RESEARCH ARTICLE

BON VIEW PUBLISHING

Electroencephalogram Signal Acquisition System with Machine Learning for Robotic Prosthesis Control: In Vivo Dataset

Bruno J. Santos[1,2,3,*] , **Alysson Avila[1]**, **Marcelo Barboza[1]**, **Evandro Drigo[1]**, **Tarcisio Leão[1]** and **Eduardo Bock[1]**

[1]*Department of Electric, Federal Institute of São Paulo, Brazil*

[2]*Department of Information Technology, University São Judas Tadeu, Brazil*

[3]*Department of Electric, University of São Paulo, Brazil*

Abstract: This study addresses the increasing global need for upper limb prostheses (ULPs), particularly those controlled by electroencephalogram (EEG) signals, due to the rising number of amputations. Focusing on an EEG signal acquisition system integrated with a machine learning (ML)-driven pattern recognition framework, the research investigates the control of a robotic ULP. The study is divided into two phases: EEG data acquisition and pattern recognition using an ensemble of K-nearest neighbors (KNN), support vector machines (SVM), and artificial neural networks (ANN) models within a brain-machine interface. Each ML model demonstrated distinct strengths—KNN in rapid pattern recognition, SVM in reliable state differentiation, and ANN in handling complex, non-linear data relationships. The ensemble ML (eML) leveraged these strengths, achieving approximately 90% accuracy in final training rounds and showing superior performance compared to individual models. The eML was successfully integrated into the robotic ULP control system, demonstrating high potential for real-world applications by efficiently processing brain activity signals and making precise control decisions.

Keywords: electroencephalogram, artificial intelligence, machine learning, ensemble AI, brain-machine interface, robotic upper limb prostheses

1. Introduction

Approximately 16% of the global population has a significant disability [1]. Upper limb prostheses (ULPs) emerge as devices designed to compensate for the partial or total absence of a limb, aiming to restore both the esthetic and functional abilities of individuals in performing daily activities such as dressing, eating, using equipment, and participating in various activities [2]. ULPs, as assistive technologies, can be categorized into cosmetic, mechanical, and myoelectric types. Myoelectric ULPs represent the pinnacle of this technology, as they utilize biosignals obtained directly from the user to enable automatic control of the prosthesis [3]. An electrical biosignal generated by brain activity can be detected using electroencephalogram (EEG) techniques and subsequently processed through an interface to control the device's functioning [4].

Although non-invasive EEG-controlled ULPs hold promise for improving the quality of life for individuals with physical disabilities, the scientific literature lacks sufficient exploration of their practical application and benefits [5, 6]. Research in this domain remains nascent, leaving critical gaps in knowledge regarding the design, efficacy, and accessibility of EEG-controlled prostheses for this population.

This study aims to address this gap by developing a (machine learning) ML-based system for acquiring, processing, and utilizing EEG signals to control a robotic ULP. It employs a structured, well-referenced methodology leveraging publicly available tools to facilitate advancements in bioengineering and assistive technologies, thereby improving accessibility for diverse users.

The exposition is organized as follows: Section 1 provides an introduction, setting the stage for the study. Section 2 comprehensively reviews related works, focusing on Robotic ULP and EEG signal acquisition systems. Section 3 details the materials and methods used in the research. The findings are outlined in Section 4, with an in-depth discussion in Section 5. Finally, Section 6 concludes the exposition with a summary of key insights.

2. Related Work

2.1. Robotic ULP

The history of ULP has deep roots in human civilization. A prosthesis was discovered on the forearm of an Egyptian mummy dating back over 2000 BCE. Around 400 BCE, the hand hook was introduced as a prosthetic device to replace missing limbs.

*Corresponding author: Bruno J. Santos, Department of Electric, Federal Institute of São Paulo and Department of Electric, University of São Paulo, Brazil. Email: brunojsantos@usp.br

Advances in ironworking made prosthetics more durable and capable of handling heavier objects, as evidenced by a Roman general's use of an iron prosthesis in the first century CE. By 1500 CE, a craftsman developed a steel arm for German knights, enabling independent movement of each finger phalange. A skeleton dated between 1450 and 1620, found in southern Freising, Germany, featured a metal prosthetic hand covered with leather and attached to the arm with bandages. This metal prosthesis is one of approximately 50 found in Central Europe, dating from the late Middle Ages to the early Modern era, some of which were notable for their movable and sophisticated components [7–9].

The role of prosthetic devices in non-military settings became more prominent over time, indicating an expansion in societal applications and a shift in priorities toward improving patient care and daily functionality. In the 17th century, non-military prosthetics emerged, such as "Le Petit Loraine", developed by military physician Ambroise Paré, featuring movable fingers and flexion-extension movement at the elbow. In 1818, German dentist Peter Baliff created a self-propelled prosthesis operated by leather straps connected to muscles. In the 1860s, Count Beaufort utilized levers and contralateral shoulder power to operate prosthetics. By 1916, German surgeon Sauberbruch described a limb design controlled by muscle movements. The first clinically accepted myoelectric ULP appeared in 1960, though it faced challenges like excessive weight and low electrical reliability [7–9]. Since then, myoelectric ULP has undergone continuous improvements in functionality, materials, design, and ergonomics [2].

Myoelectric ULP are characterized by their ability to replicate natural body movements. Achieving this level of functionality requires an external power source, such as batteries, and various signal input methods, including electromyography (EMG), electrocorticography (ECoG), and EEG. These input systems enable myoelectric ULP to interpret user commands and translate them into automatic movements, emulating the function of the amputated limb [10].

EMG measures the electrical activity of skeletal muscles and is often used to diagnose neuromuscular disorders, monitor post-injury recovery, and study biomechanics. EMG electrodes, placed on the skin or inserted into muscles, capture muscle action potentials and are amplified and recorded to analyze muscle activity and provide insights into neuromuscular function [11]. ECoG, which involves recording electrical activity directly from the cortical surface, requires the implantation of electrodes on the brain's surface and is commonly used during brain surgeries for real-time monitoring of critical areas [12]. EEG, a non-invasive technique, records brain activity through scalp electrodes and is widely used in clinical and research settings to study brain activity in various conditions. Although ECoG offers more precise spatial and temporal resolution than EEG, EEG remains a valuable tool for analyzing brain wave patterns and detecting abnormal electrical events [13].

Technological advancements, particularly in signal processing, have paved the way for the development of more sophisticated control systems for prosthetics, enhancing user experience and autonomy. The article by Masson et al. [14] introduces a simplified interface for processing and analyzing myoelectric signals from the MYO armband (Thalmic Labs, Kitchener, Canada) for gesture control to control upper limb prosthetics. The MYO wearable device features eight EMG electrodes and a nine-axis inertial measurement unit. It includes open-source software (https://github.com/thalmiclabs) with pattern recognition capabilities such as raw signal display, feature extraction, and K-Nearest Neighbors classification for mapping arbitrary hand movements to prosthetics.

Gaetani et al. [15] describe the design of the electronic module for the "Adam's Hand" (BionIT Labs, Soleto, Italy), a myoelectric transradial ULP that innovatively controls five three-phalange fingers (15 degrees of freedom) using a single motor, an improvement over the conventional five/six motor approach. The prosthesis incorporates temperature and pressure sensors at the fingertips and wirelessly acquires myoelectric signals from the MYO armband. Data are transmitted via an HM-11 BLE module to Adam's Hand's custom-printed circuit board, powered by an Arduino Micro board (Arduino, Ivrea, Italy) for data processing and actuator control. A Raspberry Pi 3 (Raspberry Pi Foundation, Cambridge, UK) manages a touchscreen display for data visualization and sends data to a dedicated cloud platform, facilitating real-time monitoring by orthopedic technicians and enhancing user rehabilitation.

The "SSSA-MyHand" ULP was developed to enable movements analogous to a biological limb and offers a sensory system intended for automatic grip control, providing the user with valuable sensory feedback. Milea et al. [16] propose an innovative approach by incorporating a matrix of sensors and actuators into the myoelectric prosthesis, providing tactile feedback to the user during object manipulation. This tactile feedback is particularly beneficial for individuals with forearm amputations, allowing complex movements and promoting the development of automatic responses to everyday actions and external stimuli.

Additive manufacturing enhances prosthesis production by enabling customization, precision, and accessibility [17]. Through 3D printing, it allows tailored prosthetics to meet individual anatomical and functional needs, improving comfort and functionality [18]. Economically, it reduces material waste and lead times, particularly for customized or small-scale production. The use of clinically approved materials ensures adaptability to specific patient requirements [19].

Beyrouthy et al. [13] present a 3D-printed ULP controlled by brain commands from a portable EEG system. This arm incorporates a network of intelligent sensors and actuators, providing thoughtful feedback on the environment and object interactions. These sensors include temperature, pressure, ultrasonic proximity, accelerometers, potentiometers, strain gauges, and gyroscopes. Fuentes-Gonzalez et al. [20] report a case where a patient successfully controlled the opening and closing movements of a 3D-printed prosthetic arm with nominal strength via EEG signals, enabling the patient to perform grasping actions on everyday objects. De Araújo et al. [21] focus on developing an anthropomorphic ULP designed to meet the needs of transradial amputees. This development relies on additive manufacturing and open-source technology to provide a cost-effective solution for addressing a regional issue.

The Youbionic project combines 3D printing and robotics to create customizable, gesture-controlled prosthetics for individuals with disabilities, supported by resources for user adaptation. The InMoov project advances humanoid robotics and additive manufacturing, offering open-source 3D models for constructing customizable robots with diverse control interfaces and integration with computer vision and ML systems. Community collaboration

has established InMoov as a research platform for robotic prosthetics [22, 23].

Sidher and Shen [24] present a modified 3D-printed artificial hand and control technique to enhance trajectory smoothing. In this context, the InMoov hand was improved by adding medial rotation to the thumb, enabling its opposition. An open-loop control system was implemented in the hand, allowing quick and continuous trajectory adjustments. Additionally, cameras and proximity sensors were incorporated into the InMoov model, improving its ability to identify objects of interest for future research. The effectiveness of open-loop control was evaluated, confirming trajectory smoothing and overshoot reduction. Cherichel and de Curtis [25] introduce EMG control for the 3D-printed InMoov hand prosthesis, enabling individual finger control through EMG sensors placed on the user's forearm. Test results demonstrated the effectiveness of this approach, contributing to making prosthetics more accessible and functional, thereby increasing the autonomy of amputees.

2.2. EEG signal acquisition systems

The electrical biosignal generated by brain activity can be captured using EEG. EEG is a widely employed non-invasive technique for recording brain electrical activity. The EEG signal originates from neurons in the cerebral cortex, which produce electrical oscillations that can be measured by electrodes placed on the patient's scalp, thereby enabling the monitoring and analysis of brain function. The pattern of EEG signal oscillations varies depending on the state of wakefulness or sleep, brain activity, and other conditions. This technique has diverse applications in clinical practice and scientific research [26].

The electrical signals captured by EEG electrodes are initially considered raw data, existing in the time domain and reflecting amplitude variations in electrical voltage. However, these raw signals require processing—such as filtering and amplification—to extract meaningful information, generating standardized rhythmic oscillations called brain waves [27]. Transforming raw data into brain wave patterns is essential for understanding brain activity and its correlation with an individual's mental state. The frequency bands of brain waves are categorized as Delta (δ), Theta (θ), Alpha (α), Beta (β), and Gamma (γ), each associated with distinct mental processes [26].

Brain wave activity spans five frequency bands, each with distinct functions and regional detection patterns: Delta waves (0.5–4 Hz) support deep sleep and tissue repair across various brain regions. Theta waves (4–8 Hz), linked to light sleep, relaxation, and creativity, are primarily detected in parietal and temporal areas. Alpha waves (8–13 Hz), associated with relaxation and sensory processing, are dominant in occipital and posterior lobes. Beta waves (13–32 Hz), tied to alertness and concentration, are prominent in frontal and central regions. Gamma waves (≥32 Hz), crucial for cognitive integration and consciousness, are concentrated in front-central areas.

The acquisition of brain biosignals can be achieved through two primary approaches [6]: EEG, a non-invasive method involving the placement of electrodes on the scalp, or ECoG, an invasive technique with electrodes positioned subcortically. In the 1950s, John C. Lily recorded monkey brain signals through implanted electrodes. Later developments by David Nowlis and Joe Kamiya explored biofeedback techniques, enabling patients to control brain signals through auditory feedback. The 1960s marked significant advancements, particularly in the Neural Control Laboratory's work on artificial actuators for individuals with disabilities. A landmark study involved Brindley and Graggs, who recorded motor cortical fields from baboons to develop neuroprosthetic systems. However, progress in this field slowed until the mid-1990s, when introducing flexible multi-electrode arrays reignited research efforts [28].

EEG signals are integral to device control technologies, including advanced robotics and prosthetics via brain-machine interfaces (BMIs). The Fourier transform facilitates the conversion of EEG signals from the time to the frequency domain, enabling spectral analysis [29]. Its efficient variant, the fast Fourier transform (FFT), decomposes EEG signals into frequency components, supporting precise analysis and real-time applications [30]. BMIs function through four stages: (1) signal acquisition—amplification and analog-to-digital conversion; (2) data extraction—noise and artifact removal via filtering; (3) interpretation—pattern identification using ML algorithms; and (4) command execution—translating signals into control commands for devices [31–33].

The BMI technologies continue to evolve; modern prosthetic systems increasingly demand higher computational power, multi-degree-of-freedom control, and integrated feedback mechanisms. However, balancing these capabilities while maintaining energy efficiency and real-time performance remains a significant challenge. In response, developers are investigating hybrid models combining classification and regression techniques to enhance control precision and system functionality [34]. One illustrative example is the embedded BMI system developed by Bueno et al. [35], which integrates a microcontroller and a digital signal processor to handle data processing, storage, and communication. Their work aligns with the goals of the OpenBCI initiative (https://openbci.com/), which promotes open-source neuro-technologies to democratize access to BMI tools and foster collaborative research efforts within the field.

In parallel, developing portable EEG devices has enabled the non-invasive monitoring of brain activity in real time, supporting research in naturalistic settings. These devices offer promising opportunities for practical applications, such as controlling external equipment, while expanding the range of real-world use cases for BMI technologies [36].

The Neuroheadset Emotiv Insight (Emotiv, San Francisco, USA) captures brain electrical signals and converts them into information regarding emotions, attention levels, and other physiological conditions. The device features an advanced and optimized electronic system that produces clean and robust signals from seven semi-dry, hydrophilic polymer sensors, two of which are used for reference (CMS/DRL). In contrast, the remaining sensors acquire EEG signals operating in the frequency range of 0.16 Hz to 43 Hz, generating 128 samples per second. The electrodes are positioned on the scalp based on the international 10–20 system [37], covering the frontal, prefrontal, temporal, parietal, and occipital lobes. The device establishes a wireless connection with a computer via a USB receiver, commonly known as a USB dongle, utilizing the Bluetooth Low Energy (BLE) standard. Additionally, the company provides the EmotivPRO software to users with at least a basic license, granting access to various EEG-related tools and resources, including mental commands, raw and processed EEG graphs, facial expressions, and movements.

However, the full potential of such systems depends on access to brain data for further research and development. The Emotiv Software Development Kit allows access to brain data collected by Emotiv devices, yet some advanced functions require a paid license, which imposes limitations. Addressing this challenge, the

CyKIT project (https://github.com/CymatiCorp/CyKit) was developed in Python, enabling the acquisition and storage of raw EEG data from Emotiv Insight. This open-source initiative empowers developers to tailor brain data to their research needs by modifying the source code and enhancing customization and accessibility.

3. Proposed Methodology

Figure 1 illustrates a flowchart detailing the process for developing and validating control mechanisms of a robotic ULP using ML techniques to acquire EEG signals. Designed to clarify the methods employed in this proposal, the flowchart uses distinct symbols and colors for each stage: purple indicates data input, blue represents processing, yellow denotes decision conditions, and red signifies data storage. Arrows depict the progression between steps. The following sections will examine each component of the flowchart in detail.

The data used to formulate and validate the proposed command classification for controlling a robotic ULP based on EEG signals were obtained from an EEG data acquisition system developed in this study called EmotIF. The processes involved will be detailed in the subsequent section.

After the acquisition process is completed, the generated dataset is used in a training process for EEG pattern recognition using ML techniques termed PaRecoML, which will be described in detail in the respective section.

3.1. EmotIF: EEG data acquisition system

For this experiment, the Emotiv Insight Neuroheadset was carefully assembled regarding electrode positions and moistened with glycerin saline solution to obtain high-accuracy EEG data, as detailed in steps A, B, and C in Figure 2.

The EEG device was securely attached to the participant's head, and electrodes were placed on the scalp according to the manufacturer's instructions to ensure optimal electrical contact (Figure 2(c)). The USB dongle was then connected to the computer, and the EmotivPRO interface was initialized. The free license of EmotivPRO provides minimal access to basic functionalities such as viewing raw EEG data, frequency band analysis, facial expression detection, mental command recognition, and signal saving (a maximum of five records at a time).

The CyKIT library was employed to circumvent these limitations. This open-source tool enables the acquisition and transmission of EEG data from headsets, primarily using the accompanying USB dongle. The platform operates within a Python environment, requiring the installation of a version earlier than 3.9 and utilizing the PyCharm development environment.

CyKIT begins by importing essential libraries and modules —"queue", "signal", "time", and "logging"—which are crucial for data manipulation and interaction with the operating system. These modules facilitate the creation of temporary storage structures, control of program execution flow, event logging, and signal processing, ensuring efficient system interactions.

Two additional modules, "cyPyWinUSB" and "cyCrypto", play a fundamental role in CyKIT's functionality. The "cyPyWinUSB" module enables detection and interaction with connected human interface devices, such as the EPOC+ USB dongle, while "cyCrypto" is responsible for managing encryption operations. Specifically, "cyCrypto" implements the Advanced Encryption Standard (AES) algorithm to secure data transmission by encoding 128-bit messages, ensuring data integrity during transmission and facilitating secure decryption upon acquisition.

A core component of the CyKIT architecture is the 'signal_handler' function, which ensures controlled termination of the data acquisition process. It also defines several global variables, ensuring consistency throughout the codebase. A dedicated class has also been developed to structure methods related to device identification, encryption management, decryption of acquired data, and conversion of data into microvolts (μV). The class constructor, defined in the '__init__'

Figure 1
Flowchart of EEG signal acquisition and ML-controlled robotic ULP
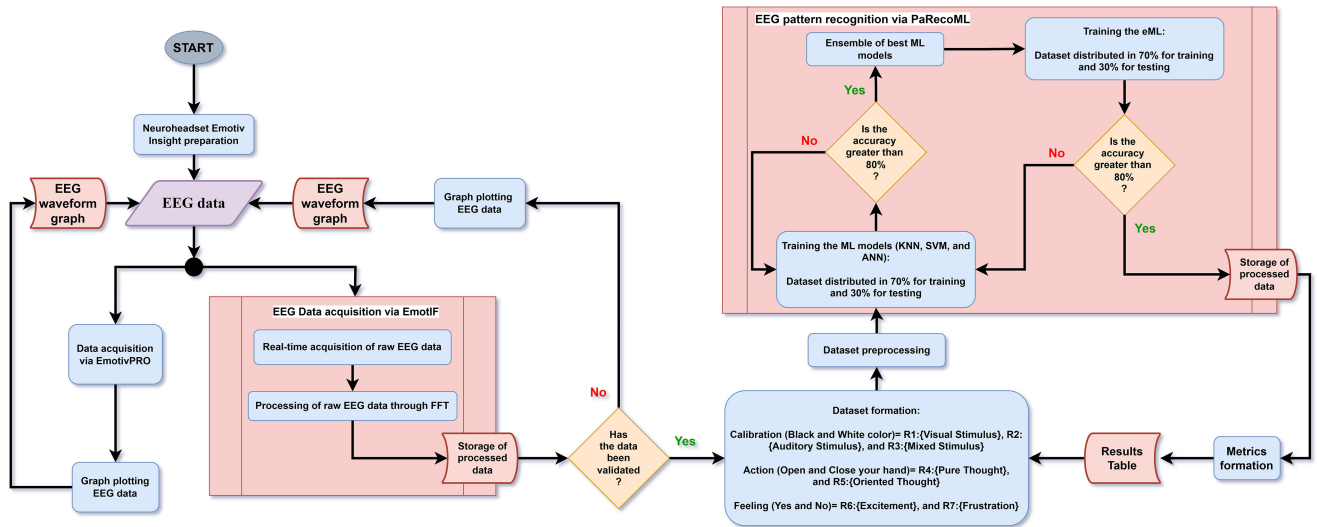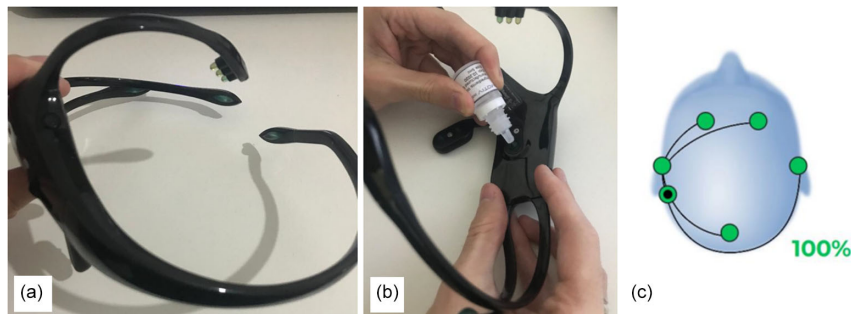
Figure 2
Preparation of the neuroheadset Emotiv Insight: (a) sensor attachment, (b) moistening with a saline glycerin solution, and (c) successful connection between the computer and EmotivPRO

method, is responsible for locating the EPOC+ device and encrypting its serial number to establish a Bluetooth connection via the USB dongle.

The data acquisition process starts by receiving encrypted Bluetooth packets containing a header with a counter field that indicates the order of data segments. During processing, irrelevant bytes from the packet header and footer are discarded, leaving only the relevant EEG data for decryption. The 'get_data' method is critical to decrypting the data using AES encryption and converting the results into µV via the 'convertEPOC_PLUS' method. To maintain data integrity and enable future analysis, the program generates a log file (.log) in real time using the "logging" module, storing all acquired data.

A module was also developed to simulate microvolt-level data and perform FFT operations as part of the project. This simulation utilizes the minimum and maximum µV intensity values provided by the CyKIT developers. The "numpy" module processes the simulated data for FFT computations. However, discrepancies were identified between the simulated data and the expected output, with the resulting data failing to reflect the anticipated intensities across three distinct brainwave patterns and the frequencies not aligning with known neurological rhythms. To resolve these issues, users are advised to use the latest version of Python and the appropriate development environment. Furthermore, properly including the "cyCrypto" and "cyPyWinUSB" modules in the project directory ensures access to the computer's USB ports and accurate decryption of the headset's transmitted data.

The main code is implemented in the 'cyKIT_INSIGHT.py' script, which begins by importing the necessary modules. The "EEG_insight" class encapsulates the methods for coordinating data acquisition. The class constructor ('__init__') initializes critical variables and identifies the USB dongle connected to the system. The 'interface__eeg' method provides a user-friendly interface that supports real-time EEG graph visualization and reading EEG data from .csv files.

The 'data_handler,' 'convert_v2', and 'get_data' methods work in unison to ensure efficient data processing, starting with the initial handling of encrypted data and culminating in its conversion into voltage units. The 'data_list' method organizes the acquired data into categorized lists, streamlining subsequent analysis. A headset object is created to integrate the functionalities of the "EEG_insight" class. At the same time, the graphical interface is managed by the 'ifkit_GRAPH.py' module, which leverages the "PyQTGraph" library for real-time data visualization.

Functions within the graphical module, such as 'graph_timer' and 'update_graph,' ensure continuous updates every five milliseconds, enabling interactive and dynamic data visualization. The 'gen_file()' and 'read_file()' functions manage the storage and retrieval of data in.csv format, allowing users to store and analyze the acquired EEG data as needed. The "CyKIT" framework represents a robust and well-structured system that seamlessly integrates data acquisition, encryption, visualization, and storage, significantly contributing to EEG data research and analysis.

For this project, the CyKIT software was customized and extended into a new program called "EmotIF" to achieve the following objectives: (i) real-time acquisition of raw EEG data; (ii) real-time graph plotting based on data from five sensors; (iii) storage of acquired data in.csv format; (iv) reading of raw EEG data from.csv files; (v) graph plotting of raw EEG data extracted from.csv files; (vi) dynamic graph plotting, allowing users to define the maximum acquisition time; (vii) processing of EEG data using FFT; and (viii) storage of processed data in.csv format.

The EmotIF enhances the functionalities of CyKIT, facilitating more flexible and efficient management of EEG data for real-time monitoring and subsequent experimentation. Upon completing this phase, we proceeded to the next stage, during which the operation of the EEG device in the experiment revealed that eye movements induced abrupt changes in the behavior of the graph generated by EmotivPRO, as illustrated in Figure 3.

Consequently, an experiment was conducted to identify the oscillation peaks in the graph corresponding to each blinking action. In this investigation, three simultaneous data acquisitions were performed using EmotivPRO and EmotIF to address this phenomenon. The first acquisition focused on recording the oscillations triggered by the left-eye blinks, the second documented the oscillations caused by right-eye blinks, and the third captured the effects of simultaneous blinking with both eyes. Each blink was performed with a one-second interval between successive actions, and the duration of each data acquisition was set to 15 s.

After this experiment, a deeper understanding was gained regarding how eye movements influence the graphs generated during the operation of the EEG device. Additionally, eye-blinking events were employed as sample markers to synchronize events across both data acquisition programs. The strategy of utilizing eye blink events as markers and synchronizers is employed due to the inadequacy of time-based synchronization. While the EmotIF

Figure 3

The interface of the proprietary software EmotivPRO highlights (in red) the abrupt change in graph behavior during eye movement
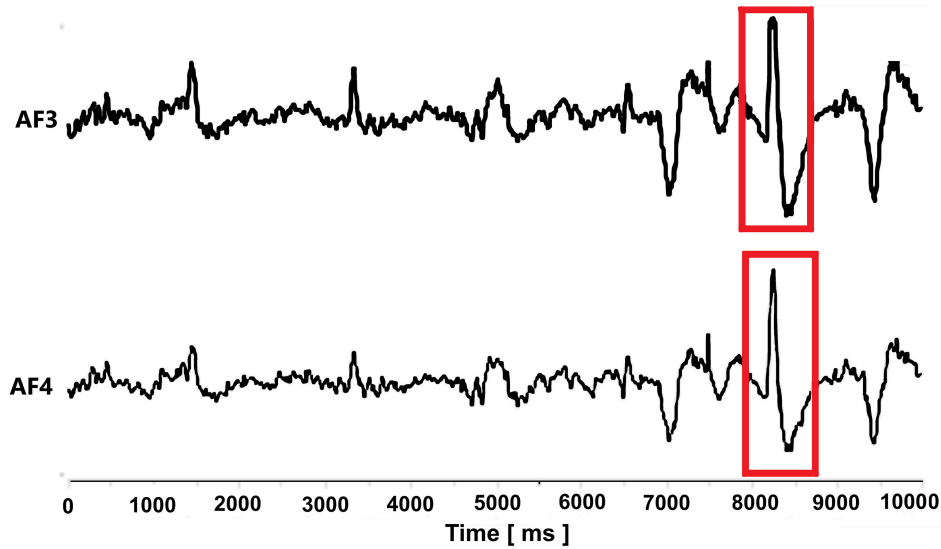
approach uses the number of samples on the x-axis, the EmotivPRO approach relies on time in seconds. Achieving temporal synchronization between the two on the x-axis is challenging because the processing time of the Python-based program differs from that of the proprietary software. This discrepancy, as the Python program is still undergoing optimization in terms of operational routines, could lead to interpretative anomalies.

The primary experimental test was conducted to collect EEG data simultaneously using EmotivPRO (the manufacturer's software, Figure 4(a)) and EmotIF (the developed software, Figure 4(b)) to validate the data acquired through the proposed code. The EEG signals were recorded during the one-minute trial, with the participant intentionally blinking every five seconds. The voltage variations, measured in µV, were plotted on the Y-axis of the resulting graphs. To ensure synchronization, data collection in both environments was performed in real time. Both software environments were closed after the acquisition period, and the collected data were subjected to a validation process that involved a morphological comparison between the graphs generated by the EmotIF program for each channel with the respective ones produced by EmotivPRO, aiming to check the consistent accuracy of the recorded data.

The EEG data obtained through the methodology described in this topic will be used to detect EEG patterns via the ML models described below.

3.2. PaRecoML: EEG pattern recognition via machine learning

After establishing the EEG data acquisition system captured through the methodology described above, the next phase focuses on leveraging this data to implement ML techniques for effective EEG pattern recognition to control a robotic ULP.

The method for control mechanisms of a robotic ULP utilizing ML techniques in the acquisition of EEG signals involves the following five steps: (1) creation of dataset; (2) preprocessing of the dataset; (3) selection of ML models; (4) training of ML models; and (5) evaluation of the performance of ML models.

The dataset generation process (Stage 1) was based on experimental trials designed to build datasets from selected categories using the EmotIF program (as detailed in the previous section). A continuous 5-minute session was conducted for each dataset category to acquire the required data. A detailed description of each category is provided below:

1) Calibration: This category aims to establish consistent parameters for color identification, facilitating inference verification and improving perceptibility. It includes three distinct datasets for each color (black and white), consisting of six data sets in total, using the following approaches:

   Visual Stimulus (Round 1): The participant observes the corresponding color to generate brainwaves in resonance with the visual perception of that color.

   Auditory Stimulus (Round 2): The participant listens to a verbal cue representing the corresponding color to form brainwaves aligned with the color's auditory memory.

   Mixed Stimulus (Round 3): A combination of visual and auditory stimuli to generate brainwaves synchrony with the visual perception and the aural memory of the color.

   These stimulus variations simulate different interaction scenarios where color recognition must occur through multiple sensory channels (vision and hearing). The calibration type focuses on establishing reference parameters to ensure the data acquisition system continues to function correctly over time.

2) Action: The Action category enables prosthesis control based on the participant's intent to open or close their hand. This category consists of four datasets, divided into two distinct routines:

   Pure Thought (Round 4): This approach relies exclusively on the participant's mental intention, without external stimuli or interference. The action is driven solely by the participant's will.

   Oriented Thought (Round 5): This approach involves using vocal commands to reinforce the participant's mental intention, improving the accuracy of prosthesis control.
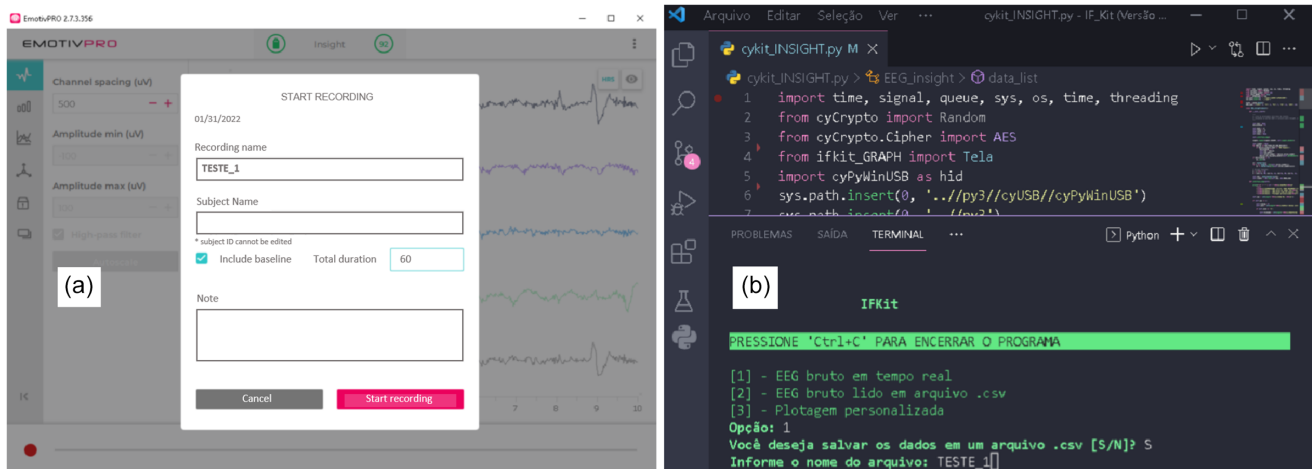
The distinction between pure and oriented thought is essential for exploring the differences between direct mental control and stimulus-reinforced control. Oriented thought offers an additional layer of reliability, helping the system correctly interpret the participant's intention, especially in cases where direct neural signal interpretation may be ambiguous. The action type prioritizes the recognition of the participant's intention, which is central to the proposed system, ensuring effective control of the robotic ULP.

3) Feeling: The Feeling category focuses on identifying the participant's emotional state during interaction with the prosthesis, particularly regarding excitement and frustration. This category consists of four datasets, divided into the following emotional states:

Excitement (Round 6): Assess the participant's enthusiasm or satisfaction when using the prosthesis.

Frustration (Round 7): Identifies potential signs of disappointment or dissatisfaction, indicating difficulties using the device.

These emotional subdivisions allow the system to function as an emotional feedback mechanism, ensuring the correct interpretation and execution of the participant's intention. This is particularly relevant since user frustration may indicate potential performance issues within the system. The feeling type enables internal assessment, verifying whether the intended control actions were effectively executed.

In the preprocessing phase (Stage 2), the fourteen datasets were reorganized into seven datasets across three specific categories, each designed to structure distinct aspects of system control and feedback (Calibration, Action, and Feeling).

The next stage was executed based on this dataset composition. The ML model selection (Stage 3) was based on representing the main niches of mathematical model architectures: K-nearest neighbors (KNN), support vector machines (SVM), and artificial neural networks (ANN).

The KNN is an instance-based classification algorithm that leverages the proximity between data points to make predictions. As a non-parametric method, it does not assume explicit data distribution, making it a versatile solution applicable to various scenarios. The core premise of KNN is that "objects located close to each other in feature space are likely to belong to the same class." During the training phase, KNN only stores the dataset without performing any active learning process. To classify a new data point, the algorithm computes the distance between this point and all the points in the training dataset using metrics such as Euclidean or Manhattan distance. It then selects the KNNs, which are the closest points to the new input. The most frequent class among these neighbors is assigned to the latest point. This simplicity makes KNN intuitive and easy to implement. However, its performance can degrade with large datasets, as each classification requires comparing the new point with all training points [38, 39].

The SVM is a supervised classification method designed to find an optimal hyperplane that separates samples from different classes while maximizing the margin between them. The margin is the distance between the hyperplane and the nearest data points from each class, known as support vectors. These vectors are pivotal in determining the decision boundary and the hyperplane's position. Maximizing the margin enhances the robustness of the model by reducing the likelihood of classification errors. When the data is linearly separable, SVM identifies a line or hyperplane that separates the classes. However, many real-world problems, such as EEG analysis, involve data that cannot be easily separated in a linear space. To address this, SVM employs kernel functions that map the data into a higher-dimensional space, where a more precise separation between classes becomes feasible. Standard kernels include the linear, RBF (Radial Basis Function), and polynomial kernels [39–41].

ANNs are inspired by the structure and functionality of the human brain, consisting of artificial neurons organized into multiple layers. Each layer comprises a set of interconnected neurons linked by weighted connections that determine the influence each neuron exerts on the others. During training, the network adjusts these weights iteratively to minimize the error between the predicted output and the expected result. A typical ANN architecture includes an input layer that receives the problem variables, hidden layers that process the data through activation functions such as ReLU (rectified linear unit) or Sigmoid, and an output layer that generates the final prediction.

The training process relies on the backpropagation algorithm, which computes the prediction error and propagates it backward through the network layers, progressively adjusting the weights to reduce the error [38, 42].

These ML models were selected as representatives of ML techniques with similar characteristics, including methods such as SVM and linear regression. To implement these algorithms, we employed the widely recognized scikit-learn library[1], which is well-regarded within the ML and deep learning communities. Python was chosen as the development language to streamline the entire acquisition and inference process, ensuring seamless integration and simplifying data analysis and compilation in embedded systems.

Grid Search was utilized to identify optimal hyperparameters for the ML models by systematically evaluating all possible combinations within a predefined search space. Hyperparameters, which control algorithm behavior without being directly learned from data, are crucial for model performance. To prevent overfitting and improve generalization, cross-validation was employed within Grid Search.

Cross-validation ensures a model's ability to generalize to unseen data. The dataset was divided into folds, with the model trained and tested iteratively across partitions. In k-fold cross-validation, data is split into k equal parts, each serving as a test set once while the rest form the training set. Performance metrics are averaged across k iterations for reliable estimates.

The datasets used for training and testing were split into 70% training and 30% testing, a widely adopted strategy that reduces the likelihood of overfitting. Each baseline model underwent twenty epochs of training (approximately one hour) using the same training dataset, comprising 76,801 inputs. In each epoch and for each model, test accuracy values were recorded and stored in a vector, allowing direct comparative evaluation of the models' performances through histogram plots.

The initial training process employed accuracy as the primary criterion to assess model progression, with a minimum acceptable threshold set at 80%. Models meeting or exceeding this accuracy during testing were preserved; otherwise, they were reverted to the training phase for further refinement. This iterative evaluation cycle ensured continuous optimization before the final assessment of performance metrics.

The final training encompassed the most optimized versions of the previously discussed ML models—namely KNN, SVM, and ANN—selected based on their superior performance during initial training and integrated into an ensemble framework. This diverse combination of models provided a consistent framework for pattern detection in EEG within a BMI. The ensemble of ML models (eML) was designed to combine multiple ML models to achieve enhanced performance. It involved model selection, parallel processing, and consensus building. By capitalizing on the unique strengths of each model, the eML optimized decision-making, leading to more precise and reliable pattern recognition [43].

The eML submodule adheres to the previously established procedure (Stages 1–4), with an additional phase dedicated to prediction integration. In this phase, the majority voting technique is employed to aggregate the predictions from the ML models. This method assigns the class with the highest number of votes from the individual models as the final prediction of the eML. Although simple, majority voting is an effective strategy for reaching a consensus among the base models. Additionally,

combining these three ML models reduces ambiguities caused by potential inference conflicts between individual models, making decision-making more streamlined and reliable.

Validation metrics (Stage 5) are computed using specific formulas. Accuracy measures the overall correctness of the model by evaluating the proportion of correctly predicted instances, including both true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), about the total number of predictions: $(TP + TN)/(TP + TN + FP + FN)$. Recall quantifies the model's ability to accurately identify all actual positive instances by determining the proportion of TP among all the positive cases, including FN. Precision assesses the model's ability to accurately identify positive instances by calculating the proportion of TP among all cases predicted as positive, including FP: $TP/(TP + FP)$. The $F1$ score is a harmonic mean of precision and recall, providing a single metric representing the model's accuracy in predicting positive instances while considering both FP and FN: $TP/(TP + 0.5*(FP + FN))$.

With the methodology established, the methods described in this section were applied to the collected dataset to assess the performance of the EEG signal acquisition system and the ML-based pattern recognition models.

4. Experimental Results

The results follow the same structure outlined in Section 3, maintaining consistency and facilitating the reader's understanding. The first subsection evaluates the EmotIF, while the subsequent subsection analyzes the performance of the PaRecoML.

4.1. EmotIF: EEG data acquisition system

Figure 5 presents a time series of signals across multiple EEG channels from electrodes on the scalp. The channels are labeled following the standard EEG nomenclature (AF3, AF4, T7, T8, and Pz). For each channel, two plots are displayed, identified by labels A (EmotIF wave) and B (EmotivPRO wave) on the left side.

In channel AF3, the curve labeled A (red) exhibits a pronounced oscillation at the beginning (between 0 and 200 ms), followed by stabilization over time. Conversely, the curve labeled B (gray) shows more minor fluctuations and lower variability. In channel T7, curve A (blue) also demonstrates an initial significant change, with a trend toward stabilization afterward, whereas curve B (purple) is smoother and exhibits minimal variability. Similarly, in channel Pz, curve A (green) follows a pattern comparable to the previous channels, showing a noticeable oscillation in the early stages and gradually stabilizing. Curve B (cyan) displays lower amplitude and minimal perceptible activity. In channel T8, a similar trend is observed: curve A (purple) oscillates more prominently at the start before stabilizing, while curve B (light green) remains less variable. Finally, in channel AF4, curve A (orange) exhibits an initial peak followed by stabilization, whereas curve B (gray) maintains a more consistent level.

Across all channels, initial peaks in the signals suggest a response to an early stimulus. At the same time, the subsequent stabilization indicates that, after the initial event, the signals enter a more stable phase. Therefore, it can be inferred that the waves generated by both EmotIF and EmotivPRO platforms can be effectively synchronized through the eye-blinking event.

An experimental test was subsequently performed to collect EEG data concurrently using both the EmotivPRO software and the custom-developed EmotIF program, corresponding to each channel of the EEG device. The data acquisition adhered to the procedures described in the

---

[1] https://scikit-learn.org/stable/

Figure 5
EEG signal waveform graph for the "eye blink" event in Voltage (μV) vs. time (ms) data acquisition for all channels (AF3, AF4, T7, T8, and Pz) via (a) EmotIF-developed program and (b) EmotivPRO proprietary program
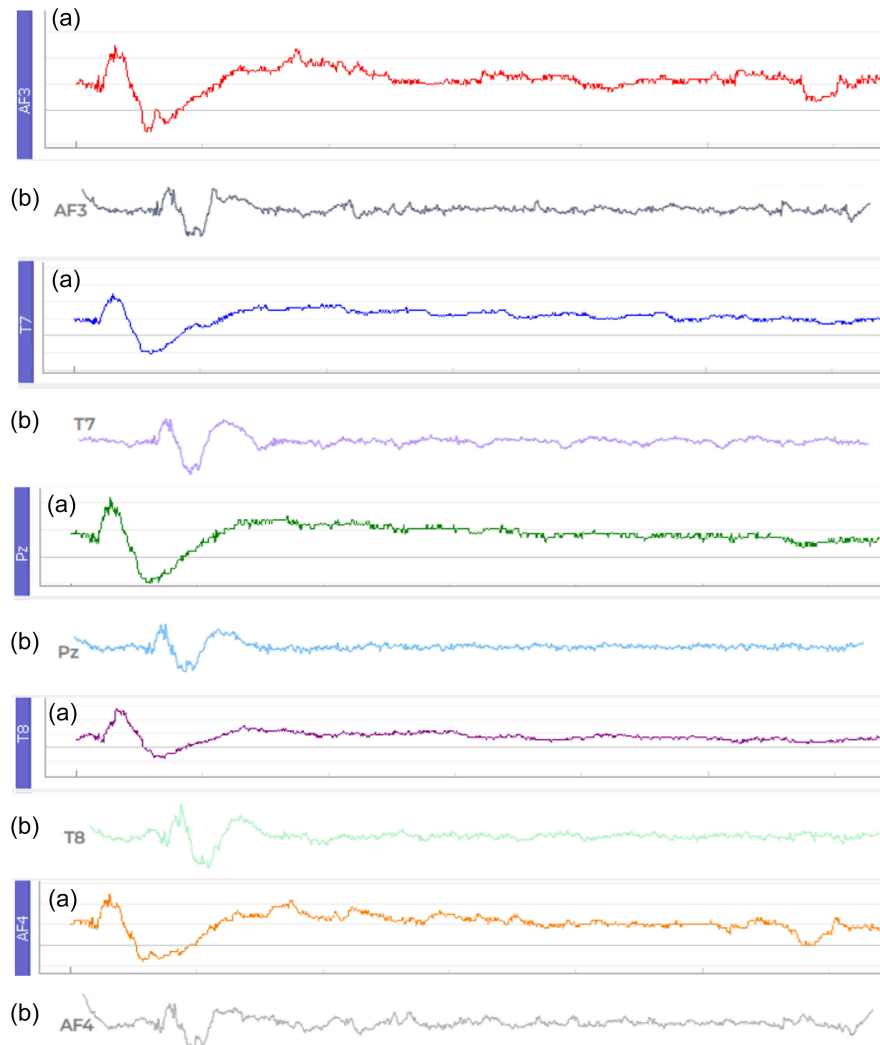
methodology section, in which the eye blink event was used to sample the marker and synchronize the events. A comparative visual analysis of the EEG graphs generated by EmotIF (A label) and EmotivPRO (B label) demonstrated consistent and comparable signal patterns across all channels: AF3 (Figure 6), T7 (Figure 7), Pz (Figure 8), T8 (Figure 9), and AF4 (Figure 10).

Figure 6 shows graph A (red) exhibiting pronounced fluctuations with sharp peaks and troughs around 2500 ms and 7500 ms, and voltage varying between 4000 μV and 4600 μV, indicating substantial signal variability. In contrast, graph B (gray) follows a similar pattern but with smoother transitions and reduced noise levels. Although distinct peaks and drops are still observable, particularly around 2500 and 7800 ms, graph B's signal is more continuous and clearly defined.

In Figure 7, graph A (orange) displays frequent voltage fluctuations with irregular peaks alternating between sharp increases and decreases around 3000 ms, with voltage ranging from 4100 μV to 4300 μV, indicating moderate signal variability. Graph B (purple) maintains a similar trend but with less abrupt transitions and more evenly distributed noise, suggesting a smoother, preprocessed signal with improved trend visibility.

Figure 8 presents graph A (gray) with irregular fluctuations, sharp peaks around 2900 ms, and sudden voltage drops between 4100 μV and 4250 μV, indicating a noisier signal. Conversely, graph B (blue) exhibits fewer abrupt changes, with more gradual transitions between peaks around 2000 ms and troughs within the same voltage range, suggesting noise reduction and signal refinement.

Figure 9 illustrates graph A (yellow) with sharp peaks and drops interspersed with brief stable periods, prominent around 2000 ms, with voltage variations between 4100 μV and 4250 μV. Graph B (light green) displays similar voltage ranges, with distinct peaks and drops near 2000 ms but smoother transitions and fewer abrupt changes, suggesting that the signal has undergone processing to reduce noise and enhance pattern clarity.

Finally, in Figure 10, graph A (blue) displays sharp voltage fluctuations, with peaks around 2500 ms and 7500 ms. The voltage fluctuates between 4000 μV and 4500 μV, reflecting high variability in the signal, followed by sharp drops. Graph B (gray) covers the same voltage range but presents smoother transitions, with peaks around 2000 ms and 7400 ms. This indicates that noise reduction and preprocessing have been applied, enhancing the clarity of the signal and making patterns more discernible.

Figure 6

EEG signal waveform for the acquisition of Voltage (*μ*V) x Time (ms) data for the AF3 channel via (a) developed software EmotIF, (b) proprietary software EmotivPRO
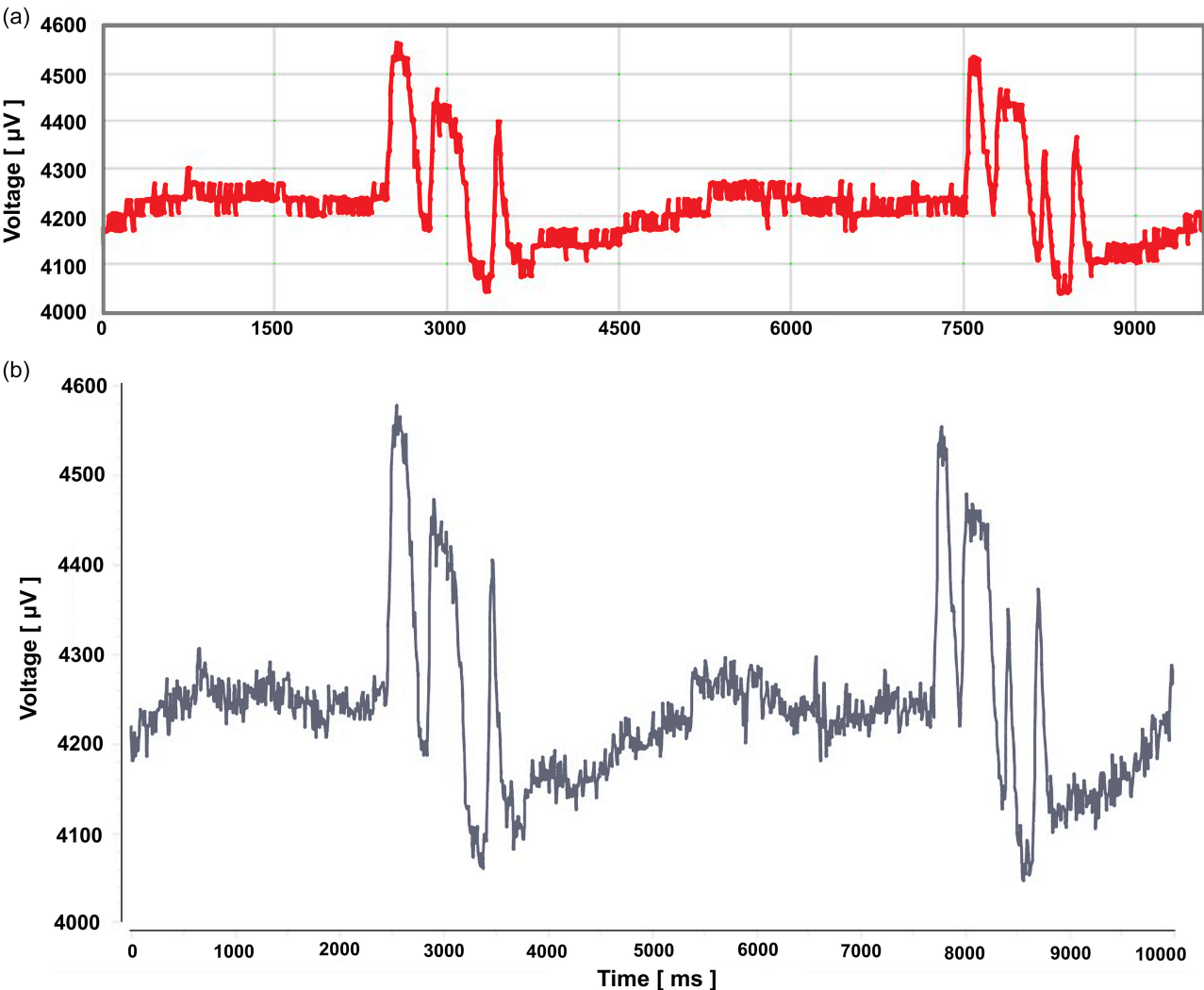
The comparison between Figures 6–10 consistently illustrates that the EmotIF graphs (labeled A) represent unfiltered data characterized by sharper fluctuations and higher noise levels. In contrast, the EmotivPRO graphs (labeled B) offer a more refined and smoothed representation of the same data. This distinction is likely attributable to preprocessing steps in EmotivPRO that reduce noise and emphasize broader trends. However, it is evident that both graphs, A and B, capture the same underlying waveform, as indicated by their similar amplitude ranges and pattern structures.

These findings confirm the efficacy of the EmotIF software for EEG signal acquisition, validating its capability to generate reliable datasets. This supports its application in training ML models for EEG pattern recognition.

4.2. PaRecoML: EEG pattern recognition via machine learning

EmotIF collected brain activity data, systematically organized and stored in ".csv" files. Tables from the experimental rounds (R1–R7) were concatenated to create a training dataset for ML models. For example, as illustrated in Figure 11, the tables from round 1—specifically "1a_black_color_VISUALSTIMULUS" and "1b_white_color_VISUALSTIMULUS"—were integrated into a single dataset labeled "1_Dataset_CALIBRATION_VISUALSTIMULUS". Additionally, a "TARGET" column was introduced, assigning a value of "0" to all entries from the first table and "1" to those from the second table.

Table 1 summarizes the signal characteristics in metrics related to the activity of different EEG channels, expressed in μV, with the highest and lowest values of each round highlighted in red and blue, respectively. The characterization of recorded brain activity involved an analysis of metrics within the inference datasets. These metrics include each channel's mean, standard deviation, and range of values, playing distinct roles in assessing brain electrical activity. The mean indicates central tendency, providing a representative value for the channel under study. The standard deviation quantifies the variability of brain electrical activity, highlighting the degree of data dispersion. Finally, the range of values, determined by the minimum and maximum values observed in each channel, reflects the amplitude of the recorded activity.

The dataset metrics in Table 1 indicate a comparative stability among the voltage measurement channels (AF3, T7, Pz, T8, AF4),

Figure 7
EEG signal waveform for the acquisition of Voltage (μV) x Time (ms) data for the T7 channel via (a) developed software EmotIF, (b) proprietary software EmotivPRO
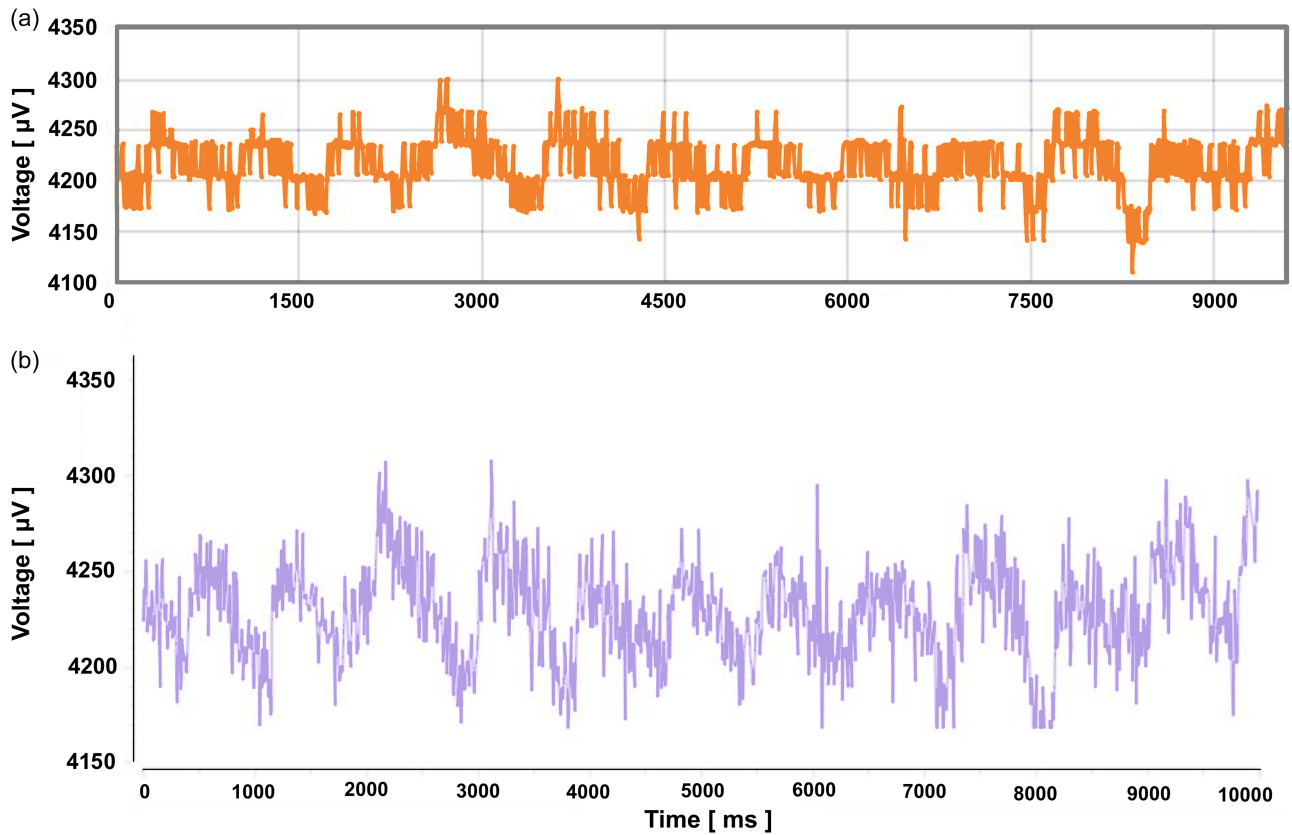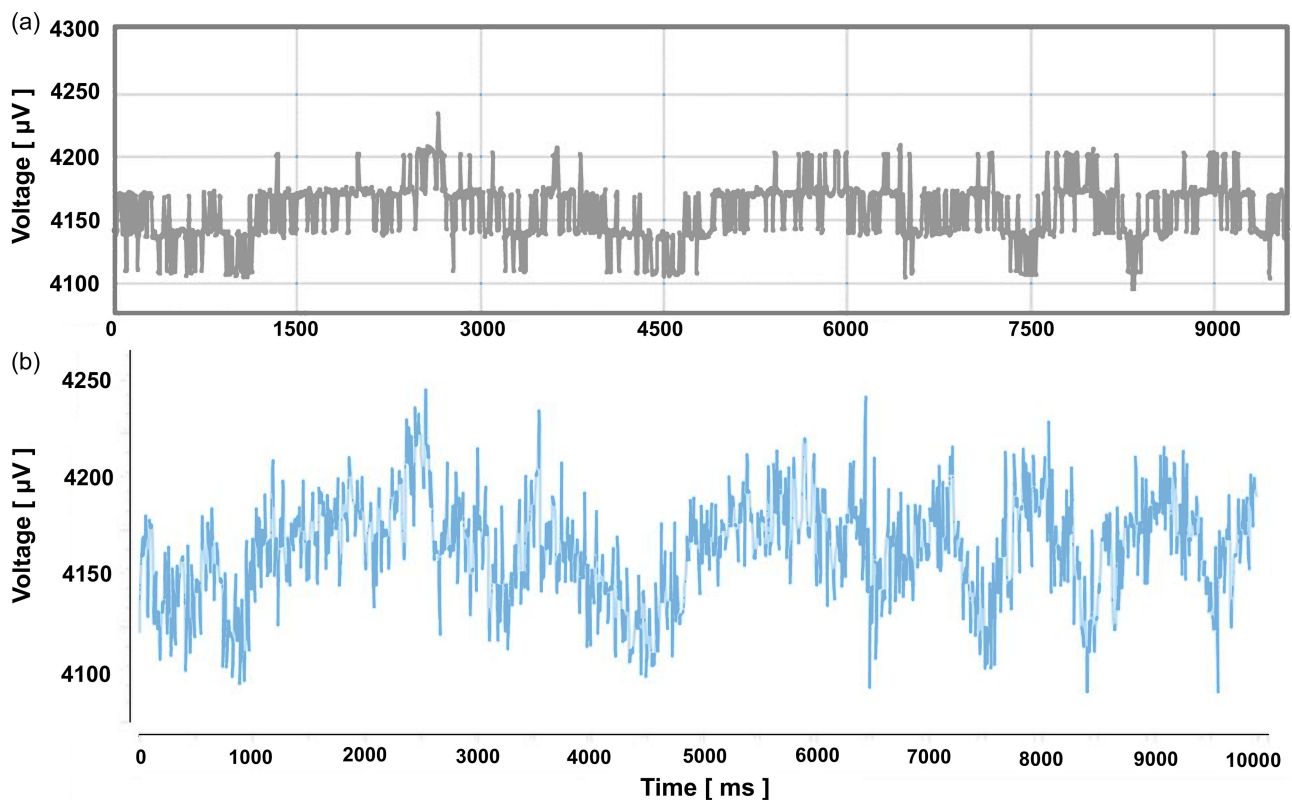

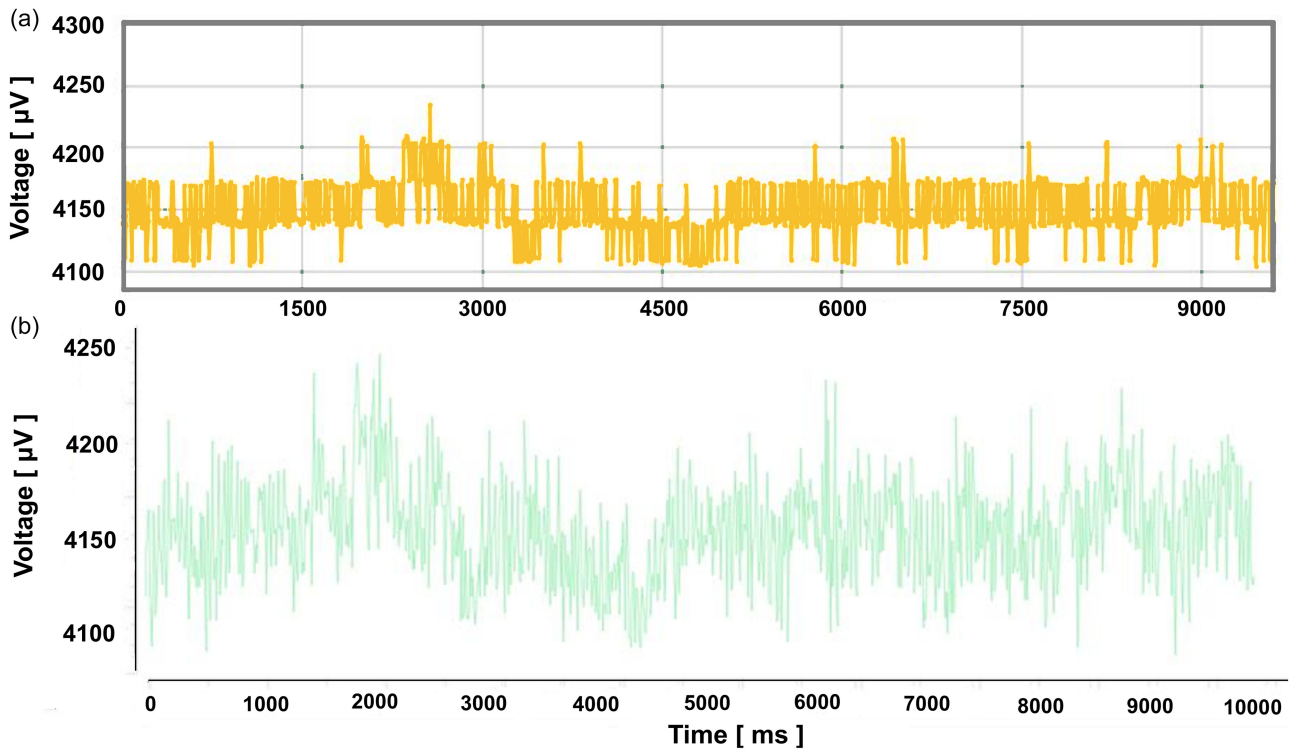**Figure 8**
EEG signal waveform for the acquisition of Voltage (μV) $\times$ Time (ms) data for the Pz channel via (a) developed software EmotIF, (b) proprietary software EmotivPRO

Figure 9
EEG signal waveform for the acquisition of Voltage (*μ*V) × Time (ms) data for the T8 channel via (a) developed program EmotIF, (b) proprietary software EmotivPRO

with AF3 exhibiting the highest average values across most tests. The standard deviations reveal a moderate variability, suggesting that the measurements are reasonably stable, albeit with some fluctuations. The measurement ranges vary considerably, with AF3 reaching a maximum of 7423 μV in R6, which may reflect variations in cerebral activity or measurement sensitivity. A slight downward trend was observed in the average values of T7, Pz, T8, and AF4 from R1 to R4, followed by fluctuations in R5 to R7, indicating changes in physiological processes or external influences. The consistency in the standard deviations across the channels suggests that all may be affected by similar levels of noise or artifacts during data collection.

Following the exploratory analysis phase of the dataset, attention was directed towards the subsequent stage, which involved the optimization of hyperparameters. This optimization aimed to identify the most effective combination for refining the ML models.

The KNN algorithm was configured using the following parameters: n_neighbors=5, weights='uniform', algorithm='auto,' leaf_size=30, metric='minkowski', and p=2 (corresponding to Euclidean distance). The choice of n_neighbors=5 reflects a commonly accepted value in the literature, offering a practical balance between bias and variance by avoiding underfitting and overfitting. A uniform weighting scheme (weights='uniform') was adopted to prevent overemphasis on closer neighbors, simplifying the decision boundary and minimizing the risk of bias towards local points. The algorithm='auto' configuration allowed scikit-learn to select the optimal algorithm based on the dataset size and structure, ensuring flexibility and efficiency. The default leaf_size=30 was also retained, as it has proven effective across various scenarios, ensuring adequate computational performance. The distance metric was defined as Minkowski with p=2,

corresponding to the Euclidean distance, which is particularly appropriate for continuous data in multi-dimensional spaces. This configuration aligns with established best practices in supervised learning, ensuring robust performance.

The SVM was parametrized with the following settings: C=1.0, kernel='rbf', degree=3, gamma='scale', coef0=0.0, shrinking= True, and probability=False. The regularization parameter C=1.0 was chosen as a reasonable trade-off between bias and variance, helping to prevent both overfitting and underfitting. The rbf kernel was selected for its ability to handle non-linear decision boundaries efficiently, making it particularly suitable for complex classification tasks. Although the parameter degree=3 does not directly impact the RBF kernel, it was retained at its default value to ensure completeness and consistency. The gamma='scale' option was used to automatically compute the optimal value of gamma based on the variance of the input data, facilitating adaptive model performance. Since the coef0 parameter is irrelevant for the RBF kernel, it was maintained at 0.0. Furthermore, the shrinking heuristic enhanced computational efficiency by focusing on crucial support vectors during optimization. Finally, probability estimation was turned off to prevent unnecessary computational overhead, ensuring faster model convergence without compromising classification accuracy.

The ANN was configured with the parameters: hidden_layer_sizes=(100,), activation='relu', solver='adam', alpha=0.0001, batch_size='auto', learning_rate_init=0.001, and max_iter=200. A single hidden layer containing 100 neurons was utilized to capture non-linear patterns while avoiding unnecessary complexity, which could lead to overfitting and increased computational demands. The ReLU activation function was selected for its efficiency and capability to mitigate the vanishing gradient problem, enhancing the stability of training in deep architectures.

Figure 10

EEG signal waveform for the acquisition of Voltage (μV) $\times$ Time (ms) data for the AF4 channel via (a) developed software EmotIF, (b) proprietary software EmotivPRO
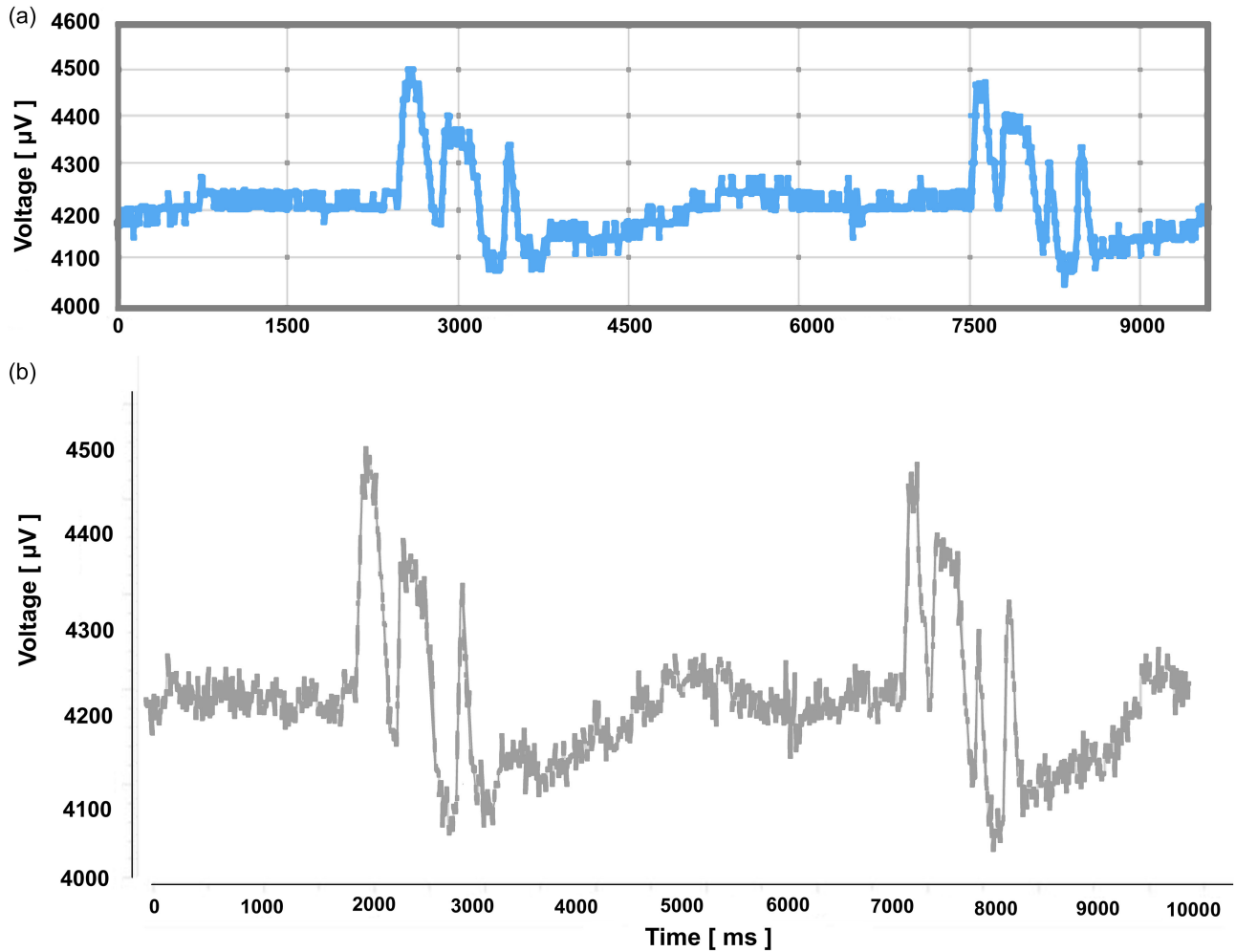
(a)

(b)

Figure 11

Representation of datasets generated by signals acquired through the EmotIF program

1a_black_color_VISUALSTIMULUS.csv 0	1_Dataset_CALIBRATION_VISUALSTIMULUS.csv
1b_white_color_VISUALSTIMULUS.csv 1	
2a_black_color_VOCALSTIMULUS.csv 0	2_Dataset_CALIBRATION_VOCALSTIMULUS.csv
2b_white_color_VOCALSTIMULUS.csv 1	
3a_black_color_MIXEDSTIMULUS.csv 0	3_Dataset_CALIBRATION_MIXEDSTIMULUS.csv
3b_white_color_MIXEDSTIMULUS.csv 1	
4a_open_hand_THOUGHT.csv 0	4_Dataset_ACTION_PURETHOUGHT.csv
4b_close_hand_THOUGHT.csv 1	
5a_open_hand_REINFORCED.csv 0	5_Dataset_ACTION_REINFORCED.csv
5b_close_hand_REINFORCED.csv 1	
6a_frustrated_THOUGHT.csv 0	6_Dataset_FEELING_PURETHOUGHT.csv
6b_excited_THOUGHT.csv 1	
7a_frustrated_REINFORCED.csv 0	7_Dataset_FEELING_REINFORCED.csv
7b_excited_REINFORCED.csv 1	

The Adam optimizer was employed for its robustness and adaptability across various learning tasks, offering efficient convergence. The regularization parameter alpha = 0.0001 was maintained to ensure moderate regularization, thereby controlling the model's capacity and preventing overfitting. The batch size was set to "auto" to allow the algorithm to dynamically determine the most appropriate size, balancing memory usage and computational performance. An initial learning rate of 0.001 was chosen, representing a stable and commonly used value supporting reliable model convergence. Finally, the maximum

Table 1
Table of signal characteristics for each routine for constructing inference datasets for training the ensemble of machine learning models

R	Metric [μV]	CHANNEL				
		AF3	T7	Pz	T8	AF4
1	Mean, Standard Deviation	3834 ± 1188	3863 ± 1186	3814 ± 1161	3796 ± 1135	3812 ± 1141
	Range (Min~Max)	0 ~ 4766	0 ~ 4666	0 ~ 4695	0 ~ 4667	0 ~ 4733
2	Mean, Standard Deviation	3835 ± 1205	3895 ± 1204	3843 ± 1150	3792 ± 1136	3803 ± 1142
	Range (Min~Max)	0 ~ 5088	0 ~ 4667	0 ~ 4533	0 ~ 4470	0 ~ 4463
3	Mean, Standard Deviation	3811 ± 1210	3836 ± 1198	3809 ± 1140	3788 ± 1142	3810 ± 1155
	Range (Min~Max)	0 ~ 4465	0 ~ 4470	0 ~ 4365	0 ~ 4497	0 ~ 4432
4	Mean, Standard Deviation	3860 ± 1192	3819 ± 1216	3735 ± 1104	3709 ± 1193	3823 ± 1169
	Range (Min~Max)	0 ~ 4729	0 ~ 4891	0 ~ 4731	0 ~ 4829	0 ~ 4665
5	Mean, Standard Deviation	3852 ± 1197	3819 ± 1209	3756 ± 1097	3744 ± 1167	3813 ± 1188
	Range (Min~Max)	0 ~ 4729	0 ~ 4891	0 ~ 4731	0 ~ 4829	0 ~ 4665
6	Mean, Standard Deviation	3826 ± 1199	3819 ± 1183	3747 ± 1174	3769 ± 1124	3821 ± 1162
	Range (Min~Max)	0 ~ 7423	0 ~ 4432	0 ~ 4301	0 ~ 4399	0 ~ 4565
7	Mean, Standard Deviation	3829 ± 1195	3817 ± 1187	3747 ± 1172	3770 ± 1119	3821 ± 1164
	Range (Min~Max)	0 ~ 7357	0 ~ 4700	0 ~ 4500	0 ~ 4632	0 ~ 4733

number of iterations was set to 200, ensuring the training process achieves a reasonable trade-off between computational efficiency and convergence quality.

An additional study using normalized values and a restructured target variable was conducted to balance the dataset's output classes (0 and 1), aiming to explore potential performance improvements in the ML models. However, no significant gains were observed, rendering adjustments to the existing hyperparameters unnecessary.

During the initial training phase with the "calibration" dataset, three rounds were performed: visual stimulus (R1), vocal stimulus (R2), and mixed stimulus (R3). Each round evaluated different data presentation modalities to assess ML model performance under varying conditions. The histogram in Figure 12 summarizes the results for each round, comparing the performance of three ML models: KNN, SVM, and ANN.

The results indicate that the KNN model achieved superior accuracy across most rounds, attaining 99.13% ± 0.04% in R1, 99.78% ± 0.04% in R2, and 98.49% ± 0.06% in R3. In contrast, the SVM model exhibited less consistent performance across the three rounds, with an accuracy of 93.14% ± 0.13% in R1, improving to 97.93% ± 0.08% in R2, but dropping significantly to 90.43% ± 0.19% in R3. Meanwhile, the ANN demonstrated relatively stable performance, yielding 95.85% ± 5.56% in R1, 98.85% ± 0.61% in R2, and 95.81% ± 3.26% in R3.

During the initial training phase with the "action" dataset, two experimental rounds—"pure thought" (R4) and "reinforced thought" (R5)—were conducted to investigate different cognitive states. These rounds aimed to assess the effectiveness of ML models in processing varying levels of mental activity, providing a comparative analysis of model performance under conditions of lower and higher cognitive engagement. The histogram in Figure 13 summarizes the results for each evaluated model (KNN, SVM, ANN).

The data reveal that the KNN model demonstrated high and consistent performance across both rounds, achieving an average accuracy of 98.22% ± 0.08% in R4 and 96.68% ± 0.12% in R5. Conversely, the SVM model yielded the lowest accuracy among the ML algorithms evaluated, with a mean accuracy of 90.51% ± 0.30% in R4 and 88.94% ± 0.69% in R5, indicating a decline in

performance as cognitive state intensity increased. The ANN model also exhibited high accuracy under both conditions, reaching 97.91% ± 0.94% in R4 and 95.50% ± 2.91% in R5, albeit with higher variability, particularly in the R2.

During the initial training phase using the dataset labeled "emotion", two distinct rounds were conducted to explore different cognitive states: "pure thought" (R6) and "reinforced thought" (R7). The objective was to evaluate the performance of ML models in processing variations in cognitive intensity associated with emotional states. The histogram shown in Figure 14 summarizes the results obtained for each model tested (KNN, SVM, and ANN).

The results indicate consistent performance from the KNN model, which achieved 97.31% ± 0.12% accuracy in R6 and slightly improved to 97.40% ± 0.12% in R7. Although less accurate than the other models, the SVM demonstrated a modest improvement between rounds, with 88.31% ± 0.56% accuracy in Round 6 and 89.27% ± 0.71% in R7, suggesting marginal adaptation to the increase in cognitive intensity. The ANN achieved 95.56% ± 3.50% accuracy in R6 but experienced a decline to 93.68% ± 4.35% in R7, accompanied by increased result variability, indicating heightened sensitivity to the complexity of emotional data in the reinforced thought state.

Figure 15 presents the confusion matrices for the KNN, SVM, ANN, and eML models. The eML model demonstrated superior performance, achieving an accuracy of 98.25% and the lowest root mean square error (RMSE) of 0.1324. Furthermore, it recorded high values for precision (98.00%), recall (98.50%), and F1-Score (98.25%), indicating a robust capability to accurately classify both positive and negative cases with a low occurrence of misclassifications.

The KNN model attained an accuracy of 98.04% and an RMSE of 0.1401, with precision, recall, and F1-Score values of 97.74%, 98.34%, and 98.04%, respectively. This indicates consistent performance, albeit slightly inferior to the eML model.

In contrast, the SVM model demonstrated the lowest accuracy at 90.76% and a significantly higher RMSE of 0.3039, reflecting a more significant error rate. Its precision (91.41%) and recall (89.95%) were also lower, indicating difficulties in accurately classifying positive and negative cases within the dataset. The

Figure 12
Histogram comparing the performance of machine learning models (KNN, SVM, and ANN) across three rounds of the "calibration" dataset: visual stimulus (R1), vocal stimulus (R2), and mixed stimulus (R3)
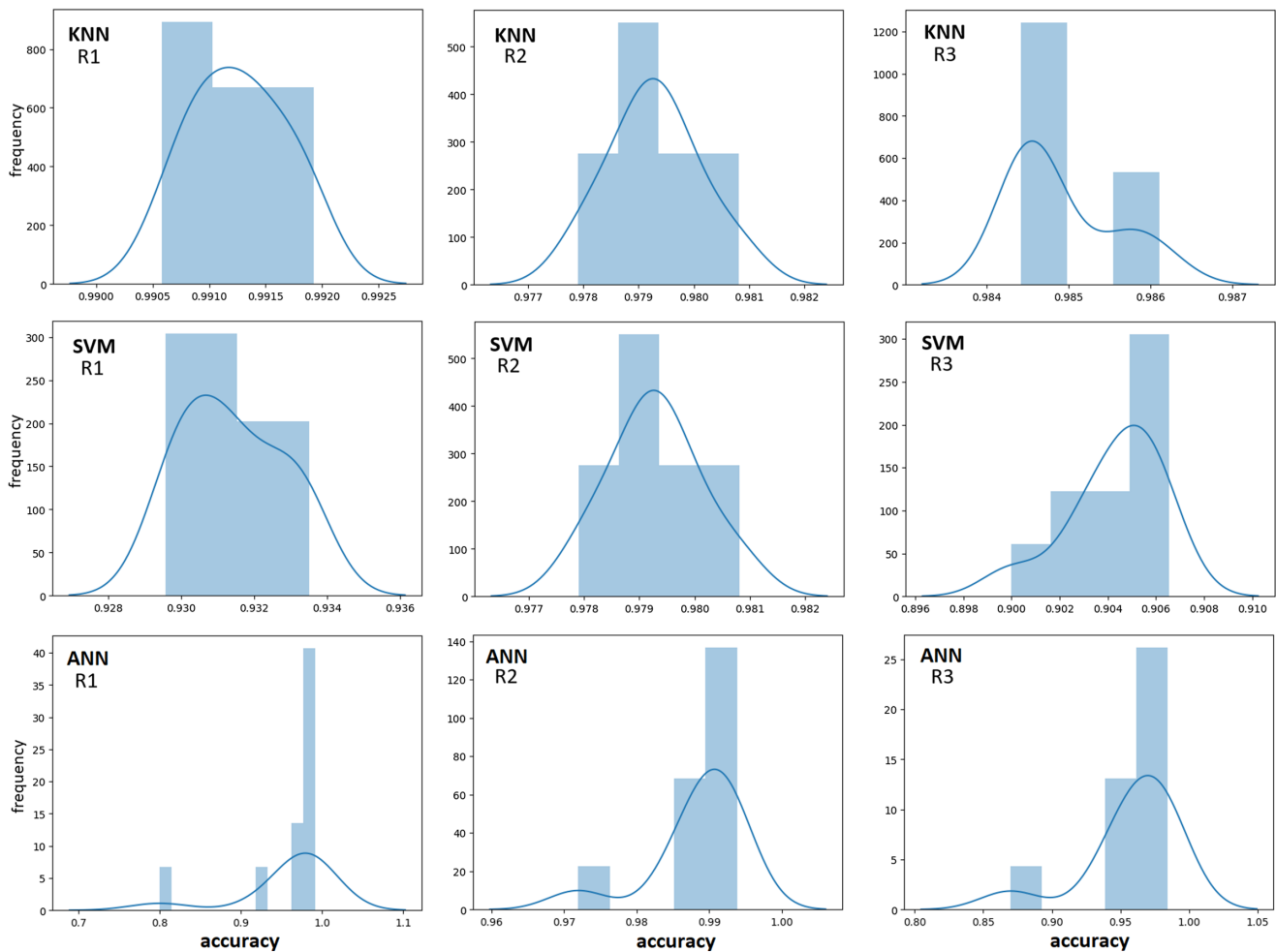
*F*1-Score of 90.68% underscores this limitation, suggesting the need for adjustments to enhance its discriminative capability.

The ANN model was competitive, achieving an accuracy of 98.38% and the lowest RMSE among the analyzed models at 0.1272. Additionally, it demonstrated high precision (97.88%), recall (98.90%), and *F*1-Score (98.39%), indicating excellent performance in accurately identifying both positive and negative cases.

The results obtained from the eML are summarized in the histogram in Figure 15. This graphical representation reveals a high correlation between the observed values and predictions. Due to this high correlation, the divergence is confined to a narrow range, reflected at the top of the graph.

5. Comparative Study

Graphs (Figures 6–10) show voltage readings (µV) over time in EmotivPRO and EmotIF, revealing consistent patterns of peaks and troughs. The similarity in these patterns confirms significant correspondence between the two approaches, validating the reliability of the custom-developed program for brain activity acquisition.

Statistical analysis of Table 1 reveals mean electrical potentials around 3850 µV, with a channel variation of ±100 µV. Standard

deviations (1100–1200 µV) indicate moderate heterogeneity across channels, but insufficient to alter the mean values significantly. Amplitude ranges (Min~Max) highlight variations across channels, with AF3 and AF4 exhibiting higher amplitudes (up to 7423 µV), suggesting greater signal fluctuation related to cognitive processes. Other channels show lower variability (4300–4900 µV), indicating stability.

Channels AF3 and T7 consistently recorded higher values, suggesting their prominent role in calibration and task-related neural activity. These results contribute to understanding the neural mechanisms underlying task execution.

Preliminary signal dispersion analysis in inference datasets informed parameter optimization for ML training. Graphs indicated potential classification patterns, though with notable complexity. Subtle interchannel relationships suggest a possible linear correlation with enhanced acquisition systems and more extensive channel data.

The results in Figure 12 indicate that the "vocal stimulus" (R2) yielded the highest overall performance among the three ML models, suggesting these algorithms were particularly effective in processing auditory data. However, the performance decline of the SVM model during the "mixed stimulus" (R3) points to potential sensitivity to mixed sensory modalities, emphasizing the need for model

Figure 13
Histogram comparing the performance of machine learning models (KNN, SVM, and ANN) across two rounds of the "action" dataset: pure thought (R4) and reinforced thought (R5)
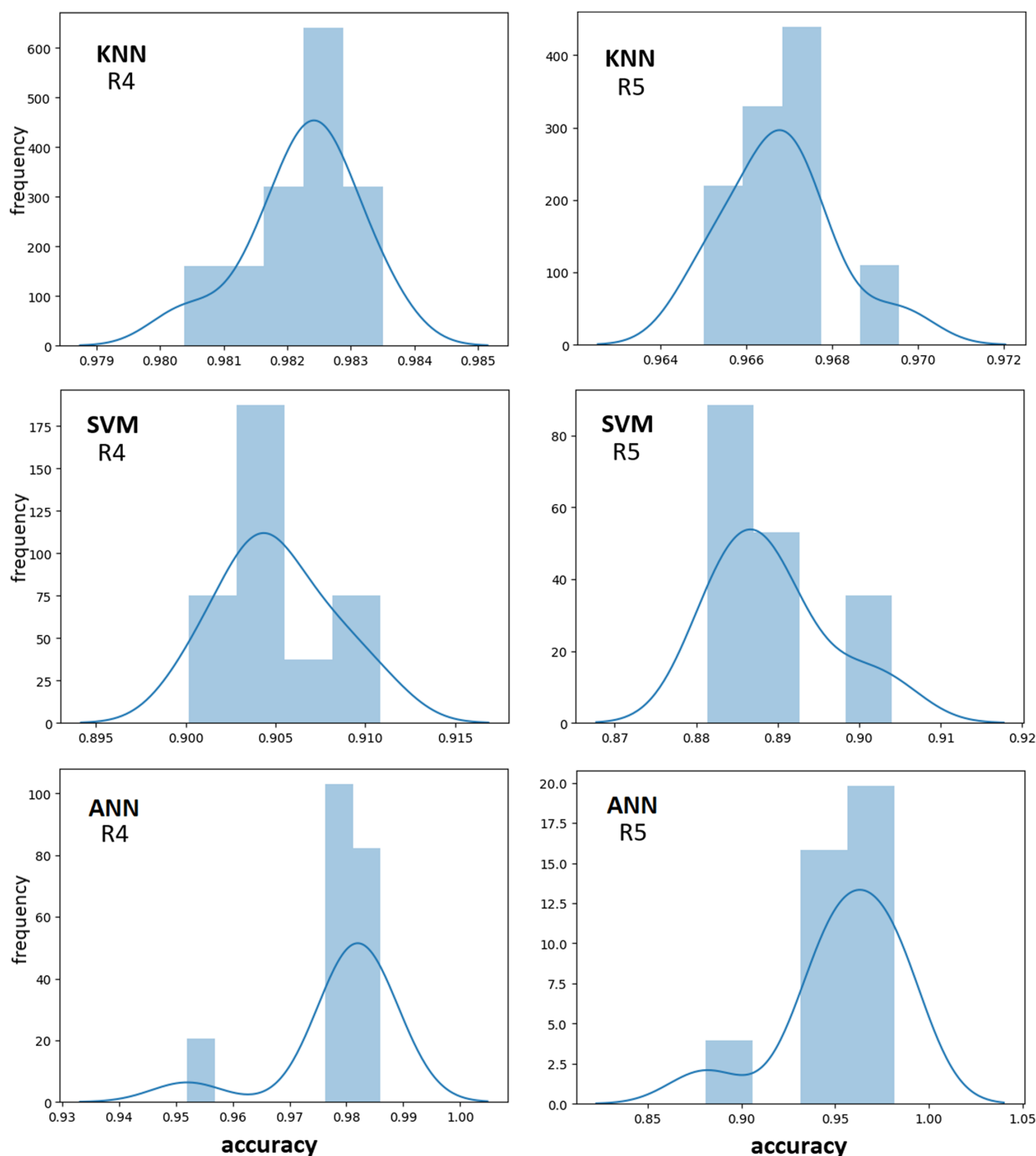

adjustments or more targeted calibration when dealing with combined stimuli. Conversely, the consistent outcomes observed in the KNN and ANN models underscore their robustness across varying conditions, with KNN demonstrating a slight advantage in handling purely visual (R1) and auditory (R2) data.

The analysis of results in Figure 13 shows that all models exhibited a modest decrease in accuracy when processing data from the "reinforced thought" state (R5). This reduction in accuracy can be attributed to increased complexity arising from heightened cognitive activity, suggesting that the engaging nature of this state may introduce additional noise into the data. Notably, the KNN model proved the most robust, displaying minimal performance variation across rounds and achieving superior accuracy under both scenarios. In contrast, the SVM model exhibited higher sensitivity to variations between cognitive states, resulting in less consistent performance. Although the ANN model experienced slightly reduced accuracy

Figure 14
Histogram comparing the performance of machine learning models (KNN, SVM, and ANN) across two rounds of the "emotion" dataset: pure thought (R6) and reinforced thought (R7)
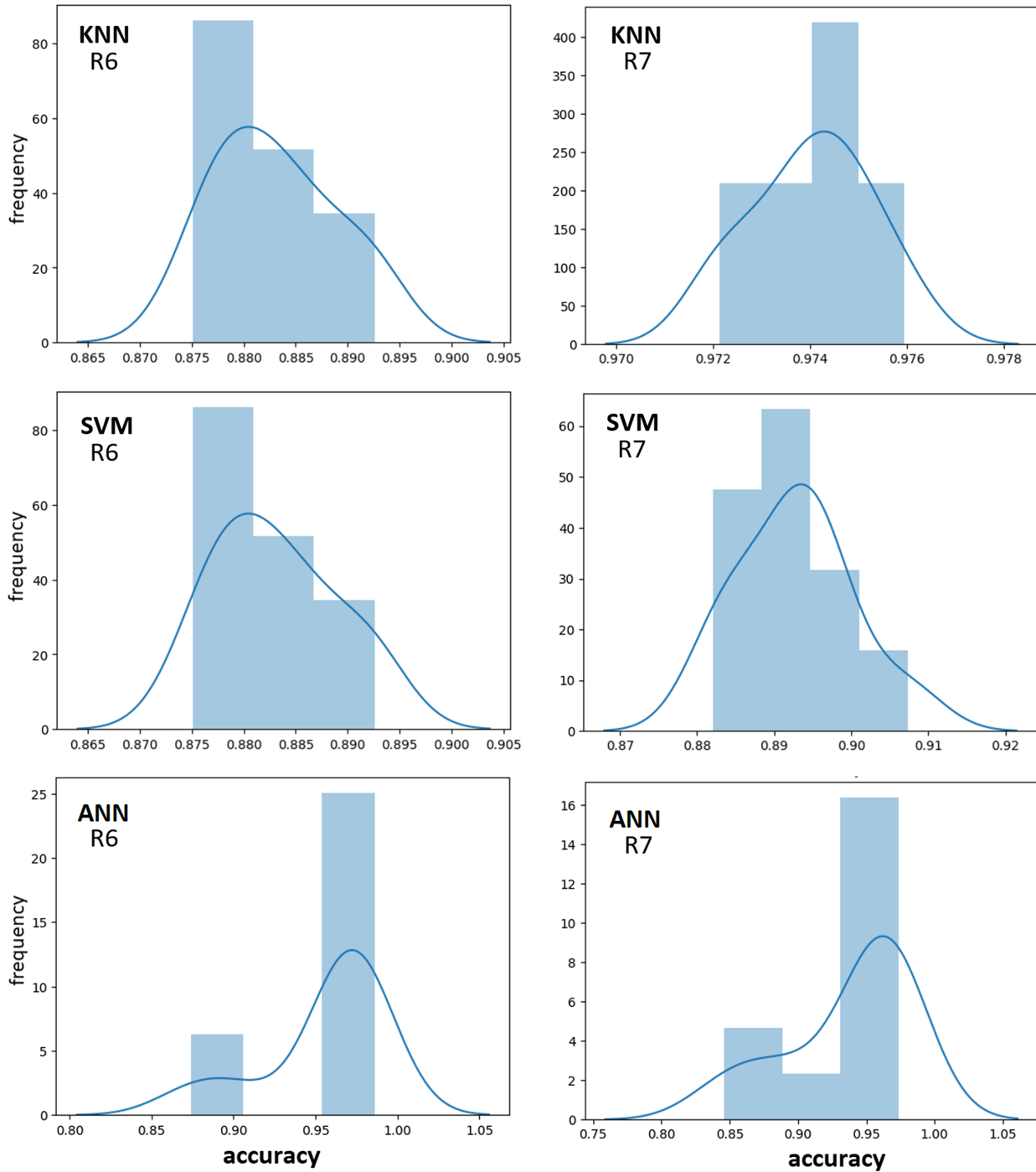

during R5, it maintained competitive performance, demonstrating a relatively strong capacity to manage the complexity of data generated from cognitively demanding states.

A comparative evaluation of Figure 14 reveals that KNN remained the most stable and efficient model across both conditions, showing minimal variation in performance between the "pure thought" (R7) and "reinforced thought" (R8) states. While the SVM model displayed lower overall accuracy, its progressive improvement across rounds suggests some capacity

for adaptation to increasing emotional complexity. On the other hand, although ANN achieved high performance during R6, it experienced a reduction in accuracy and a substantial increase in standard deviation in R7, indicating greater susceptibility to variability introduced by more intense emotional states.

The results obtained from the seven experimental rounds indicate that the KNN algorithm demonstrated the best performance in terms of accuracy, followed by ANN and SVM. The KNN achieved an average accuracy of 98.15% ± 0.08%,

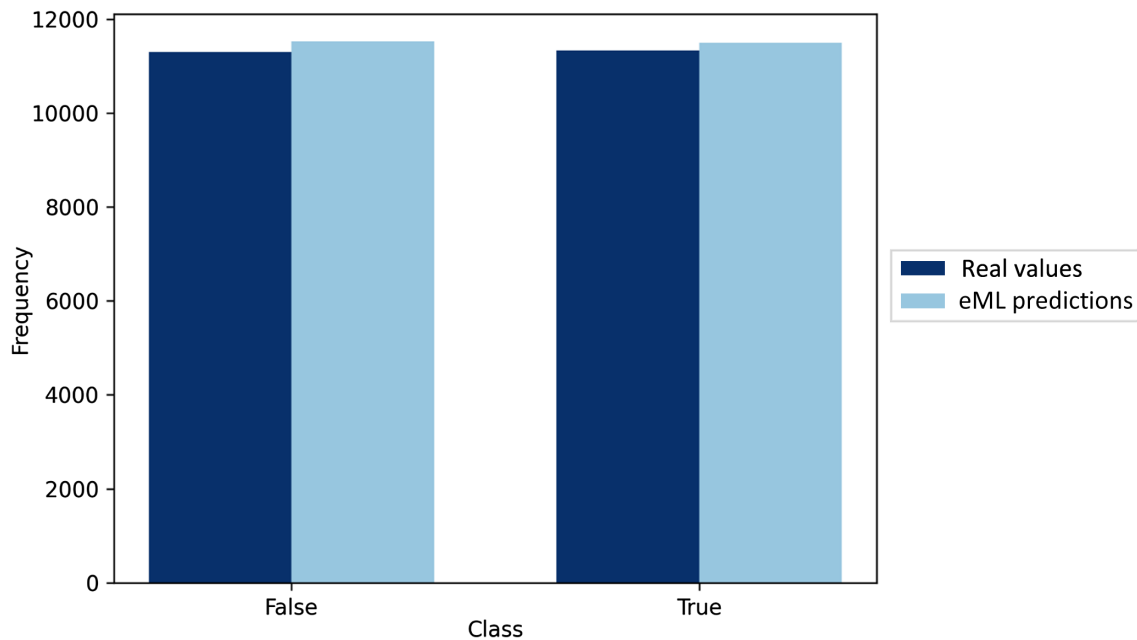**Figure 15**
Confusion matrix charts of eML, KNN, SVM, and ANN
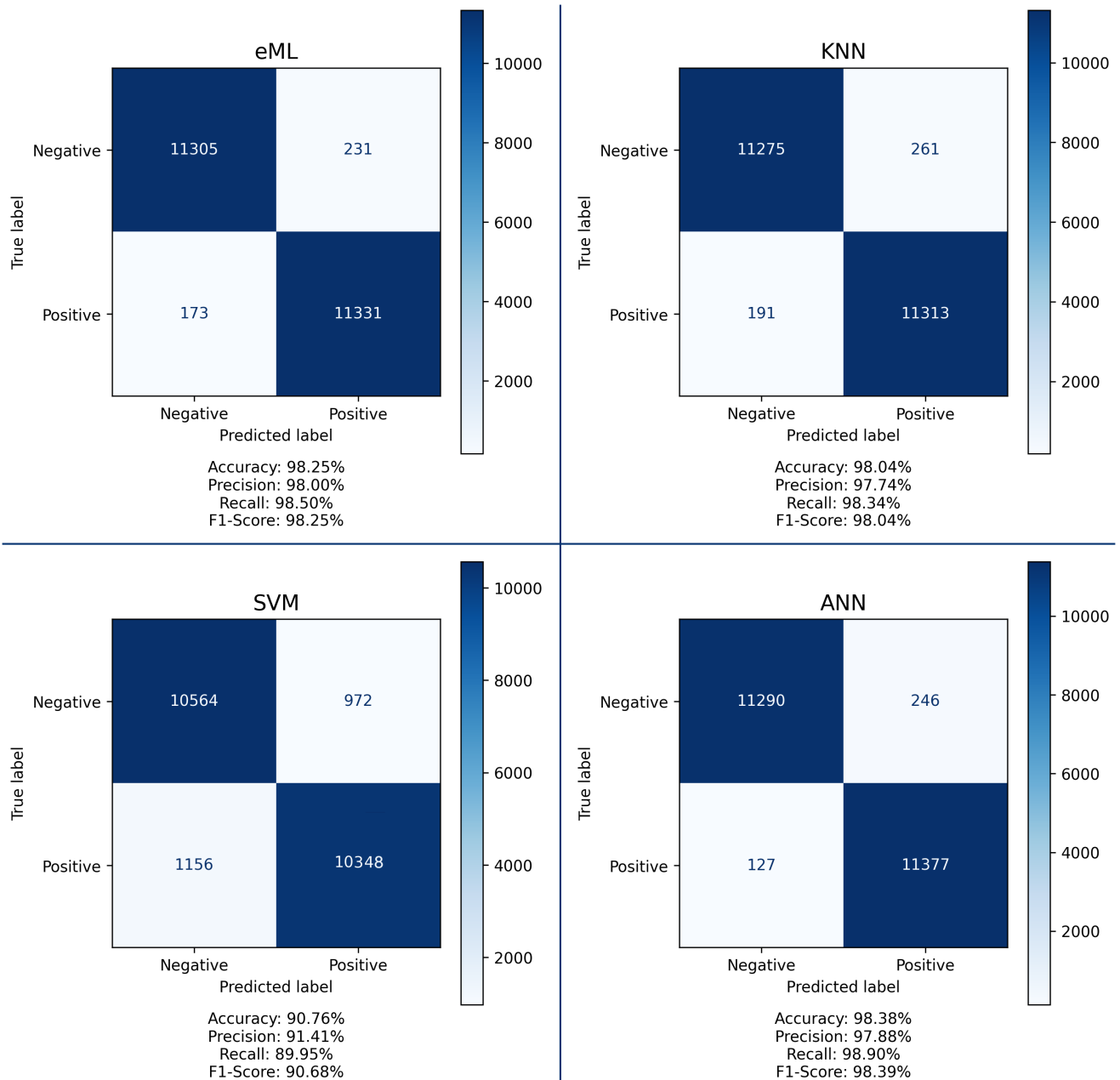
indicating high precision and consistency in data classification. The ANN achieved an average accuracy of 96.02% ± 3.02%, demonstrating good generalization and adaptability to the data, though with more significant variability in the results. The SVM achieved an average accuracy of 90.93% ± 0.38%, revealing lower efficiency and robustness in data classification but less variability in the results. These findings reinforce the robustness of the KNN model across diverse emotional scenarios and indicate that further adjustments may be necessary to optimize the performance of the SVM and ANN models when exposed to cognitively more demanding conditions.

These results can be explained by the nature of the algorithms and the data used. KNN, being an instance-based algorithm, classifies data by leveraging similarities between data points.

This algorithm is well-suited for high-dimensional data with minimal class overlap, as was the case with the data used in this experiment. The ANN is a deep learning algorithm that learns complex patterns in the data and classifies them. While capable of handling non-linear and noisy data, ANN requires large amounts of data and parameters for proper training. SVM, a margin-based algorithm, utilizes support vectors to define hyperplanes that separate classes within the data. As observed in this experiment, linear data with few outliers is efficient but may struggle with non-linear and numerous outliers. Thus, the choice between these algorithms should consider not only the type and complexity of the problem but also the available computational resources and the time required for processing and training.

Figure 16
Histogram comparing the predictions of the eML

Based on the obtained metrics, as illustrated in Figure 15, both the eML and the ANN models emerged as the most robust options, with the eML model exhibiting a slight advantage in overall balance. The KNN model also demonstrated satisfactory results, lagging only marginally behind. In contrast, the SVM faced significant challenges, suggesting it may not be the optimal choice for this type of task or that more refined adjustments are necessary to enhance its performance. The results indicate that the eML model outperforms the individual models (ANN, KNN, and SVM) in the data classification task. This distinction is evidenced by the accuracy and RMSE values, which represent the rate of correct classifications and the mean of the errors' deviations, respectively. The eML model achieved accuracies exceeding 98% and RMSE values below 0.14, indicating greater precision and consistency in distinguishing between true positives and negatives and false positives and negatives. This superior performance, as presented in Figure 16, can be attributed to the eML's capacity to integrate information from multiple classifiers, thereby enhancing the robustness and generalization capabilities of the models. Consequently, the results suggest that the eML is more suitable for addressing this classification problem than the ML models.

The studies by Beyrouthy et al. [13] and Fuentes-Gonzalez et al. [20] leverage EEG signals for prosthetic control, underscoring the growing relevance of EEG and other biosignals as technological advancements and domain knowledge progress. A thorough literature review reveals a notable gap in studies utilizing ML to detect brain patterns from EEG signals for ULP control. This absence complicates direct comparisons with prior research; however, the findings of this study suggest significant potential for this application.

This research presents several limitations that must be acknowledged. First, the experiments were conducted under controlled conditions, and the system's performance in real-world environments—where EEG signals are inherently noisier and more variable—was not evaluated. Thus, the accuracy of the ML models may have been affected by the controlled nature of data collection, potentially influencing the reported metrics. Furthermore, despite extensive testing, the risk of overfitting remains a concern, as the

models may have adapted too closely to the training data, limiting their capacity to generalize to unseen datasets. These factors emphasize the need for future studies to assess the robustness of the proposed approach under more realistic and challenging conditions.

Real-world applications introduce complexities beyond controlled lab conditions, such as perspiration, head movement, and patient-related factors like stress and multitasking, which degrade the physical interface between the scalp and electrodes, reducing EEG signal quality [44]. ML offers solutions to address these challenges by using large datasets to enhance pattern detection and classification, improving control system reliability in dynamic environments. ML-based frameworks can mitigate noise and disturbances, facilitating personalized systems tailored to individual patient needs [45]. However, ML faces challenges, including algorithmic complexity and model biases [46], necessitating further validation studies.

Integrating ML algorithms into portable devices is challenging due to high computational demands, which often exceed the size and power constraints of wearable systems [47]. Recent advances in ML inference optimization enable more efficient integration [48]. Additionally, ML-enabled systems' reliance on internet-based solutions makes them vulnerable to cyber threats [49], highlighting the need for robust security frameworks in real-world deployment.

6. Conclusion

An EEG signal acquisition system using KNN, SVM, ANN, and eML controlled a robotic ULP with promising results across metrics. The custom EmotIF software effectively captured EEG signals, validated against manufacturer software, with minor conversion discrepancies noted for further investigation.

Regarding PaRecoML performance, the eML demonstrated high reliability with an accuracy of 98.25% and an RMSE of 0.1324. The ANN slightly outperformed other models with a peak accuracy of 98.38% and the lowest RMSE of 0.1272, followed closely by KNN, which achieved 98.04% accuracy with an RMSE

of 0.1401. Although exhibiting comparatively lower performance, the SVM reached 90.76% accuracy with an RMSE of 0.3039. Integrating KNN, SVM, and ANN into the eML combined the strengths of individual models to accurately detect EEG patterns, achieving approximately 90% accuracy in the final training rounds.

The study demonstrates ML's potential in prosthetics, encouraging further exploration. Project software is fostering collaboration. Future work will optimize signal conversion, integrate supervisory control for real-time monitoring, conduct in vitro trials, and expand applications to assistive devices, leveraging advanced ML methods like reinforcement or deep learning for improved EEG interpretation and prosthetic control.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data that support the findings of this study are openly available in GitHub at https://github.com/S-Brno/ARTIFICIAL_INTELLIGENCE__EEG_UPPERLIMB_PROSTHESIS.

Author Contribution Statement

Bruno J. Santos: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Visualization. **Alysson Avila:** Conceptualization, Software, Validation, Formal analysis, Investigation, Data curation. **Marcelo Barboza:** Writing – review & editing. **Evandro Drigo:** Resources, Writing – review & editing. **Tarcisio Leão:** Supervision. **Eduardo Bock:** Resources, Supervision, Project administration.

References

[1] World Health Organization. (2023). *Disability*. https://www.who.int/news-room/fact-sheets/detail/disability-and-health

[2] Guatibonza, A., Solaque, L., Velasco, A., & Peñuela, L. (2024). Assistive robotics for upper limb physical rehabilitation: A systematic review and future prospects. *Chinese Journal of Mechanical Engineering*, *37*(1), 69. https://doi.org/10.1186/s10033-024-01056-y

[3] Jiang, N., Chen, C., He, J., Meng, J., Pan, L., Su, S., & Zhu, X. (2023). Bio-robotics research for non-invasive myoelectric neural interfaces for upper-limb prosthetic control: A 10-year perspective review. *National Science Review*, *10*(5), nwad048. https://doi.org/10.1093/nsr/nwad048

[4] Huang, Z., & Wang, M. (2021). A review of electro-encephalogram signal processing methods for brain-controlled robots. *Cognitive Robotics*, *1*, 111–124. https://doi.org/10.1016/j.cogr.2021.07.001

[5] Brack, R., & Amalu, E. H. (2021). A review of technology, materials and R&D challenges of upper limb prosthesis for improved user suitability. *Journal of Orthopaedics*, *23*, 88–96. https://doi.org/10.1016/j.jor.2020.12.009

[6] Benevides, A. B., da Paz Floriano, A. S., Sarcinelli-Filho, M., & Bastos-Filho, T. F. (2020). Review of the human brain and EEG signals. In T. Bastos-Filho (Ed.), *Introduction to non-invasive EEG-based brain-computer interfaces for assistive technologies* (pp. 1–49). CRC Press. https://doi.org/10.1201/9781003049159-1

[7] Neumann, B. (2010). Being prosthetic in the first world war and Weimar Germany. *Body & Society*, *16*(3), 93–126. https://doi.org/10.1177/1357034X10373403

[8] Farina, D., Vujaklija, I., Brånemark, R., Bull, A. M., Dietl, H., Graimann, B., . . . , & Aszmann, O. C. (2023). Toward higher-performance bionic limbs for wider clinical use. *Nature Biomedical Engineering*, *7*(4), 473–485. https://doi.org/10.1038/s41551-021-00732-x

[9] Briko, A., Kapravchuk, V., Kobelev, A., Hammoud, A., Leonhardt, S., Ngo, C., . . . , & Shchukin, S. (2022). A way of bionic control based on EI, EMG, and FMG signals. *Sensors*, *22*(1), 152. https://doi.org/10.3390/s22010152

[10] Zhou, H., & Alici, G. (2022). Non-invasive human-machine interface (HMI) systems with hybrid on-body sensors for controlling upper-limb prosthesis: A review. *IEEE Sensors Journal*, *22*(11), 10292–10307. https://doi.org/10.1109/JSEN.2022.3169492

[11] Triwiyanto, T., Rahmawati, T., Pawana, I. P. A., Lamidi, L., Hamzah, T., Pudji, A., . . . , & Luthfiyah, S. (2021). State-of-the-art method in prosthetic hand design: A review. *Journal of Biomimetics, Biomaterials and Biomedical Engineering*, *50*, 15–24. https://doi.org/10.4028/www.scientific.net/JBBBE.50.15

[12] Nicolelis, M. A. (2003). Brain-machine interfaces to restore motor function and probe neural circuits. *Nature Reviews Neuroscience*, *4*(5), 417–422. https://doi.org/10.1038/nrn1105

[13] Beyrouthy, T., Al Kork, S. K., Korbane, J. A., & Abdulmonem, A. (2016). EEG mind controlled smart prosthetic arm. In *2016 IEEE International Conference on Emerging Technologies and Innovative Business Practices for the Transformation of Societies*, 404–409. https://doi.org/10.1109/EmergiTech.2016.7737375

[14] Masson, S., Fortuna, F., Moura, F., & Soriano, D. (2016). Integrating Myo armband for the control of myoelectric upper limb prosthesis. In *Proceedings of the XXV Congresso Brasileiro de Engenharia Biomédica*, 1642–1645.

[15] Gaetani, F., Primiceri, P., Antonio Zappatore, G., & Visconti, P. (2019). Hardware design and software development of a motion control and driving system for transradial prosthesis based on a wireless myoelectric armband. *IET Science, Measurement & Technology*, *13*(3), 354–362. https://doi.org/10.1049/iet-smt.2018.5108

[16] Milea, P. L., Dascalu, M., Franti, E. D., Barbili , A., & Stoica, I. C. (2017). Tactile feedback experiments for forearm prosthesis with myoelectric control. *Romanian Journal of Information Science and Technology*, *20*(2), 101–114.

[17] Wang, Y., Tan, Q., Pu, F., Boone, D., & Zhang, M. (2020). A review of the application of additive manufacturing in prosthetic and orthotic clinics from a biomechanical perspective. *Engineering*, *6*(11), 1258–1266. https://doi.org/10.1016/j.eng.2020.07.019

[18] ten Kate, J., Smit, G., & Breedveld, P. (2017). 3D-printed upper limb prostheses: A review. *Disability and Rehabilitation: Assistive Technology*, *12*(3), 300–314. https://doi.org/10.1080/17483107.2016.1253117

[19] Savsani, S., Singh, S., & Mali, H. S. (2023). Additive manufacturing for prostheses development: State of the art. *Rapid Prototyping Journal*, *29*(4), 741–765. https://doi.org/10.1108/RPJ-01-2022-0029

[20] Fuentes-Gonzalez, J., Infante-Alarcón, A., Asanza, V., & Loayza, F. R. (2021). A 3D-printed eeg based prosthetic arm. In *2020 IEEE International Conference on E-health Networking, Application & Services*, 1–5. https://doi.org/10.1109/HEALTHCOM49281.2021.9399035

[21] de Araújo, F. M. A., Rocha, F. V. T., da Silva Souza, A., & Dantas, M. A. (2021). Desenvolvimento de uma mão robótica construída com manufatura aditiva e tecnologia open source controlada a partir de estímulos musculares [Development of a robotic hand constructed with additive manufacture and controlled open source technology from muscle stimulus]. *Brazilian Journal of Development*, *7*(3), 27895–27903. https://doi.org/10.34117/bjdv7n3-480

[22] Escribà Montagut, G. (2016). *Inmoov robot: Building of the first open source 3D printed life-size robot* [Bachelor's Thesis]. Repositori Obert Udl.

[23] Cortes Zarta, J. F., Giraldo Tique, Y. A., & Vergara Ramírez, C. F. (2021). Convolutional neural network for spatial perception of inMoov robot through stereoscopic vision as an assistive technology. *Enfoque UTE*, *12*(4), 88–104. https://doi.org/10.29019/enfoqueute.776

[24] Sidher, A., & Shen, Y. (2017). Improving a 3D-printed artificial anthropomorphic hand using the human hand model. In *2017 IEEE International Conference on Real-time Computing and Robotics*, 739–744. https://doi.org/10.1109/RCAR.2017.8311952

[25] Cherichel, P., & de Curtis, R. (2020). EMG-controlled mechanical hand. *Journal of Innovative Ideas in Engineering and Technology*, *1*(1), 1–38.

[26] Sanei, S., & Chambers, J. A. (2021). *EEG signal processing and machine learning*. USA: John Wiley & Sons. https://doi.org/10.1002/9781119386957

[27] Li, G.-L., Wu, J.-T., Xia, Y.-H., He, Q.-G., & Jin, H.-G. (2020). Review of semi-dry electrodes for EEG recording. *Journal of Neural Engineering*, *17*(5), 051004. https://doi.org/10.1088/1741-2552/abbd50

[28] Cheng, G., Ehrlich, S. K., Lebedev, M., & Nicolelis, M. A. (2020). Neuroengineering challenges of fusing robotics and neuroscience. *Science Robotics*, *5*(49), eabd1911. https://doi.org/10.1126/scirobotics.abd1911

[29] Zhang, X.-D. (2023). *Modern signal processing*. China: Tsinghua University Press. https://doi.org/10.1515/9783110475562

[30] Bhardwaj, H., Tomar, P., Sakalle, A., & Ibrahim, W. (2021). EEG-based personality prediction using fast fourier transform and deepLSTM model. *Computational Intelligence and Neuroscience*, *2021*(1), 6524858. https://doi.org/10.1155/2021/6524858

[31] Torres, E. P., Torres, E. A., Hernández-Álvarez, M., & Yoo, S. G. (2020). EEG-based BCI emotion recognition: A survey. *Sensors*, *20*(18), 5083. https://doi.org/10.3390/s20185083

[32] Kawala-Sterniuk, A., Browarska, N., Al-Bakri, A., Pelc, M., Zygarlicki, J., Sidikova, M., . . . , & Gorzelanczyk, E. J. (2021). Summary of over fifty years with brain-computer interfaces—A review. *Brain Sciences*, *11*(1), 43. https://doi.org/10.3390/brainsci11010043

[33] Värbu, K., Muhammad, N., & Muhammad, Y. (2022). Past, present, and future of EEG-based BCI applications. *Sensors*, *22*(9), 3331. https://doi.org/10.3390/s22093331

[34] Igual, C., Pardo Jr, L. A., Hahne, J. M., & Igual, J. (2019). Myoelectric control for upper limb prostheses. *Electronics*, *8*(11), 1244. https://doi.org/10.3390/electronics8111244

[35] Bueno, L., Pons, J. L., & Bastos Filho, T. F. (2013). An embedded system for an EEG based BCI. In *2013 ISSNIP Biosignals and Biorobotics Conference: Biosignals and Robotics for Better and Safer Living,* 1–5. https://doi.org/10.1109/BRC.2013.6487457

[36] Balderas, D., Ponce, P., Lopez-Bernal, D., & Molina, A. (2021). Education 4.0: Teaching the basis of motor imagery classification algorithms for brain-computer interfaces. *Future Internet*, *13*(8), 202. https://doi.org/10.3390/fi13080202

[37] Lioumis, P., & Rosanova, M. (2022). The role of neuronavigation in TMS–EEG studies: Current applications and future perspectives. *Journal of Neuroscience Methods*, *380*, 109677. https://doi.org/10.1016/j.jneumeth.2022.109677

[38] Choubey, H., & Pandey, A. (2021). A combination of statistical parameters for the detection of epilepsy and EEG classification using ANN and KNN classifier. *Signal, Image and Video Processing*, *15*(3), 475–483. https://doi.org/10.1007/s11760-020-01767-4

[39] Sha'abani, M. N. A. H., Fuad, N., Jamal, N., & Ismail, M. F. (2020). kNN and SVM classification for EEG: A review. In *InECCE2019: Proceedings of the 5th International Conference on Electrical, Control & Computer Engineering*, 555–565. https://doi.org/10.1007/978-981-15-2317-5_47

[40] Hosseini, M.-P., Hosseini, A., & Ahi, K. (2021). A review on machine learning for EEG signal processing in bioengineering. *IEEE Reviews in Biomedical Engineering*, *14*, 204–218. https://doi.org/10.1109/RBME.2020.2969915

[41] Huang, Q., Wang, C., Ye, Y., Wang, L., & Xie, N. (2023). Recognition of EEG based on improved black widow algorithm optimized SVM. *Biomedical Signal Processing and Control*, *81*, 104454. https://doi.org/10.1016/j.bspc.2022.104454

[42] Ramírez-Arias, F. J., García-Guerrero, E. E., Tlelo-Cuautle, E., Colores-Vargas, J. M., García-Canseco, E., López-Bonilla, O. R., . . . , & Inzunza-González, E. (2022). Evaluation of machine learning algorithms for classification of EEG signals. *Technologies*, *10*(4), 79. https://doi.org/10.3390/technologies10040079

[43] Dong, X., Yu, Z., Cao, W., Shi, Y., & Ma, Q. (2020). A survey on ensemble learning. *Frontiers of Computer Science*, *14*, 241–258. https://doi.org/10.1007/s11704-019-8208-z

[44] Chen, Z., Min, H., Wang, D., Xia, Z., Sun, F., & Fang, B. (2023). A review of myoelectric control for prosthetic hand manipulation. *Biomimetics*, *8*(3), 328. https://doi.org/10.3390/biomimetics8030328

[45] Santos, B., Barboza, M., Leão, T., Santos, D., Andrade, A., & Bock, E. (2023). Intelligent embedded system for physiological control of ventricular assist devices in Health 4.0 background. *International Journal of Advances in Medical Biotechnology*, *5*(2), 8–21. https://doi.org/10.52466/ijamb.v5i2.112

[46] de Sousa, M. A. D. A. (2025). The shift of Artificial Intelligence research from academia to industry: Implications and possible future directions. *AI & Society*, *40*, 805–814. https://doi.org/10.1007/s00146-024-01924-0

[47] dos Santos, B. J., Leão, T. F., Silva, M. B., da Silva, E. D., & Bock, E. G. P. (2024). Embedded cyber-physical system for physiological control of ventricular assist devices. *Journal of Electronics and Electrical Engineering*, *3*(1), 160–178. https://doi.org/10.37256/jeee.3120244199

[48] Santos, B. J., & Cestari, I. A. (2025). Intelligent reflux and suction detection system for ventricular assist devices: In

silico study. *Research on Biomedical Engineering*, *41*(1), 8. https://doi.org/10.1007/s42600-024-00393-0

[49] Santos, B. J., Tabacow, R. P., Barboza, M., Leão, T. F., & Bock, E. G. (2022). Cyber security in health: Standard protocols for IoT and supervisory control systems. In Information Resources Management Association (Ed.), *Research anthology on securing medical systems and records* (pp. 238–254). IGI Global Publishing. https://doi.org/10.4018/978-1-6684-6311-6.ch012