IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 31, 2021, accepted February 22, 2021, date of publication February 26, 2021, date of current version March 8, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3062329

Biologically-Inspired Legged Robot Locomotion Controlled With a BCI by Means of Cognitive Monitoring

PATRICIA BATRES-MENDOZA[1], ERICK ISRAEL GUERRA-HERNANDEZ[1], ANDRES ESPINAL[2], EDUARDO PÉREZ-CARETA[3], (Member, IEEE), AND HORACIO ROSTRO-GONZALEZ[3]

[1]School of Biological Systems and Technological Innovation, Universidad Autonoma Benito Juarez de Oaxaca, Oaxaca 68120, Mexico
[2]Department of Organizational Studies, DCEA-Universidad de Guanajuato, Guanajuato 36000, Mexico
[3]Department of Electronics, DICIS-Universidad de Guanajuato, Salamanca 36885, Mexico

Corresponding author: Horacio Rostro-Gonzalez (hrostrog@ugto.mx)

ABSTRACT Brain-computer interfaces (BCI) are a mechanism to record the electrical signals of the brain and translate them into commands to operate an output device like a robotic system. This article presents the development of a real-time locomotion system of a hexapod robot with bio-inspired movement dynamics inspired in the stick insect and tele-operated by cognitive activities of motor imagination. Brain signals are acquired using only four electrodes from a BCI device and sent to computer equipment for processing and classification by the iQSA method based on quaternion algebra. A structure consisting of three main stages are proposed: (1) signal acquisition, (2) data analysis and processing by the iQSA method, and (3) bio-inspired locomotion system using a Spiking Neural Network (SNN) with twelve neurons. An off-line training stage was carried out with data from 120 users to create the necessary decision rules for the iQSA method, obtaining an average performance of 97.72%. Finally, the experiment was implemented in real-time to evaluate the performance of the entire system. The recognition rate to achieve the corresponding gait pattern is greater than 90% for BCI, and the time delay is approximately from 1 to 1.5 seconds. The results show that all the subjects could generate their desired mental activities, and the robotic system could replicate the gait pattern in line with a slight delay.

INDEX TERMS Bio-inspired robot, brain-computer interface (BCI), electroencephalography, hexapod robot, iQSA method, motor imagery, spiking neural network, central pattern generator.

I. INTRODUCTION

Brain-Computer Interfaces (BCIs) are systems that provide a communication and control channel between the human brain and the outside world by means of **electroencephalography (EEG)** [1]. Originally, BCIs are developed to help patients suffering from severe motor impairments [2]–[4]. However, several applications have emerged outside of the medical field, like the integration of BCIs with other immersive technologies such as virtual reality (VR), augmented reality (AR), and computer games [5]–[8]. Similarly, there are researches related to integrating BCIs and external devices to robot systems [9]. In this last case, researchers have used several strategies to control a robot with a BCI such as Visual Evoked Potentials (VEPs) [10]–[13], Event-Related Potential (ERP) [14]–[18], Slow Cortical Potentials (SCPs) [19]–[21] and Sensorimotor Rhythm μ *and* β [22]–[24].

For instance, in [25], authors present a **Steady State Visually Evoked Potentials (SSVEP)** based hierarchical architecture for controlling a humanoid robot with mind. This architecture is tested in a multi-task experiment to drive the robot through obstacles and picking up a balloon. Similarly, Zhao S. *et al.*, in [26] describe the development of a teleoperation control framework of multiple coordinated mobile robots through a brain-machine interface. The online BCI system uses SSVEP, analyzes the EEG data using AdaBoost with the **Support Vector Machines (SVM)** classifier, and motion commands are produced for the teleoperated robot with an average recognition accuracy of 85%.

The associate editor coordinating the review of this manuscript and approving it for publication was Ludovico Minati.

In their research, George *et al.* [27] present a BCI headset controlling a service robot's navigation using a SVM classifier while an operator performs motor imagery tasks. The performance obtained is between 98% and 100% for classification accuracy in predicting the desired direction. Bousseta, Outset, *et al.* [28] propose a BCI system that controls a robot arm based on the user's thoughts. They used the Principal Component Analysis (PCA) method combined with the Fast Fourier Transform (FFT) spectrum within the frequency band responsible for sensorimotor rhythms (8Hz to 22 Hz) and the SVM classifier with the **Radial Basis Function (RBF)** kernel, whose outputs were translated into commands to control the robot arm.

Xu *et al.* [29] developed an EEG-based teleoperation robot control system with motor imagery in their research. The imagined movement's EEG signals were translated into a continuous two-dimensional control signal and transmitted to the remote robotic arm using TCP/IP and then moved through the control signal in real-time. Barbosa *et al.* [30] developed a BCI to activate a 120lb mobile robot's movements associating four different mental tasks to robot commands. In one of these implementations, a 91% average hit rate was obtained with only 1.25% wrong commands after 400 attempts to control the mobile robot. Qui *et al.* [31] present a teleoperated robotic system based on visual feedback and a local adaptive fuzzy controller to drive the exoskeleton tracking the intended trajectories in the human operator's mind, obtaining a recognition accuracy of about 80%. Saduanov *et al.* [32] present the design, evaluation, and control of a six-degree-of-freedom robot manipulator by motor imagery-based BCI, allowing users to perform reach-grasp-release activities in 3D space. The experimental results tested on seven healthy individuals have shown the feasibility of controlling the robot with an accuracy range between 64% - 86%. Muller-Putz *et al.* [33] performed an analysis of events related to synchronization and desynchronization (ERD/ERS) of 160 EEG tests to generate motor imagination patterns for controlling an implanted neuroprosthesis in a 42-year-old patient with spinal cord injury. Cichock and Choi [34] implemented a Motor Imagery based algorithm to control a wheelchair by mental instructions.

In all the researches mentioned above, the EEG signals are analyzed and classified to get the mental command to control the output device. During this process, the features extracted by the BCI classifier are induced by the user as a response to an external stimulus. Still, unlike event-related brain activities, the μ rhythm can be voluntarily modulated. For this, the user must produce mentioned brain activities using some mental strategy such as **Motor Imagination (MI)**, which is the mental simulation of specific movements [35], [36]. MI consciously triggers brain regions involved in planning and executing similar actions when performed in a real way.

Additionally, robotic systems based on neuromorphic hardware have been recently studied [37]–[39], resulting that legged robot locomotion may be achieved by plausible neural mechanisms known as Central Pattern Generators (CPGs). The CPGs are specialized neural networks that allow, among other things, the locomotion of living beings through the production of rhythmic patterns without the need of sensory inputs (endogenously). These patterns allow us to control and coordinate repetitive activities, such as breathing, chewing, walking, swimming, and running [40], [41]. Most CPG implementations are built using neuron models with a low biological plausibility, e.g., oscillators [42], [43], and spiking neuron models with different levels of biological plausibility [44]–[46].

In this regard, our work aims to control a bio-inspired hexapod robot in real-time with the help of a BCI based on motor imagery. To achieve locomotion in the hexapod robot, we used **Spiking Neural Networks** (SNN) that mimic a CPG behavior to reproduce and execute the orders received by the BCI in real-time [47]. The EEG signals generated in the BCI were classified using the **improved Quaternion-based Signal Analysis method** (iQSA) [48], which bases its operation on quaternions' algebra, while evaluates the rotations and orientations that the EEG signal performs over time. Experimental results show how our approach generates BCI commands performed by a hexapod robot in real-time. This article is organized as follows: Section 2 introduces the CPG system and the iQSA method; Section 3 includes a description of the architecture developed in this work; Section 4 presents the implementation and results; finally, we conclude in Section 5.

II. MATERIALS AND METHODS

This article covers all aspects from the acquisition of EEG signals to the execution of the cognitive commands and real-time validation on legged robots. In this regard, we use the iQSA method to classify the EEG signals and to design the CPG with a SNN capable of reproducing a specific rhythmic locomotion pattern (gait).

A. THE iQSA METHOD

The iQSA method is a technique used in this work for modeling bioelectric signals through quaternion-based rotations and orientations. Analyzing EEG signals and extracting their characteristics in a complex number space, allows to consider them as a single entity without leaving aside information that may be relevant to the classification stage. In addition, with this method, ambiguities in the data are avoided; it also performs fewer calculations than other rotational techniques and get a more accurate representation of the data. A quaternion [49] is defined as a set of four constituents (one real component and three imaginary) of the form: $q = w + ix + jy + kz$, where $w, x, y, z \in \mathbb{R}$ and i, j, k are symbols of three imaginary quantities known as imaginary units. So, a quaternion can be described as $q = (s+a)$, $a = (x, y, z)$ where s and a are known as the quaternion's scalar and vector, respectively. Based on the expanded Euler's formula [50], the rotation for quaternion around the axis $a = [a_x, a_y, a_z]$ by a theta angle is defined as

follows (See [48], [51] for further details):

$$q = \cos\left(\frac{\theta}{2}\right) + \left(a_x \cdot i + a_y \cdot j + a_z \cdot k\right) \sin\left(\frac{\theta}{2}\right) \quad (1)$$

Furthermore, the operation to be performed on a vector r to produce a rotated vector r' is given by the equation (1), which is an useful representation that makes easier the rotation of a vector. We can see that r is the original vector, r' is the rotated quaternion, and q is the quaternion that defines the rotation.

$$r' = qrq^{-1}$$

replacing q and q^{-1} the equation is rewritten as follows:

$$r' = \left(\cos\left(\frac{\theta}{2}\right) + a\sin\left(\frac{\theta}{2}\right)\right) r \left(\cos\left(\frac{\theta}{2}\right) - a\sin\left(\frac{\theta}{2}\right)\right) \quad (2)$$

The iQSA method takes a set of four signals as input and convert them in a pure quaternion to evaluate the orientations and rotations made by the EEG signal in the time domain. Table 1 shows the algorithm of the iQSA method towards real-time applications, whose aims to calculate the rotation and module of the signal and build an array of features with the mean (μ), variance (σ_2), contrast (Ct), and homogeneity (H). For this purpose, a feature matrix (M) is defined from the description of the quaternion q and a vector r, where each of them corresponds to an array of 4 and 3 EEG channels, respectively.

Algorithm 1 can be described as follows: the required input data (*quat*) are the signals to be analyzed, and includes signals from the four channels. Delta movements in time (*dt*), *ns* represents a segment of the signals to be analyzed, *t_disp* is a displacement in time t of samples, and *pr* is a flag to indicate whether the sample is being used during validation or training. Line 1 calculates the segments matrix $y(t)$ defined by the changes between classes defined in *quat*. Line 2 dictates that for each segment (*ns*) in y_i, $q(ns)$ and $r(nr)$ are formed considering that $q(ns)$ is an array with n quaternions and $r(ns)$ is an array with n pure quaternions moved according to a *dt* value. After this, the rotation $q_{rot}(ns)$ is calculated using quaternions, which produces an array of n rotated quaternions. Similarly, it calculates the scalar array $q_{mod}(ns)$ from the module $q_{rot}(ns)$, which contains n scalar elements that will be used to form a matrix with $M_{i,j}$ features, where index i corresponds to the analyzed segment and index j is one of the m features to be analyzed using the equations included in Table 2.

B. CENTRAL PATTERN GENERATOR

The principles of CPGs emerged a century ago when Brown *et al.* [52] determined that the central nervous system can generate stereotyped movements, whose main characteristic is the alternation between the skeletal muscles known as flexors and extensors. In 2012, Grabowska *et al.* [53] presented a study on the coordination patterns of the stick insect Carausius Morosus's limbs (Figure 1a) when walking freely along a straight path on a flat, horizontal surface. Grabowska deciphers the rules and mechanisms that control leg coordination in hexapods in that investigation, identifying several patterns like the one in Figure 1b.

TABLE 1. iQSA method.

Algorithm 1. iQSA Method

Require: $quat = [q1, q2, q3, q4, c], dt, ns, t\_disp, task$
(1) $y(t) \leftarrow$ segments of signals
(2) for each ns in $y_i(t)$ do:
  $q(ns) \leftarrow quat(ns)$
  $r(ns) \leftarrow quat(ns - dt)$
  $q_{rot}(ns) \leftarrow n_{rot}(q(ns), r(ns))$
  $q_{mod}(ns) \leftarrow mod(q_{rot}(ns))$
  $M_{i,j} \leftarrow f_j(q_{mod}(ns))\ \{j = 1, ..., m\}$
  $c_i \leftarrow \{c = (1, 2, 3, ..., n) \mid y_i(t) \in c\}$
  $ns = ns + tdisp;$
 end for
(3) $M_{k,j} \leftarrow \left\{M_{i,j} \mid \frac{\#k}{\#i} = \%t\right\}$
(4) $M_{l,j} \leftarrow \left\{M_{i,j} \mid \{l\} \notin \{k\}, \frac{\#l}{\#i} = 1 - \%t\right\}$
(5) $[\hat{C}_k, pr] = Classify(pr, M_{k,j}, c_k)$
(6) $[\hat{C}_l, pr] = Classify(pr, M_{i,j}, c_l)$
$\%rt = \#\left\{\hat{C}_k \mid \hat{C}_k = \hat{C}_k\right\} / \#\{C_k\}$
$\%rv = \#\left\{\hat{C}_l \mid \hat{C}_l = \hat{C}_l\right\} / \#\{C_l\}$

Function: Classify(pr, Mt, c)
(1) if pr then:
  $R = training(Mt, c)$
(2) $else:$
$R = classify(Mt)$
end if
$return[R, pr]$

The emulation of these biological motor activities has inspired many researchers to adopt this locomotion mechanism on robotic devices, mainly due to the insect's ability of easy navigation in uneven terrain. There are different types of mathematical models developed to study biological CPG's [42]. One of the most recent is spiking neurons, considered the third generation of artificial neural networks. Spiking neurons are processing units described by differential equations that emulate the electrochemical processes occurred in the brain [54]. An essential fact of these models, and the main reason for their application in this work, is that using a network of spiking neurons, we can generate periodic patterns just like the insect mentioned above's movements, see Figure 1b.

To do this, we used the "Integrate-and-Fire" (IF) model, the best-known example of a spiking neural model, capable of reproducing most of the neuronal dynamics observed in the brain [55]. The IF model describes a neuron's state in terms of the membrane potential, determined by the synaptic inputs and an injected current.

In this model, the neural dynamics can be represented as spike trains. For this model, the membrane potential V_i and the firing state Z_i of the ith neuron at time k are given by the follows equations:

$$V_i[k] = \gamma V_i[k-1](1 - Z_i[k-1]) + \sum_{j=1}^{N} W_{i,j} Z_j[k-1] + I_i^{ext} \quad (3)$$

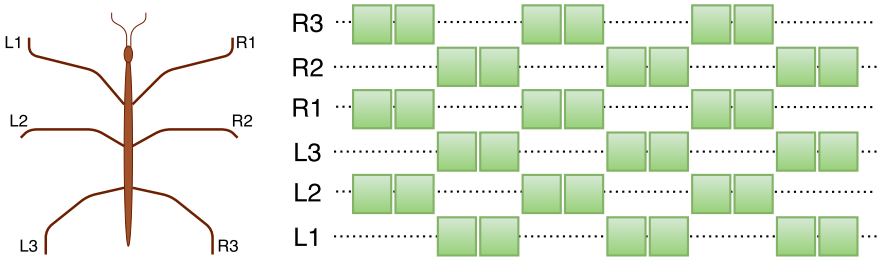


FIGURE 1. Stick insect. a)Arrangement of the limbs of the stick insect, b)Schematic drawing of hexapod walking patterns. Arrangement of the limbs of the stick insect. The letters *R* and *L* refer to the insect's left and right legs. The green bars indicate the oscillation phase of each leg of the animal (when the corresponding limb is resting on the ground).

and,

$$Z_i[k] = \begin{cases} 1 & if\ V \geq \theta\ (firing) \\ 0 & otherwise \end{cases} \tag{4}$$

where $\gamma \in (0, 1)$ defines the leak rate, $V_i[k]$ represents the membrane potential of the ith neuron at time k, Z represents the firing state, N is the number of neurons (one per servo motor), W is the matrix of synaptic weights, and I^{ext} represents an external stimulus. Hence, when $V_i[k]$ reaches a given threshold θ, a spike occurs in $Z_i[k]$ and the neuron i is reset by the term $(1 - Z_i[k])$ in equation 3 (See [47] for further details).

Eq. 3 estimates the membrane potential for a neural network with N neurons, and Eq. 4 allows to generate the output of the SNN basis this membrane potential.

In our case, N is equal to 12 for the hexapod robot (see Fig. 4), since we have a one-to-one correspondence between neurons and servomotors (or Degrees Of Freedom). The SNN is described in the following section.

1) HARDWARE

We use the robotic platform MH2 of Lynxmotion, a hexapod robot with 12 degrees of freedom (DOF) with each leg of two DOF's (see Figure 2). Each articulation is motioned by a servomotor Hitec of high torque. The figure shows the red ovals tagging each of the articulations (servos), giving them the name of coxa and femur. Talking about the electronics, the central part of the module is integrated by Opal Kelly ZEM4310 (Figure 3a), based on a **Field Programmable Gate Arrays** (FPGA) of family Cyclone IV of Altera, with 55,856 logic elements (LEs), 2,340 Kbits of embedded memory and 154 embedded 18 × 18 multipliers. The integration module is formed by a USB 3.0 High transfer connection, as the 128 MB of DDR2 integrated memory.

To control the servos in the robot a controller SSC-32 (Figure 3b) was used.

This module consists of 32 outputs to control servo motors using **Pulse Width Modulation (PWM)** with a 1ms resolution and a serial input. This card acts like a bypass between the internal communication system and the servomotors. Therefore, the system locomotion only needs a string of characters coded in ASCII, sent by a serial controller

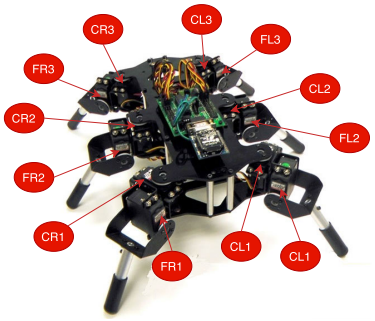


FIGURE 2. Hexapod robot structure. Tags *C* and *F* correspond to coxa and femur, respectively, and the letters *R* and *L* correspond to right leg and left leg.

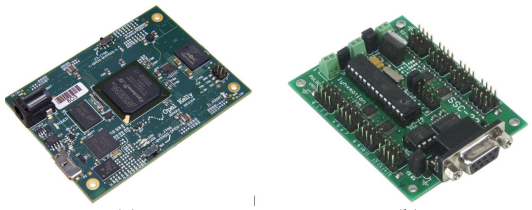


FIGURE 3. Hardware. (a) FPGA ZEM4310 module. (b) SSC-32 servo controller card.

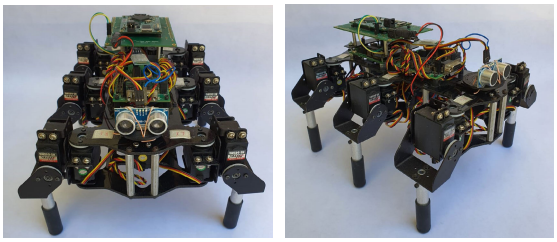


FIGURE 4. Hexapod Robot: (a) Frontal view. (b) Lateral view.

protocol SSC-32 that generates the corresponding PWM signals. The card allows the use of extended range servos. A change in a unit produces a positional change of 1μs in the PWM pulse width, according to this, the position resolution is $0.09°/unit$. An HC-06 module was used for bluetooth communication.

Figure 4 shows the hexapod robot integration with the FPGA ZEM4310 and the SSC-32 servo controller card.

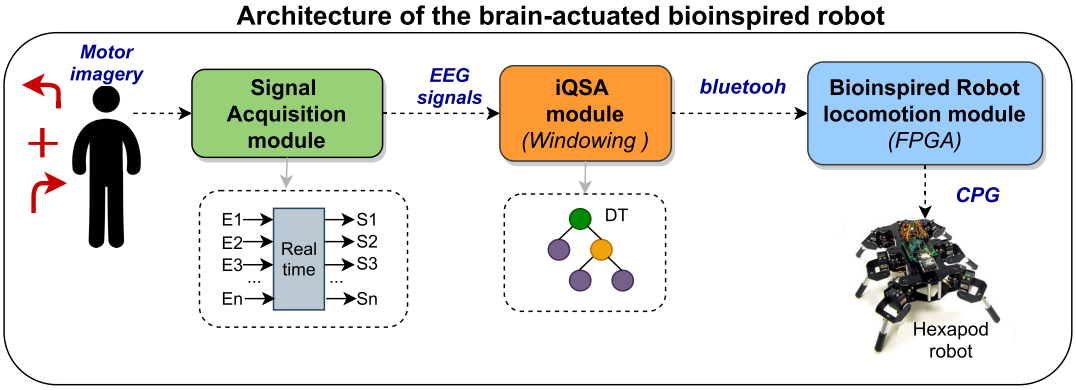


FIGURE 5. Communication architecture for iQSA-BCRI.

III. ARCHITECTURAL OVERVIEW

Figure 5 describes the software architecture implemented to communicate the BCI and the hexapod robot through EEG signals of motor imagery obtained from subjects. The proposed system consists of three modules: (1) EEG Signal Acquisition, (2) iQSA, and (3)Bioinspired Robot Locomotion.

A. EEG SIGNAL ACQUISITION

In this research, we used the Emotiv Epoc headset to record the EEG signals (Figure 6) during motor imagery activity. **The headset has a gyroscopic sensor and 14 EEG electrodes distributed in the right and left hemispheres of the brain, a sample rate of 128 Hz and wireless EEG system.** Although the headband provides 14 electrodes, **the proposed system only requires four sensors (F3, F4, FC5, and FC6) to identify the imagined command. These four electrodes were selected due to their location near the lower parietal lobe and the premotor cortex**, where a group of neurons from the cortex called mirror neurons which are activated during the execution or imagination of movement. **We implemented an experiment to acquire users' EEG signals, train the recognition algorithm, and build the real-time experiment classifier (The details of the experiment are mentioned in secion IV).** After that, a multiprocessing system was implemented in Python 2.7 to manage the EEG data acquired by the BCI device in real-time. In this system, a process named producer registers readings EEG signals every 1 second and stores the information in a FIFO data structure (Firs-In, First-Out). Simultaneously, another process (consumer) continuously checks if elements are waiting to be analyzed. If so, the consumer process will remove that element and send it to the iQSA module for analysis.

B. iQSA MODULE

The iQSA module aims to analyze, process, and classify the acquired EEG signal blocks, identifying their most relevant characteristics to determine the action imagined by the user. According to Figure 7, EEG data block of 1s (128 samples)

FIGURE 6. Emotiv Epoc headset has a gyroscopic sensor and 14 EEG electrodes distributed in the right and left hemispheres of the brain, a sample rate of 128 Hz and wireless EEG system.

TABLE 2. Statistical characteristics adapted for implementation in the iQSA method.

Statistical Features	
Mean	$\mu = \frac{\sum(q_{mod})}{N}$
Variance	$\sigma^2 = \frac{(\sum(q_{mod})^2 - \mu)^2 + \sum(q_{mod})^2}{2N}$
Contrast	$Ct = \frac{\sum(q_{mod})^2}{N}$
Homogeneity	$H = \sum \frac{1}{1+(q_{mod})^2}$

is analyzed with the iQSA module; for this, it was performed an offset in the signal every 8ms (T_disp) to obtain small data samples (w_n) of 0.5s and form a new sample (w_{n+1}). Additionaly, this process performs an overlap sampling (windowing), producing a greater amount of samples to reinforce the learning stage of the algorithm.

Then, every time a Window (w_n) is formed, the queue structure waiting to be evaluated enters to calculate the rotation of the vector $q_{rot}(t)$ using the quaternion q and the vector r, where q is an m-by-4 matrix containing m quaternions, and r is an m-by-3 matrix containing m quaternions displaced according to a dt value. Later, the modulus is applied to the quaternion q resulting in the vector q_{mod}. After that, the scalar array q_{mod} is calculated from the module q_{rot}, which will be used to form a matrix features $M_{i,j}$, where i corresponds to the analyzed segment, and j is one of the four features to be analyzed using the equations in Table 2.

In this study, we use a decision tree classifier for the training and validation process. To train our model, we separate the initial data set considering 70% of data samples for the training stage and the remaining 30% for the test stage. Each sample (ns) was executed 20 times to obtain its set's performance and the decision rules in the training stage.

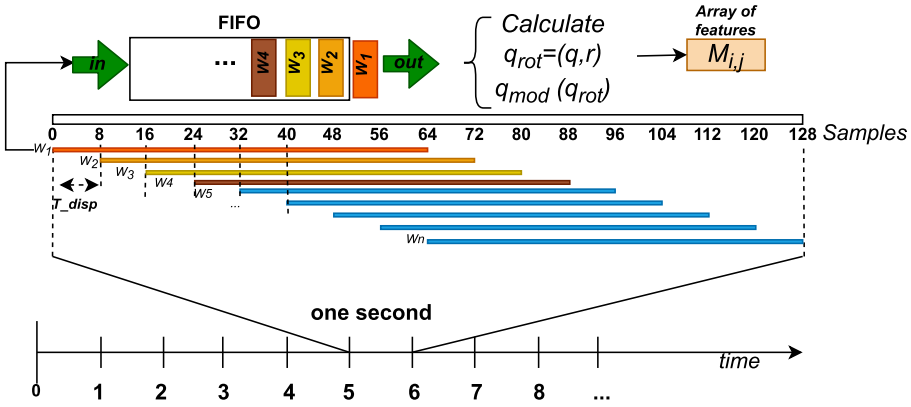


FIGURE 7. EEG signal subsampling with window and window scrolling.

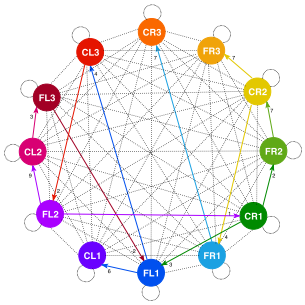


FIGURE 8. Topology of the pulsating neural network.

In the validation stage, the remaining 30% of the data set was evaluated with the decision rules generated in the training stage to obtain final recognition and the movement imagined by the user. Finally, action imagined is sent to the locomotion module for implementation.

C. BIO-INSPIRED ROBOT' LOCOMOTION MODULE

We defined the locomotion mechanism for leg coordination of the robot based on a spiking neural network, which acts as a CPG. The CPG consists of a spiking neural network, which produces rhythmic patterns known as locomotion gaits. For this research, we defined three gait patterns: turn-left, turn-right and walk forward, that topology can be seen in Figure 8. Where each neuron is shown in a different color, synaptic connections are represented by arrows of the color corresponding to the presynaptic neuron and synaptic weights are indicated with numerical values in black. In total, the network has 13 synaptic connections (12 excitatory and 1 inhibitory).

With the aim of evaluating the design and performance of the CPG, the simulation of the SNN in Matlab was made, obtaining the simulation shown in Figure 9, where each neuron is represented with the same color of the network topology and the pattern generated by the simulation is identical to the one to be replicated (See [47] for further details).

In order to generate the bio-inspired robot locomotion, we defined a locomotion system block diagram (Figure 10) designed from three modules: (1) a HC-05 Bluetooth module that receives the data from the iQSA module and sends them to the FGPA; (2) FGPA module consist of three sub-modules: CPG, Bluetooth and RS-232; and (3) Robot Module with three submodules: left servos, right servos and servos controller card (SSC32).

In the block diagram, the HC-05 module communicates with the FPGA module through the Bluetooth sub-module in charge of receiving the iQSA module's commands and interpreting them, so the CPG submodule generates the corresponding gait pattern. In this way, the FPGA module sends the generated locomotion patterns to the SSC32 controller card to activate synchronically the left and right servo-motors.

Figure 11 shows the electronic design at the registers transfer level (RTL) of Quartus for the implementation of the pulsating neural model, which receives as input the number of neurons in the network (N), the matrix of synaptic weights ($W_{i,j}$) the initial values vector of membrane voltages (V_0), the initial values vector of the neuron state (Z_0) and the threshold θ. As output, the spiking activity Z and membrane voltages V are obtained. For the implementation, the synthesizer used four 16-bit 2 to 1 multiplexer, a 4-bit 12 to 1 multiplexer, two 16-bit adders, two 16-bit registers, a MOD 12 counter, a 16-bit comparator, and a block of memory for the synaptic weights of 12 16-bit memory locations, two inverters, an AND gate, and an OR gate.

Regarding the RS-232 sub-module, it receives as input the output vector Z generated by the CPG and it is responsible for translating such outputs into instructions for the SSC32 servo controller. RS-232 is an asynchronous serial transmission protocol, where characters are transmitted in ASCII code. In this case, the transmission speed used is 9600 baud. The instruction format sent to the SSC32 module is as follows:

#<ch>P<pwm> . . . #<ch>P<pwm>T<time><cr>

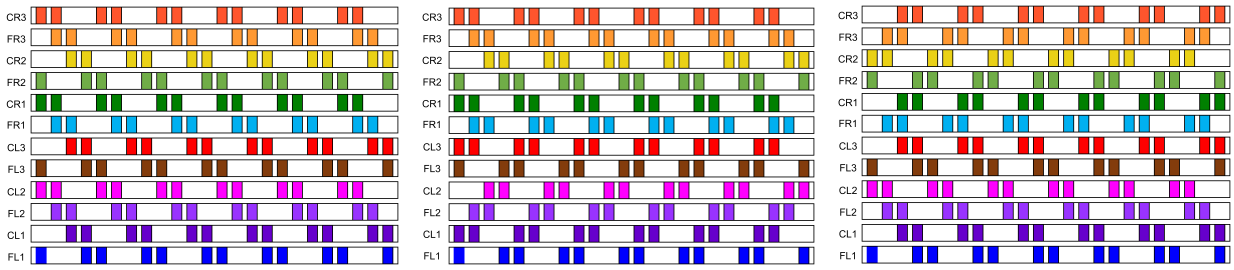
where

<ch> = channel, decimal number between 0 and 31.

<pwm> = Pulse width in microseconds, from 500 to 2500.

<time> = Time in milliseconds to complete a servo movement.

<cr> = carriage return, character ASCII 13.



(a) walk forward (b) turn-left (c) turn-right

FIGURE 9. Gait patterns: turn-left, turn-right, and walk forward. Raster plot generated in Matlab (pulsating neural network simulation).

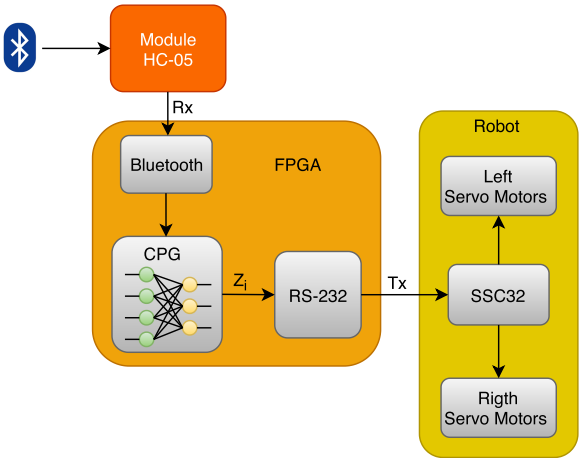


FIGURE 10. Locomotion system block diagram.

The Bluetooth sub-module is also based on the RS-232 protocol; in this case, ASCII code instructions from the iQSA module are used to indicate to the CPG submodule the pattern which the robot must generate (turn left, turn right or stop). The locomotion patterns, turn-left, and turn-right are shown in Figure 9. These patterns are obtained by denying the exits of neurons corresponding to the left and right coxa, respectively, in the walk forward pattern.

IV. IMPLEMENTATION AND RESULTS

The implementation of the proposed system was carried out in 2 phases: (1) Offline experiment for the generation of decision rules and (2) Real-time experiment.

A. OFFLINE EXPERIMENT

This phase is essential to train the recognition algorithm. It includes the MI training process to familiarize the user with the imagined commands using visual stimulation, receive feedback to speed up the learning, and improve the real-time experiment's performance.

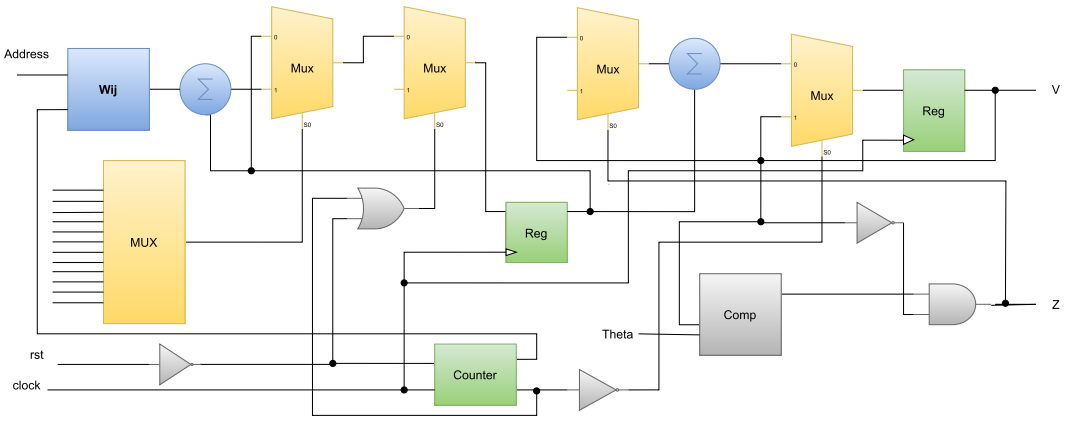
1) EEG DATA ACQUISITION AND PROCESSING

For the EEG data acquisition, we recruited 120 healthy people, **51 women and 69 men between 23 and 44 years of age, who were instructed about the experiment and its duration. After, subjects were asked to fill out a form to collect information about the current state in which they are presented to the test, such as insomnia, stress, fatigue, headache, among others, which may influence information processing. It is essential to mention the experiment followed ethical guidelines obtaining the participants' consent and guaranteeing their privacy.**

We enabled a place to avoid distractions and eliminate external audible or visual noise. The experimental protocol was as follows: The Emotiv-Epoc headband was placed on each participant (Figure 13), taking care of the electrodes' correct placement with the scalp. After that, we asked each subject to perform mentally three actions (turn-left, turn-right, and stop) according to the visual stimulus displayed on the screen randomly. The image of a red arrow sliding to left and right corresponded to the mental imagination turn-left (TL) and turn-right (TR), respectively. The image of a fixed cross in the center of the screen signal to the user a state of rest (action mental: stop (S)).

Figure 12, shows the outline of the training session. In the beginning, appears the visual stimulus "cross" for $3s$ to indicate to the user the state of rest and waiting for the next stimulus. Then, from the second $t = 4s$, one of the two arrows was shown: TL or TR with its respective displacement for $5s$. Each stimulus, indicated by the red arrow instructs the subject to imagine movement in the specified direction until the "cross" stimulus reappeared at time $t = 9s$. The experimental process is randomly repeated for 5 minutes, showing a total of 38 arrows (TL, TR) and 38 cross stimuli (S), with 3 seconds of rest between them. Thus, we record 38,400 samples corresponding to each user's experiment time, and 4,608,000 samples from the 120 sessions carried out.

Next, from the registered data, an analysis was conducted using 64 samples in the iQSA module to obtain the quaternions and their rotations in each iteration. This process reduced the matrix of original data to an array of quaternions with 612,121 final samples. Here, statistical measurements presented in Table 2 were carried out to obtain the array of characteristics $M_{i,j}$, where we took 70-30% of the data for the training and validation stage respectively. The data was classified using a decision tree to obtain: decision rules, recognition rates (RT) and error (ET). This process was



(a)

FIGURE 11. Pulsating neuron design diagram.

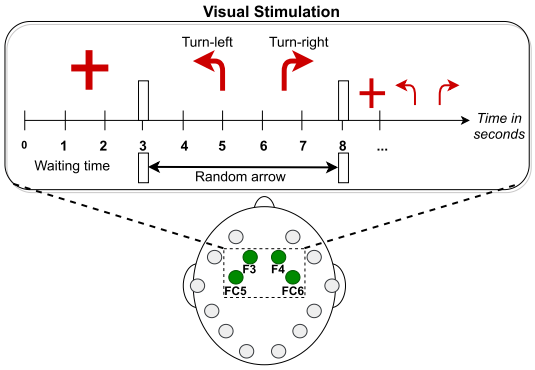


FIGURE 12. Experiment performed on users for data acquisition to train the classifier.

performed 20 times by selecting randomly the data for each repetition. The obtained decision rules were coded in Python and integrated into the iQSA online module for the real-time validation stage.

2) BCI PERFORMANCE

The classification results of all users were averaged and are shown in Table 3. These results were evaluated using the confusion matrix output of the Decision Tree classifier. The table shows the percentage of recognition obtained from the three motor imagery activities, carried out by each participant in the experiment. We can see that subject 33 got the precision highest of 99.87%, while subject 11 had the lowest precision of 46.21%. Finally, all participants' average recognition rate was 97.72% and an average error rate of 2.27%.

We use other assessment metrics such as sensitivity(S) and specificity(Sp) (Table 4). The sensitivity metric shows that the classifier can recognize samples from the relevant class, and the specificity measures whether the classifier can recognize samples that do not belong to the relevant class. In this test, the sensitivity average for class 0($S0$) associated with the waiting time mental state was 99.12%, and the sensitivity average for class 1($S1$) and class 2($S2$) associated with turn-left and turn-right was 99.17% and 99.57% respectively. Then, it is implied that the classifier had no problem categorising classes. In turn, the specificity rate for class 0($Sp0$) shows that 99.37% of the samples classified as negative were actually negative, while class 1($SP1$) performed at 99.35% and class 2($SP2$) at 99.15%. The precision (P) of each class was 96.48%, 95.73%, and 99.57% for classes 0, 1, and 2 respectively.

Once the iQSA module's decision rules were obtained, they were codified and implemented in Python to evaluate the functioning of the robot online.

3) COMPARISON WITH OTHER SYSTEMS

To evaluate the performance of our approach, we have carried out a comparison with other BCI systems. Table 5 shows performances reported under similar experimental conditions using a left-right imagery engine. Li and Daeglau obtained 71.72% and 63% of accuracy controlling a humanoid robot. Jiang and Barbosa controlled a mobile robot with motor imagery, getting 95% and 90% of mean accuracy, and our study shows a better accuracy to control a bio-inspired hexapod robot. Besides, response times reported since the subject initiates the command until it is executed by the robot are around 1.6 and 2.6 seconds. With our method, the response time is of 1.5 seconds, shorter than reported in other studies.

To analyze our results, we performed a significant statistical test making use of the STAC (Statistical Tests for Algorithms Comparison) web platform [56]. Here, we chose the Friedman test with a significance level of 0.05 to rank the algorithms and check if the differences between them are statistically significant.

Table 6 shows the Friedman test ranking results obtained with the *p-value* approach. We can observe that our proposal gets the lowest ranking; that is, the iQSA model has the best accuracy among all the algorithms.

In order to compare whether the differences between iQSA and the other methods are significant, a Li post hoc procedure

TABLE 3. Accuracy obtained after apply the iQSA method (RT and ET refer to the recognition and error rates respectively).

Classification Accuracy											
User	RT	ET	User	RT	ET	User	RT	ET	User	RT	ET
1	0.99560	0.00440	31	0.99588	0.00412	61	0.99141	0.00859	91	0.99577	0.00423
2	0.99583	0.00417	32	0.99444	0.00556	62	0.99453	0.00547	92	0.99678	0.00322
3	0.99844	0.00156	33	0.99870	0.00130	63	0.99806	0.00194	93	0.99481	0.00519
4	0.99339	0.00661	34	0.99374	0.00626	64	0.99615	0.00385	94	0.99596	0.00404
5	0.56686	0.43314	35	0.99518	0.00482	65	0.99600	0.00400	95	0.99524	0.00476
6	0.99067	0.00933	36	0.99756	0.00244	66	0.99547	0.00453	96	0.99532	0.00468
7	0.59856	0.40144	37	0.99823	0.00177	67	0.99387	0.00613	97	0.99424	0.00576
8	0.99805	0.00195	38	0.99323	0.00677	68	0.99355	0.00645	98	0.99700	0.00300
9	0.99686	0.00314	39	0.99680	0.00320	69	0.99446	0.00554	99	0.99748	0.00252
10	0.99580	0.00420	40	0.99461	0.00539	70	0.99858	0.00142	100	0.99774	0.00226
11	0.99723	0.00277	41	0.99782	0.00218	71	0.99482	0.00518	101	0.99266	0.00734
12	0.46217	0.53783	42	0.99782	0.00218	72	0.99777	0.00223	102	0.99597	0.00403
13	0.99623	0.00377	43	0.99370	0.00630	73	0.99656	0.00344	103	0.99787	0.00213
14	0.99247	0.00753	44	0.51837	0.48163	74	0.99758	0.00242	104	0.99349	0.00651
15	0.99569	0.00431	45	0.99670	0.00330	75	0.99368	0.00632	105	0.99546	0.00454
16	0.99617	0.00383	46	0.99683	0.00317	76	0.99545	0.00455	106	0.99439	0.00561
17	0.99241	0.00759	47	0.99760	0.00240	77	0.99426	0.00574	107	0.99662	0.00338
18	0.99255	0.00745	48	0.99629	0.00371	78	0.99498	0.00502	108	0.99494	0.00506
19	0.99631	0.00369	49	0.99490	0.00510	79	0.99789	0.00211	109	0.99570	0.00430
20	0.99750	0.00250	50	0.99763	0.00237	80	0.99175	0.00825	110	0.99394	0.00606
21	0.99528	0.00472	51	0.99517	0.00483	81	0.99576	0.00424	111	0.99648	0.00352
22	0.99366	0.00634	52	0.99722	0.00278	82	0.99246	0.00754	112	0.99705	0.00295
23	0.99428	0.00572	53	0.99679	0.00321	83	0.99550	0.00450	113	0.99708	0.00292
24	0.99831	0.00169	54	0.99597	0.00403	84	0.99536	0.00464	114	0.99657	0.00343
25	0.99780	0.00220	55	0.99550	0.00450	85	0.99580	0.00420	115	0.99462	0.00538
26	0.63436	0.36564	56	0.99560	0.00440	86	0.99657	0.00343	116	0.99716	0.00284
27	0.99522	0.00478	57	0.99212	0.00788	87	0.99426	0.00574	117	0.99566	0.00434
28	0.99503	0.00497	58	0.99482	0.00518	88	0.99314	0.00686	118	0.99776	0.00224
29	0.99736	0.00264	59	0.99353	0.00647	89	0.99289	0.00711	119	0.99534	0.00466
30	0.99790	0.00210	60	0.99608	0.00392	90	0.99519	0.00481	120	0.99372	0.00628

TABLE 4. Assessment Metrics obtained after applying iQSA method.

Assessment Metrics			
Metric	[Class 0]	[Class 1]	[Class 2]
S	0.99126	0.99179	0.99575
Sp	0.99379	0.99354	0.99152
P	0.96483	0.95732	0.99575

was performed (Table 7). The differences are statistically significant because the p-values are below 0.1.

B. REAL-TIME EXPERIMENT

To evaluate the real-time proposed system's performance, we experimented with three subjects who were instructed to imagine all three states to control the hexapod robot: turn-left, turn-right, and relax for actions TL, TR, and S, respectively. Each time the user performs a motor imagination activity, the developed system displays the command detected on the screen as visual feedback of the imagined movements. Simultaneously, the command is sent to the hexapod robot to execute the corresponding locomotion pattern (Figure13).

An environment like the one in Figure 14 was used to instruct the subjects to control the hexapod robot. The robot must walk from one initial position until a target position on a 3mx2m ground. It is essential to mention that the robot was configured to execute the walk forward pattern gait, besides it was included an ultrasound sensor to the robot to prevent it from colliding with some obstacle. In this way, if the robot found an obstacle, then it stops, waiting for instructions to "turn-left", "turn-right", or "stop".

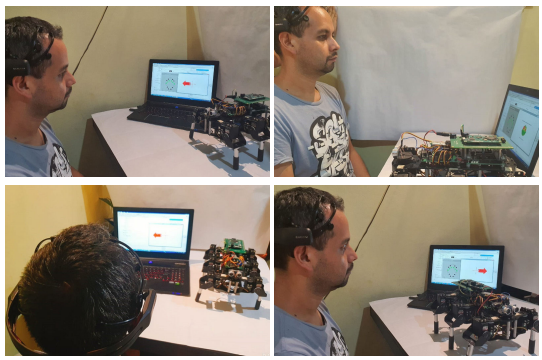


FIGURE 13. User electrode placement and training stage.

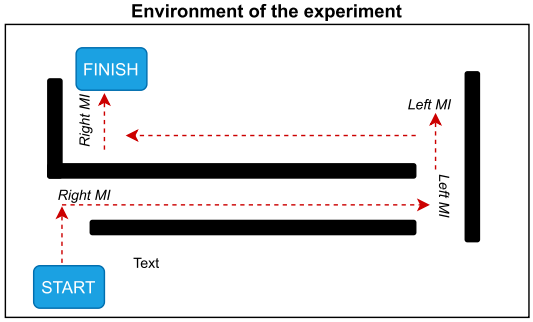


FIGURE 14. Environment of the experiment of 3*m* × 2*m* area.

In the experiment, subjects were asked to complete four cognitive tasks (two left and two right). Table 8 shows the

TABLE 5. Comparative BCRI performances.

Study	Subjects	Method applied	Acurracy online	Robot	Response Time
Present study	120	iQSA-DecicionTree	97.72	Hexapod	1.5s
Saduano et al. [32]	7	Soft-voting classification	80.8	Robotic Manipulator	–
Jian et al. [57]	4	CSP, LDA	95.0	Mobile robot	2.6s
Li et al. [58]	3	CSP-based SVM	71.72	Humanoid	1.6s
Barbosa et al. [30]	4	Hierarchical Model, MLP	90.00	Mobile robot	2s
Bousseta et al. [28]	4	PCA, FFT, SVM(RBF)	85.45	Robot arm	–
Daeglau et al. [59]	27	LDA	63	Two Humanoid	–

TABLE 6. Li Post Hoc adjusted *p-value* for the test error ranking.

Algorithm	Ranking
iQSA	1.000
Jian	2.500
Bousseta	3.000
Saduano	3.500

TABLE 7. Li Post Hoc adjusted *p-value* for the test error ranking.

Comparison	p-value
iQSA vs Saduano	0.00681
iQSA vs Bousseta	0.03066
iQSA vd Jian	0.10035

TABLE 8. Experimental result in real-time.

Assessment Metrics		
Subject	Commands	[Action Executed]
A	4	2 TL, 2 TR
B	4	2 TL, 2 TR
C	2	2 TL

TABLE 9. Execution time of iQSA process.

iQSA Process	Time in seconds offline	online
iQSA quaternion (1)	0.20540	0.00999
iQSA Classification (2)	0.14006	0.00001
Total Time	0.34546	0.01000

number of mental actions performed, the percentage of precision obtained regarding recognized mental activities, and the average time from intention to the command's execution. We can see each subject's total time to complete the entire established path in the table. Subject A and B managed to complete four commands, but subject C found it quite challenging to complete the path; this might be due to a lack of concentration or desperation to accomplish the task. Another aspect to be consider is that motor imagination activities require a lot of attention and training, so that the system needs to be trained to adapt it to each individual in particular and achieve better results.

A real-time system must have the ability to perform tasks in a response time determined. We show the time required for each of the functions performed by the iQSA algorithm (Table 9): (1) quaternion (2) and classification. According to the experiment, for offline classification (processes 1 and 2), the processing time required was 0.34546 seconds. However, the time online executed to recognize the motor imagery activity generated by each subject was of 0.01 seconds, considering that the learning trees were already developed, even when the processing phase was done in real-time.

Finally, the delay time since the iQSA module identifies an action command until that is received and executed by the robot is between 1 to 1.5s

V. CONCLUSION

The relationship between neuroscience and robotics has led to various investigations related to living beings' locomotion and adapting them to robotic systems. This research combines the following elements: BCI interface, iQSA method, imagery engine, CPG's systems, pulsating neural networks, and a hexapod robot to establish a communication system between them. That is, we developed a system to control a bio-inspired hexapod robot in real-time with three cognitive instructions given by the user: turn-left, turn-right, and stop. We used spiking neural networks that mimic a CPG behavior to reproduce and execute the orders received. The EEG signals generated in the BCI were classified using the iQSA method, which bases its operation on quaternions' algebra. The average accuracy rate obtained in tests performed offline was 97.72%, with an average time of 0.34s since the EEG signals were acquired until their processing by the iQSA method. Instead, the online time only requires 0.01s to capture the order and execute the movement.

Performance results obtained in the real-time experiment and offline processing times showed that the system developed effectively controls a bio-inspired hexapod robot based on motor imagination activities. Furthermore, we can also conclude that the iQSA method is ideal for controlling a robotic system in real-time. So far, the system responds to three cognitive activities. Still, it is intended as future work to extend the number of commands and perform algorithm processing in hardware without relying on a central computer.

REFERENCES

[1] J. Wolpaw, N. Birbaumer, D. McFarland, G. Pfurtscheller, and T. Vaughan, "Brain-computer interfaces for communication and control," *Clin. Neurophys.*, vol. 113, no. 6, pp. 767–791, 2002.

[2] I. Lazarou, S. Nikolopoulos, P. C. Petrantonakis, I. Kompatsiaris, and M. Tsolaki, "EEG-based Brain–Computer interfaces for communication and rehabilitation of people with motor impairment: A novel approach of the 21st century," *Frontiers Human Neurosci.*, vol. 12, p. 14, Jan. 2018.

[3] K. Bowsher *et al.*, "Brain–computer interface devices for patients with paralysis and amputation: A meeting report," *J. Neural Eng.*, vol. 13, no. 2, p. 23001, Apr. 2016.

[4] U. Chaudhary, N. Birbaumer, and A. Ramos-murguialday, "Brain–computer interfaces for communication and rehabilitation," *Nature Rev. Neurol.*, vol. 12, no. 9, pp. 513–525, 2016.

[5] Y. Baram and A. Miller, "Virtual reality cues for improvement of gait in patients with multiple sclerosis," *Neurology*, vol. 66, no. 2, pp. 178–181, Jan. 2006.

[6] J. Faller, G. Müller-Putz, D. Schmalstieg, and G. Pfurtscheller, "An application framework for controlling an avatar in a desktop-based virtual environment via a software SSVEP brain–computer interface," *Presence Teleoperators Virtual Environ.*, vol. 19, no. 1, pp. 25–34, 2010.
[7] A. Benitez-Andonegui, R. Burden, R. Benning, R. Möckel, M. Lührs, and B. Sorger, "An augmented-reality fNIRS-based brain-computer interface: A proof-of-concept study," *Frontiers Neurosci.*, vol. 14, p. 346, Apr. 2020.
[8] H. Si-Mohammed, F. Argelaguet, G. Casiez, N. Roussel, and A. Lécuyer, "Brain-computer interfaces and augmented reality: A state of the art," in *Proc. 7th Graz Brain-Comput. Interface*, Graz, Austria, Sep. 2017, pp. 447–452.
[9] A. Lenhardt and H. Ritter, "An augmented-reality based brain-computer interface for robot control," in *Neural Information Processing. Models and Applications*, vol. 6444. Berlin, Germany: Springer, 2010, pp. 58–65.
[10] E. Lalor, S. Kelly, C. Finucane, R. Burke, R. Smith, R. Reilly, and G. McDarby, "Steady-state vep-based brain-computer interface control in a inmersive 3D gaming environment," *EURASIP J. Appl. Signal Process.*, vol. 19, pp. 3156–3164, Dec. 2004.
[11] M. Middendorf, G. Mcmillan, G. Calhoun, and K. S. Jones, "Brain-computer interfaces based on the steady-state visual-evoked response," *IEEE Trans. Rehabil. Eng.*, vol. 8, no. 2, pp. 211–214, Jun. 2000.
[12] R. J. Rodriguez, "Electroencephalogram (EEG) based authentication leveraging visual evoked potentials (VEP) resulting from exposure to emotionally significant images," in *Proc. IEEE Symp. Technol. Homeland Secur. (HST)*, May 2016, pp. 1–6.
[13] S.-C. Chen, S.-C. Hsieh, and C.-K. Liang, "An intelligent brain computer interface of visual evoked potential EEG," in *Proc. 8th Int. Conf. Intell. Syst. Design Appl.*, Nov. 2008, pp. 343–346.
[14] U. Hoffman, J. M. Vesin, and K. Diserens, "An efficient p300-based brain-computer interface for disabled subjects," *J. Neurosci. Methods*, vol. 167, no. 1, pp. 115–125, 2008.
[15] J. D. Bayliss, "Use of the evoked potential P3 component for control in a virtual apartment," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 11, no. 2, pp. 113–116, Jun. 2003.
[16] Y. Liu, H. Ayaz, B. Onaral, and P. A. Shewokis, "EEG band powers for characterizing user engagement in P300-BCI," in *Proc. 6th Int. IEEE/EMBS Conf. Neural Eng. (NER)*, Nov. 2013, pp. 1066–1069.
[17] A. Curtin, H. Ayaz, Y. Liu, P. A. Shewokis, and B. Onaral, "A P300-based EEG-BCI for spatial navigation control," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2012, pp. 3841–3844.
[18] Y. Li, J. Long, T. Yu, Z. Yu, C. Wang, H. Zhang, and C. Guan, "An EEG-based BCI system for 2-D cursor control by combining Mu/Beta rhythm and P300 potential," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 10, pp. 2495–2505, Oct. 2010.
[19] H. Gevensleben, B. Albrecht, H. Lsütcke, T. Auer, W. I. Dewiputri, R. Schweizer, G. Moll, H. Heinrich, and A. Rothenberger, "Neurofeedback of slow cortical potentials: Neural mechanisms and feasibility of a placebo-controlled design in healthy adÓults," *Frontiers Hum. Neurosci.*, vol. 8, p. 990, Dec. 2014.
[20] N. Birbaumer, L. E. Roberts, W. Lutzenberger, B. M. Rockstroh, and T. Elbert, "Area-specific self-regulation of slow cortical potentials on the sagital midline and its effects on behavior," *Electroencephalogr. Clin. Neurophysiol*, vol. 84, no. 4, p. 353, 1992.
[21] N. Robinson, A. P. Vinod, C. Guan, K. Keng Ang, and T. Keng Peng, "A wavelet-CSP method to classify hand movement directions in EEG based BCI system," in *Proc. 8th Int. Conf. Inf., Commun. Signal Process.*, Singapore, Dec. 2011, pp. 1–5.
[22] A. Chatterjee, V. Aggarwal, A. Ramos, S. Acharya, and N. V. Thakor, "A brain-computer interface with vibrotactile biofeedback for haptic information," *J. NeuroEngineering Rehabil.*, vol. 4, no. 1, pp. 4–40, Dec. 2007.
[23] B. Kamousi, A. N. Amini, and B. He, "Classification of motor imagery by means of cortical current density estimation and von neumann entropy," *J. Neural Eng.*, vol. 4, no. 2, pp. 17–25, Jun. 2007.
[24] G. Pfurtscheller, C. Brunner, A. Schlögl, and F. H. Lopes da Silva, "Mu rhythm (de)synchronization and EEG single-trial classification of different motor imagery tasks," *NeuroImage*, vol. 31, no. 1, pp. 153–159, May 2006.
[25] J. Zhao, Q. Meng, W. Li, M. Li, and G. Chen, "SSVEP-based hierarchical architecture for control of a humanoid robot with mind," in *Proc. 11th World Congr. Control Automat.*, Shenyang, China, 2014, pp. 2401–2406.
[26] S. Zhao, Z. Li, R. Cui, Y. Kang, F. Sun, and R. Song, "Brain–Machine interfacing-based teleoperation of multiple coordinated mobile robots," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5161–5170, Jun. 2017.
[27] L. Georgescu, D. Wallace, D. Kyong, A. Chun, K. Chun, and P. Oh, "The future of work: Towards service robot control through brain-computer interface," in *Proc. 10th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Las Vegas, NV, USA, Jan. 2020, pp. 0932–0937.
[28] R. Bousseta, I. El Ouakouak, M. Gharbi, and F. Regragui, "EEG based brain computer interface for controlling a robot arm movement through thought," *IRBM*, vol. 39, no. 2, pp. 129–135, Apr. 2018.
[29] B. Xu, W. Li, X. He, Z. Wei, D. Zhang, C. Wu, and A. Song, "Motor imagery based continuous teleoperation robot control with tactile feedback," *Electronics*, vol. 9, no. 1, p. 174, Jan. 2020.
[30] A. O. G. Barbosa, D. R. Achanccaray, and M. A. Meggiolaro, "Activation of a mobile robot through a brain computer interface," in *Proc. IEEE Int. Conf. Robot. Autom.* Anchorage, AK, USA, May 2010, pp. 4815–4821.
[31] S. Qui, Z. Li, W. He, L. Zhang, Ch. Yang, and C. Su, "Brain–machine interface and visual compressive sensing-based teleoperation control of an exoskeleton robot," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 1, pp. 58–69, Feb. 2017.
[32] B. Saduanov, D. Tokmurzina, K. Kunanbayev, and B. Abibullaev, "Design and optimization of a real-time asynchronous BCI control strategy for robotic manipulator assistance," in *Proc. 8th Int. Winter Conf. Brain-Comput. Interface (BCI)*, Gangwon, Korea (South), Feb. 2020, pp. 1–5.
[33] G. R. Müller-Putz, R. Scherer, G. Pfurtscheller, and R. Rupp, "EEG-based neuroprosthesis control: A step towards clinical practice," *Neurosci. Lett.*, vol. 382, nos. 1–2, pp. 169–174, Jul. 2005.
[34] A. Cichocki and K. Choi, *Control of a Wheelchair by Motor Imagery in Real Time*. Berlin, Germany: Springer, 2008, pp. 330–337.
[35] J. McFarland, L. A. Miner, T. M. Vaughan, and J. R. Wolpaw, "Mu and beta rhythm topographies during motor imagery and actual movements," *Brain Topography*, vol. 12, no. 3, pp. 177–186, 2000.
[36] O. Bai, D. Huang, D.-Y. Fei, and R. Kunz, "Effect of real-time cortical feedback in motor imagery-based mental practice training," *NeuroRehabilitation*, vol. 34, no. 2, pp. 355–363, May 2014.
[37] J. Krichmar and H. Wagatsuma, *Neuromorphic and Brain-Based Robots*. Cambridge, U.K. Cambridge, Univ. Press, 2011.
[38] C. Bartolozzi, F. Rea, C. Clercq, D. B. Fasnacht, G. Indiveri, M. Hofstatter, and G. Metta, "Embedded neuromorphic vision for humanoid robots," in *Proc. CVPR WORKSHOPS*, Colorado Springs, CO, USA, Jun. 2011, pp. 129–135.
[39] S. Menon, S. Fok, A. Neckar, O. Khatib, and K. Boahen, "Controlling articulated robots in task-space with spiking silicon neurons," in *Proc. 5th IEEE RAS/EMBS Int. Conf. Biomed. Robot. Biomechatronics*, Aug. 2014, pp. 181–186.
[40] L. H. Scott, "Central pattern generator," *Current Biol.*, vol. 10, no. 2, pp. 176–177, 2000.
[41] C. M. A. Pinto, "Central pattern generator for legged locomotion: A mathematical approach," in *Proc. Workshop Robot. Math.*, vol. 16, 2007, pp. 1–6.
[42] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: A review," *Neural Netw.*, vol. 21, no. 4, pp. 642–653, May 2008.
[43] J. Yu, M. Tan, J. Chen, and J. Zhang, "A survey on CPG-inspired control models and system implementation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 3, pp. 441–456, Mar. 2014.
[44] A. Russell, G. Orchard, and R. Etienne-Cummings, "Configuring of spiking central pattern generator networks for bipedal walking using genetic algorithms," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2007, pp. 1525–1528.
[45] H. Rostro-Gonzalez, P. A. Cerna-Garcia, G. Trejo-Caballero, C. H. Garcia-Capulin, M. A. Ibarra-Manzano, J. G. Avina-Cervantes, and C. Torres-Huitzil, "A CPG system based on spiking neurons for hexapod robot locomotion," *Neurocomputing*, vol. 170, pp. 47–54, Dec. 2015.
[46] A. Espinal, H. Rostro-Gonzalez, M. Carpio, E. I. Guerra-Hernandez, M. Ornelas-Rodriguez, H. J. Puga-Soberanes, M. A. Sotelo-Figueroa, and P. Melin, "Quadrupedal robot locomotion: A biologically inspired approach and its hardware implementation," *Comput. Intell. Neurosci.*, vol. 2016, p. 13, Jun. 2016.
[47] E. I. Guerra-Hernandez, A. Espinal, P. Batres-Mendoza, C. H. Garcia-Capulin, R. De J. Romero-Troncoso, and H. Rostro-Gonzalez, "A FPGA-based neuromorphic locomotion system for multi-legged robots," *IEEE Access*, vol. 5, pp. 8301–8312, 2017.

[48] P. Batres-Mendoza, M. A. Ibarra-Manzano, E. I. Guerra-Hernandez, D. L. Almanza-Ojeda, C. R. Montoro-Sanjose, R. J. Romero-Troncoso, and H. Rostro-Gonzalez, "Improving EEG-based motor imagery classification for real-time applications using the QSA method," *Comput. Intell. Neurosci.*, vol. 2017, pp. 1–16, May 2017.
[49] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," Stanford Univ., Stanford, CA, USA, Tech. Rep., 2006.
[50] W. R. Hamilton, "On quaternions," *Proc. Roy. Acad.*, vol. 3, no. 48, pp. 1–16, 1847.
[51] A. Janota, V. Šimák, D. Nemec, and J. Hrbček, "Improving the precision and speed of euler angles computation from low-cost rotation sensor data," *Sensors*, vol. 15, no. 3, pp. 7016–7039, Mar. 2015.
[52] T. G. Brown, "On the nature of the fundamental activity of the nervous centres; together with an analysis of the conditioning of rhythmic activity in progression, and a theory of the evolution of function in the nervous system," *J. Physiol.*, vol. 48, no. 1, pp. 18–46, Mar. 1914.
[53] M. Grabowska, E. Godlewska, J. Schmidt, and S. Daun-Gruhn, "Quadrupedal gaits in hexapod animals–inter-leg coordination in free-walking adult stick insects," *J. Experim. Biol.*, vol. 215, no. 24, pp. 4255–4266, Dec. 2012.
[54] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge, U.K.: Cambridge Univ. Press, 2002.
[55] A. Destexhe, "Conductance-based integrate-and-fire models," *Neural Comput.*, vol. 9, no. 3, pp. 503–514, Mar. 1997.
[56] I. Rodríguez-Fdez, A. Canosa, M. Mucientes, and A. Bugarin, "STAC: A Web platform for the comparison of algorithms using statistical test," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Istanbul, Turkey, Aug. 2015, pp. 1–8.
[57] J. Jiang, B. Zhao, P. Zhang, Y. Bai, and X. Chen, "Brain-actuated humanoid robot based on brain-computer interface (BCI)," in *Proc. IEEE Int. Conf. Autom., Electron. Electr. Eng. (AUTEEE)*, Nov. 2018, pp. 319–322.
[58] J. Li, Z. Li, Y. Feng, Y. Liu, and G. Shi, "Development of a human–robot hybrid intelligent system based on brain teleoperation and deep learning SLAM," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 4, pp. 1664–1674, Oct. 2019.
[59] M. Daeglau, F. Wallhoff, S. Debener, I. Condro, C. Kranczioch, and C. Zich, "Challenge accepted? Individual performance gains for motor imagery practice with humanoid robotic EEG neurofeedback," *Sensors*, vol. 20, no. 6, p. 1620, Mar. 2020.

PATRICIA BATRES-MENDOZA received the B.E. and M.S. degrees in software engineering from the Instituto Tecnologico de Orizaba, in 2001 and 2003, respectively, and the D.Sc. degree (Hons.) in electrical engineering from the Universidad de Guanajuato, Mexico, in 2018. Since 2008, he has been a Professor with the Universidad Autonoma Benito Juarez de Oaxaca, Mexico. His current research interests include the brain signal processing, computational neuroscience, bio-inspired algorithms, and artificial neural networks.

ERICK ISRAEL GUERRA-HERNANDEZ received the B.S. degree in electronics and the M.S. degree (Hons.) in electronics from the Benemerita Universidad Autónoma de Puebla, Mexico, in 2003 and 2007, respectively, and the D.Sc. degree in electrical engineering from the Universidad de Guanajuato, Mexico, in 2019. Since 2008, he has been a Professor with the Universidad Autonoma Benito Juarez de Oaxaca, Mexico. His current research interests include digital designs in FPGAs, bio-inspired systems, robotics, computational vision, and fuzzy logic.

ANDRES ESPINAL received the B.S. degree in computational systems engineering and the master's degree in computer science from the Tecnologico Nacional de Mexico, in 2009 and 2011, respectively, and the Ph.D. degree in computer science from the Tecnologico Nacional de Mexico, Tijuana. He is currently a full-time Professor with the Department of Organizational Studies, Universidad de Guanajuato. He has authored several journals and proceedings articles. His research interests include evolutionary algorithms, artificial neural networks, computer vision, digital image processing, and bio inspired algorithms.

EDUARDO PÉREZ-CARETA (Member, IEEE) received the M.S. degree in electrical engineering from the Universidad de Guanajuato, in 2008, where he is currently pursuing the Ph.D. degree in electrical engineering. He was an Instrumentation Engineer with Petroleos Mexicanos, Salamanca, from 2008 to 2016.

HORACIO ROSTRO-GONZALEZ received the B.S. degree in electronic engineering from the Instituto Tecnologico de Celaya, Mexico, in 2003, the M.E. degree (Hons.) in electrical engineering from the Universidad de Guanajuato, Mexico, in 2006, and the D.Sc. degree in computational neuroscience from the University of Nice Sophia Antipolis, France, in 2011. From 2011 to 2012, he was a Research Fellow with the University of Cyprus, where he was involved in the FP7 EU SCANDLE Project. Since 2012, he has been a Full Professor with the Department of Electronics, Universidad de Guanajuato. He has authored around 40 journals and conferences proceeding articles. His current research interests include reconfigurable electronic, neuromorphic engineering, computational neuroscience, bio-inspired algorithms, parallel computing, and signal processing.