Investigating the feasibility of a BCI-driven robot-based writing agent for handicapped individuals

Chanan S. Syan¹, Randy E. S. Harnarinesingh and Rishi Beharry
Department of Mechanical and Manufacturing Engineering
The University of the West Indies
St. Augustine, Trinidad

E-mail: chanan.syan@sta.uwi.edu

Abstract. Brain-Computer Interfaces (BCIs) predominantly employ output actuators such as virtual keyboards and wheelchair controllers to enable handicapped individuals to interact and communicate with their environment. However, BCI-based assistive technologies are limited in their application. There is minimal research geared towards granting disabled individuals the ability to communicate using written words. This is a drawback because involving a human attendant in writing tasks can entail a breach of personal privacy where the task entails sensitive and private information such as banking matters. BCI-driven robot-based writing however can provide a safeguard for user privacy where it is required.

This study investigated the feasibility of a BCI-driven writing agent using the 3 degree-of-freedom Phantom Omnibot. A full alphanumerical English character set was developed and validated using a teach pendant program in MATLAB. The Omnibot was subsequently interfaced to a P300-based BCI. Three subjects utilised the BCI in the online context to communicate words to the writing robot over a Local Area Network (LAN). The average online letter-wise classification accuracy was 91.43%. The writing agent legibly constructed the communicated letters with minor errors in trajectory execution. The developed system therefore provided a feasible platform for BCI-based writing.

1. Introduction

A Brain-Computer Interface (BCI) is a system that allows users to control external devices using only their inherent brain activity. BCI users are required to perform mental or physical tasks that embody device commands. For example, users can focus on directional stimuli on a computer screen in order to issue motive commands to a mouse cursor. BCIs are responsible for recording brain data and identifying the resulting modulation of brain features that signify user intention. BCIs are an attractive option for disabled individuals because they rely on cognitive ability without necessarily requiring sensorimotor functions [1].

BCIs have been used to control virtual keyboards thereby allowing disabled individuals an avenue for communication [2,3]. However, the current work on BCI-based physical writing systems is limited. Human attendants can assist in writing tasks for disabled individuals. However, this can entail a breach of privacy where the task entails sensitive information such as personal letters and legal contracts. BCI-based writing therefore provides a safeguard for the user’s privacy where it is required.

¹ To whom any correspondence should be addressed.
A previous study utilized the 5-axis Mitsubishi RV-2AJ robot [4] for BCI-based writing. The robot was used to write letters in real time over the internet using a BCI-based speller. This paper investigates the feasibility of a BCI-based writing agent using the lower cost 3-axis Quanser Phantom Omnibot for a full alphanumerical English character set.

The BCI is based on the P300 signal. The P300 is positive deflection in the Electroencephalogram (EEG) that is evoked approximately 300-400ms following the presentation of a rare, deviant or target stimulus [5]. The P300 wave can be evoked due to visual, auditory or somatosensory stimulation. However, this study uses the visually evoked P300.

2. BCI Experimental Study

2.1. Subject demographics
Three subjects hereinafter referred to as S1, S2 and S3 each participated in 1 training session and 1 testing session using the P300 Speller. All of the subjects were male and had normal or corrected-to-normal vision. Subjects S2 and S3 were naïve to BCI experiments and Subject S1 was non-naïve.

2.2. Data Recording Circumstances
The g.tec g.MOBILAB was used to record EEG at 8 sites according to the 10/20 electrode placement guideline at a sampling frequency of 256 Hz. Subject EEG cap sizes were chosen based on head circumference and the CZ position was centred along the Nasion to Inion points and the left to right pre-auricular areas. The 8 EEG sites used for data collection were P7, P3, P4, P8, FZ, CZ, PZ and OZ. Electrical reference was taken at the left mastoid and ground was taken at the Inion.

2.3. Stimulus Presentation Paradigm
The Stimulus Presentation Paradigms (SPPs) used for stimulus delivery were programmed as Graphical User Interfaces (GUIs) in MATLAB® 7.8.0.347. Subjects were seated approximately 80cm from a 22-inch computer monitor in a dimly lit room for the experiments.

2.3.1. Training session. Training sessions were used in order to obtain features to train classifiers to recognise subject-specific P300 responses. The GUI used for training sessions is presented in Figure 1. The matrix consisted of 36 grey alphanumerical elements against a white background. Target characters were highlighted in red for 2 seconds and subjects were instructed to mentally count the number of times the target characters were evoked in the subsequent flashing sequence.

Each character of the matrix was highlighted in black for 100ms followed by a 75ms Inter-Stimulus Interval in which no character was highlighted. The order of character presentation was random and the presentation of all 36 characters is referred to as a trial. Five trials were used to communicate each target character. Each character appeared once as the target. This generated a total of 36 5-trial segments.
2.3.2. Testing sessions. The trained classifiers were implemented in online testing sessions in which the subjects selected characters on their own volition. In these testing sessions, the BCI had no prior knowledge of subject intention. This provided a basis for the unbiased determination of classifier performance. The GUI used for online testing session is presented in Figure 2.

![Stimulus Presentation Paradigm GUI for Testing session](image)

**Figure 2.** Stimulus Presentation Paradigm GUI for Testing session
Preliminary simulations were performed to select the appropriate number of trials to utilise for online testing for each subject. Three trials of online presentation were used to communicate characters for Subject S1. However, 5 trials of online presentation were used for Subjects S2 and S3. The classified letters were sequentially ordered in a text box at the top of the GUI. Eight letters were communicated in total to send to the writing robot. A “backspace” functionality was programmed into the GUI to allow for deletion of misclassified letters to ensure that only the intended letters were written by the robot.

2.4. Pre-processing, Feature Extraction and Pattern Classification
The EEG was collected raw with no pre-processing applied for the offline sessions. N-fold Cross-Validation (NFCV) was used to grid search pre-processing and feature extraction parameters for Fishers Linear Discriminant Analysis (FLDA) classifiers for Subjects S1-S3. The NFCV parameter search is described further.

Each EEG channel was forward-reverse filtered using a digital Butterworth filter of order $n$ with 3DB cut-off frequencies at 1Hz and 12Hz respectively. The 500ms segments subsequent to stimulus delivery were down-sampled by a factor of $K$ to create a set of temporal features for each channel. The temporal features were combined across the 8 channels to obtain a single spatiotemporal feature vector for each post-stimulus segment. The values of $n$ and $K$ were determined using a NFCV search on the offline data with N=3 for single trial P300 classification using FLDA.

3. Robot Forward Kinematics
Forward kinematic modelling was utilised to determine the end-effector position of the Omnibot I terms of the individual joint angles. The coordinate frame assignment for the Phantom Omnibot is presented in Figure 3.

![Figure 3. Phantom Omnibot coordinate frame assignments](image)

The Denavit-Hartenberg (DH) [6] parameters for the Omnibot joints are presented in Table 1.
Table 1. DH joint parameters for Omnibot

| Link | a (m) | d (m) | α    | Θ    |
|------|-------|-------|------|------|
| 1    | 0     | 0     | -π/2 | q1   |
| 2    | 0.132 | 0     | 0    | q2   |
| 3    | 0.132 | 0     | 0    | q3-π/2 |

The DH matrix relating joint ‘i-1’ to joint ‘i’ takes the form shown in (1).

\[
H_{i-1}^i = \begin{pmatrix}
\cos \theta & -\sin \theta \cos a & \sin \theta \sin a & \cos \theta \\
\sin \theta & \cos \theta \cos a & -\cos \theta \sin a & \sin \theta \\
0 & \sin a & \cos a & d \\
0 & 0 & 0 & 1
\end{pmatrix}
\] (1)

In this study, a whiteboard marker was fixed to the final joint of the Omnibot. The marker tip can be referenced to the base frame of the robot using the Forward Kinematic model as shown in (2). The coordinates of the marker tip referenced to joint 3 are given as tipₙ, tipᵧ, and tipₚz.

\[
pos = H_{0}^3H_{1}^3H_{2}^3 \begin{pmatrix}
tip_{x} \\
tip_{y} \\
tip_{z}
\end{pmatrix}
\] (2)

4. Robot Inverse Kinematics and Controller Design

A flat whiteboard surface was used for character writing. The vertical z-coordinate of the whiteboard was determined by measurement. This allowed for the specification of the xy coordinates of character points whilst leaving the z-coordinate fixed. Finding the joint angles that satisfy the forward kinematics matrix multiplication for a given end-effector position solves the inverse kinematics problem.

A Proportional-Derivative (PD) controller was used to obtain the joints angles identified by Inverse Kinematics. The PD position controller for a joint is provided in Figure 4 where ‘J’ and ‘B’ are the inertial and frictional terms respectively. The placement of the derivative block in the inner loop averts the generation of impulses due to step changes in the set-point.

![Figure 4. Feedback control block diagram for joint positioning](image-url)

The controllers for 3 Omnibot joints were designed to meet the specifications in (3) and (4).
Percentage Overshoot (PO) = 25%  
Peak time (tp) = 0.14s  

The closed loop transfer function for the feedback loop in Figure 4 is given in (5).

\[
\frac{\theta_m}{\theta_d} = \frac{K_p}{s^2 + \left(\frac{B + K_d}{f}\right)s + \frac{K_p}{f}}
\]

This is recognized as a reference 2nd order transfer function with natural frequency \(\omega_n\) and damping ratio \(\zeta\) whose relationships with the closed loop parameters are given in (6) and (7).

\[
\omega_n^2 = \frac{K_p}{j}
\]

\[
2\zeta\omega_n = \frac{B + K_d}{j}
\]

The peak overshoot, \(M_p\) and peak time, \(t_p\) for a second order system are given in (8) and (9).

\[
M_p = e^{-\pi\zeta}
\]

\[
t_p = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}}
\]

This permits the simultaneous solution of \(\zeta\) and \(\omega_n\) using the design criteria specified in (3) and (4)

\[
\zeta = \frac{\ln 4}{\sqrt{\pi^2 + (\ln 4)^2}} = 0.404
\]

\[
\omega_n = \frac{\pi}{0.14\sqrt{1-\zeta^2}}
\]

The PD parameters were obtained using Equations (6)-(11) are presented in Table 2.

| Link | J (kgm²) | B (Nmsrad⁻¹) | Kp (Nmrad⁻¹) | Kd (Nmsrad⁻¹) |
|------|---------|--------------|-------------|-------------|
| 1    | 0.0031  | 0.0089       | 1.86        | 0.0525      |
| 2    | 0.0022  | 0.0170       | 1.32        | 0.0266      |
| 3    | 0.0009  | 0.0058       | 0.54        | 0.012       |

5. Character generation

The alphanumerical characters were defined using a set of xy coordinates on the whiteboard that were spatially interpolated to traced the desired character. The characters points were referenced to the base frame of the robot. The Simulink program determined the intermediate points between the user-defined points at equally spaced intervals for the purpose of interpolation. These points were input to
the inverse kinematics system for the determination of joint angles and the PD controller was used to realize the required joint angles. The representative plots for the ‘E’ and ‘J’ characters are shown in Figure 5.

![Figure 5. MATLAB Character generation of ‘E’ and ‘J’](image)

6. Results
The NFCV searched parameters for Subjects S1-S3 are presented in Table 3.

| Subject | K | n |
|---------|---|---|
| S1      | 10 | 1 |
| S2      | 12 | 1 |
| S3      | 16 | 2 |

The online BCI accuracies obtained for letter classification are presented in Table 4.

| Subject | Number of letters correctly communicated | Number of misclassified letters | Total number of letters communicated | Percentage classification accuracy |
|---------|-----------------------------------------|-------------------------------|-----------------------------------|----------------------------------|
| S1      | 24                                      | 2                             | 26                                | 92.31%                           |
| S2      | 16                                      | 3                             | 19                                | 84.21%                           |
| S3      | 24                                      | 1                             | 25                                | 96.00%                           |
| TOTAL   | 64                                      | 6                             | 70                                | 91.43%                           |
Words were sent to the writing robot over the LAN to another computer connected to the Omnibot after the successful classification of 8 letters for Subjects S1-S3. Figures 6-8 present the Omnibot realisation of 8 letter words and phrases communicated by Subjects S1-S3.

Figure 6. Omnibot generation of ‘POSTPONE’ communicated by Subject S1

Figure 7. Omnibot generation of ‘FISHFARM’ communicated by Subject S2
7. Discussion
The average online BCI classification accuracy for Subjects S1-S3 was 91.43%. Subject S3 attained an online classification accuracy of 96% which was significant for a naïve BCI subject. In contrast, Subject S1 was non-naïve and attained an accuracy of 92.31%. This seemed like a counter-intuitive result since naïve subjects in general perform worse at BCI usage than non-naïve subject. However, Subject S1 used fewer trials for online communication than Subject S3. Subject S1 used 3 trials for online communication where Subject S3 used 5 trials. Trial averaging has been shown to reduce misclassification errors in BCI [7] and provides an explanation for the observed online accuracies.

The 3-axis Omnibot was programmed to write 36 English characters consisting of 25 letters, numbers 0-9 and a hyphen. The robot-written characters were legible for the developed character set. However, curved realization of linear characters segments were observed for Omnibot written characters. This can be seen in Figures 6-8 and can be attributed to multiple reasons.

First, the transience inherent to second order controller response manifests itself as errors in trajectory execution. Second, the controllers were designed in the continuous-time domain however the MATLAB/Simulink program was discrete-time. The approximations in controller design can also manifest themselves as trajectory errors. Finally, the whiteboard surface used was not perfectly 2-dimensional. Ridges and sinks on the surfaces would have appeared as output perturbations for the control system which can degrade the response at those regions.

8. Conclusion
This paper investigated the feasibility of an online BCI-based robot writing agent using the 3-axis Phantom Omnibot. A full English alphanumerical character set was developed in MATLAB. The writing robot was interfaced to the BCI on another PC over a Local Area Network (LAN). Three subjects utilised the developed system to communicate words and phrases to the writing robot. The generated characters were legible with observed deviations in programmed trajectory due to controller
transience, approximation in controller design and uneven writing surfaces. These issues however did not significantly demerit the legibility of the written characters can be addressed for future implementation.

References
[1] Itturate I, Antelis J, Kubler A and Minguez J 2009 A Non-Invasive Brain-Actuated Wheelchair Based on a P300 Neurophysiological Protocol and Automated Navigation IEEE Transactions on Robotics 25(3) 617-627
[2] Andrews S, Palaniappan R, Teoh A and Kiong L C 2008 Enhancing P300 Component by Spectral Power Ratio Principal Components for a Single Trial Brain-Computer Interface American Journal of Applied Science 5(6) 639-644
[3] Citi L, Poli R, Cineland C and Supulveda F 2008 P300-Based BCI Mouse With Genetically-Optimized Analogue Control IEEE transactions on Neural Systems and Rehabilitation Engineering 16(1) 51-61
[4] Perez-Marcos D, Buitrage J A and Velasquez F D 2011 Writing through a robot: a proof of concept for a brain-machine interface Med. Eng. Phys. 33(10) 1314-1320
[5] Edlinger G, Wach P and Pfurtscheller G 1998 On the Realization of an Analytic High-Resolution EEG IEEE Transactions on Biomedical Engineering 45(6) 736-745
[6] Denavit J and Hartenberg R S 1955 A kinematic notation for lower-pair mechanisms based on matrices Trans ASME J. Appl. Mech. 23 215-221
[7] Pires G, Nunes U and Castelo-Branco M 2008 Brain Computer Interface Approaches to Control Mobile Robotic Devices Proceedings of the 11th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines (Coimbra, Portugal, 8-10 September 2008) pp 489-496