Brain-Actuated Robotics: A Logic-Based Approach for Multimodal Programming and Operation of Assistive Humanoid Robots

Abstract

Whether due to a missing limb or damage to the central nervous system, physical disabilities can limit or eliminate an individual's capability to perform volitional movements. Brain-computer interfaces (BCIs) can be used to bridge the biological gap from brain to body: assisting and enhancing user mobility.

This report demonstrates how non-invasive electroencephalography, electromyography, and voice commands can be used in a multimodal approach for developing flexible assistive technologies. Biosignals were induced via focused attention, jaw clenching, and eye blinking. The study also examines the use of a natural language user-interface capable of supporting programming robot actions and decisions. Further, it validates a framework for building intelligence into the BCI system, thereby allowing automated decision making and task performance.

Consistent classification results with accuracies as high as 99.17% and minimal user fatigue imply that that the four investigated BCI features (multimodal non-invasive brain signal generation, natural language interfacing, programming, and automated reasoning and decision making) are in fact essential facets for developing user-friendly, cost-effective assistive technologies for people with physical disabilities.

Introduction

According to the World Health Organization, over one billion people worldwide live with a disability, nearly 200 million of whom experience difficulties in physical functioning. Studies have revealed large gaps in service provision for those in needs of assistive devices, particularly in developing countries. For example, no more than 15% of the 70 million people who require the use of a wheelchair have access to one (World Health Organization & World Bank, 2011). This points to a pressing need across the globe to develop cost-effective technologies that can further enable mobility in people with physical disabilities. The long-term aim of the research reported herein is to build commercially-viable user-friendly assistive technologies that help people with physical disabilities perform day-to-day tasks effectively and independently.

The term “physical disability” is broadly encompassing. While some people are prevented from performing motor tasks because of a lost or missing limb, others retain damage to the central nervous system that may corrupt or entirely sever communication from the brain to other areas of the body. In both cases, the implementation of a brain-computer interface could serve to bridge this biological gap. A brain-computer interface (BCI) provides a direct control pathway from the user’s brain to an external device, such as a robotic limb. Since the term was coined by Jacques Vidal in 1973, the use of biosignals for directing electromechanical devices has been a subject of active research investigation (Vidal, 1973).

There are two distinct methods of recording the electrical signals that emanate from the human brain: invasive and non-invasive. The invasive method requires surgically implanting electrodes directly into the gray matter of the brain. The non-invasive method uses electrodes generally placed on the scalp, eliminating surgery cost and potential medical complications at the expense of signal quality (Rao, 2013). This recording method, called electroencephalography

(EEG), has been used extensively for BCI work. Electromyography (EMG) is another popular non-invasive technique that involves recording electrical signals that are produced by skeletal muscles. Although not all potential users retain voluntary muscle control, a majority of the people mentioned in the above statistic would be able to utilize EMG signals in conjunction with EEG signals to control assistive devices (World Health Organization & The World Bank, 2011). Listed below are several formative studies in the area of brain-computer interfacing.

With an invasive approach, researchers demonstrated the application of motor imagery signals for mapping the user's intent of how to move objects into robot arm actions. The user's brainwaves were recorded with an array of implanted microelectrodes as he was instructed to perform various mental visualizations of movement (Aflalo et al., 2015). This can be viewed as an extension of earlier research in which signals generated by the visualization of fist clenching were used to aid a tetraplegic subject in controlling a robotic arm (Hochburg et al., 2012).

Alternatively, some researchers have chosen to explore the non-invasive approach to brain-computer interfacing. The most widely used strategies for generating EEG-readable brain signals include visual evoked potentials, motor imagery, cognitive tasks, and focused attention. Perhaps the most intuitive of these for brain-computer interfacing is motor imagery. Interestingly, the mental visualization of movement alone causes neurons to fire in the motor cortex of the brain. Researchers have taken advantage of this phenomenon for brain-machine communication. For example, one study utilized motor imagery of clenching fists to control the four-directional movement of a quadcopter (LaFleur et al., 2013). This technique can also enable the user to distinguish between multiple classes of motor movement for both directional and speed controls (Sengupta, Bhattacharyya, & Janarthanan, 2011). To further reduce user effort, algorithms have also been employed to map few motor imagery states into a larger number of motion commands. This method was used to navigate a miniature humanoid robot (Chae, Jeong, & Jo, 2012).

Some researchers attempt to integrate the aforementioned techniques, among others, in a hybrid BCI system. One such investigation involved the use of motor imagery coupled with cognitive tasks such as mental math, word association, and focused attention to navigate a mobile robot (Millán, Renkens, Mouriño, & Gerstner, 2004).

To date, a significant majority of the work done in the area of brain-computer interfacing has demonstrated an average success rate of no more than 80%, despite most utilizing medical-grade EEG equipment for recording and processing signals. Clearly, a better performance has to be achieved if brain-computer interfacing is to expand its reach beyond research labs, becoming a viable tool for the millions of people who live with a physical disability.

In pursuing this goal, this work seeks to address several important questions that have yet to be resolved by the still growing field. For one, is it possible to incorporate natural language with BCI technologies so as to offer its users a more expressive and flexible user interface? This should also prove advantageous in being less mentally taxing to operate as it more closely mimics human decision making and thought processes. Further, how would it be possible for the user to program complex, personalized commands using his or her brainwaves in a system with preexisting controls? Lastly, in order to reduce mental taxation on the user, assistive technologies need to exhibit a certain level of intelligence for routine decision making and task execution. For example, an assistive wheelchair should be able to plan a safe navigation path and traverse it without expecting active, second-by-second involvement of the user. How could BCI systems employ such intelligence?

Given that various types and degrees of physical disabilities exist, this work was focused on developing a multimodal BCI capable of handling different methods of user-system communication. Specifically, combining EEG, EMG, and voice control to address the aforementioned questions while using inexpensive, off-the-shelf technology to develop an effective (i.e., >90% success rate) BCI for people with physical disabilities.

Materials & Methods

The underlying process of the BCI involves converting biosignals into movements of robotic prosthetics for manipulating objects. This entails four subprocesses that transform:

1. Biosignals into natural language statements.
2. Natural language statements into computable logical expressions.
3. Logical expressions into robot commands.
4. Robot commands into actions of a humanoid robot.

The methods employed for each of the above subprocesses are described in this section. This work was conducted as an independent project, where all uncited research was performed by and tested on the Competition Entrant.

A. Transforming Biosignals into Natural Language Statements

Four techniques were used to generate biosignals for user-system communication: eye blinking, jaw clenching, focused attention, and voice. The EMG (blinking and clenching) and EEG (focused attention) signals were recorded using an EEG headset. Voice was captured using a conventional microphone. The subsequent transformation of the biosignals into natural language sentences was accomplished with the help of a graphical user interface. The EEG recording device, the user interface, and the underlying computational methods for this transformation are described below.

A.1. EEG Headset

A cost-effective consumer-grade EEG headset (Neurosky Mindwave) was used in this investigation. The device, shown in Figure 2, uses a single electrode to record EEG and EMG signals from near the left prefrontal cortex. The headset collects raw data with a frequency of 512 Hertz and transmits it wirelessly to any computer that supports Bluetooth communication. The device is readily available on the market and costs approximately 80 U.S. dollars. An off-the-shelf BCI software environment (OpenVibe) was used to receive and process data from the headset. Its use is described in detail in subsection A.3.

A.2. User Interface

The user can formulate natural language sentences word-by-word using the interface shown in Figure 4. It groups keywords into nine primary clusters as depicted below.

Cluster 1: {left_side, right_side, walk, waist}

Cluster 2: {arm, leg, shoulder, elbow, wrist}

Cluster 3: {one_step, two_steps, n_steps}

Cluster 4: {full, three_fourth, one_half, one_fourth}

Cluster 5: {up, down, in, out, forward, backward, open, close}

Cluster 6: {clockwise, counter_clockwise}

Cluster 7: {next_to, above, below, touch_left, touch_right}

Cluster 8: {apple, bottle, banana, ball, pear}

Cluster 9: {process_query, program_sentence, save_program, discard, continue}

The clusters form a hierarchy in which a higher numbered cluster can be reached only through a lower numbered one. A sentence is formed using two or more clusters. For example, to formulate the sentence “left_side shoulder one_half up”, clusters 1, 2, 4, and 5 need to be traversed in that order. The sentence indicates the user’s desire to move the left shoulder up halfway.

At any instance in time only one cluster is active and remains so until the user selects a keyword. To make keywords available for selection, the interface uses two independent visual markers: a large black rectangle and a small gray half-circle. The large rectangle sequentially highlights keywords in the active cluster for 5-second intervals. This marker is synchronized for focused attention; when the user focuses while a keyword is being highlighted, the word is selected. At this point, the next cluster becomes active. Simultaneously, the smaller gray marker alternates sequentially among the active cluster’s keywords in 2-second intervals. This marker is linked to EMG signals. If an eye-blinking or jaw-clenching event occurs when the marker points to a keyword, it is selected and the next cluster in the hierarchy becomes active.

The user interface is multilayered in the sense that both EMG and EEG signals can be used interchangeably for word selection. Furthermore, if desired, the user can activate the voice mode at any time to formulate sentences using spoken language.

In addition, a word cluster is provided in support of programming. The user can define and store up to 25 sets of sentences (“paragraphs”) as programs to be retrieved and executed at a later time. The user interface was implemented in Adobe Flash using Actionscript 3.0.

A.3. Signal Processing

Three algorithms were implemented for detecting biosignals using the EEG headset: one each for focused attention, blinking, and jaw clenching.

A.3.1. Focused Attention Algorithm

Generally, EEG signals can be separated into several distinct frequency bands. Activity in the α range (8-13 Hz) increases in parietal and occipital areas of the brain during periods of relaxation and inattentiveness. In contrast, brainwaves in the β range (13-30 Hz) become more prominent in the frontal lobe when one is alert and engaged in active thinking. Therefore, the α-to-β ratio can be used as a measure of attentiveness (Liu, Chiang, and Chu 2013).

Raw transmitted data from the EEG headset was divided into five frequency bands: θ (4-8 Hz), α (8-13 Hz), low β (13-18 Hz), high β (18-30 Hz), and γ (30-45 Hz) using a fourth order Butterworth bandpass filter. The signal in each frequency band was then partitioned into ½ second intervals with a $^{1}/_{16}$ second offset. Each epoch was mapped from the time domain into the frequency domain using a Fast Fourier Transform algorithm. The average power (S) per frequency band was then calculated using the following equations, in which P_f is the energy value (μv^2) of frequency f (Hz).

$$S_\theta = \sum_{f=4}^{8} \frac{P_f}{5} \quad S_\alpha = \sum_{f=8}^{13} \frac{P_f}{6} \quad S_{\beta_{low}} = \sum_{f=13}^{18} \frac{P_f}{6} \quad S_{\beta_{high}} = \sum_{f=18}^{30} \frac{P_f}{13} \quad S_\gamma = \sum_{f=30}^{45} \frac{P_f}{16}$$

The α-to-β ratio was defined as: $\rho = \frac{S_\alpha}{S_{\beta_{low}}}$. For data classification purposes, a four-dimensional (feature) vector space was defined: $(S_\theta, \rho,\ S_{\beta_{high}}, S_\gamma)$

Linear discriminant analysis was then used for data classification. This technique involves constructing a hyperplane that divides data into two subspaces. One (positive) subspace consists of feature vectors that correspond to focused attention while the other (neutral) subspace consists of those that represent inattentiveness. To define the classification hyperplane (i.e., “train” the algorithm), two distinct 60-second signals were used. The first was generated as the user continuously focused on the highlighted keywords displayed by the user interface. The second was generated as the user relaxed or otherwise did not mentally focus on any task. The constructed hyperplane was then used for data classification during run time. The focused attention algorithm, implemented using OpenVibe, is shown in Figure 5C.

A.3.2. Eye Blinking Detection Algorithm

Muscle contraction around the eyes generally results in a high-amplitude spike in neuronal potential in the α wave frequency band **(**Rao, 2013, p. 28). The algorithm utilized to detect blinking involved filtering the raw signal using a fourth order Butterworth bandpass filter to contain only frequencies between 8 and 13 Hertz. This modified signal was then partitioned into ½ second intervals with a $^1/_{16}$ second offset. To improve classification performance, the signal was then amplified by squaring its amplitude before the average was calculated. A threshold value of 3000 μV^2 was experimentally determined to distinguish blinking. The algorithm looks for two consecutive blinks within two seconds in order to classify a blinking event. This algorithm, implemented in OpenVibe, is shown in Figure 5A.

A.3.3. Jaw Clenching Detection Algorithm

Similar to eye blinking, jaw clenching involves the contraction of facial muscles. This movement can be detected due to an increase in λ waves received from the EEG headset. The adopted algorithm used a fourth order Butterworth bandpass filter to remove all frequencies except those in the 45 to 50 Hertz range. The resulting signal was then epoched into ½ second intervals with no offset. A Fast Fourier Transform technique was used to analyze the partitioned signal and to generate the underlying frequency spectrum. The average power of the frequency band was calculated and used to identify a threshold value of 400 μV^2 for classification purposes. The algorithm also times the duration of a jaw clenching event, thereby supporting an n-ary classification scheme. One to five second clenches were used to generate various user-system commands in this study. The OpenVibe algorithm for detecting clenching is shown in Figure 5B.

A.4. Voice Processing

An off-the-shelf dictation software (Dragon Dictate) was used to enable voice control. The system was trained to only recognize keywords supported by the interface that are spoken during the designated voice-activated period. The user can enter this mode by saying "activate voice input" at any time. The mode is deactivated when the word "deactivate" is spoken. For example, when the microphone is on, if the following sentences are read aloud, the robot would turn its waist clockwise and move its right arm forward: "Activate voice input. Waist one-half clockwise. Right-side arm full forward. Deactivate."

B. Transforming Natural Language Statements into Logical Expressions

Predicate logic was used as a formalism for knowledge representation. An object-relation model was assumed where, for example, the sentence "left_hand next_to ball." is expressed using predicate next_to(left_hand, ball). When it becomes necessary to represent multiple sentences in predicate logic, conjunction (∧) or disjunction (∨) operators can be used. For example, sentences "left_hand next_to ball. Left_hand open one_half." can be expressed as the following logical expression: next_to(left_hand,ball) ∧ open(left_hand, one_half).

The algorithm developed for transforming natural language sentences from the user interface into logical expressions first parses each sentence into tokens. It then uses the cluster hierarchy defined in section A.2 to order the tokens before synthesizing them into predicates. This algorithm was implemented using the C programming language.

C. Transforming Logical Expressions into Robot Commands

All relevant objects, their attributes, and the relationships among them are expressed as logical statements. Collectively, these statements define the knowledge base. The transformation from logical expressions to robot commands is accomplished via theorem proving. In this process, the output query from the previous step is treated as a theorem to be proven true or false in the context of the knowledge base. A representative set of knowledge base statements are shown below.

1. next_to(X, Y):- robot_object(X, P) ∧ camera_object(Y, Q) ∧ constraint(P, Q, R).
2. robot_object(left_hand, P):- robot_object(left_shoulder, Q) ∧ robot_object(left_elbow, R) ∧ constraint(Q, R, P).
3. robot_object(left_shoulder, P):- servo(left_shoulder, N) ∧ servo_state(N, Q) ∧ constraint(P, Q).
4. servo_state(N, P):- limits_of(N, L, U) ∧ *servoState(N, P) ∧ constraint(L, P, Q).
5. servo(left_shoulder, 5).
6. limits_of(5,900,2100).

The predicates defined by an asterisk (*), called action predicates, are directly responsible for actuating the robot. They also provide access to the internal state of the robot's actuators and sensors. For example, predicate *servoState(1,1200) processed as a query forces Servo 1 to assume position 1200. Alternatively, query *servoState(1,X), when processed, binds variable X with the current position of Servo 1.

SWI-Prolog, a free, off-the-shelf implementation of logic programming, was used to build the knowledge base and perform theorem proving. The action predicates were implemented using C and compiled as an SWI-Prolog library for inclusion in the knowledge base.

D. Transforming Robot Commands into Object Manipulation Actions

A life-size humanoid robot was designed and built as a test bed for validating the described approach to robot control (see Figure 1). Its skeleton consists of lightweight aluminum channels and tubes, set to enable the robot to perform object manipulation tasks while remaining upright. For this purpose, the upper body was given 7 degrees of freedom. Each arm was fitted with 3 servo gear boxes for moving the shoulder, the elbow, and the wrist. In addition, a gear box was placed at the waist permitting the upper body to turn relative to the lower body. This configuration ensured that the robot hand could reach a target object on a table regardless of the object's location.

The anthropomorphic hand, as shown in Figure 3, was designed and constructed from light Styrene tubes. It consists of four slim fingers, each with three phalanges that curl closed when an attached string is pulled by a linear actuator. To improve the overall grabbing ability, the hand was fitted with a slip resistant skin of black rubber over silk mesh.

To facilitate the hand's positional movement, an inverse kinematics algorithm was implemented for calculating shoulder and elbow servo positions. This allowed the user to focus on high-level commands such as "move the hand forward" while the system automatically performed low-level servo calculations using the following two equations, where P_{2x} and P_{2y} define the coordinates of the end effector, P_{1x} and P_{1y} the coordinates of the elbow, and L_1 and L_2 the lengths of the upper and lower arm, respectively.

$$\theta_{elbow} = \cos^{-1}\left(\frac{{P_{2x}}^2 + {P_{2y}}^2 - {L_1}^2 - {L_2}^2}{2L_1L_2}\right) \quad \theta_{shoulder} = \frac{\pi}{2} - \tan^{-1}\left(\frac{P_{2x}}{P_{2y}}\right) - \sin^{-1}\left(\frac{L_2\sin(\pi - \theta_{elbow})}{\sqrt{{P_{2x}}^2 + {P_{2y}}^2}}\right)$$

All servos and linear actuators are controlled using the SSC-32 microcontroller, which communicated wirelessly with the computer via XBee radio modules. The robot is also capable of providing the user with auditory feedback via a text-to-voice application on a tablet that acts as the humanoid's head.

Results

Results of this investigation pertain to the use of the four user-system interaction modes, the viability of predicate logic as a computational formalism for brain-computer interfacing, and the overall effectiveness of the paradigm for object manipulation.

A. Focused Attention, Blinking, Jaw Clenching, and Voice Controls

The effectiveness of each mode of interaction (focused attention, blinking, jaw clenching, and voice) was tested quantitatively using individual word selection and sentence formation. A total of 400 trials (words) were selected in testing each mode, as tabulated below.

Word Selection Data				
	Eye Blinking	Jaw Clenching	Focused Attention	Voice
Words Selected Correctly	386	391	389	394
Words Selected Incorrectly	11	6	7	5
No Event Registered	3	2	1	1
False Positive Registration	0	1	3	0
% Correct	96.5%	97.75%	96.4%	98.5%

For testing for sentence composure, 100 trials of arbitrarily predefined sentences were attempted with each of the four methods. The results are tabulated below.

Sentence Composure Data				
	Eye Blinking	Jaw Clenching	Focused Attention	Voice
100% Correct Composure	94	97	92	95
75% Correct Composure (4-word sentence)	4	2	5	3
67% Correct Composure (3-word sentence)	2	1	1	2
Average (Weighted) % Correct	98.34%	99.17%	96.42%	98.67%

B. Predicate Calculus

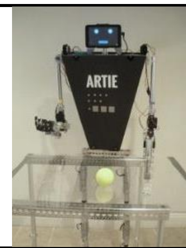
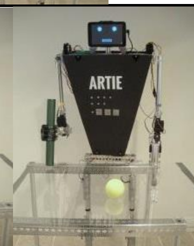
SWI-Prolog was determined to be effective for the symbolic representation of the following: system actuators and their physical constraints, pertinent dimensions of the robot, parent-child relationships between robot components, the mapping between robot parts and their linguistic description, and various spatial relationships between the robot and objects in the environment. However, the ability of the platform to model and solve algebraic equations was found to be lacking. This ability is critical for computations involving multiple equations, such as direct and inverse kinematics for gait control and arm movement, respectively.

Further problems are posed with SWI-Prolog in supporting computations that require feedback, such as active gait control. The necessary robot commands have to be transmitted to the onboard microcontroller to drive the actuators. Sensor data pertaining to the resulting movements must be collected and used to calculate the next set of actuation instructions. This feedback/calculation cycle must take place within a fraction of a second if the robot is to maintain its balance while walking. A series of experiments were performed to determine this response time in SWI-Prolog. It was found that the software platform spends between 2 and 3 seconds in a typical computation cycle. This is sufficiently longer than the required speed and thus was deemed inadequate for active gait control. To overcome these deficiencies, SWI-Prolog was extended using external functions (built-in predicates) defined in C. These fast numeric-based libraries satisfy the needed computational demands of robot kinematics while maintaining the declarative nature of the knowledge representation framework.

C. Object Manipulation

The basic premise of being able to control robot movements using a consumer-grade EEG headset via natural language was put to test by targeting a set of basic robot movements. The individual move instructions and the resulting humanoid robot postures are tabulated below. These results reveal that the humanoid robot was effectively and time-efficiently controllable for object manipulation using the BCI system.

Pose	Description	Mode(s)	Elapsed time	Outcome
Pose 0 (Initial)	Starting position			

Pose 1	Move right hand down half way, turn wrist up, open fist.	Voice	3 seconds	
Pose 2	Grab object, turn half way, release object.	Clenching, focused attention	8 seconds	 
Pose 3	Move right hand to the top of the ball.	Focused attention	3 seconds	
Pose 4	Grab object, raise hand half way, move hand forward, release object.	Blinking, clenching, focused attention	10 seconds	  

Illustrations

Figure 1. Anthropomorphic Robot Trained In Electroencephalography (ARTIE)

Figure 2. Neurosky Mindwave Headset

Figure 3. Anthropomorphic Robot Hand

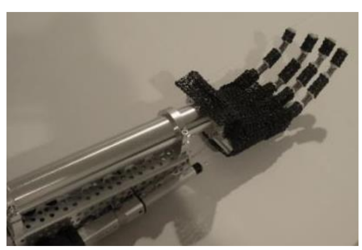
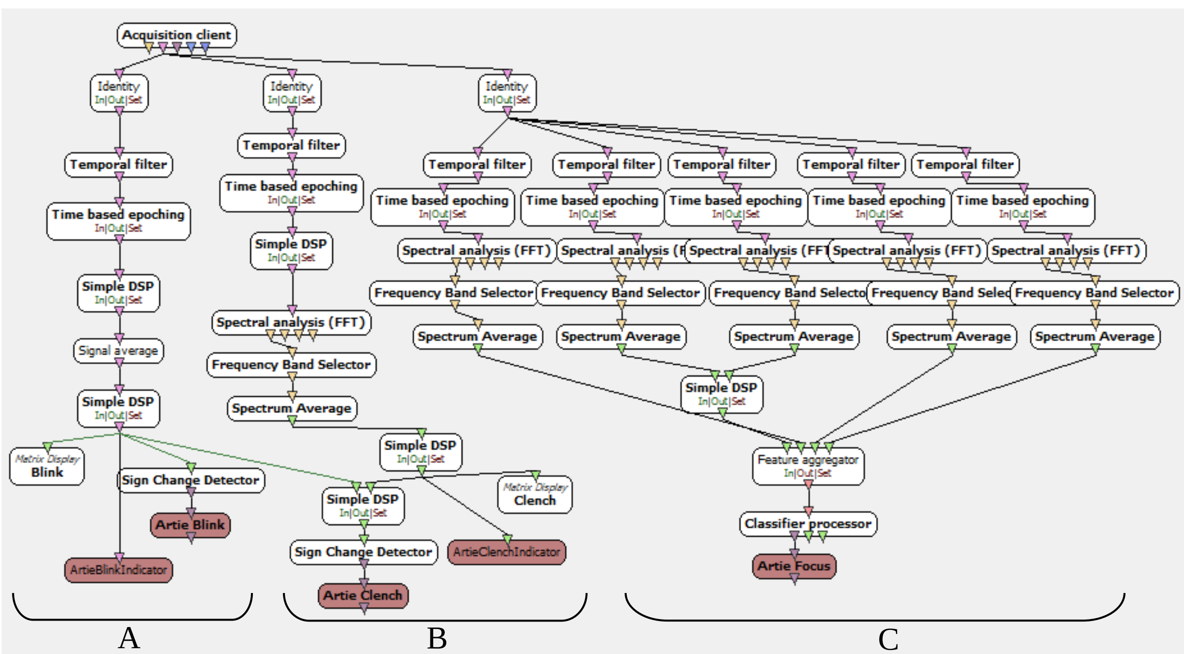


Figure 4. The Graphical User Interface

Figure 5. Eye Blinking, Jaw Clenching, and Focused Attention Detection Algorithms Implemented in OpenVibe



Discussion

The results presented herein confirm that it is possible to use a consumer BCI headset to record EEG and EMG signals for controlling a robotic system. This finding paves the way for the future development and commercialization of cost-effective assistive technologies for people with physical disabilities.

In two of the previously cited research works (Aflalo et al., 2015; Hochburg et al., 2012), participants controlled a robotic arm with electrodes that required brain surgery to implant. Both subjects had tetraplegia and sustained paralysis of all limbs, but retained control of facial muscles for eye and jaw movements. This research investigation points to an alternative approach to brain-computer interfacing that comparatively reduces medical complications to zero. The non-invasive approach demonstrated in this work is adaptable, portable, and cost-effective, making viable its use, distribution, and commercialization.

To this point, user fatigue is also an important factor to consider. The use of a single interaction mode, such as focused attention alone, could result in a diminished rate of performance over time. The multimodal approach presented can be used to combat fatigue and thus enable the user to interact with the system for longer durations. In addition, this could make the BCI more flexible and user-friendly by allowing individuals to traverse it based on personal preference and abilities.

The graphical interface developed in this work provides the user a visual map for all permissible processes, rather than require him or her to memorize various command options. Again, this decreases mental taxation on the user. This process is aided by the use of natural language: an intuitive means for externalizing intent. Further, when the user has the ability to use his or her voice, this offers an even more natural mode of interaction with the system.

Currently, the interface uses a research software platform (OpenVibe) for signal processing. Although OpenVibe has a small footprint and can be installed and run on a laptop, it limits the efficiency and portability of the BCI system. Instantaneous dynamic control of robotic prosthetics requires a more time-efficient computational platform that operates within a fraction of a second. Recent advances in microelectronics could make possible the development of specialized yet inexpensive chips for this purpose.

BCI systems could also benefit from automated reasoning and decision making in order to lessen burdens on the user. Although some progress has been made in this area, more work remains to be done. A robust logic programming environment (SWI-Prolog) serves as a viable platform for building knowledge bases to automate processes. The flexibility of logic programming allows the use of special functions such as built-in predicates for computationally intensive and time-sensitive tasks without sacrificing the expressive power of the language.

Finally, a practical assistive system must offer the option of user programming. That is, the individual must be able to write programs for extending system capabilities using his or her brainwaves. Imagine a case in which a robotic arm is used for a repetitive, frequent task such as unlocking one's door. If the system is not programmed to do so, the user has to expend daily mental effort in performing the same fine-tuned arm manipulations to precisely align their key with the lock. In this case, it would be beneficial for the user to perform and save the task once, thereafter being able to call it with the ease of selecting a single word. The developed interface offers the user this option in storing a set of sentences to be retrieved and executed at a later time. This simple approach to programming is a byproduct of the declarative nature of predicate logic and its deductive computational power. Although this feature needs to be studied further, the ease with which it can be incorporated into a BCI system is promising.

To test the described system, this work involved designing and building a life-size humanoid robot. It was shown that such a robot can indeed be controlled by biosignals that are recorded by a consumer-grade EEG headset. Although the challenge of controlling the robot's gait cycle, with or without obstacles, remains to be addressed, results obtained to-date are encouraging and exciting.

Conclusions and Future Work

The work described in this report points to the viability of developing cost-effective, consumer-grade assistive technologies for people with physical disabilities. In the short run, the future goal of this work is to further extend the present findings in three related directions:

1. Develop a portable robotic arm equipped with sensors that can be mounted on a wheelchair and controlled by the existing graphical user interface.
2. Design and develop the prototype of a transformable/expandable wheelchair that can accommodate a person in both a seated and an upright position.
3. Explore the use of machine vision techniques for making the robotic system aware of its surroundings and investigate the use of predicate logic for automated reasoning and decision making in order to assist the user with task completion.

In the long-term, a related goal is developing a fully functional exoskeleton for similar purposes. The overarching and guiding objective of this undertaking is to develop commercially-viable, economically-feasible, brain-actuated systems to enable mobility and increased independence for people with physical disabilities.

References

Aflalo, T., Kellis, S., Klaes, C., Lee, B., Shi, Y., Pejsa, K., . . . Andersen, R. A. (2015). Decoding motor imagery from the posterior parietal cortex of a tetraplegic human. *Science*, *348*(6237), 906-910.

Chae, Y., Jeong, J., & Jo, S. (2012). Toward brain-actuated humanoid robots: Asynchronous direct control using an EEG-based BCI. *IEEE Transactions on Robotics*, *28*(5), 1131-1144. http://dx.doi.org/10.1109/TRO.2012.2201310

Hochburg, L. R., Bacher, D., Jarosiewicz, B., Masse, N. Y., Simeral, J. D., Vogel, J., . . . Donoghue, J. P. (2012). Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, *485*, 372-377. http://dx.doi.org/10.1038/nature11076

LaFleur, K., Cassady, K., Doud, A., Shades, K., Rogin, E., & He, B. (2013). Quadcopter control in three-dimensional space using a noninvasive motor imagery-based brain-computer interface. *Journal of Neural Engineering*, *10*(4), 1-15. http://dx.doi.org/10.1088/1741-2560/10/4/046003

Millán, J. D. R., Renkens, F., Mouriño, J., & Gerstner, W. (2004). Noninvasive brain-actuated control of a mobile robot by human EEG. *IEEE Transactions on Biomedical Engineering*, *51*(6), 1026-1033. http://dx.doi.org/0018-9294/04$20.00

Rao, R. P. N. (2013). *Brain-computer interfacing: An introduction*. New York, NY: Cambridge University Press.

Sengupta, A., Bhattacharyya, S., & Janarthanan, R. (2011). Implementation of EEG based control of remote robotic systems. *Recent Trends in Information Systems*.

Vidal, J. J. (1973). Toward direct brain-computer communication. *Annual Review of Biophysics and Bioengineering*, *2*, 157-180. http://dx/doi.org/10.1146/annurev.bb.02.060173.001105

World Health Organization, & World Bank. (2011). World report on disability. Retrieved August 27, 2015, from http://www.who.int/disabilities/world_report/2011/report.pdf?ua=1