GENERAL INSTRUCTION

- **Authors: Carefully check the page proofs (and coordinate with all authors); additional changes or updates WILL NOT be accepted after the article is published online/print in its final form. Please check author names and affiliations, funding, as well as the overall article for any errors prior to sending in your author proof corrections. Your article has been peer reviewed, accepted as final, and sent in to IEEE. No text changes have been made to the main part of the article as dictated by the editorial level of service for your publication.**
- **Authors: We cannot accept new source files as corrections for your article. If possible, please annotate the PDF proof we have sent you with your corrections and upload it via the Author Gateway. Alternatively, you may send us your corrections in list format. You may also upload revised graphics via the Author Gateway.**

QUERIES

Q1. Author: Please confirm or add details for any funding or financial support for the research of this article.
Q2. Author: Please check and confirm whether the author affiliations in the first footnote are correct as set.
Q3. Author: Please provide the expansion for the acronyms AI and ROC at the instance when they are first mentioned in the text.
Q4. Author: Please provide the page range in Ref. [1].
Q5. Author: Please provide the complete page range in Refs. [15], [32], and [42].
Q6. Author: Please update Refs. [37], [39], and [41].

Q1: Ok, confirmed.
Q2: Yes, they are correct.
Q3: AI (Artificial Intelligence) used only once in the “Index Terms” section, on page 1.
ROC (Receiver Operating Characteristic) used first on line 337 on page 5.
Q4: Ref [1], Pages 1-8
Q5: Ref [15] Page Range 208-221,
Ref [32] Page Range 267-281,
Ref [42] Page Range 36-46
Q6: Ref [37] Replace with @article{Cullen2018,
doi = {10.1016/j.bpsc.2018.06.010},
issn = {24519030},
journal = {Biological Psychiatry: Cognitive Neuroscience and Neuroimaging},
month = {sep},
number = {9},
pages = {809--818},
pmid = {30082215},
publisher = {Elsevier Inc},
title = {{Active Inference in OpenAI Gym: A Paradigm for Computational Investigations Into Psychiatric Illness}},
volume = {3},
year = {2018}
}
Ref [39], Replace with @article{doi:10.1080/01621459.2020.1831925,
author = { Xinkun Nie and Emma Brunskill and Stefan Wager },
title = {Learning When-to-Treat Policies},
journal = {Journal of the American Statistical Association},
volume = {0},
number = {0},
pages = {1-18},
year = {2020},
publisher = {Taylor & Francis},
doi = {10.1080/01621459.2020.1831925},
URL = {https://doi.org/10.1080/01621459.2020.1831925},eprint = {https://doi.org/10.1080/01621459.2020.1831925}
}
Ref [41], Replace with @article{Aldayel2020,
author = {Aldayel, Mashael S. and Ykhlef, Mourad and Al-Nafjan, Abeer N.},
doi = {10.1109/access.2020.3027429},
issn = {2169-3536},
journal = {IEEE Access},
month = {sep},
pages = {176818--176829},
publisher = {Institute of Electrical and Electronics Engineers (IEEE)},
title = {{Electroencephalogram-Based Preference Prediction Using Deep Transfer Learning}},
volume = {8},
year = {2020}
}

Training a Gaming Agent on Brainwaves

Bartolomé Francisco, Moreno Juan, Navas Natalia, Vitali José, Ramele Rodrigo, *Member, IEEE*, and Santos Juan Miguel

***Abstract*—Error-related potentials (ErrPs) are a particular type of event-related potential elicited by a person attending a recognizable error. These electroencephalographic signals can be used to train a gaming agent by a reinforcement learning algorithm to learn an optimal policy. The experimental process consists of an observational human critic (OHC) observing a simple game scenario while their brain signals are captured. The game consists of a grid, where a blue spot has to reach a desired target in the fewest amount of steps. Results show that there is an effective transfer of information and that the agent successfully learns to solve the game efficiently, from the initial 97 steps on average required to reach the target to the optimal number of eight steps. Our results are expressed in threefold: the mechanics of a simple grid-based game that can elicit the ErrP signal component; the verification that the reward function only penalizes wrong steps, which means that type II error (not properly identifying a wrong movement) does not affect significantly the agent learning process; collaborative rewards from multiple OHCs can be used to train the algorithm effectively and can compensate low classification accuracies and a limited scope of transfer learning schemes.**

***Index Terms*—Agent, AI, brain–computer interface (BCI), electroencephalographic (EEG), error-related potential (ErrP), reinforcement learning (RL).**

I. Introduction

THE effectiveness of today's human–machine interaction and artificial intelligence is limited by a communication bottleneck, as humans are required to translate high-level concepts into a machine-mandated sequence of instructions [1], [2]. Hence, new interaction methods are required to increase the communication bandwidth between computers and humans or to produce alternative communications systems to increase the efficiency of this channel. In this respect, video games have been widely used as test tools to assess new means of interactions [3]–[5]. Video gaming agents are computer programs that can sense the computer game environment, process information, and react accordingly within the environment. They are used in the context of testing and evaluating artificial intelligence algorithms that aim to win the game or to behave like a real user player [6]. In this work, the feedback obtained from an observational human critic (OHC) in the form of electroencephalographic (EEG) signals is used to evaluate the operational performance of a gaming agent. OHCs are silent subjects observing a computer gaming agent playing the game.

The feasibility of a distinct nonbiological communication channel between the central nervous system and a computer device has been previously proven with brain–computer interfaces (BCI) or brain–machine interfaces [7]. BCI systems provide a new input modality that can be used in computer games [8]–[10]. This advancement is relevant in the context of the accessibility for video games [11] and the growing area of e-sports [12].

In this study, gaming agents are trained using only signal components called error-related potentials (ErrPs) that can be identified in the observer's brainwaves. These types of signals can be found on EEG traces and are elicited when subjects are aware of the presence of an unexpected outcome, which they identify as an error. The analysis of ErrP signals is currently an extensive area of research in the neuroscience community [13]. ErrPs can be detected by signal processing and machine learning techniques [14] and are also used in BCIs to implement or enhance artificial communication channels [15].

Given the scenario, reinforcement learning (RL) [16] stands out as a natural method to train the agent. RL refers to an algorithmic learning strategy inspired on how biological agents learn by exploring their environment while getting negative or positive feedback rewards. The method aims to maximize positive rewards while minimizing negative feedback. Thus, the learning problem is posed as a stochastic optimization strategy [17]. Recently, this technique has been used extensively in artificial intelligence [18]. The influence of DeepBrain's AlphaGo project cannot be neglected, since it was the first to reach a very high proficiency when it won the complex game Go against several world champions [19].

Previous research has explored the usage of RL with reward signals based on brain activity, recorded by an EEG-based BCI system during task execution. The papers [20]–[22] have successfully demonstrated that a robot can be controlled with brain signals from a person who is observing a robot to solve a task. Moreover, a growing number of studies have demonstrated the feasibility of using ErrPs as rewards for RL schemes such as to enhance robotic behavior [23], to assess air traffic controller's decisions [24] or to categorize actions as errors [25]. Other approaches have used these signals as important feedback for human–robot interaction or to implement shared-control strategies [26], [27]. Additionally, ErrPs have also been used

Manuscript received December 9, 2019; revised August 25, 2020 and October 21, 2020; accepted December 2, 2020. This work was supported by the Grant ITBACyT-15 issued by ITBA University. *(Bartolomé Francisco, Moreno Juan, Navas Natalia, and Vitali José contributed equally to this work.) (Corresponding author: Ramele Rodrigo.)*

The authors are with the Department of Computer Engineering, Instituto Tecnológico de Buenos Aires, Buenos Aires 1106, Argentina (e-mail: fbartolo@itba.edu.ar; jumoreno@itba.edu.ar; nnavas@itba.edu.ar; jvitali@itba.edu.ar; rramele@itba.edu.ar; jsantos@itba.edu.ar).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TG.2020.3042900.

Digital Object Identifier 10.1109/TG.2020.3042900

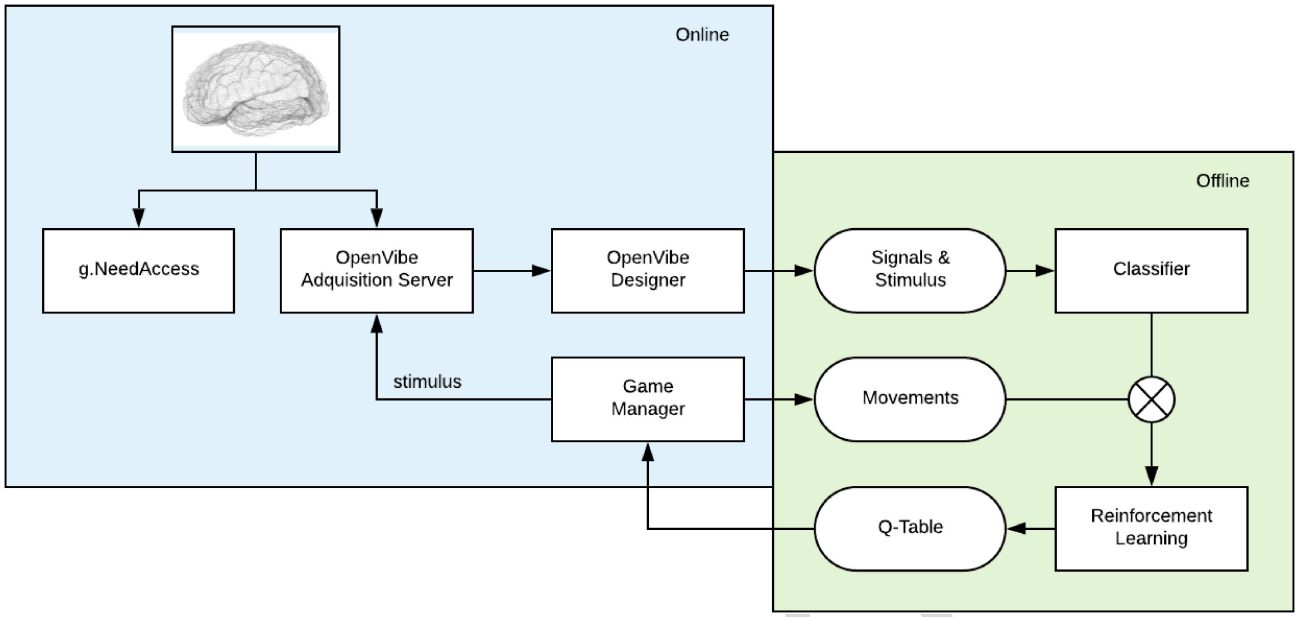

Fig. 1. Overview of the experimental procedure. Brainwaves are obtained by the OpenVibe Acquisition Server. The Game Manager is responsible for generating the game screen, the game mechanics, and the game movements performed by the gaming agent. It is also connected to the Acquisition Server to send stimulus information. The captured information is stored by the OpenVibe Designer. Offline, EEG signals are classified and they are linked to each game movement calculated by the Game Manager to determine proper rewards for each action. This information is used by an RL algorithm that iteratively trains a Q-Table in order to improve the performance of the agent that plays the game.

in games as an additional feedback channel that can be explored to improve gaming experience [28], [29].

Therefore, we aim to use the information extracted from brainwaves to enhance the performance of a gaming agent. The three contributions are as follows:

1) a simple game mechanics and agent that can elicit the ErrP potential;
2) results that confirm that even when ErrP classification accuracy is low but with a high specificity, enough information is generated for an agent to learn the optimal policy and solve a simple game;
3) collaborative rewards from multiple observational human observers can compensate the lack of classification accuracy or the inefficacy of transfer learning procedures for brainwaves signals.

This work unfolds as follows. In Section II, the general layout of the cognitive game is described. Sections II-A and II-B outline the cognitive game procedure used to obtain rewards in the form of ErrP components. Section II-F describes the gaming agent learning procedure. Results are described in Sections III. Section IV concludes this article.

II. Materials and Methods

The experimental procedure is summarized in Fig. 1. The proposed system has two distinct parts. This first part consists of the collection of brainwave signals from an OHC that is watching an agent play a game. The agent knows the game rules but not how to win it. The second part, the gaming agent learning phase, is where the agent can learn the winning strategy using the OHC's feedback to improve its own performance.

A. Brainwave Session

The retrieval of the OHC's brain activity, called the brainwave session, is one of the most critical parts of the study. Subjects are recruited voluntarily and given a form with questions regarding their health (previous health issues and particular visual sensitivity), habits (sleeping hours, caffeine and alcohol consumption), and a written informed consent petition to collect the required data. The experiment is conducted anonymously in accordance with the Declaration of Helsinki published by the World Health Organization. No monetary compensation is handed out. This study is approved by the Departamento de Investigación y Doctorado, Instituto Tecnológico de Buenos Aires. The brainwave sessions are performed with eight subjects, five males, and three females, with an average age of 25.12 years, a standard deviation of 1.54 years, and a range of 22–28 years. All subjects have normal vision, are right-handed and no history of neurological disorders.

After the form is filled out, a short description of the procedure is given to each subject. They are only told that the objective of the agent is to reach the goal and the four movements that the agent can make. When this concludes, the subject is introduced to the wireless digital EEG device (g.Nautilus, g.Tec, Austria) that she/he has to wear during the brainwave session. It has eight electrodes (g.LADYbird, g.Tec, Austria) on the positions Fz, Cz, Pz, Oz, P3, P4, PO7, and PO8, identified according to the 10–20 international system, with a reference set to the right ear lobe and ground set as the AFz position. The electrode contact points are adjusted applying conductive gel until the impedance values displayed by the program g.NeedAccess (g.Tec, Austria) are within the desired range. This process takes between 10–15 min.

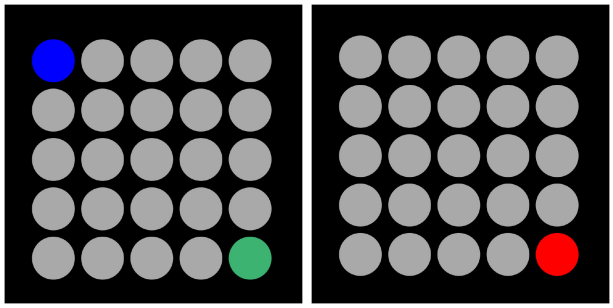

Fig. 2. Grid system representation used in the cognitive game. The blue spot represents the initial location, whereas the green spot represents the target location. Once the agent reaches the target spot, its color turns red to indicate the end of the play.

After this step, the subject is instructed to close their eyes, make eye movements and muscle chew in order to check the program and guarantee that the live channel values are accurate.

Once the headset is correctly applied, the OpenVibe Acquisition Server program, from the OpenVibe platform [30], is launched and configured with a sampling rate of 250 Hz. A 50-Hz notch filter is applied to filter out power line noise. An additional bandpass filter between 0.5 and 60 Hz is applied. Data are handled and processed with the OpenVibe Designer, from the same platform, using 8 channels for the brain data (one channel per electrode) and an additional channel to record the stimulus, which corresponds to a game movement performed by the agent. After everything is connected, the subject is seated in a comfortable chair in front of a computer screen. The brightness of the screen is set to the maximum setting to avoid any visual inconvenience in which the subject cannot distinguish the components of the game that appear on the screen.

The Acquisition Server receives and synchronizes the signal data from the headset and any event information from the game, and transfers it to the OpenVibe Designer application. When the subject is ready, the Game Manager and the OpenVibe Designer programs are launched and configured to communicate with the previously mentioned Acquisition Server. A brainwave session consists of several matches, each one being a gameplay. In the end, the sequence of game movements and the signal data generated for each match are saved for offline processing.[1]

B. Cognitive Game Procedure

The game parsimoniously consists of a $5x5$ grid of gray circular spots with a black background. It is similar to the one proposed by Iturrate et al. [27]. A blue spot indicates the current position of the agent and a green spot represents the goal, as shown in Fig. 2. The agent's objective is to reach the goal. The circular spot representing the goal remains static at the bottom-right position of the grid, whereas the one representing the position of the agent starts at the upper left position of the grid and moves in each iteration. When the agent reaches the goal, the position where the agent and the goal are located turns red, showing that the match has ended. There are four possible movements that the agent can perform: it can go upwards, downwards, toward the left and the right, and those movements are bounded to avoid the agent from leaving the grid. The movement direction is selected randomly and is executed once every 2 s. After each gameplay, there is a pause of 5 s until the next match starts. Each time an agent moves, the Game Manager program sends an event marker to the Acquisition Server. This is considered a stimulus to the OHC. The game is designed as to be evident whenever there is an error (i.e., the agent moves away from the objective) so the subject can notice it immediately after the stimulus is presented, possibly triggering the expected cognitive response, which can be imprinted as an ErrP component within the EEG stream.

[1]The brainwave data set has been published on the IEEE DataPort initiative [31].

C. Signal Processing, Segmentation, and Classification

To aid the detection of the ErrP response, an offline processing pipeline and classifier is constructed to identify whether the action taken by the agent is an error or not, from the human observer's point of view. It is developed in Python using the "MNE" software platform [32], which is a package designed specifically for processing EEG and magnetoencephalography data, and built upon the machine learning library Scikit-Learn [33].

This pipeline consists of the offline processing of the collected signals used to train a classifier that can decide whether an error potential is triggered. First, the output of a brainwave session is read and an additional band-pass filter of 0.1–20.0 Hz is applied to the signal. Samples that correspond to the start of an event are tagged using the data from the stimulus channel.

After the raw data are loaded and tagged, epochs are extracted. Epochs consist of all the sample points that take place during the 2 s from the start of the event, 2 s corresponding to the time it takes for each action to take place, resulting in 500 samples per channel, as the sampling frequency is 250 Hz. Thus, each epoch is composed of a matrix 500×8 channels.

Samples that do not correspond to an epoch (located beyond the 2 s frame after the onset of the event) are not used. Also, epochs referring to the start or finish of each match are excluded.

In this way, the raw data of a brainwave session are processed into an array of matches where each element is an array of epochs tagged with a number specifying the prediction of the classifier, i.e., if the epoch corresponds to an action that made the agent move further from the goal (hit) or an action that made the agent move closer to the goal (no-hit). The ErrP is expected to be found in hits. To get the data ready for classification, the stimulus channel is removed to classify the signals using only the EEG data. Each epoch is regularized using a MinMaxScaler, i.e., subtracting the minimum value in the epoch and dividing by the signal peak-to-peak amplitude [34]. The eight channels are concatenated using the MNE Vectorizer function, which transforms the data matrix into a single array sample. Lastly, these data are used by the classification module as information to train and test a classifier. Five different classification algorithms are used: logistic regression, multilayer perceptron with a hidden layer of 100 neurons (i.e., default values for the Scikit-Learn

MLPClassifier), random forest, *K*-neighbors with $k = 3$ and finally a linear kernel support vector classifier [35].

D. Reinforcement Learning

Each match consists of a list of game movement configurations and the associated epochs obtained from OHC's brainwaves. The set of matches of each OHC is split into training and testing. Training matches are used to train the classifier to identify the ErrP signal. After a classifier is trained, the epochs extracted from the test matches are classified as hit or no-hit. A reward for each movement in the match is generated based on the prediction from the classifier for that movement. The reward can either be -1 when the event is classified as a hit or 0 when it is classified as a no-hit. The accuracy of these rewards depends on the performance of the classifier. The list of game movements and their associated reward information is used to train the agent by a variant of RL called Q-Learning algorithm.

E. Q-Learning

Q-Learning [36] is a form of model-free RL, where an agent tries an action at a particular state and evaluates its consequences in terms of the reward or penalty it receives. To represent rewards, a matrix $Q(s, a)$ is used, where rows correspond to all the possible states, and columns represent all possible actions. This matrix is known as the Q-Table. The algorithm proceeds by randomly choosing what action to do and iteratively updating the Q-Table based on the received reward r by the following equation:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma * \max_{\tilde{a}} Q(\tilde{s},\tilde{a}) - Q(s,a)] \quad (1)$$

where s is the current state, a the action, α the learning rate, and γ the discount factor, a value between 0 and 1 that determines the importance of long term results versus immediate rewards. Hence, $Q(s, a)$ is the expected value of the sum of discounted rewards that the agent will receive if in the s state, it takes the action a according to this policy. Once the environment has been extensively explored and the Q-Table has been optimized, the action chosen for a given state is the one that maximizes the expected reward according to the Q-Table matrix.

The algorithm is developed in Python and uses the OpenAI Gym toolkit [37]. Gym is a toolkit for developing and comparing RL algorithms. It makes no assumptions about the structure of an agent, and is compatible with any numerical computation library, such as TensorFlow or Theano [38].

F. Gaming Agent Learning Procedure

The gaming agent learning procedure uses the testing matches from brainwave sessions produced during the cognitive game procedure phase, and their components are schematized in Fig. 1.

This phase is divided into a sequence of run sessions and gaming agent training matches. During the run session, the agent plays 200 matches guided by a specific Q-Table with a 5% chance of randomly selecting a movement, to reduce deadlocks and loops. Following the run session, the agent performs a single gaming agent training match. The gaming agent starts first with a Q-Table initialized with zeros, so the initial policy for the agent is randomized. For the agent to learn from the feedback generated by the OHC, movement actions are determined by the replay of the agent's actions that were taken during one brainwave session match, in an offline RL scheme [39]. This allows the Q-Table to be learned based on the OHC's feedback from the movements the agent took, which were executed pseudorandomly during the brainwave session. The previously mentioned feedback is not explicit as it comes from the interpreted brain signal data. This implies that the reward is determined by the OHC's brain activity.

Hence, following the iterative procedure based on (1), the Q-Table is updated in each gaming agent training match. After the algorithm finishes replicating all the steps from the brainwave session match, the Q-Table is stored and used by the agent in the next run session.

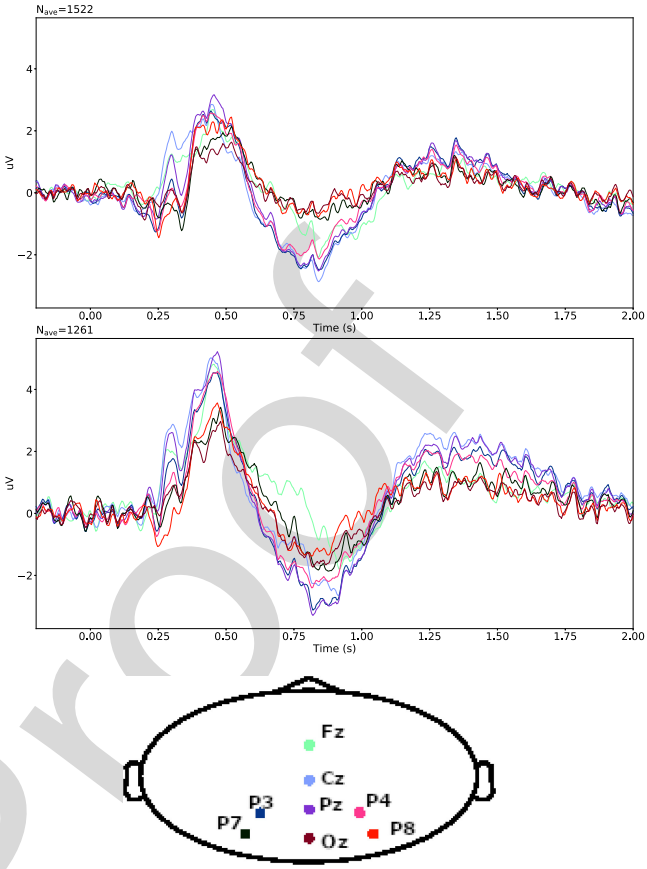

Fig. 3. Grand average of 2-s time-locked segments for all OHCs for the "move closer" (top) and "move further" (middle) condition. Zero time on *x*-axis corresponds to the onset of the stimulus, i.e., when the gaming agent moves. The increased height of the first peak around 0.4 s reflects the ErrP response particularly on Pz, based on the electrode layout (bottom).

III. Results

Grand averaged time-locked signal segments of 2-s length for all the OHCs can be seen in Fig. 3. The ErrP can be noticed more clearly on parieto-central areas (Pz electrode), around 0.4 s with a more prominent positive peak.

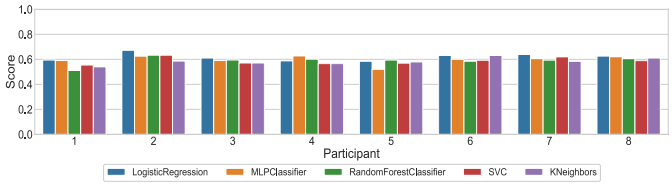

Fig. 4. Binary single trial classification score using five different classifiers while recognizing ErrP potentials for the eight OHCs. Chance level is 0.5.

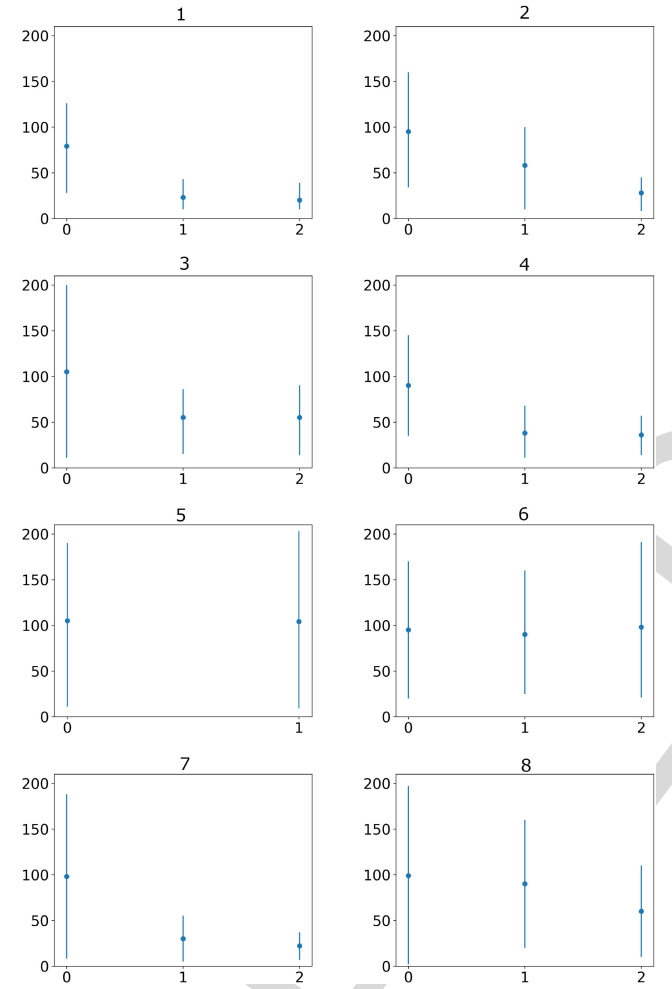

Fig. 5. Average number of steps for the agent to reach the goal when trained with rewards generated from brainwaves from OHCs 1–8. *Y*-axis shows the averaged number of steps for a run session, whereas *x*-axis shows the number of game matches used to cumulative train the Q-Table.

Fig. 4, on the other hand, shows the binary classification accuracy obtained for the eight OHCs using five different classification algorithms and using a tenfold cross-validation procedure. The best overall performance is obtained using logistic regression.

Complementary, Fig. 5 shows the average amount of steps it takes for the agent to reach the goal for each OHC, as the Q-Table is progressively trained using the reward information obtained from the prediction of the trained classifier. Each point corresponds to a run session of 200 game play repetitions. The

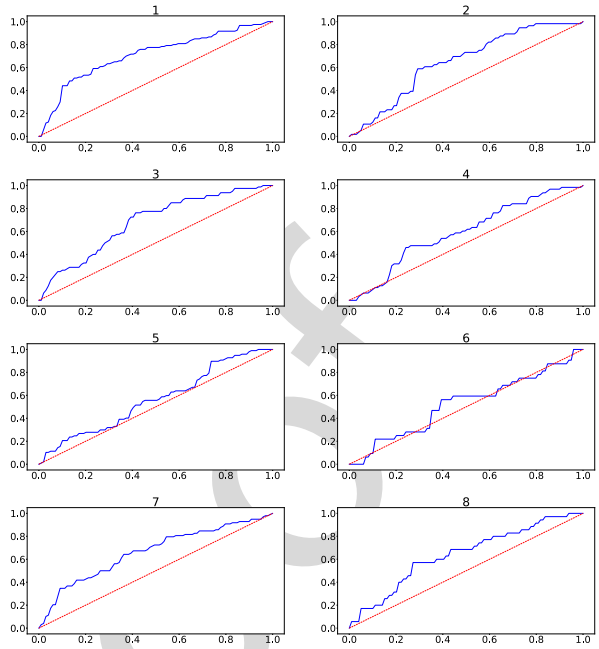

Fig. 6. ROC curves for OHCs 1–8. True positive rate is on the vertical axis and false positive rate on the horizontal axis. The ROC curves for ErrP identification for subjects 5 and 6 show low classification scores.

y-values represent the average number of steps the agent takes to reach the goal using a specific Q-Table for a run session. The first point, at the *x*-value 0, represents the number of steps the agent takes to reach the goal with an untrained Q-Table, where movements are decided randomly. The next point corresponds to the amount of steps it takes to reach the goal using a policy derived from a Q-Table trained after one brainwave session match, and so on.

The results show that as the Q-Table is progressively trained the average amount of steps decreases, meaning that the agent learns. However, the rate at which it learns varies per OHC, depending on the classification accuracy of the extracted brainwaves. For example, results for OHC 1 show faster learning than those of OHC 8 (see Fig. 5).

In the case for OHC 5 and 6, the reward information obtained from the brainwaves is not enough to train the agent effectively. Fig. 5 for OHC 5 and 6 shows no apparent learning, as the amount of steps to reach the goal does not decrease when trained. These results are also consistent with the classification ROC curves, shown in Fig. 6, where the area under the curve for OHCs 5 and 6 is close to chance level. Both OHCs have less recorded data from the sessions in comparison to the rest of the OHCs. This variation in performance for different OHCs has been studied extensively in BCI [15]. Besides low data samples, there are other reasons affecting the classification accuracy: cognitive reasons (i.e., the OHC not paying extensive attention to the game dynamics), very low SNR of the ErrP component or even the BCI-illiteracy phenomena where the specific OHC's signals do not contain the expected component response [40]. Fig. 7 shows the result of an agent successively trained with brainwave session matches where the EEG is generated with random signals. In this case,

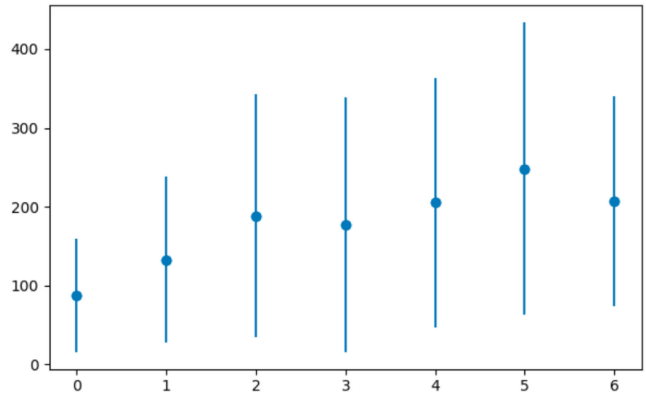

Fig. 7. Average number of steps for the agent to reach the goal when trained with a classifier produced from sham EEG signals. *X*-axis shows the number of gaming agent training matches used to train the Q-Table.

Fig. 8. Heatmap for the transfer learning experiment. Values represent the reduction in the average number of steps required to reach the goal. Negative values represent net improvements.

random EEG signals are generated using OpenVibe Acquisition Server signal generator for all channels, as if they were produced by an OHC who does not pay attention to the game. The agent learns nothing, and regardless of the number of matches that are used to learn the Q-Table, the number of steps required to reach the goal does not decrease. This pattern is also obtained when the game matches from OHCs 5 and 6 are used, showing that the reward labeling predicted by the trained classifier for those cases worked like a random classifier.

EEG signals have high intersubject variability [15]. This is evidenced in Fig. 8 where the agent training is performed with rewards obtained by classifying epochs from one Tester OHC with a classifier that was trained using the brainwaves from a different Trainer OHC. The figure shows the cumulative variation for all run sessions on the average number of steps required to reach the goal after training the agent with all the available matches from the brainwave session. Enhancements are shown as negative values. Only the diagonal of the heatmap matrix shows a clear improvement in terms of the reduction of the required number of steps to reach the goal (averaged per 200 runs) that corresponds to the same information for each OHC

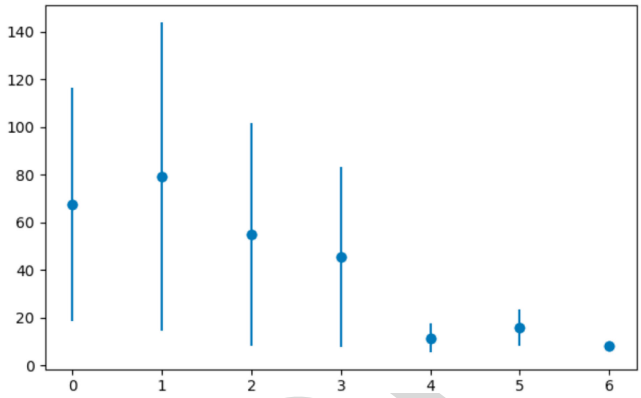

Fig. 9. Average steps using Q-Table trained with brainwave session matches from six different OHCs. *X*-axis shows the progressive number of gaming agent training matches used to train the Q-Table. These matches correspond to all subjects, excluding subjects 5 and 6, which individually show no significant learning progress.

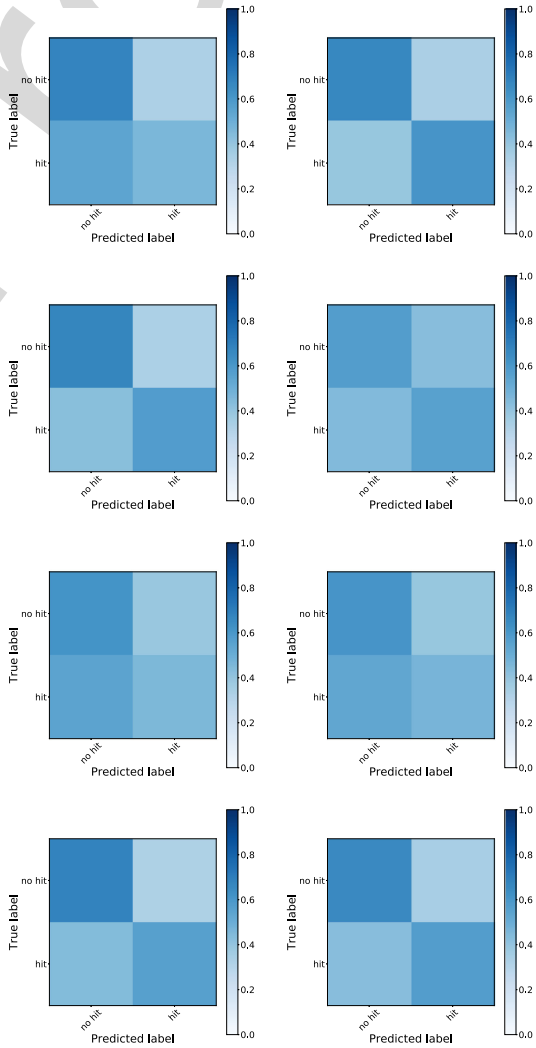

Fig. 10. Confusion matrix for OHCs 1–8. Darker colors show higher values. It can be seen in the lower percentage of false positives (upper right corner of each chart).

shown in Fig. 5. For this transfer learning experiment [41], no performance gain is evidenced, the agents learn nothing and this implies that the reward information provides no value.

Finally, Fig. 9 shows the result of training an agent with cumulative brainwave session matches from OHCs 1, 2, 3, 4, 7, and 8. It can be seen that the overall performance of the agent improves as long as there is information to produce rewards, regardless of the fact that they were generated from classifiers trained with different OHC's signals.

IV. Conclusion

Each time the gaming agent plays this simple game, it takes on average around 100 steps to reach the target spot. A classifier can be trained to recognize error potential from OHCs that observe the agent playing the game. The gaming agent plays randomly and movements are marked based on the identification of an error potential from the OHC. Those movements are used as rewards in an RL scheme to train a Q-Table. The gaming agent plays the game again, but this time the number of required steps to solve the game is reduced. If rewards are provided based on random signals, no reduction is achieved and the average number of steps does not change. This shows that there is an effective transfer of information from the brainwaves to the agent. As this process is repeated, the agent keeps improving solving the game effectively, i.e., performing the minimum number of required steps to reach the goal.

This work aims to propose a simple game mechanics that can use the ErrP component to train a gaming agent using an RL model. As described in the literature [15], ErrPs can be used to transmit subjective feedback to a computer. The collected data show that ErrP signals can in fact be classified and used to train an agent effectively. This proposal tries to keep the system as simple as possible, emphasizing information flow from the subjective error perception of the OHC.

One additional aspect to remark is the robustness of the learning strategy based on Q-Learning [42], [43]. The obtained accuracy to discriminate ErrPs is low, though on the same level to other similar results [27], [44]. In this regard, considering the variability of the ErrP response in terms of different cognitive experiments, levels of 90% accuracy in identifying it, are reported [15]. In this work, despite the low accuracy in ErrP identification, the RL algorithm was able to extract meaningful information from rewards that were helpful to improve, and often maximize, the agent's performance. Additionally, classification results show a low percentage of false positives (see Fig. 10), entailing a high specificity. On the other hand, the percentage of false negatives is generally higher. Even though this implies that the agent misses frequently when a wrong action takes place, this is not hindering the overall performance and the agent is still learning. Although they may be scarce, accurate rewards are very useful for the RL algorithm.

At the same time, effective agent training depends on the OHC's training data. Results confirm the futility or complexity of using Transfer Learning [41]: training a classifier with data obtained from one OHC, and using the same classifier to identify ErrPs for another OHC does not increase the performance of the agent. Despite that, the rewards generated from different subject's classifiers can be used to train the same Q-Table to improve its performance, which may lead to strategies where the overall performance is enhanced based on the information from different OHCs at the same time. We hypothesize that as long as there are accurate rewards obtained from high specificity classifiers, there is extra information that can be used by the agent to get an improved Q-Table. Additionally, it seems to be an agreement in terms of the subjective interpretation of what may be an appropriate movement to reach the goal, and different OHCs produce rewards for the same type of movements.

The simple setup of the grid-based game allows further experimentation, using the reduction on the number of average steps to reach the goal as a validation of the achieved information transfer. It will be of research interest to verify if the smooth progression toward the end alters the shape of the ErrP response, how the ErrP response is triggered in relation with different shapes and colors of the board markers [45], or if there is a differential ErrP signal component in relation to up, down, left and right movements. In addition, the outcome of manipulating the stimulus could be further studied as well as the influence on the results if incentives are given to participants.

Further work will be conducted in order to increase the complexity of the game to allow the possibility that the target position is dynamically changed. Although we found that the best performing classifier is logistic regression, there is room for improvement. The classifier could be enhanced to recognize the error potential [46] more effectively or could be pretrained to allow higher accuracy [47].

Acknowledgment

The authors would like to thank the Laboratory Centro de Inteligencia Computacional and ITBA University.

References

[1] D. Xu, M. Agarwal, F. Fekri, and R. Sivakumar, "Playing games with implicit human feedback," in *Proc. AAAI Workshop Reinforcement Learn. Games*, New York, NY, USA, 2020. [Online]. Available: https://www.semanticscholar.org/paper/Playing-Games-with-Implicit-Human-Feedback-Xu-Agarwal/faba141483180e53f461c6035ce95041cfed9a8f

[2] T. O. Zander, L. R. Krol, N. P. Birbaumer, and K. Gramann, "Neuroadaptive technology enables implicit cursor control based on medial prefrontal cortex activity," *Proc. Nat. Acad. Sci. USA*, vol. 113, no. 52, pp. 14898–14903, Dec. 2016. [Online]. Available: http://www.pnas.org/lookup/doi/10.1073/pnas.1605155114

[3] M. Carter, J. Downs, B. Nansen, M. Harrop, and M. Gibbs, "Paradigms of games research in HCI: A review of 10 years of research at CHI," in *Proc. Annu. Symp. Comput.–Hum. Interact. Play*, 2014, pp. 27–36. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2658537.2658708

[4] J. Frey, C. Mühl, F. Lotte, and M. Hachet, "Review of the use of electroencephalography as an evaluation method for human-computer interaction," in *Proc. Int. Conf. Physiol. Comput. Syst.*, Nov. 2014, pp. 214–223. [Online]. Available: http://arxiv.org/abs/1311.2222

[5] P. Barr, J. Noble, and R. Biddle, "Video game values: Human-computer interaction and games," *Interacting Comput.*, vol. 19, no. 2, pp. 180–195, Mar. 2007. [Online]. Available: https://academic.oup.com/iwc/article-lookup/doi/10.1016/j.intcom.2006.0 8.008

[6] Y. Zhao *et al.*, "Winning is not everything: Enhancing game development with intelligent agents," *IEEE Trans. Games*, vol. 12, no. 2, pp. 199–212, Jun. 2020.

[7] G. A. M. Vasiljevic and L. C. de Miranda, "Brain–Computer interface games based on consumer-grade EEG devices: A systematic literature review," *Int. J. Hum.–Comput. Interact.*, vol. 36, no. 2, pp. 105–142, Jan. 2020. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/10447318.2019.1612213

Q4

[8] R. Scherer, M. Pröll, B. Allison, and G. R. Müller-Putz, "New input modalities for modern game design and virtual embodiment," in *Proc. IEEE Virtual Reality*, Mar. 2012, pp. 163–164.

[9] D. Marshall, D. Coyle, S. Wilson, and M. Callaghan, "Games, gameplay, and BCI: The state of the art," *IEEE Trans. Comput. Intell. AI Games*, vol. 5, no. 2, pp. 82–99, Jun. 2013.

[10] A. Nijholt and D. Tan, "Playing with your brain: Brain-computer interfaces and games," in *Proc. ACM Int. Conf. Proc. Ser.*, 2007, vol. 203, pp. 305–306. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1255047.1255140

[11] J. Aguado-Delgado, J. M. Gutiérrez-Martínez, J. R. Hilera, L. De-Marcos, and S. Otón, "Accessibility in video games: A systematic review," *Universal Access Inf. Soc.*, vol. 19, no. 1, pp. 169–193, Mar. 2020. [Online]. Available: http://link.springer.com/10.1007/s10209-018-0628-2

[12] L. Yakovlev, N. Syrov, G. Nikolai, and A. Kaplan, "BCI-controlled motor imagery training can improve performance in e-sports," in *Proc. Int. Conf. Hum.–Comput. Interact.*, Jul. 2020, pp. 581–586. [Online]. Available: http://link.springer.com/10.1007/978-3-030-50726-8_76

[13] C. B. Holroyd, O. E. Krigolson, R. Baker, S. Lee, and J. Gibson, "When is an error not a prediction error? An electrophysiological investigation," *Cogn., Affect. Behav. Neurosci.*, vol. 9, no. 1, pp. 59–70, Mar. 2009. [Online]. Available: http://www.springerlink.com/index/10.3758/CABN.9.1.59

[14] G. Dornhege, *Toward Brain-Computer Interfacing*. Cambridge, MA, USA: MIT Press, 2007. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6281216

[15] R. Chavarriaga, A. Sobolewski, and J. d. R. Millán, "Errare machinale est: The use of error-related potentials in brain-machine interfaces," *Frontiers Neurosci.*, vol. 8, p. 208, Jul. 2014. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2014.00208/abstract

[16] R. S. Sutton and A. G. Barto, *Reinforcement Learning, Second Edition: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[17] J. M. Santos and C. Touzet, "Exploration tuned reinforcement function," *Neurocomputing*, vol. 28, no. 1–3, pp. 93–105, Oct. 1999. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231298001179

[18] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 3826–3839, Sep. 2020.

[19] D. Silver *et al.*, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, Oct. 2017. [Online]. Available: http://www.nature.com/articles/nature24270

[20] I. Iturrate, L. Montesano, and J. Mínguez, "Robot reinforcement learning using EEG-based reward signals," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 4822–4829.

[21] S. K. Kim, E. A. Kirchner, A. Stefes, and F. Kirchner, "Intrinsic interactive reinforcement learning—Using error-related potentials for real world human-robot interaction," *Sci. Rep.*, vol. 7, no. 1, Dec. 2017, Art. no. 17562. [Online]. Available: http://www.nature.com/articles/s41598-017-17682-7

[22] J. Omedes, I. Iturrate, L. Montesano, and J. Minguez, "Using frequency-domain features for the generalization of EEG error-related potentials among different tasks," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Jul. 2013, pp. 5263–5266.

[23] T. J. Luo, Y. C. Fan, J. T. Lv, and C. L. Zhou, "Deep reinforcement learning from error-related potentials via an EEG-based brain-computer interface," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, Dec. 2019, pp. 697–701.

[24] S. K. Goh, N. P. Tran, D. T. Pham, S. Alam, K. Izzetoglu, and V. Duong, "Construction of air traffic controller's decision network using error-related potential," in *Augmented Cognition* (Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)). Cham, Switzerland: Springer, Jul. 2019, vol. 11580, pp. 384–393. [Online]. Available: http://link.springer.com/10.1007/978-3-030-22419-6_27

[25] C. Wirth, P. M. Dockree, S. Harty, E. Lacey, and M. Arvaneh, "Towards error categorisation in BCI: Single-trial EEG classification between different errors," *J. Neural Eng.*, vol. 17, no. 1, Dec. 2020, Art. no. 016008. [Online]. Available: https://iopscience.iop.org/article/10.1088/1741-2552/ab53fe

[26] L. Schiatti, J. Tessadori, N. Deshpande, G. Barresi, L. C. King, and L. S. Mattos, "Human in the loop of robot learning: EEG-based reward signal for target identification and reaching task," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2018, pp. 4473–4480.

[27] I. Iturrate, L. Montesano, and J. Minguez, "Shared-control brain-computer interface for a two dimensional reaching task using EEG error-related potentials," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2013, pp. 5258–5262.

[28] D. Plass-Oude Bos *et al.*, "Brain-computer interfacing and games," in *Brain–Computer Interfaces, Applying Our Minds to Human Computer Interaction.* London, U.K.: Springer, 2010, pp. 149–178. [Online]. Available: http://link.springer.com/10.1007/978-1-84996-272-8_10

[29] S. E. Kober, M. Ninaus, E. V. Friedrich, and R. Scherer, "BCI and games: Playful, experience-oriented learning by vivid feedback?" in *Brain–Computer Interfaces Handbook*. Boca Raton, FL, USA: CRC Press, 2018, pp. 209–234.

[30] Y. Renard *et al.*, "OpenViBE: An open-source software platform to design, test, and use brain–computer interfaces in real and virtual environments," *Presence, Teleoperators Virtual Environ.*, vol. 19, no. 1, pp. 35–53, Feb. 2010. [Online]. Available: http://www.mitpressjournals.org/doi/10.1162/pres.19.1.35

[31] F. Bartolomé, J. Moreno, N. Navas, and J. Vitali, "ERRP-Dataset," 2019. [Online]. Available: http://dx.doi.org/10.21227/6emh-wb46

[32] A. Gramfort *et al.*, "MEG and EEG data analysis with MNE-Python," *Frontiers Neurosci.*, vol. 7, p. 267, Dec. 2013. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2013.00267/abstract

[33] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[34] T. Zhou and J. P. Wachs, "Spiking neural networks for early prediction in human–robot collaboration," *Int. J. Robot. Res.*, vol. 38, no. 14, pp. 1619–1643, Dec. 2019. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364919872252

[35] F. Lotte *et al.*, "A review of classification algorithms for EEG-based brain-computer interfaces: A 10 year update," *J. Neural Eng.*, vol. 15, no. 3, Jun. 2018, Art. no. 031005. [Online]. Available: http://stacks.iop.org/1741-2552/15/i=3/a=031005?key=crossref.9cd2b15ab6 5c8ad34b475584b43dc509

[36] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3/4, pp. 279–292, May 1992. [Online]. Available: http://link.springer.com/10.1007/BF00992698

[37] G. Brockman *et al.*, "OpenAI gym," 2016, *arXiv:1606.01540*.

[38] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: http://tensorflow.org/

[39] S. Levine, A. Kumar, G. Tucker, and J. Fu, "Offline reinforcement learning: Tutorial, review, and perspectives on open problems," May 2020, *arXiv:2005.01643*.

[40] R. Yousefi, A. Rezazadeh Sereshkeh, and T. Chau, "Development of a robust asynchronous brain-switch using ErrP-based error correction," *J. Neural Eng.*, vol. 16, no. 6, Nov. 2019, Art. no. 066042. [Online]. Available: https://iopscience.iop.org/article/10.1088/1741-2552/ab4943

[41] D. Wu, Y. Xu, and B. Lu, "Transfer learning for EEG-based brain-computer interfaces: A review of progress made since 2016," *IEEE Trans. Cogn. Develop. Syst.*, to be published.

[42] R. Bauer and A. Gharabaghi, "Reinforcement learning for adaptive threshold control of restorative brain-computer interfaces: A Bayesian simulation," *Frontiers Neurosci.*, vol. 9, p. 36, Feb. 2015. [Online]. Available: http://journal.frontiersin.org/Article/10.3389/fnins.2015.00036/abstract

[43] J. Rubin, O. Shamir, and N. Tishby, "Trading value and information in MDPs," in *Intelligent Systems Reference Library*. Berlin, Germany: Springer, 2012, vol. 28, pp. 57–74. [Online]. Available: http://link.springer.com/10.1007/978-3-642-24647-0_3

[44] S. Ehrlich and G. Cheng, "A neuro-based method for detecting context-dependent erroneous robot action," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Dec. 2016, pp. 477–482.

[45] M. Eimer, "An event-related potential (ERP) study of transient and sustained visual attention to color and form," *Biol. Psychol.*, vol. 44, no. 3, pp. 143–160, 1997. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0301051196052179

[46] F. Iwane, R. Chavarriaga, I. Iturrate, and J. Del Millan, "Spatial filters yield stable features for error-related potentials across conditions," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.* Oct. 2017, pp. 661–666.

[47] M. Spüler, M. Bensch, S. Kleih, W. Rosenstiel, M. Bogdan, and A. Kübler, "Online use of error-related potentials in healthy users and people with severe motor impairment increases performance of a P300-BCI," *Clin. Neurophysiol.*, vol. 123, no. 7, pp. 1328–1337, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1388245711009059

GENERAL INSTRUCTION

- **Authors: Carefully check the page proofs (and coordinate with all authors); additional changes or updates WILL NOT be accepted after the article is published online/print in its final form. Please check author names and affiliations, funding, as well as the overall article for any errors prior to sending in your author proof corrections. Your article has been peer reviewed, accepted as final, and sent in to IEEE. No text changes have been made to the main part of the article as dictated by the editorial level of service for your publication.**
- **Authors: We cannot accept new source files as corrections for your article. If possible, please annotate the PDF proof we have sent you with your corrections and upload it via the Author Gateway. Alternatively, you may send us your corrections in list format. You may also upload revised graphics via the Author Gateway.**

QUERIES

Q1. Author: Please confirm or add details for any funding or financial support for the research of this article.
Q2. Author: Please check and confirm whether the author affiliations in the first footnote are correct as set.
Q3. Author: Please provide the expansion for the acronyms AI and ROC at the instance when they are first mentioned in the text.
Q4. Author: Please provide the page range in Ref. [1].
Q5. Author: Please provide the complete page range in Refs. [15], [32], and [42].
Q6. Author: Please update Refs. [37], [39], and [41].

Training a Gaming Agent on Brainwaves

Bartolomé Francisco, Moreno Juan, Navas Natalia, Vitali José, Ramele Rodrigo, *Member, IEEE*, and Santos Juan Miguel

***Abstract*—Error-related potentials (ErrPs) are a particular type of event-related potential elicited by a person attending a recognizable error. These electroencephalographic signals can be used to train a gaming agent by a reinforcement learning algorithm to learn an optimal policy. The experimental process consists of an observational human critic (OHC) observing a simple game scenario while their brain signals are captured. The game consists of a grid, where a blue spot has to reach a desired target in the fewest amount of steps. Results show that there is an effective transfer of information and that the agent successfully learns to solve the game efficiently, from the initial 97 steps on average required to reach the target to the optimal number of eight steps. Our results are expressed in threefold: the mechanics of a simple grid-based game that can elicit the ErrP signal component; the verification that the reward function only penalizes wrong steps, which means that type II error (not properly identifying a wrong movement) does not affect significantly the agent learning process; collaborative rewards from multiple OHCs can be used to train the algorithm effectively and can compensate low classification accuracies and a limited scope of transfer learning schemes.**

***Index Terms*—Agent, AI, brain–computer interface (BCI), electroencephalographic (EEG), error-related potential (ErrP), reinforcement learning (RL).**

I. Introduction

THE effectiveness of today's human–machine interaction and artificial intelligence is limited by a communication bottleneck, as humans are required to translate high-level concepts into a machine-mandated sequence of instructions [1], [2]. Hence, new interaction methods are required to increase the communication bandwidth between computers and humans or to produce alternative communications systems to increase the efficiency of this channel. In this respect, video games have been widely used as test tools to assess new means of interactions [3]–[5]. Video gaming agents are computer programs that can sense the computer game environment, process information, and react accordingly within the environment. They are used in the context of testing and evaluating artificial intelligence algorithms that aim to win the game or to behave like a real user player [6]. In this work, the feedback obtained from an observational human critic (OHC) in the form of electroencephalographic (EEG) signals is used to evaluate the operational performance of a gaming agent. OHCs are silent subjects observing a computer gaming agent playing the game.

The feasibility of a distinct nonbiological communication channel between the central nervous system and a computer device has been previously proven with brain–computer interfaces (BCI) or brain–machine interfaces [7]. BCI systems provide a new input modality that can be used in computer games [8]–[10]. This advancement is relevant in the context of the accessibility for video games [11] and the growing area of e-sports [12].

In this study, gaming agents are trained using only signal components called error-related potentials (ErrPs) that can be identified in the observer's brainwaves. These types of signals can be found on EEG traces and are elicited when subjects are aware of the presence of an unexpected outcome, which they identify as an error. The analysis of ErrP signals is currently an extensive area of research in the neuroscience community [13]. ErrPs can be detected by signal processing and machine learning techniques [14] and are also used in BCIs to implement or enhance artificial communication channels [15].

Given the scenario, reinforcement learning (RL) [16] stands out as a natural method to train the agent. RL refers to an algorithmic learning strategy inspired on how biological agents learn by exploring their environment while getting negative or positive feedback rewards. The method aims to maximize positive rewards while minimizing negative feedback. Thus, the learning problem is posed as a stochastic optimization strategy [17]. Recently, this technique has been used extensively in artificial intelligence [18]. The influence of DeepBrain's AlphaGo project cannot be neglected, since it was the first to reach a very high proficiency when it won the complex game Go against several world champions [19].

Previous research has explored the usage of RL with reward signals based on brain activity, recorded by an EEG-based BCI system during task execution. The papers [20]–[22] have successfully demonstrated that a robot can be controlled with brain signals from a person who is observing a robot to solve a task. Moreover, a growing number of studies have demonstrated the feasibility of using ErrPs as rewards for RL schemes such as to enhance robotic behavior [23], to assess air traffic controller's decisions [24] or to categorize actions as errors [25]. Other approaches have used these signals as important feedback for human–robot interaction or to implement shared-control strategies [26], [27]. Additionally, ErrPs have also been used

Manuscript received December 9, 2019; revised August 25, 2020 and October 21, 2020; accepted December 2, 2020. This work was supported by the Grant ITBACyT-15 issued by ITBA University. *(Bartolomé Francisco, Moreno Juan, Navas Natalia, and Vitali José contributed equally to this work.) (Corresponding author: Ramele Rodrigo.)*

The authors are with the Department of Computer Engineering, Instituto Tecnológico de Buenos Aires, Buenos Aires 1106, Argentina (e-mail: fbartolo@itba.edu.ar; jumoreno@itba.edu.ar; nnavas@itba.edu.ar; jvitali@itba.edu.ar; rramele@itba.edu.ar; jsantos@itba.edu.ar).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TG.2020.3042900.

Digital Object Identifier 10.1109/TG.2020.3042900

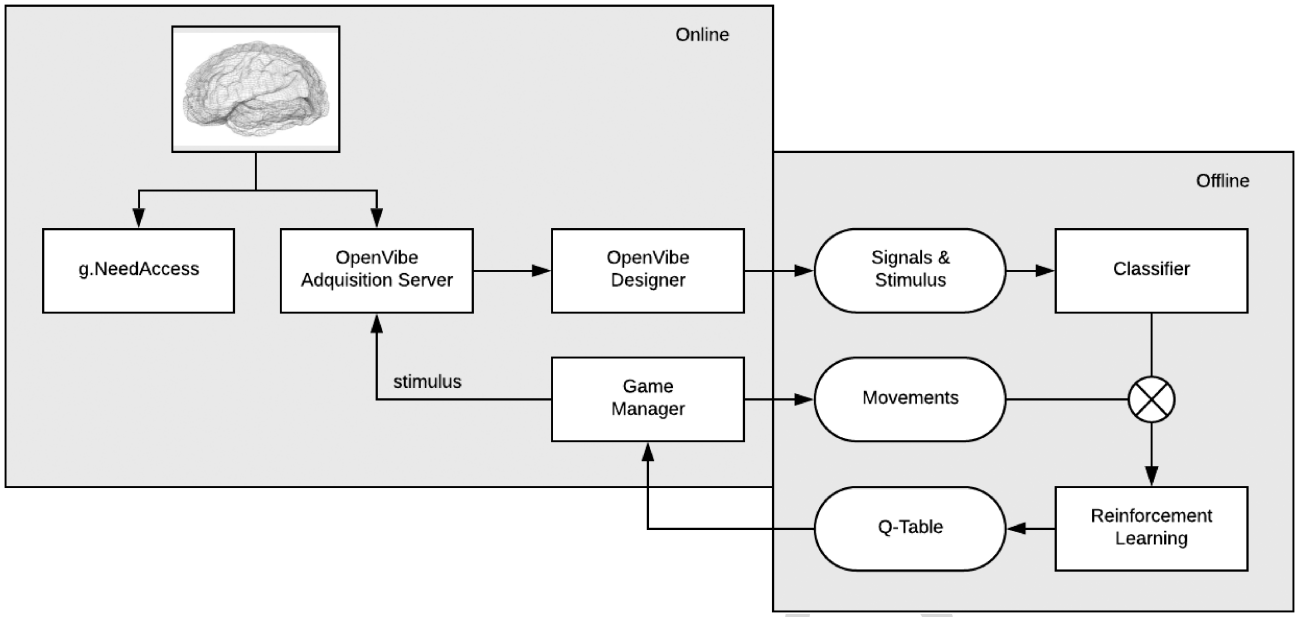

Fig. 1. Overview of the experimental procedure. Brainwaves are obtained by the OpenVibe Acquisition Server. The Game Manager is responsible for generating the game screen, the game mechanics, and the game movements performed by the gaming agent. It is also connected to the Acquisition Server to send stimulus information. The captured information is stored by the OpenVibe Designer. Offline, EEG signals are classified and they are linked to each game movement calculated by the Game Manager to determine proper rewards for each action. This information is used by an RL algorithm that iteratively trains a Q-Table in order to improve the performance of the agent that plays the game.

in games as an additional feedback channel that can be explored to improve gaming experience [28], [29].

Therefore, we aim to use the information extracted from brainwaves to enhance the performance of a gaming agent. The three contributions are as follows:

1) a simple game mechanics and agent that can elicit the ErrP potential;
2) results that confirm that even when ErrP classification accuracy is low but with a high specificity, enough information is generated for an agent to learn the optimal policy and solve a simple game;
3) collaborative rewards from multiple observational human observers can compensate the lack of classification accuracy or the inefficacy of transfer learning procedures for brainwaves signals.

This work unfolds as follows. In Section II, the general layout of the cognitive game is described. Sections II-A and II-B outline the cognitive game procedure used to obtain rewards in the form of ErrP components. Section II-F describes the gaming agent learning procedure. Results are described in Sections III. Section IV concludes this article.

II. Materials and Methods

The experimental procedure is summarized in Fig. 1. The proposed system has two distinct parts. This first part consists of the collection of brainwave signals from an OHC that is watching an agent play a game. The agent knows the game rules but not how to win it. The second part, the gaming agent learning phase, is where the agent can learn the winning strategy using the OHC's feedback to improve its own performance.

A. Brainwave Session

The retrieval of the OHC's brain activity, called the brainwave session, is one of the most critical parts of the study. Subjects are recruited voluntarily and given a form with questions regarding their health (previous health issues and particular visual sensitivity), habits (sleeping hours, caffeine and alcohol consumption), and a written informed consent petition to collect the required data. The experiment is conducted anonymously in accordance with the Declaration of Helsinki published by the World Health Organization. No monetary compensation is handed out. This study is approved by the Departamento de Investigación y Doctorado, Instituto Tecnológico de Buenos Aires. The brainwave sessions are performed with eight subjects, five males, and three females, with an average age of 25.12 years, a standard deviation of 1.54 years, and a range of 22–28 years. All subjects have normal vision, are right-handed and no history of neurological disorders.

After the form is filled out, a short description of the procedure is given to each subject. They are only told that the objective of the agent is to reach the goal and the four movements that the agent can make. When this concludes, the subject is introduced to the wireless digital EEG device (g.Nautilus, g.Tec, Austria) that she/he has to wear during the brainwave session. It has eight electrodes (g.LADYbird, g.Tec, Austria) on the positions Fz, Cz, Pz, Oz, P3, P4, PO7, and PO8, identified according to the 10–20 international system, with a reference set to the right ear lobe and ground set as the AFz position. The electrode contact points are adjusted applying conductive gel until the impedance values displayed by the program g.NeedAccess (g.Tec, Austria) are within the desired range. This process takes between 10–15 min.

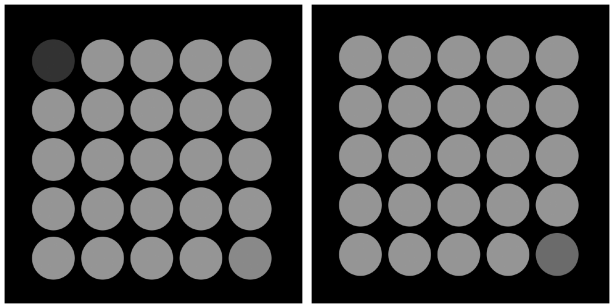

Fig. 2. Grid system representation used in the cognitive game. The blue spot represents the initial location, whereas the green spot represents the target location. Once the agent reaches the target spot, its color turns red to indicate the end of the play.

After this step, the subject is instructed to close their eyes, make eye movements and muscle chew in order to check the program and guarantee that the live channel values are accurate.

Once the headset is correctly applied, the OpenVibe Acquisition Server program, from the OpenVibe platform [30], is launched and configured with a sampling rate of 250 Hz. A 50-Hz notch filter is applied to filter out power line noise. An additional bandpass filter between 0.5 and 60 Hz is applied. Data are handled and processed with the OpenVibe Designer, from the same platform, using 8 channels for the brain data (one channel per electrode) and an additional channel to record the stimulus, which corresponds to a game movement performed by the agent. After everything is connected, the subject is seated in a comfortable chair in front of a computer screen. The brightness of the screen is set to the maximum setting to avoid any visual inconvenience in which the subject cannot distinguish the components of the game that appear on the screen.

The Acquisition Server receives and synchronizes the signal data from the headset and any event information from the game, and transfers it to the OpenVibe Designer application. When the subject is ready, the Game Manager and the OpenVibe Designer programs are launched and configured to communicate with the previously mentioned Acquisition Server. A brainwave session consists of several matches, each one being a gameplay. In the end, the sequence of game movements and the signal data generated for each match are saved for offline processing.[1]

B. Cognitive Game Procedure

The game parsimoniously consists of a $5x5$ grid of gray circular spots with a black background. It is similar to the one proposed by Iturrate et al. [27]. A blue spot indicates the current position of the agent and a green spot represents the goal, as shown in Fig. 2. The agent's objective is to reach the goal. The circular spot representing the goal remains static at the bottom-right position of the grid, whereas the one representing the position of the agent starts at the upper left position of the grid and moves in each iteration. When the agent reaches the goal, the position where the agent and the goal are located turns red, showing that the match has ended. There are four possible movements that the agent can perform: it can go upwards, downwards, toward the left and the right, and those movements are bounded to avoid the agent from leaving the grid. The movement direction is selected randomly and is executed once every 2 s. After each gameplay, there is a pause of 5 s until the next match starts. Each time an agent moves, the Game Manager program sends an event marker to the Acquisition Server. This is considered a stimulus to the OHC. The game is designed as to be evident whenever there is an error (i.e., the agent moves away from the objective) so the subject can notice it immediately after the stimulus is presented, possibly triggering the expected cognitive response, which can be imprinted as an ErrP component within the EEG stream.

[1]The brainwave data set has been published on the IEEE DataPort initiative [31].

C. Signal Processing, Segmentation, and Classification

To aid the detection of the ErrP response, an offline processing pipeline and classifier is constructed to identify whether the action taken by the agent is an error or not, from the human observer's point of view. It is developed in Python using the "MNE" software platform [32], which is a package designed specifically for processing EEG and magnetoencephalography data, and built upon the machine learning library Scikit-Learn [33].

This pipeline consists of the offline processing of the collected signals used to train a classifier that can decide whether an error potential is triggered. First, the output of a brainwave session is read and an additional band-pass filter of 0.1–20.0 Hz is applied to the signal. Samples that correspond to the start of an event are tagged using the data from the stimulus channel.

After the raw data are loaded and tagged, epochs are extracted. Epochs consist of all the sample points that take place during the 2 s from the start of the event, 2 s corresponding to the time it takes for each action to take place, resulting in 500 samples per channel, as the sampling frequency is 250 Hz. Thus, each epoch is composed of a matrix 500×8 channels.

Samples that do not correspond to an epoch (located beyond the 2 s frame after the onset of the event) are not used. Also, epochs referring to the start or finish of each match are excluded.

In this way, the raw data of a brainwave session are processed into an array of matches where each element is an array of epochs tagged with a number specifying the prediction of the classifier, i.e., if the epoch corresponds to an action that made the agent move further from the goal (hit) or an action that made the agent move closer to the goal (no-hit). The ErrP is expected to be found in hits. To get the data ready for classification, the stimulus channel is removed to classify the signals using only the EEG data. Each epoch is regularized using a MinMaxScaler, i.e., subtracting the minimum value in the epoch and dividing by the signal peak-to-peak amplitude [34]. The eight channels are concatenated using the MNE Vectorizer function, which transforms the data matrix into a single array sample. Lastly, these data are used by the classification module as information to train and test a classifier. Five different classification algorithms are used: logistic regression, multilayer perceptron with a hidden layer of 100 neurons (i.e., default values for the Scikit-Learn

MLPClassifier), random forest, *K*-neighbors with $k = 3$ and finally a linear kernel support vector classifier [35].

D. Reinforcement Learning

Each match consists of a list of game movement configurations and the associated epochs obtained from OHC's brainwaves. The set of matches of each OHC is split into training and testing. Training matches are used to train the classifier to identify the ErrP signal. After a classifier is trained, the epochs extracted from the test matches are classified as hit or no-hit. A reward for each movement in the match is generated based on the prediction from the classifier for that movement. The reward can either be -1 when the event is classified as a hit or 0 when it is classified as a no-hit. The accuracy of these rewards depends on the performance of the classifier. The list of game movements and their associated reward information is used to train the agent by a variant of RL called Q-Learning algorithm.

E. Q-Learning

Q-Learning [36] is a form of model-free RL, where an agent tries an action at a particular state and evaluates its consequences in terms of the reward or penalty it receives. To represent rewards, a matrix $Q(s, a)$ is used, where rows correspond to all the possible states, and columns represent all possible actions. This matrix is known as the Q-Table. The algorithm proceeds by randomly choosing what action to do and iteratively updating the Q-Table based on the received reward r by the following equation:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma * \max_{\tilde{a}} Q(\tilde{s},\tilde{a}) - Q(s,a)] \quad (1)$$

where s is the current state, a the action, α the learning rate, and γ the discount factor, a value between 0 and 1 that determines the importance of long term results versus immediate rewards. Hence, $Q(s, a)$ is the expected value of the sum of discounted rewards that the agent will receive if in the s state, it takes the action a according to this policy. Once the environment has been extensively explored and the Q-Table has been optimized, the action chosen for a given state is the one that maximizes the expected reward according to the Q-Table matrix.

The algorithm is developed in Python and uses the OpenAI Gym toolkit [37]. Gym is a toolkit for developing and comparing RL algorithms. It makes no assumptions about the structure of an agent, and is compatible with any numerical computation library, such as TensorFlow or Theano [38].

F. Gaming Agent Learning Procedure

The gaming agent learning procedure uses the testing matches from brainwave sessions produced during the cognitive game procedure phase, and their components are schematized in Fig. 1.

This phase is divided into a sequence of run sessions and gaming agent training matches. During the run session, the agent plays 200 matches guided by a specific Q-Table with a 5% chance of randomly selecting a movement, to reduce deadlocks and loops. Following the run session, the agent performs a single gaming agent training match. The gaming agent starts first with a Q-Table initialized with zeros, so the initial policy for the agent is randomized. For the agent to learn from the feedback generated by the OHC, movement actions are determined by the replay of the agent's actions that were taken during one brainwave session match, in an offline RL scheme [39]. This allows the Q-Table to be learned based on the OHC's feedback from the movements the agent took, which were executed pseudorandomly during the brainwave session. The previously mentioned feedback is not explicit as it comes from the interpreted brain signal data. This implies that the reward is determined by the OHC's brain activity.

Hence, following the iterative procedure based on (1), the Q-Table is updated in each gaming agent training match. After the algorithm finishes replicating all the steps from the brainwave session match, the Q-Table is stored and used by the agent in the next run session.

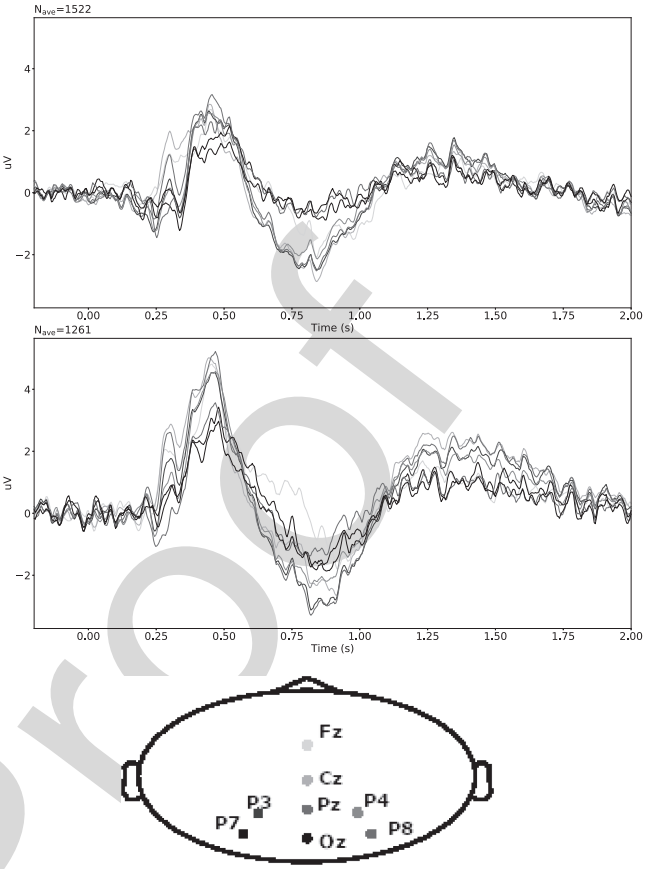

Fig. 3. Grand average of 2-s time-locked segments for all OHCs for the "move closer" (top) and "move further" (middle) condition. Zero time on *x*-axis corresponds to the onset of the stimulus, i.e., when the gaming agent moves. The increased height of the first peak around 0.4 s reflects the ErrP response particularly on Pz, based on the electrode layout (bottom).

III. Results

Grand averaged time-locked signal segments of 2-s length for all the OHCs can be seen in Fig. 3. The ErrP can be noticed more clearly on parieto-central areas (Pz electrode), around 0.4 s with a more prominent positive peak.

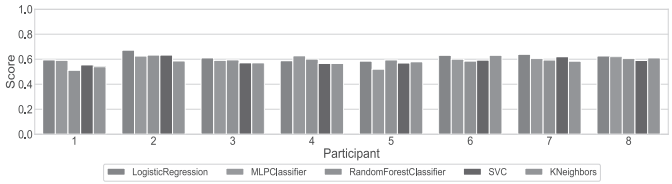

Fig. 4. Binary single trial classification score using five different classifiers while recognizing ErrP potentials for the eight OHCs. Chance level is 0.5.

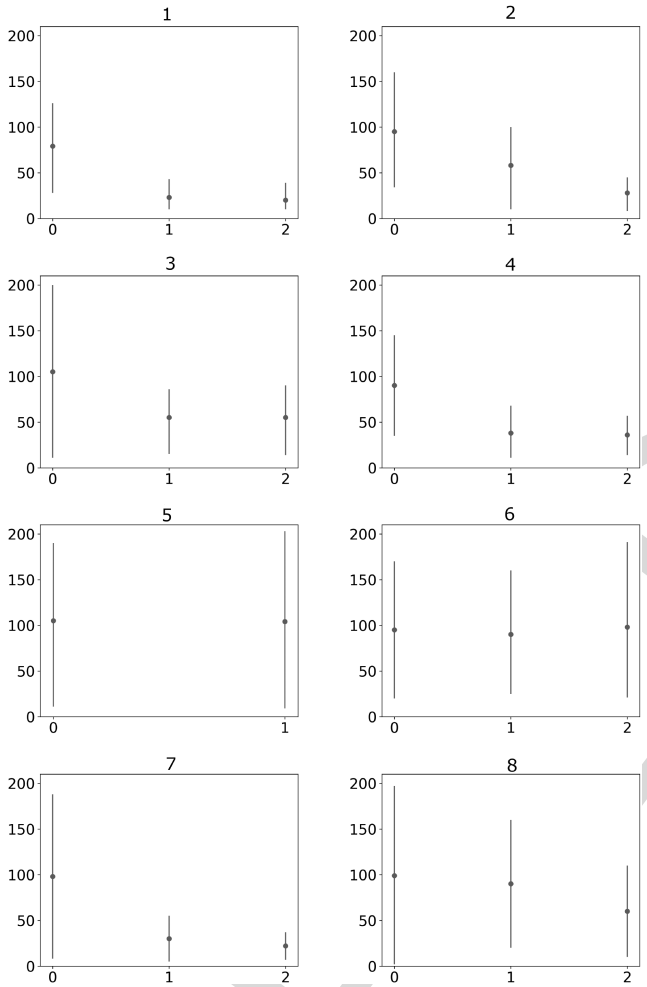

Fig. 5. Average number of steps for the agent to reach the goal when trained with rewards generated from brainwaves from OHCs 1–8. *Y*-axis shows the averaged number of steps for a run session, whereas *x*-axis shows the number of game matches used to cumulative train the Q-Table.

Fig. 4, on the other hand, shows the binary classification accuracy obtained for the eight OHCs using five different classification algorithms and using a tenfold cross-validation procedure. The best overall performance is obtained using logistic regression.

Complementary, Fig. 5 shows the average amount of steps it takes for the agent to reach the goal for each OHC, as the Q-Table is progressively trained using the reward information obtained from the prediction of the trained classifier. Each point corresponds to a run session of 200 game play repetitions. The

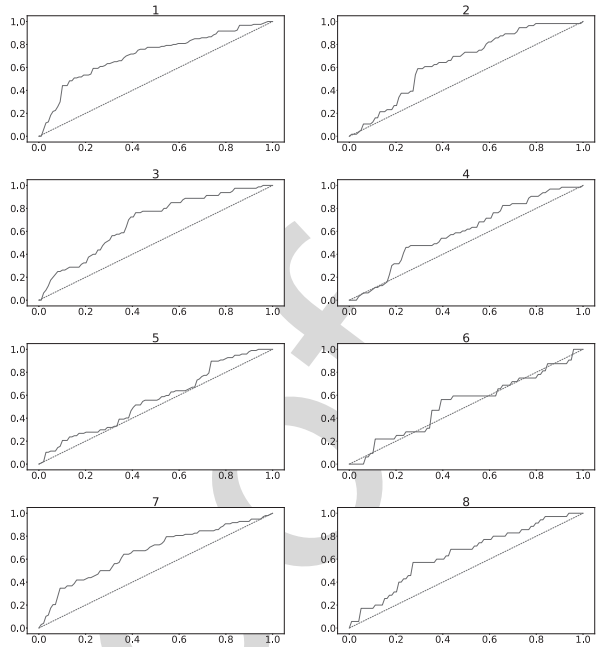

Fig. 6. ROC curves for OHCs 1–8. True positive rate is on the vertical axis and false positive rate on the horizontal axis. The ROC curves for ErrP identification for subjects 5 and 6 show low classification scores.

y-values represent the average number of steps the agent takes to reach the goal using a specific Q-Table for a run session. The first point, at the *x*-value 0, represents the number of steps the agent takes to reach the goal with an untrained Q-Table, where movements are decided randomly. The next point corresponds to the amount of steps it takes to reach the goal using a policy derived from a Q-Table trained after one brainwave session match, and so on.

The results show that as the Q-Table is progressively trained the average amount of steps decreases, meaning that the agent learns. However, the rate at which it learns varies per OHC, depending on the classification accuracy of the extracted brainwaves. For example, results for OHC 1 show faster learning than those of OHC 8 (see Fig. 5).

In the case for OHC 5 and 6, the reward information obtained from the brainwaves is not enough to train the agent effectively. Fig. 5 for OHC 5 and 6 shows no apparent learning, as the amount of steps to reach the goal does not decrease when trained. These results are also consistent with the classification ROC curves, shown in Fig. 6, where the area under the curve for OHCs 5 and 6 is close to chance level. Both OHCs have less recorded data from the sessions in comparison to the rest of the OHCs. This variation in performance for different OHCs has been studied extensively in BCI [15]. Besides low data samples, there are other reasons affecting the classification accuracy: cognitive reasons (i.e., the OHC not paying extensive attention to the game dynamics), very low SNR of the ErrP component or even the BCI-illiteracy phenomena where the specific OHC's signals do not contain the expected component response [40]. Fig. 7 shows the result of an agent successively trained with brainwave session matches where the EEG is generated with random signals. In this case,

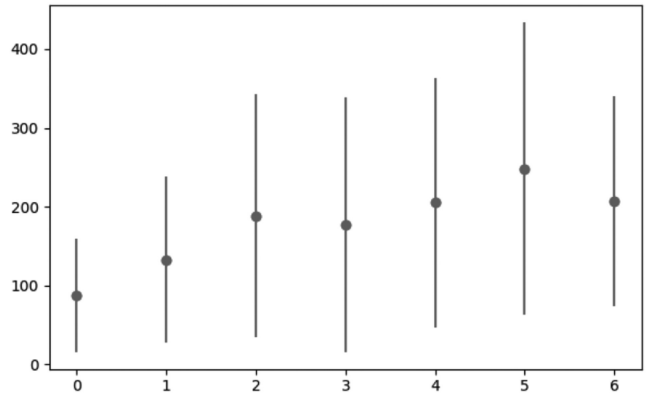

Fig. 7. Average number of steps for the agent to reach the goal when trained with a classifier produced from sham EEG signals. *X*-axis shows the number of gaming agent training matches used to train the Q-Table.

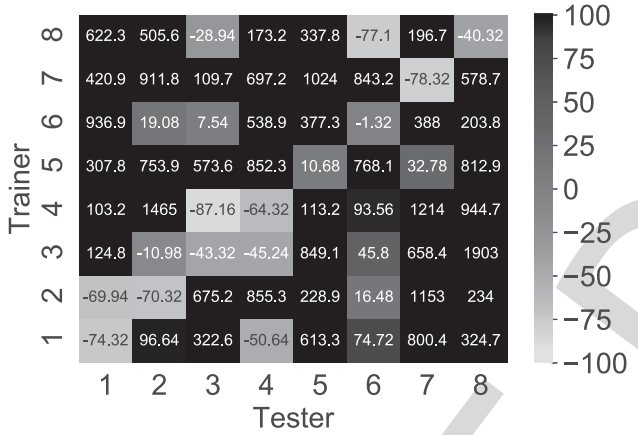

Fig. 8. Heatmap for the transfer learning experiment. Values represent the reduction in the average number of steps required to reach the goal. Negative values represent net improvements.

random EEG signals are generated using OpenVibe Acquisition Server signal generator for all channels, as if they were produced by an OHC who does not pay attention to the game. The agent learns nothing, and regardless of the number of matches that are used to learn the Q-Table, the number of steps required to reach the goal does not decrease. This pattern is also obtained when the game matches from OHCs 5 and 6 are used, showing that the reward labeling predicted by the trained classifier for those cases worked like a random classifier.

EEG signals have high intersubject variability [15]. This is evidenced in Fig. 8 where the agent training is performed with rewards obtained by classifying epochs from one Tester OHC with a classifier that was trained using the brainwaves from a different Trainer OHC. The figure shows the cumulative variation for all run sessions on the average number of steps required to reach the goal after training the agent with all the available matches from the brainwave session. Enhancements are shown as negative values. Only the diagonal of the heatmap matrix shows a clear improvement in terms of the reduction of the required number of steps to reach the goal (averaged per 200 runs) that corresponds to the same information for each OHC

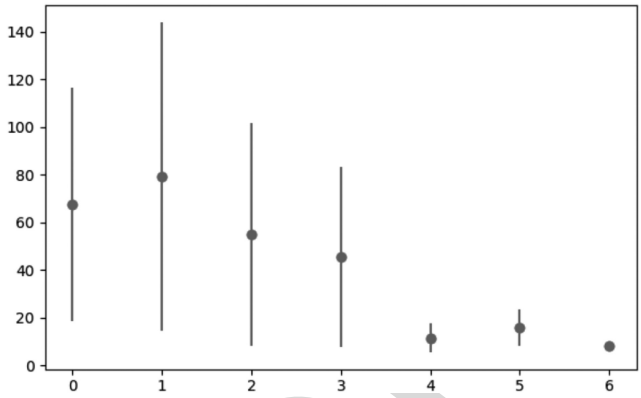

Fig. 9. Average steps using Q-Table trained with brainwave session matches from six different OHCs. *X*-axis shows the progressive number of gaming agent training matches used to train the Q-Table. These matches correspond to all subjects, excluding subjects 5 and 6, which individually show no significant learning progress.

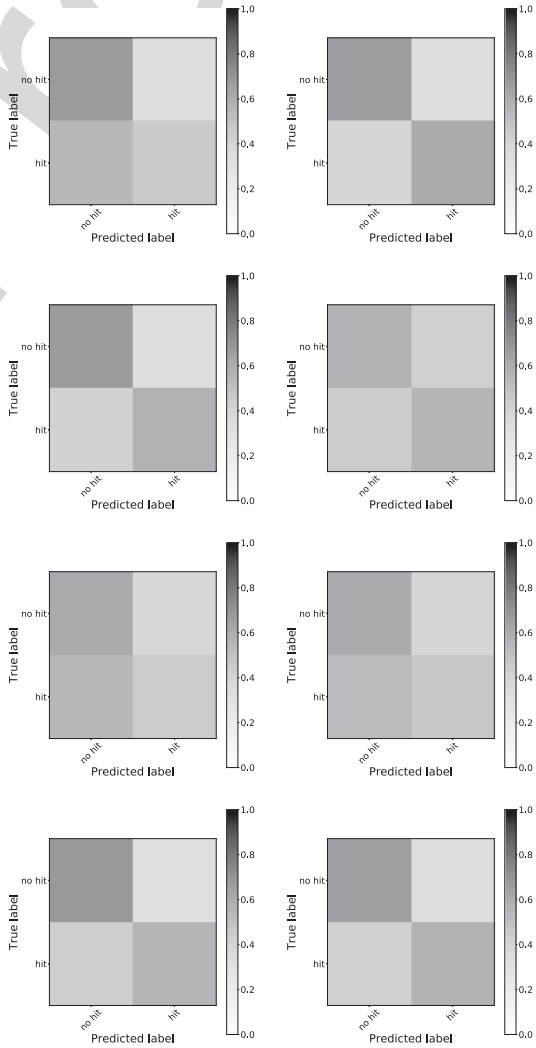

Fig. 10. Confusion matrix for OHCs 1–8. Darker colors show higher values. It can be seen in the lower percentage of false positives (upper right corner of each chart).

shown in Fig. 5. For this transfer learning experiment [41], no performance gain is evidenced, the agents learn nothing and this implies that the reward information provides no value.

Finally, Fig. 9 shows the result of training an agent with cumulative brainwave session matches from OHCs 1, 2, 3, 4, 7, and 8. It can be seen that the overall performance of the agent improves as long as there is information to produce rewards, regardless of the fact that they were generated from classifiers trained with different OHC's signals.

IV. Conclusion

Each time the gaming agent plays this simple game, it takes on average around 100 steps to reach the target spot. A classifier can be trained to recognize error potential from OHCs that observe the agent playing the game. The gaming agent plays randomly and movements are marked based on the identification of an error potential from the OHC. Those movements are used as rewards in an RL scheme to train a Q-Table. The gaming agent plays the game again, but this time the number of required steps to solve the game is reduced. If rewards are provided based on random signals, no reduction is achieved and the average number of steps does not change. This shows that there is an effective transfer of information from the brainwaves to the agent. As this process is repeated, the agent keeps improving solving the game effectively, i.e., performing the minimum number of required steps to reach the goal.

This work aims to propose a simple game mechanics that can use the ErrP component to train a gaming agent using an RL model. As described in the literature [15], ErrPs can be used to transmit subjective feedback to a computer. The collected data show that ErrP signals can in fact be classified and used to train an agent effectively. This proposal tries to keep the system as simple as possible, emphasizing information flow from the subjective error perception of the OHC.

One additional aspect to remark is the robustness of the learning strategy based on Q-Learning [42], [43]. The obtained accuracy to discriminate ErrPs is low, though on the same level to other similar results [27], [44]. In this regard, considering the variability of the ErrP response in terms of different cognitive experiments, levels of 90% accuracy in identifying it, are reported [15]. In this work, despite the low accuracy in ErrP identification, the RL algorithm was able to extract meaningful information from rewards that were helpful to improve, and often maximize, the agent's performance. Additionally, classification results show a low percentage of false positives (see Fig. 10), entailing a high specificity. On the other hand, the percentage of false negatives is generally higher. Even though this implies that the agent misses frequently when a wrong action takes place, this is not hindering the overall performance and the agent is still learning. Although they may be scarce, accurate rewards are very useful for the RL algorithm.

At the same time, effective agent training depends on the OHC's training data. Results confirm the futility or complexity of using Transfer Learning [41]: training a classifier with data obtained from one OHC, and using the same classifier to identify ErrPs for another OHC does not increase the performance of the agent. Despite that, the rewards generated from different subject's classifiers can be used to train the same Q-Table to improve its performance, which may lead to strategies where the overall performance is enhanced based on the information from different OHCs at the same time. We hypothesize that as long as there are accurate rewards obtained from high specificity classifiers, there is extra information that can be used by the agent to get an improved Q-Table. Additionally, it seems to be an agreement in terms of the subjective interpretation of what may be an appropriate movement to reach the goal, and different OHCs produce rewards for the same type of movements.

The simple setup of the grid-based game allows further experimentation, using the reduction on the number of average steps to reach the goal as a validation of the achieved information transfer. It will be of research interest to verify if the smooth progression toward the end alters the shape of the ErrP response, how the ErrP response is triggered in relation with different shapes and colors of the board markers [45], or if there is a differential ErrP signal component in relation to up, down, left and right movements. In addition, the outcome of manipulating the stimulus could be further studied as well as the influence on the results if incentives are given to participants.

Further work will be conducted in order to increase the complexity of the game to allow the possibility that the target position is dynamically changed. Although we found that the best performing classifier is logistic regression, there is room for improvement. The classifier could be enhanced to recognize the error potential [46] more effectively or could be pretrained to allow higher accuracy [47].

Acknowledgment

The authors would like to thank the Laboratory Centro de Inteligencia Computacional and ITBA University.

References

[1] D. Xu, M. Agarwal, F. Fekri, and R. Sivakumar, "Playing games with implicit human feedback," in *Proc. AAAI Workshop Reinforcement Learn. Games*, New York, NY, USA, 2020. [Online]. Available: https://www.semanticscholar.org/paper/Playing-Games-with-Implicit-Human-Feedback-Xu-Agarwal/faba141483180e53f461c6035ce95041cfed9a8f

[2] T. O. Zander, L. R. Krol, N. P. Birbaumer, and K. Gramann, "Neuroadaptive technology enables implicit cursor control based on medial prefrontal cortex activity," *Proc. Nat. Acad. Sci. USA*, vol. 113, no. 52, pp. 14898–14903, Dec. 2016. [Online]. Available: http://www.pnas.org/lookup/doi/10.1073/pnas.1605155114

[3] M. Carter, J. Downs, B. Nansen, M. Harrop, and M. Gibbs, "Paradigms of games research in HCI: A review of 10 years of research at CHI," in *Proc. Annu. Symp. Comput.–Hum. Interact. Play*, 2014, pp. 27–36. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2658537.2658708

[4] J. Frey, C. Mühl, F. Lotte, and M. Hachet, "Review of the use of electroencephalography as an evaluation method for human-computer interaction," in *Proc. Int. Conf. Physiol. Comput. Syst.*, Nov. 2014, pp. 214–223. [Online]. Available: http://arxiv.org/abs/1311.2222

[5] P. Barr, J. Noble, and R. Biddle, "Video game values: Human-computer interaction and games," *Interacting Comput.*, vol. 19, no. 2, pp. 180–195, Mar. 2007. [Online]. Available: https://academic.oup.com/iwc/article-lookup/doi/10.1016/j.intcom.2006.0 8.008

[6] Y. Zhao *et al.*, "Winning is not everything: Enhancing game development with intelligent agents," *IEEE Trans. Games*, vol. 12, no. 2, pp. 199–212, Jun. 2020.

[7] G. A. M. Vasiljevic and L. C. de Miranda, "Brain–Computer interface games based on consumer-grade EEG devices: A systematic literature review," *Int. J. Hum.–Comput. Interact.*, vol. 36, no. 2, pp. 105–142, Jan. 2020. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/10447318.2019.1612213

Q4

[8] R. Scherer, M. Pröll, B. Allison, and G. R. Müller-Putz, "New input modalities for modern game design and virtual embodiment," in *Proc. IEEE Virtual Reality*, Mar. 2012, pp. 163–164.

[9] D. Marshall, D. Coyle, S. Wilson, and M. Callaghan, "Games, gameplay, and BCI: The state of the art," *IEEE Trans. Comput. Intell. AI Games*, vol. 5, no. 2, pp. 82–99, Jun. 2013.

[10] A. Nijholt and D. Tan, "Playing with your brain: Brain-computer interfaces and games," in *Proc. ACM Int. Conf. Proc. Ser.*, 2007, vol. 203, pp. 305–306. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1255047.1255140

[11] J. Aguado-Delgado, J. M. Gutiérrez-Martínez, J. R. Hilera, L. De-Marcos, and S. Otón, "Accessibility in video games: A systematic review," *Universal Access Inf. Soc.*, vol. 19, no. 1, pp. 169–193, Mar. 2020. [Online]. Available: http://link.springer.com/10.1007/s10209-018-0628-2

[12] L. Yakovlev, N. Syrov, G. Nikolai, and A. Kaplan, "BCI-controlled motor imagery training can improve performance in e-sports," in *Proc. Int. Conf. Hum.–Comput. Interact.*, Jul. 2020, pp. 581–586. [Online]. Available: http://link.springer.com/10.1007/978-3-030-50726-8_76

[13] C. B. Holroyd, O. E. Krigolson, R. Baker, S. Lee, and J. Gibson, "When is an error not a prediction error? An electrophysiological investigation," *Cogn., Affect. Behav. Neurosci.*, vol. 9, no. 1, pp. 59–70, Mar. 2009. [Online]. Available: http://www.springerlink.com/index/10.3758/CABN.9.1.59

[14] G. Dornhege, *Toward Brain-Computer Interfacing*. Cambridge, MA, USA: MIT Press, 2007. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/6281216

[15] R. Chavarriaga, A. Sobolewski, and J. d. R. Millán, "Errare machinale est: The use of error-related potentials in brain-machine interfaces," *Frontiers Neurosci.*, vol. 8, p. 208, Jul. 2014. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2014.00208/abstract

[16] R. S. Sutton and A. G. Barto, *Reinforcement Learning, Second Edition: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[17] J. M. Santos and C. Touzet, "Exploration tuned reinforcement function," *Neurocomputing*, vol. 28, no. 1–3, pp. 93–105, Oct. 1999. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231298001179

[18] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 3826–3839, Sep. 2020.

[19] D. Silver *et al.*, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, Oct. 2017. [Online]. Available: http://www.nature.com/articles/nature24270

[20] I. Iturrate, L. Montesano, and J. Mínguez, "Robot reinforcement learning using EEG-based reward signals," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2010, pp. 4822–4829.

[21] S. K. Kim, E. A. Kirchner, A. Stefes, and F. Kirchner, "Intrinsic interactive reinforcement learning—Using error-related potentials for real world human-robot interaction," *Sci. Rep.*, vol. 7, no. 1, Dec. 2017, Art. no. 17562. [Online]. Available: http://www.nature.com/articles/s41598-017-17682-7

[22] J. Omedes, I. Iturrate, L. Montesano, and J. Minguez, "Using frequency-domain features for the generalization of EEG error-related potentials among different tasks," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Jul. 2013, pp. 5263–5266.

[23] T. J. Luo, Y. C. Fan, J. T. Lv, and C. L. Zhou, "Deep reinforcement learning from error-related potentials via an EEG-based brain-computer interface," in *Proc. IEEE Int. Conf. Bioinf. Biomed.*, Dec. 2019, pp. 697–701.

[24] S. K. Goh, N. P. Tran, D. T. Pham, S. Alam, K. Izzetoglu, and V. Duong, "Construction of air traffic controller's decision network using error-related potential," in *Augmented Cognition* (Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)). Cham, Switzerland: Springer, Jul. 2019, vol. 11580, pp. 384–393. [Online]. Available: http://link.springer.com/10.1007/978-3-030-22419-6_27

[25] C. Wirth, P. M. Dockree, S. Harty, E. Lacey, and M. Arvaneh, "Towards error categorisation in BCI: Single-trial EEG classification between different errors," *J. Neural Eng.*, vol. 17, no. 1, Dec. 2020, Art. no. 016008. [Online]. Available: https://iopscience.iop.org/article/10.1088/1741-2552/ab53fe

[26] L. Schiatti, J. Tessadori, N. Deshpande, G. Barresi, L. C. King, and L. S. Mattos, "Human in the loop of robot learning: EEG-based reward signal for target identification and reaching task," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2018, pp. 4473–4480.

[27] I. Iturrate, L. Montesano, and J. Minguez, "Shared-control brain-computer interface for a two dimensional reaching task using EEG error-related potentials," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 2013, pp. 5258–5262.

[28] D. Plass-Oude Bos *et al.*, "Brain-computer interfacing and games," in *Brain–Computer Interfaces, Applying Our Minds to Human Computer Interaction.* London, U.K.: Springer, 2010, pp. 149–178. [Online]. Available: http://link.springer.com/10.1007/978-1-84996-272-8_10

[29] S. E. Kober, M. Ninaus, E. V. Friedrich, and R. Scherer, "BCI and games: Playful, experience-oriented learning by vivid feedback?" in *Brain–Computer Interfaces Handbook*. Boca Raton, FL, USA: CRC Press, 2018, pp. 209–234.

[30] Y. Renard *et al.*, "OpenViBE: An open-source software platform to design, test, and use brain–computer interfaces in real and virtual environments," *Presence, Teleoperators Virtual Environ.*, vol. 19, no. 1, pp. 35–53, Feb. 2010. [Online]. Available: http://www.mitpressjournals.org/doi/10.1162/pres.19.1.35

[31] F. Bartolomé, J. Moreno, N. Navas, and J. Vitali, "ERRP-Dataset," 2019. [Online]. Available: http://dx.doi.org/10.21227/6emh-wb46

[32] A. Gramfort *et al.*, "MEG and EEG data analysis with MNE-Python," *Frontiers Neurosci.*, vol. 7, p. 267, Dec. 2013. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2013.00267/abstract

[33] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[34] T. Zhou and J. P. Wachs, "Spiking neural networks for early prediction in human–robot collaboration," *Int. J. Robot. Res.*, vol. 38, no. 14, pp. 1619–1643, Dec. 2019. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364919872252

[35] F. Lotte *et al.*, "A review of classification algorithms for EEG-based brain-computer interfaces: A 10 year update," *J. Neural Eng.*, vol. 15, no. 3, Jun. 2018, Art. no. 031005. [Online]. Available: http://stacks.iop.org/1741-2552/15/i=3/a=031005?key=crossref.9cd2b15ab6 5c8ad34b475584b43dc509

[36] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3/4, pp. 279–292, May 1992. [Online]. Available: http://link.springer.com/10.1007/BF00992698

[37] G. Brockman *et al.*, "OpenAI gym," 2016, *arXiv:1606.01540*.

[38] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available: http://tensorflow.org/

[39] S. Levine, A. Kumar, G. Tucker, and J. Fu, "Offline reinforcement learning: Tutorial, review, and perspectives on open problems," May 2020, *arXiv:2005.01643.*

[40] R. Yousefi, A. Rezazadeh Sereshkeh, and T. Chau, "Development of a robust asynchronous brain-switch using ErrP-based error correction," *J. Neural Eng.*, vol. 16, no. 6, Nov. 2019, Art. no. 066042. [Online]. Available: https://iopscience.iop.org/article/10.1088/1741-2552/ab4943

[41] D. Wu, Y. Xu, and B. Lu, "Transfer learning for EEG-based brain-computer interfaces: A review of progress made since 2016," *IEEE Trans. Cogn. Develop. Syst.*, to be published.

[42] R. Bauer and A. Gharabaghi, "Reinforcement learning for adaptive threshold control of restorative brain-computer interfaces: A Bayesian simulation," *Frontiers Neurosci.*, vol. 9, p. 36, Feb. 2015. [Online]. Available: http://journal.frontiersin.org/Article/10.3389/fnins.2015.00036/abstract

[43] J. Rubin, O. Shamir, and N. Tishby, "Trading value and information in MDPs," in *Intelligent Systems Reference Library*. Berlin, Germany: Springer, 2012, vol. 28, pp. 57–74. [Online]. Available: http://link.springer.com/10.1007/978-3-642-24647-0_3

[44] S. Ehrlich and G. Cheng, "A neuro-based method for detecting context-dependent erroneous robot action," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Dec. 2016, pp. 477–482.

[45] M. Eimer, "An event-related potential (ERP) study of transient and sustained visual attention to color and form," *Biol. Psychol.*, vol. 44, no. 3, pp. 143–160, 1997. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0301051196052179

[46] F. Iwane, R. Chavarriaga, I. Iturrate, and J. Del Millan, "Spatial filters yield stable features for error-related potentials across conditions," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.* Oct. 2017, pp. 661–666.

[47] M. Spüler, M. Bensch, S. Kleih, W. Rosenstiel, M. Bogdan, and A. Kübler, "Online use of error-related potentials in healthy users and people with severe motor impairment increases performance of a P300-BCI," *Clin. Neurophysiol.*, vol. 123, no. 7, pp. 1328–1337, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1388245711009059