Deep reinforcement learning from error-related potentials via an EEG-based brain-computer interface

Tian-jian Luo, Ya-chao Fan, Ji-tu Lv, Chang-le Zhou*
Department of Cognitive Science, School of Informatics
Xiamen University
Xiamen, China
{createopenbci, moriartyfan1212, lvjitu000}@gmail.com, dozero@xmu.edu.cn

***Abstract*—Reinforcement learning (RL) from human preferences suffered from temporal and interaction environments' limitations, which rule out real-time and real-world robotic applications of deep RL. To overcome the limitations, this study introduced the electroencephalography (EEG)-measured error-related potentials (ErrPs) to train a robotic RL system based on a brain-computer interface (BCI). We decoded ErrP signals by selecting human preferences in real-time to train robotic behavior by deep RL during a binary object selection task. Twelve healthy subjects participated in the ErrP experiments, in which they were asked to select and adjust self-favored behavior after a machine's random selections. The decoded ErrP signals classified by a convolutional neural network (CNN)architecture to achieve an average classification accuracy and an area under the ROC curve of 67.49% and 0.639, respectively. By using the well-trained ErrP signals classifier to train the deep RL system, our final results for training robotic behavior through ErrP-based preferences showed an average of 15.21% improvement in efficiency while obtaining acceptable rewards in RL. Thus, the work used brain signals instead of pressing or clicking buttons as the rewards of RL, and constructed a real-time and free from interaction interference intuitive RL system.**

***Index Terms*—reinforcement learning, error-related potentials, brain-computer interface, deep neural networks, intuitive system**

I. INTRODUCTION

Approaches to sophisticated reinforcement learning (RL) from human instructions can be roughly divided into two categories: RL from human ratings or rankings as reward values [1]–[3] and RL from preferences rather than absolute reward values [4]–[6]. State-of-the-art deep RL from human preferences [7] applies a convolutional neural network (CNN) for training policies and rewards and obtains preference segments from human comparison. However, obtaining human intuitive preferences to train robotic behaviors is highly time demanding since all subjects need time to compare different behaviors to see which is preferred. In addition, the selection procedure has a limited interaction environment that requires the subject to sit in front of a screen and use a mouse to select their preferences. Therefore, the time complexity and interaction environments' limitations are two major challenges for designing a real-time system of deep RL from human preferences.

*The corresponding author: *dozero@xmu.edu.cn*

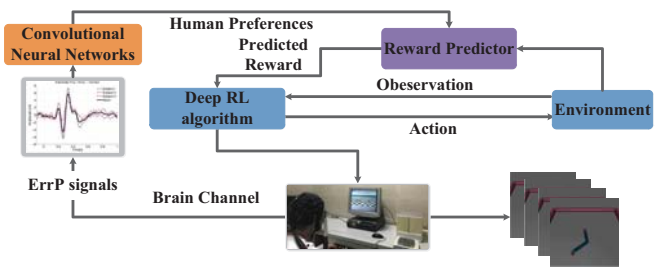


Fig. 1. Instead of obtaining rewards from human's intuitive selection, this study introduced a unique method of using the "brain channel" instead of the conventional interaction modality to overcome the limits of the environment and improve the efficiency.

This study introduced a unique method of using the electroencephalography (EEG) based error-related potential (ErrP) [8], [9] signals as the "brain channel" instead of the conventional interaction modality to overcome the limits of the interaction environment (see Figure 1). The advantage of "brain channel" is that it performs within one second, which makes it suitable for real-time systems. Although evoking and classifying such signals by current techniques present considerable challenges, recent researches on ErrP signals have shown that ErrP signals can be regarded as a brain channel for human-robot interaction in response to observing a mistake [10], [11].

To this end, an interactive system is developed for transferring objective human preferences to deep RL via the online identification of ErrP signals. The "brain channel" recognized preferences within one second and alleviated interaction environments' limitations for constructing a real-time and command-free control interaction. Therefore, the two contributions of this study are as follows:

(1)A carefully experiment paradigm was designed to demonstrate the existence and applicability of ErrP signals by a judgment between a subject's intentional selection and a machine's random selection. The extracted ErrP signals are then used as preference rewards to train a deep RL system without interaction environments' limitations.

978-1-5386-5488-0/18/$31.00 ©2018 IEEE

(2)By using the high efficiency CNN architecture to classify the ErrP signals, the results were applied as the subject's preference segment to train a real-time deep RL system. The high efficiency of ErrP-based preference rewards as a "brain channel" approach will contribute to constructing a real-world and real-time interaction for any novel robotic behavior.

II. MATERIALS AND METHODS

A. Stimuli and subjects

The stimuli for evoking ErrP signals were constructed by the project "RL-teacher [1]" [7]. In this study, we used the standard behavior of MuJoCo [12] simulated robotics within four different agents ("Reacher," "Cheetah," "Hopper," and "Ant") for training. To generate suitable stimuli for the EEG experiments, each agent randomly generated 1000 pairs of segments from "RL-teacher" with default parameters. Twelve right-handed healthy human subjects (eight males and four females, mean age=25.083, SD=1.832) with no prior BCI experience were invited to participate in the deep RL experiments from human preferences based on ErrP signals. All subjects provided written informed consent in accordance with the Declaration of Helsinki. Among the four different agents, each was presented to two males and one female subjects in the experiments. Each subject wearing an EEG cap was seated individually in a comfortable armchair 100 cm in front of a computer monitor. The experiments were performed in a dimly lit, sound-attenuated room, and the subjects were asked to read the instructions carefully on the screen. All subjects are asked to keep their body still during experiments to make the EEG signals are cleared for the Electrocardiograph (ECG) and Electromyography (EMG) artifacts.

B. Experiments

There are two main sub-experiments for the ErrP signals extraction, and Figure 2 illustrated the experimental paradigms for evoking ErrP signals and training deep RL algorithm. In the offline experiment (Figure 2(a)), due to the previous study [10], the experimental paradigm for evoking ErrP signals involved a subject passively observing whether an agent performing a binary-choice task made correct or incorrect decisions. The preferences of an agent's behavior can be defined as binary-choice tasks to evoke ErrP signals.T he subjects were asked which segment was consistent with the standard behavior of the MuJoCo simulated robotics in their minds. In the exhibition phase, two crosses were initially displayed to obtain the subject's focus. Then, a pair of trajectory segments was then randomly selected without replacement from the stimulus dataset to be shown on the screen three times. After three times exhibitions, a 1.5 s break was provided to ensure that the subject completed his/her mental selection. In the selection phase, the machine's target answer was given by a random 50/50 biased selection. When the machine's selection was exhibited, the ErrP was evoked and recorded within the brain in response to an unexpected error caused by the mismatch of

[1]https://github.com/nottombrown/rl-teacher

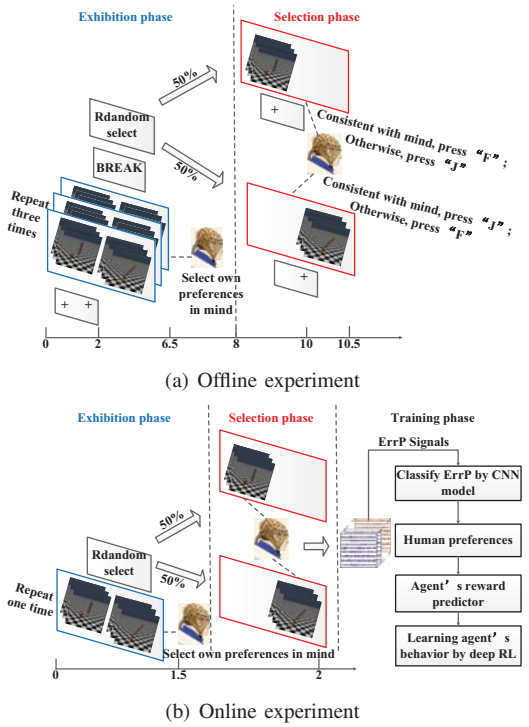


Fig. 2. Experimental paradigms for evoking ErrP signals and training deep RL algorithm.(a)Offline: The subjects were asked which segment was consistent with the standard behavior of the robotics in their minds to cause the ErrP signals. (b)Online: The same procedure is used to obtain subject's correct and ErrP trials by exhibiting one time, since the subject is familiar with stimuli.

the subject's mental selection. Otherwise, a correct trial will be recorded as a negative sample for training the CNN classifier. Whether recorded as an ErrP or correct trial, the corresponding labels were recorded by the subject's keyboard response. If the target of any trials was difficult to judge, the subject could skip the trial by pressing "SPACE." The offline experiment stopped when each subject completed 750 trials. The total trials of EEG data were divided into two parts: 60% for training, and 20% for testing, and the 10 iterations of 10-fold cross-validation strategy will be adapted for each subject.

The online experiment was initialized after well-trained ErrP classifiers from offline experiment. The selection phase was the same as in the offline experiment to extract correct trials and ErrP trials within 1 s. After extracting the ErrP signals, the human preference was classified by the well-trained CNN classifier. Instead of conventional deep RL from human preferences, in this study, we used an ErrP detector to generate human preferences rather than subject selections by clicking a mouse. As shown in Figure 2(b), the predicted results of human preferences generated by the ErrP constructed from the subjects' selections of two trajectory segments were used to train the agent's reward predictor. Finally, the agent's reward predictor was presented to learn the standard behavior of the MuJoCo simulated robotics for each agent by deep RL. Each human preference made by the ErrP required less than 2.6 s, which was faster than the 3-5 s in the original approach.

C. Data acquisition and ErrP analysis

EEG recordings for all subjects were performed on a "NeuroScan SynAmps" device with a "NeuroScan QuikCap international 10-20 system," and all 64 electrode impedances remained below 5 $k\omega$. The EEG signals were referenced to the nose, grounded at the frontal position (Fpz), and sampled at $500Hz$. A notch filter of $[48, 52Hz]$ and a bandpass filter of $[0.1, 30Hz]$ were used to pre-process the EEG signals [13]. Since the CNN architecture learns both a descriptive feature representation and a discriminative classifier by joint optimization [14], [15], and the cropped training strategy [16] will ensure the samples quantity, this study introduced the CNN architecture with a cropped training strtegy for ErrP analysis. Due to the nonlinear and nonstationary characteristics of EEG signals, the loss function is designed to penalize discrepancies between predictions of neighboring crops. The loss of the current crop is related to the cross-entropy of the next neighboring crop and the negative log likelihood of the target label. Therefore, the loss function was desinged by minimizing the sum of the per-crop losses:

$$\theta^* = \underset{\theta}{argmin} \sum_{j=1}^{N} loss\left(y^j, p_{f,k}\left(X^j_{t,\ldots,t+T'}\right)\right) \qquad (1)$$

As shown in Figure 3, our CNN architecture has two convolutional layers, one pooling layer, and one dense layer. The first convolutional layer was across the discretized samples to extract temporal features (width = 276), and the second convolutional layer was across the electrodes to extract spatial features (width = 276). The temporal convolutional layer has a smaller kernel size (25×1) than the spatial convolutional layer (64×40), thus allowing for a larger range of transformations in this layer. We used exponential linear units [17] as the activation function for these two convolutional layers. The following mean pooling layer was introduced to prevent over-fitting with a kernel size (75×1). The last dense layer contained a (40×12) size feature map and output two predicted labels (correct or incorrect) by using a logarithmic activation function. To reduce the computational load due to the increased training set size, we decode a group of neighboring crops together and reuse the intermediate convolution outputs.

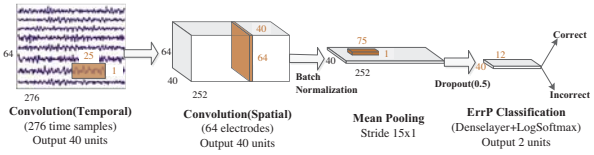


Fig. 3. CNN architecture for classifying ErrP signals in this study. The architecture has two convolutional layers, one pooling layer, and one dense layer. We used exponential linear units as the activation function for these two convolutional layers. The architectural choices are "Dropout," "batch normalization," for EEG signals.

The architectural choices are "Dropout [18]," "batch normalization [19]," for EEG signals. We present batch normalization for the output of convolutional layers before the nonlinearity; this strategy facilitates optimization by keeping the inputs of layers closer to a normal distribution during training. The probability of the "Dropout" for the inputs of two convolutional layers is set to 0.5 to overcome overfitting. The loss function is adapted from Equation 1, which is specific-defined from the previous section to improve the classification performance. The optimization method for the loss function uses the Adam [20] method together with the early stopping method following current research. The construction, training and classification of CNN model for ErrP signals were based on the project of "Brain-decode[2]."

III. Results and Discussion

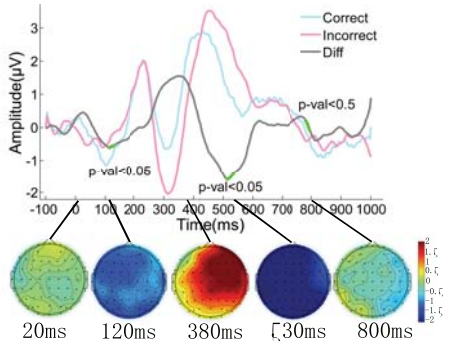
A. Occurrence of ErrPs for deep RL tasks

To determine whether an ErrP occurred, Figure 4(a) shows a representative average ErrP signal detected in all agent's experiments. The mean FCz electrode traces from all subjects with corresponding agents are shown when the selection of two trajectory segments was incorrect (pink trace) and when the selection of two trajectory segments was correct (blue trace). The gray trace is the difference between the correct trace and the incorrect trace, and a negative peak occurs approximately $[300, 350ms]$ after showing the ErrP procedure, a positive peak occurs at approximately $[500, 550ms]$, and a long negative tail occurs at approximately $[700, 1000ms]$. The results of the average ErrP shape are consistent with the theory of ErrP [11], and the negative peak and positive peak show significant differences (**p<0.05**) between the correct and incorrect traces within all subjects based on an ANOVA analysis. Therefore, the time range of $[-50, 500ms]$ was used to train the CNN classifier. The bottom plots are a brain electrical activity mapping (BEAM) of EEG signals' difference on the scalp created by interpolating 64 standard electrode locations, and the color bar represents the range $[-2, 2\mu V]$. From the BEAM, we found that the signal was centered toward the middle of the scalp in the key time samples for the ErrP, which is consistent with the theory of ErrP [11].

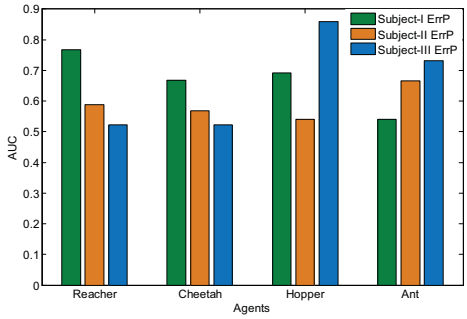
B. ErrP offline performance

Table 1 illustrates the ErrP offline classification performance in terms of accuracy in the training and testing. Subjects for the "Reacher," "Cheetah," "Hopper," and "Ant" agents achieved average classification accuracies of 63.88, 64.59, 74.37, and 67.20, respectively. The standard deviations were within a range of $[2, 5]$. The results showed that the averaged chances were 7.38, 6.43, 15.89, and 11.87 for the respective agents. The classification of certain subjects was over 75% while that of others was less than 60%. Furthermore, by using the True Positive Rates (TPRs) and False Positive Rates (FPRs) values, Figure 4(b) separates the area under the curve (AUC) values of the ROC curve into individual subjects. The figure shows that at least one subject approached or exceeded an AUC of 0.7 for each agent. Therefore, at least one best-performing subject had sufficient classification performance for the real-time online

[2]https://github.com/robintibor/braindecode



(a) Average shape of ErrP and BEAM



(b) AUC values of the ROC for individual subjects

Fig. 4. The shape and classification results of ErrP. (a)The plot is time-locked to the selection of two trajectory segments when the machine first displays the target of its random selection. The bottom plots are a BEAM to show whether the shape is consistent with the theory of ErrP.(b) The figure shows that at least one subject approached or exceeded an AUC of 0.7 for each agent. Therefore, at least one best-performing subject had sufficient classification performance for the real-time online experiments for each agent.

experiments for each agent. Among the four agents, we found ErrP signals performed better for the behaviors that easy to judge and select.

C. ErrP online performance for training the deep RL algorithm

In the online experiment, except for the 750 subject ErrP queries of three different subjects for each agent, the human queries were based on other subjects who did not take part in the EEG experiments, and the synthetic queries obtained by calculating the preferences over trajectories to exactly reflect the rewards in the underlying task were used to compare for the online performance of training the deep RL system. Due to limitations of the computing resources, the default parameters of project "RL-teacher" were presented in the online experiment. Using the three different preference queries to train the deep RL algorithm for each agent, the rewards of the agent's behavior were calculated through training timesteps. Figure 5 illustrates the reward results of each agent for five different comparison experiments. From the four plots of online experimental performance, the 1400 synthetic queries achieved the best reward. Due to the misclassification of ErrP signals, the reward results based on the ErrP queries were lower than those of the synthetic queries and human queries. However, if the number of ErrP queries increased to trade off the misclassification of ErrP signals, the ErrP queries will

TABLE I
CLASSIFICATION PERFORMANCE OF VALIDATION AND TEST. CHANCE IS DETERMINED BY RANDOMLY SHUFFLING TRIAL LABELS. THE BEST PERFORMANCE OF SUBJECTS WAS BOLD.

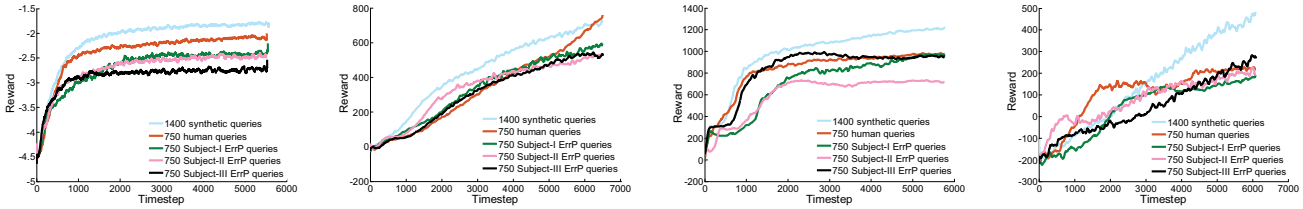
Accuracy(%)	Training	Test		
Subjects	Mean±Std.Dev.	Mean±Std.Dev.	Chance	Above
	Agent	**Reacher**		
Subject-I	**90.00±2.92**	**70.67±2.82**		**14.18**
Subject-II	88.00±3.42	62.67±4.14	56.49	6.18
Subject-III	86.00±3.63	58.28±5.36		1.79
	Agent	**Cheetah**		
Subject-I	**91.33±2.86**	**68.21±3.19**		**10.05**
Subject-II	81.33±3.55	60.67±3.24	58.16	2.51
Subject-III	85.33±3.42	64.90±3.35		6.74
	Agent	**Hopper**		
Subject-I	90.00±2.88	76.00±2.68		17.59
Subject-II	87.33±3.03	66.23±3.75	58.41	7.82
Subject-III	**94.67±2.72**	**80.67±3.24**		**22.26**
	Agent	**Ant**		
Subject-I	86.00±3.19	59.60±5.15		4.27
Subject-II	**92.67±2.94**	**79.33±3.40**	55.33	**24.00**
Subject-III	88.00±3.55	62.67±4.26		7.34

TABLE II
EXTRA NUMBER OF QUERIES AND TIME CONSUMPTION FOR ALL AGENTS AND SUBJECTS.

	Human Preference	Subject-I ErrP	Subject-II ErrP	Subject-III ErrP	ErrP improvement
		Agent	**Reacher**		
Queries	750	970	1030	1030	-
Time (s)	2794	2445	2596	2679	7.90%
		Agent	**Cheetah**		
Queries	750	989	1045	1014	-
Time (s)	3083	2493	2634	2556	16.93%
		Agent	**Hopper**		
Queries	750	930	1004	895	-
Time (s)	3116	2344	2531	2256	23.72%
		Agent	**Ant**		
Queries	750	1053	906	1030	-
Time (s)	2863	2654	2283	2596	12.29%

approaching the results of human queries. Since the ErrP preference rewards were more efficiency than human preference rewards, the trade off is reasonable to ensure the accuracy of preference with the improvement of efficiency.

Table 2 illustrated the trade off between queries and efficiency for the online experiment, and the extra number from ErrP queries was added to trade off the misclassification to ensure enough right preferences. After enough right preferences, the results of training RL by ErrP signals preference rewards will approach to human preferences. For the time complexity comparison, the query time of ErrP based human preferences was obtained in only 2.52 s (1.5 s for exhibition, 1s for ErrP-evoking, and 20ms for classification) because of the well-trained classifiers, which was faster than the average query time complexity of 3-5 s based on the subjects' response time among the human queries. Therefore, whatever the number



(a) Agent "Reacher" (b) Agent "Cheetah" (c) Agent "Hopper" (d) Agent "Ant"

Fig. 5. Reward results of each agent for five different comparison experiments. The reward results based on the ErrP queries were lower than those of the synthetic queries and human queries. Extra number of ErrP queries will improve the performance.

of queries increased for ErrP, the time consumption of ErrP queries was still lower than the human queries. A final average efficiency improvement among all four agents and 12 subjects was 15.21%, and the deep RL was trained without interaction environments' limitations.

IV. Conclusion

In this paper, ErrP signals are introduced to overcome the time costs and interaction environments' limitations for the deep RL agents meaningfully trained from human preferences. A high efficiency CNN classification model was well trained to decode the EEG activity of the ErrP by using enough correct trials and ErrP trials for subjects without prior BCI experience. The decoded ErrP signals achieve an average classification accuracy and an area under the ROC curve of 67.49% and 0.639, respectively. The ErrP signals were treated as preference rewards and deep RL training was completed based on ErrP preference rewards by an online BCI system. Except for alleviating interaction limitations of training RL system, an average of 15.21% efficiency improvement was made by the ErrP signals and experimental paradigm. Thus, obtaining human preferences from ErrP queries to train an agent's behavior via deep RL will facilitate the construction of intuitive human-robot interaction systems. Future work may be able to improve the performance of classifying ErrP signals, construct more efficient ErrP-evoking paradigms, and expand the range of tasks to which the paradigms can be applied.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No.61673322).

References

[1] Layla El Asri, Bilal Piot, Matthieu Geist, Romain Laroche, and Olivier Pietquin. Score-based inverse reinforcement learning. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 457–465. International Foundation for Autonomous Agents and Multiagent Systems, 2016.

[2] Aaron Wilson, Alan Fern, and Prasad Tadepalli. A bayesian approach for policy learning from trajectory preference queries. In *Advances in neural information processing systems*, pages 1133–1141, 2012.

[3] Christian Wirth, J Furnkranz, Gerhard Neumann, et al. Model-free preference-based reinforcement learning. In *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, pages 2222–2228, 2016.

[4] Patrikk D Sørensen, Jeppeh M Olsen, and Sebastian Risi. Breeding a diversity of super mario behaviors through interactive evolution. In *Computational Intelligence and Games (CIG), 2016 IEEE Conference on*, pages 1–7. IEEE, 2016.

[5] Riad Akrour, Marc Schoenauer, Michèle Sebag, and Jean-Christophe Souplet. Programming by feedback. In *International Conference on Machine Learning*, pages 1503–1511. JMLR. org, 2014.

[6] Johannes Fürnkranz, Eyke Hüllermeier, Weiwei Cheng, and Sang-Hyeun Park. Preference-based reinforcement learning: a formal framework and a policy iteration algorithm. *Machine learning*, 89(1-2):123–156, 2012.

[7] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, pages 4302–4310, 2017.

[8] Iñaki Iturrate, Luis Montesano, and Javier Minguez. Robot reinforcement learning using eeg-based reward signals. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 4822–4829. IEEE, 2010.

[9] Andres F Salazar-Gomez, Joseph DelPreto, Stephanie Gil, Frank H Guenther, and Daniela Rus. Correcting robot mistakes in real time using eeg signals. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 6570–6577. IEEE, 2017.

[10] Ricardo Chavarriaga and José del R Millán. Learning from eeg error-related potentials in noninvasive brain-computer interfaces. *IEEE transactions on neural systems and rehabilitation engineering*, 18(4):381–388, 2010.

[11] Pierre W Ferrez and José del R Millán. You are wrong!—automatic detection of interaction errors from brain waves. In *Proceedings of the 19th international joint conference on Artificial intelligence*, number EPFL-CONF-83269, 2005.

[12] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 5026–5033. IEEE, 2012.

[13] Tian-jian Luo, Jitu Lv, Fei Chao, and Changle Zhou. Effect of different movement speed modes on human action observation: An eeg study. *Frontiers in neuroscience*, 12:219, 2018.

[14] Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11):5391–5420, 2017.

[15] Domenico Mirarchi, Patrizia Vizza, Pietro Cinaglia, Giuseppe Tradigo, and Pierangelo Veltri. meeg: A system for electroencephalogram data management and analysis. In *Bioinformatics and Biomedicine (BIBM), 2017 IEEE International Conference on*, pages 1642–1646. IEEE, 2017.

[16] Tian-jian Luo, Fei Chao, et al. Exploring spatial-frequency-sequential relationships for motor imagery classification with recurrent neural network. *BMC bioinformatics*, 19(1):344, 2018.

[17] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[18] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[19] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.