaces still require improvements. From the current development stage we can highlight the great precision in uniaxial movements offered by *Reever Control* as well as its great robustness to artifacts like *false-clicks*. In contrast, *Camera Mouse* offers good omnidirectional control and no need of external devices. The practicality of the system is counter-balanced by frequent artifacts occurrence and a fragile calibration process, which often loses reference and makes the control process unfeasible.

As additional considerations, it is worth mentioning that both interfaces still have considerable limitations. The use of electrodes in *Reever Control* may offer some discomfort when used for a long period. Moreover, *Reever Control* is connected to the PC through a USB port and a wireless communication system would offer more practicality.

Camera Mouse, is an interface based on image processing, thus, it depends severely from many environmental conditions, the lighting, the camera and the pc used and also have the limitation that the camera must stand in front of the user, always. The simple act of turning the lights out turns the use of *Camera Mouse* completely unfeasible. Another serious limitation is that the Camera Mouse performs the click action incessantly whenever the cursor is left in an approximate position, which impedes user's simple activities, such as watching some video in a media player, without pause the player incessantly by false-clicking the screen. The occurrence of false clicks is one of the biggest problems of this interface as well as the absence of the right click. Another problem observed was the loss of reference of the image, forcing the users to recalibrate the system.

ACKNOWLEDGMENT

The authors would like to kindly acknowledge Professor Leia Bagesteiro for the sEMG device used in this paper and also all the volunteers who participated.

REFERENCES

[1] M. Ison and P. Artemiadis, "The Role of Muscle Synergies in Myoelectric Control: Trends and Challenges for Simultaneous Multifunction Control," *J. Neural Eng.*, vol. 11, no. 5, p. 51001, 2014.

[2] P. Geethanjali, "Comparative study of PCA in classification of multichannel EMG signals," *Australas. Phys. Eng. Sci. Med.*, vol. 38, no. 2, pp. 331–343, 2015.

[3] V. H. Cene and A. Balbinot, "Optimization of Features to Classify Upper - Limb Movements Through sEMG Signal Processing," *Brazilian J. Instrum. Control*, vol. 4, no. 3, pp. 14–20, 2016.

[4] V. H. Cene, G. Favieiro, and A. Balbinot, "Upper-limb movement classification based on sEMG signal validation with continuous channel selection," *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, vol. 2015–Novem, pp. 486–489, 2015.

[5] X. Zheng, W. Chen, and B. Cui, "Multi-gradient Surface Electromyography (SEMG) Movement Feature Recognition Based on Wavelet Packet Analysis and Support Vector Machine (SVM)," pp. 1–4, 2011.

[6] V. H. Cene and A. Balbinot, "Upper-Limb Movement Classification Through Logistic Regression sEMG Signal Processing," no. 3, 2015.

[7] M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A.-G. M. Hager, S. Elsig, G. Giatsidis, F. Bassetto, and H. Müller, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Sci. Data*, vol. 1, 2014.

[8] A. Moschetti, L. Fiorini, D. Esposito, P. Dario, and F. Cavallo, "Recognition of Daily Gestures with Wearable Inertial Rings and Bracelets," *Sensors*, vol. 16, no. 8, p. 1341, 2016.

[9] M. Betke, J. Gips, and P. Fleming, "The Camera Mouse: Visual tracking of body features to provide computer access for people with severe disabilities," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 10, no. 1, pp. 1–10, 2002.