

# PreMovNet: Pre-Movement EEG-based Hand Kinematics Estimation for Grasp and Lift Task

Anant Jain<sup>1</sup>, and Lalan Kumar<sup>1,2</sup>

<sup>1</sup>Department of Electrical Engineering, Indian Institute of Technology Delhi, New Delhi 110016 India

<sup>2</sup>Bharti School Of Telecommunication Technology And Management, Indian Institute of Technology Delhi, New Delhi 110016 India

**Abstract**—Kinematics decoding from brain activity helps in developing rehabilitation or power-augmenting brain-computer interface devices. Low-frequency signals recorded from non-invasive electroencephalography (EEG) are associated with the neural motor correlation utilised for motor trajectory decoding (MTD). In this communication, the ability to decode motor kinematics trajectory from pre-movement delta-band (0.5-3 Hz) EEG is investigated for the healthy participants. In particular, two deep learning-based neural decoders called PreMovNet-I and PreMovNet-II, are proposed that make use of motor-related neural information existing in the pre-movement EEG data. EEG data segments with various time lags of 150 ms, 200 ms, 250 ms, 300 ms, and 350 ms before the movement onset are utilised for the same. The MTD is presented for grasp-and-lift task (WAY-EEG-GAL dataset) using EEG with the various lags taken as input to the neural decoders. The performance of the proposed decoders are compared with the state-of-the-art multi-variable linear regression (mLR) model. Pearson correlation coefficient and hand trajectory are utilized as performance metric. The results demonstrate the viability of decoding 3D hand kinematics using pre-movement EEG data, enabling better control of BCI-based external devices such as exoskeleton/exosuit.

**Index Terms**—brain computer interface (BCI), electroencephalography (EEG), multi-variable linear regression (mLR), deep learning, pre-movement.

## I. INTRODUCTION

Brain-computer interface (BCI) incorporates brain activity to control external devices. In particular, Electroencephalography (EEG) signals are commonly utilised due to its non-invasive nature, mobility, and low cost. EEG signal has been widely utilized for motor classification [1], [2], commanding robotic devices [3] and instinctual control of prosthetic devices based on continuous classification [4]. However, it is to note that continuous kinematic parameter prediction based motor trajectory decoding (MTD) would provide better control of the external devices such as neural prosthetic, exosuit or exoskeleton devices. Recent literature supports EEG-based MTD for upper-limb in offline[5]–[8], and online mode[9]. The technique utilized for this purpose is multi-variable linear regression (mLR) [5], [6]. However, mLR technique is based on linear relationship between input-output variables and is sensitive to outliers. This shortcoming of mLR technique can be overcome by deep learning based decoding models. Deep learning models utilize non-linear features and are robust to the outliers. Deep learning based frameworks have been used for MTD [7]–[9] using EEG signals. Convolutional neural network (CNN)-Bidirectional long short-term memory (BiLSTM) based arm trajectory decoding for robotic arm control, was established using EEG in [9] with an average correlation reported as 0.47. Visual stimuli was considered as reference point in [7] for CNN-LSTM based trajectory estimation. However, visual stimuli may not be always present in real world applications. Hence, movement onset as reference point is explored for MTD in this work. In particular, delta frequency band is utilized as input features to neural decoder for efficient decoding [10], [11].

In this study, two neural decoding models, PreMovNet-I (MLP based) and PreMovNet-II (CNN-LSTM based), are proposed for robust MTD. It makes use of motor-related neural information existing

in the pre-movement EEG data. It may be noted that the motor movement is encoded in the EEG signal around 300 ms prior to actual movement [7]. Hence, EEG with an appropriate lag from the movement onset is taken as input to neural decoders. The inclusion of this neural information helps improve the performance of MTD. In particular, 3D hand trajectory decoding is performed for the grasp-and-lift task [12]. Twelve right-handed participants' data sets were included. For the inclusion of pre-movement neural information, EEG lags of 150 ms, 200 ms, 250 ms, 300 ms, and 350 ms from movement onset were taken for evaluating the performance of the proposed neural decoder for 3D hand movement. Thus, the input to the neural decoder is EEG with time lags. The channels were selected around the motor-cortex region, which is mapped to the hand movement. EEG signal recordings are initially preprocessed for noise removal and filtered in the delta frequency band (0.5–3 Hz). The 3D hand position kinematics data is also preprocessed prior to neural decoder training and testing. Both EEG and kinematics data have been down-sampled to 100 Hz to reduce the computation complexity of the neural decoder. The proposed deep learning frameworks, PreMovNet-I and PreMovNet-II, are compared with state-of-the-art mLR for the reach-and-grasp task.

## II. METHODOLOGY

In this Section, an explicit description of the utilised dataset, data preprocessing steps, and EEG-based kinematics estimation framework is presented. Fig. 1 illustrates the process flow of the proposed hand kinematics estimation framework.

### A. Experimental Setup

The WAY-EEG-GAL database [12] is utilized for the performance evaluation of the MTD framework with the kinematics data of twelve subjects. A 32-channel ActiCap (Brainproducts) was utilized to record the EEG data and wrist position (XYZ Cartesian coordinates) of the

This work was supported in part by DRDO - JATC project with project number RP04191G.

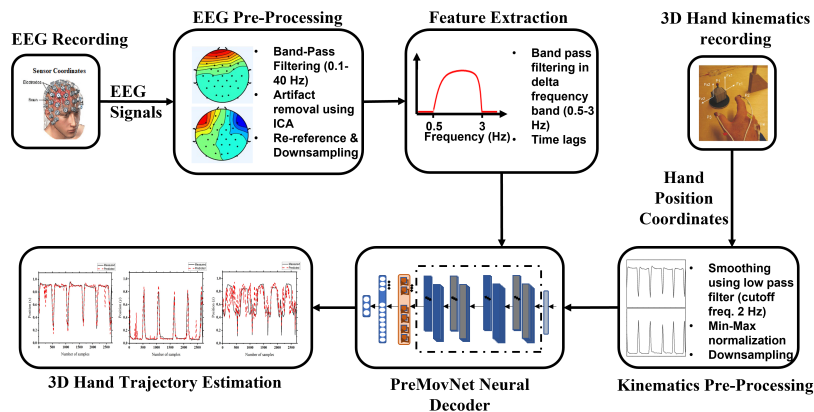


Fig. 1. Flowchart of pre-movement EEG based hand kinematics estimation framework.

subjects was recorded using a 3D position sensor with a sampling rate as 500 Hz. The subjects performed a series of grasp and lift task with various loads (165, 330, or 660 g) and surface frictions (silk, suede, or sandpaper).

Each trial began when the LED was turned on. The participant reached for the object, grasped it with forefinger and thumb, and then lifted it in the space stably for two seconds. Upon turning off the LED, the participant had to lower the hand, put the object in the original position and retracting his or her arm to its initial resting position.

### B. Data Pre-Processing & Feature Extraction

The recorded EEG data was first band-pass filtered using a zero-phase FIR filter in the frequency band of 0.1–40 Hz in order to discard baseline drifts. The data was then re-referenced to the average of all scalp electrodes. Then, the eye movement artefacts were removed using the independent component analysis (ICA) technique. The processed EEG data was further down-sampled to 100 Hz to reduce the computational cost. All the EEG data preprocessing was performed using the EEGLAB [13]. The EEG data was further filtered using a zero-phase windowed FIR filter with a Hamming window in the delta band (0.5-3 Hz). A total of 21 electrodes from the motor cortex region ('F3', 'Fz', 'F4', 'FC5', 'FC1', 'FC2', 'FC6', 'C3', 'Cz', 'C4', 'CP5', 'CP1', 'CP2', 'CP6', 'P7', 'P3', 'Pz', 'P4'), and occipital region ('O1', 'Oz', 'O2') of the brain were considered. Each EEG channel data is standardized using z-normalization technique.

The kinematic data was smoothed using a zero-phase low pass FIR filter with a cut-off frequency of 2 Hz. The filtered data was further normalized between 0 and 1 using the min-max normalization technique. This data was further down-sampled to 100 Hz in order to match the sampling rate of the EEG data.

### C. Data preparation

The kinematics data segment was initiated from the movement onset till the subject puts their hand in the resting position. It has been observed that the brain activity appears in the motor cortex region prior to the movement onset [7]. We consolidated the motor cortex activity information prior to the motor activity by incorporating the time lags of the selected EEG channels from the movement onset. In particular, time lags of 15 samples, 20 samples, 25 samples, 30 samples, and 35 samples were included in this study. With sampling rate as 100 Hz, EEG segment corresponding to time lags of 15 samples will be (-150ms to 0 ms) where 0ms is taken from the movement onset. For each kinematic vector, an EEG input matrix of dimension  $T \times (L * N)$  was generated where  $T$  is the data segment,  $L$  is the time lag, and  $N$  is total EEG channels selected.

### D. Multi-variable Linear Regression (mLR) model

Multi-variable linear regression (mLR) based kinematic decoding is widely being used for brain-computer interface [5], [6], [14]. The mLR model utilises multiple inputs to predict hand position directions. The input-output mapping for the mLR model is given as:

$$H_x[t] = \alpha_x + \sum_{n=1}^N \sum_{l=0}^L \beta_x^{(nl)} V_n[t-l] \quad (1)$$

$$H_y[t] = \alpha_y + \sum_{n=1}^N \sum_{l=0}^L \beta_y^{(nl)} V_n[t-l] \quad (2)$$

$$H_z[t] = \alpha_z + \sum_{n=1}^N \sum_{l=0}^L \beta_z^{(nl)} V_n[t-l] \quad (3)$$

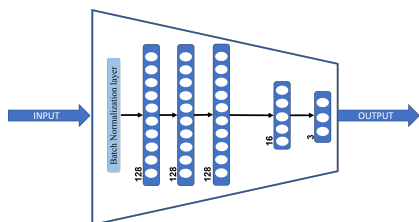


Fig. 2. PreMovNet-I based on Multi-layer Perceptron model structure for 3D hand kinematics decoding.

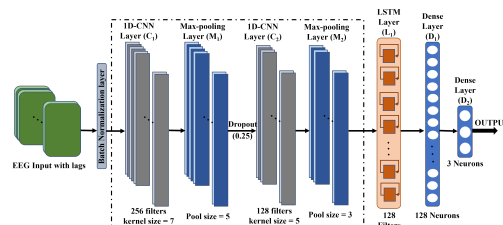


Fig. 3. PreMovNet-II based on CNN-LSTM architecture for 3D hand kinematics decoding.

TABLE 1. PCC analysis of (a) mLR, (b) PreMovNet-I (MLP-based), and (c) PreMovNet-II (CNN-LSTM based) neural decoders with time lags of 150 ms, 200 ms, 250 ms, 300 ms and 350 ms.

PARTICIPANT	Direction	150 ms			200 ms			250 ms			300 ms			350 ms		
		mLR	PreMovNet-I	PreMovNet-II	mLR	PreMovNet-I	PreMovNet-II	mLR	PreMovNet-I	PreMovNet-II	mLR	PreMovNet-I	PreMovNet-II	mLR	PreMovNet-I	PreMovNet-II
P1	x	0.6692	0.7249	0.7350	0.6537	0.7335	0.7397	0.6542	0.8035	0.8131	0.8410	0.7729	0.7926	0.8396	0.7955	0.7777
	y	0.6621	0.7122	0.7284	0.6470	0.7198	0.7336	0.6462	0.7930	0.8098	0.8350	0.7639	0.7893	0.8335	0.7767	0.7596
	z	0.4453	0.6134	0.5948	0.4438	0.6506	0.6671	0.4349	0.7116	0.7164	0.4360	0.6921	0.7186	0.3984	0.6949	0.7045
P2	x	0.4527	0.6915	0.6956	0.4612	0.6794	0.6899	0.4369	0.6963	0.7608	0.4700	0.7181	0.7763	0.4460	0.7398	0.7465
	y	0.4513	0.7114	0.7037	0.4641	0.6943	0.7049	0.4466	0.7056	0.7561	0.4744	0.7329	0.7758	0.4561	0.7500	0.7438
	z	0.2393	0.4266	0.3945	0.2695	0.5251	0.5301	0.2810	0.5072	0.5112	0.3077	0.5599	0.5122	0.3056	0.5492	0.5198
P3	x	0.5157	0.6392	0.6605	0.5877	0.6768	0.6798	0.6028	0.6972	0.7601	0.6510	0.6950	0.6903	0.6985	0.7329	0.7478
	y	0.5273	0.6381	0.6703	0.6067	0.6812	0.6794	0.6165	0.6965	0.7604	0.5661	0.7011	0.6732	0.6100	0.7391	0.7469
	z	0.4845	0.6361	0.6221	0.5158	0.6433	0.6457	0.4917	0.6402	0.6885	0.4304	0.6334	0.6620	0.4437	0.6577	0.6885
P4	x	0.6720	0.8263	0.8352	0.6689	0.8343	0.8628	0.6681	0.8439	0.8249	0.8512	0.8183	0.8040	0.8389	0.7663	0.7559
	y	0.6972	0.8438	0.8343	0.6941	0.8549	0.8629	0.6913	0.8587	0.8473	0.8786	0.8353	0.8220	0.8604	0.7913	0.7688
	z	0.4784	0.6704	0.7228	0.4715	0.7280	0.6966	0.4573	0.7023	0.7020	0.4317	0.7167	0.7048	0.4086	0.6814	0.7205
P5	x	0.5274	0.7311	0.7992	0.5380	0.7858	0.8106	0.5408	0.7810	0.7924	0.5304	0.7602	0.7870	0.5016	0.7606	0.8005
	y	0.5422	0.7667	0.8171	0.5588	0.8150	0.8306	0.5633	0.8089	0.8185	0.5548	0.8120	0.8106	0.5294	0.7990	0.8383
	z	0.5154	0.6162	0.6579	0.5279	0.6229	0.6469	0.5052	0.6378	0.5452	0.4796	0.6658	0.5381	0.4499	0.6409	0.6554
P6	x	0.3589	0.7838	0.8033	0.3744	0.7708	0.8090	0.3641	0.8046	0.8380	0.4541	0.7740	0.8288	0.4173	0.7922	0.7655
	y	0.3711	0.7602	0.7860	0.3873	0.7529	0.7951	0.3783	0.7888	0.8308	0.4606	0.7713	0.8118	0.4263	0.7735	0.7597
	z	0.2968	0.5971	0.5612	0.3133	0.5822	0.5939	0.3166	0.6001	0.5752	0.3255	0.6137	0.6023	0.3184	0.6217	0.6383
P7	x	0.4059	0.6981	0.7390	0.4384	0.7168	0.7626	0.4030	0.7158	0.7791	0.3952	0.7587	0.7846	0.3898	0.7689	0.7902
	y	0.4078	0.7082	0.7557	0.4373	0.7409	0.8057	0.3951	0.7330	0.8015	0.3407	0.7714	0.8045	0.3676	0.7964	0.8083
	z	0.4347	0.5484	0.4968	0.4471	0.5573	0.5617	0.4191	0.5602	0.5386	0.4041	0.5659	0.5580	0.3809	0.5847	0.6007
P8	x	0.4956	0.7650	0.7925	0.4518	0.6919	0.7262	0.6095	0.8012	0.8273	0.6601	0.8174	0.8078	0.6505	0.8032	0.8343
	y	0.5028	0.7615	0.7989	0.5091	0.7007	0.7299	0.6078	0.8081	0.8228	0.6534	0.8174	0.8026	0.6419	0.8017	0.8356
	z	0.2775	0.6320	0.6276	0.1486	0.5701	0.5744	0.2740	0.7039	0.6741	0.2867	0.6925	0.7115	0.2827	0.6968	0.8823
P9	x	0.4903	0.7545	0.7868	0.5221	0.7958	0.7987	0.5440	0.7699	0.8223	0.5202	0.8024	0.7903	0.5441	0.8108	0.7938
	y	0.4877	0.7605	0.7946	0.5207	0.7970	0.7978	0.5429	0.7758	0.8185	0.5305	0.8039	0.7926	0.5469	0.8137	0.7959
	z	0.3442	0.5782	0.5141	0.3436	0.5994	0.5408	0.3631	0.5667	0.5735	0.3532	0.6155	0.5194	0.3498	0.6458	0.5861
P10	x	0.6767	0.8200	0.8393	0.3779	0.7508	0.7984	0.3714	0.7629	0.8164	0.3860	0.7417	0.7821	0.4483	0.7609	0.7978
	y	0.7040	0.8337	0.8565	0.3989	0.7629	0.8149	0.3943	0.7755	0.8347	0.4091	0.7525	0.7949	0.4817	0.7773	0.8085
	z	0.3393	0.6634	0.6376	0.2487	0.6374	0.6359	0.2465	0.6426	0.6629	0.1981	0.6228	0.6384	0.2053	0.6503	0.6182
P11	x	0.5094	0.7233	0.8140	0.4823	0.7367	0.7677	0.4992	0.7669	0.7800	0.4530	0.7317	0.7567	0.4423	0.7358	0.7566
	y	0.5342	0.7531	0.8360	0.5131	0.7747	0.7950	0.5203	0.7873	0.8048	0.4801	0.7601	0.7853	0.4665	0.7608	0.7841
	z	0.3693	0.6467	0.6142	0.3944	0.6354	0.4744	0.4280	0.5378	0.5991	0.3547	0.5616	0.5991	0.3547	0.5676	0.6581
P12	x	0.3132	0.6455	0.6736	0.3429	0.6996	0.6673	0.3182	0.5975	0.6750	0.3159	0.6596	0.6527	0.3482	0.6428	0.6811
	y	0.3328	0.6616	0.6778	0.3688	0.6966	0.6991	0.3442	0.5953	0.6904	0.3482	0.6154	0.6522	0.3742	0.6504	0.6911
	z	0.3486	0.4625	0.4169	0.3688	0.4368	0.4161	0.3869	0.4375	0.4377	0.3904	0.4560	0.4052	0.4034	0.4619	0.4318
Average	x	0.5072	0.7336	0.7644	0.4974	0.7310	0.7618	0.5010	0.7534	0.7908	0.5006	0.7500	0.7711	0.5051	0.7610	0.7706
	y	0.5184	0.7426	0.7733	0.5088	0.7409	0.7724	0.5122	0.7605	0.7990	0.5113	0.7614	0.7762	0.5162	0.7692	0.7784
	z	0.3805	0.5778	0.5575	0.3745	0.5907	0.5822	0.3834	0.6056	0.6005	0.3724	0.6163	0.5983	0.3668	0.6206	0.6193

the outputs,  $H_x[t]$ ,  $H_y[t]$ , and  $H_z[t]$  are the horizontal, vertical, and depth positions of the hand at time  $t$ , respectively.  $V_n[t-l]$  is the standardized EEG potential at time lag  $l$ , where the number of time lags is varied from 0 to  $L$ . The regression coefficients,  $\alpha$  and  $\beta$ , are the coefficients of the mLR model.

### E. Proposed PreMovNet-I: Multi-Layer Perceptron (MLP) based

The multi-layer perceptron (MLP) based PreMovNet model is presented in Fig. 2. The model consists of six layers, including one batch normalisation layer, four dense layers, and one output layer. The first three dense layers consist of 128 neurons each, and the last dense layer consists of 16 neurons. The output layer consists of three neurons corresponding to the hand trajectory values in the x, y, and z directions as shown in Fig. 2.

### F. Proposed PreMovNet-II: CNN-LSTM based

In this Section, a CNN-LSTM based PreMovNet model, as shown in Fig. 3, is presented for hand kinematics decoding for grasp and lift task. It consists of nine layers that include of batch normalisation layer, two convolution layers, two max-pooling layers, one dropout layer, one LSTM layer, and two dense layers. The batch-normalization layer is represented as  $B_1$ . The convolutional layer  $C_1$  has a kernel size of 7 and 256 filters, while  $C_2$  layer has a kernel size of 5 and 128 filters. The zero padding is done for both  $C_1$  and  $C_2$  layers so that the size of input and output remains the same. Each convolution layer is followed by a ReLU activation unit.  $M_1$  and  $M_2$  are max-pooling layers with a window size of 5 and 3, respectively. The layer,  $D_1$  is a dropout layer with a 0.25 dropout rate. The layer  $L_1$  is the LSTM layer with 128 cells, followed by the ReLU activation function. The last two layers,  $D_1$  and  $D_2$  are dense layers with 128 and 3 neurons, respectively. The last layer with three output neurons yields the predicted hand kinematics in the x, y, and z-directions as shown in Fig. 3.

### G. Training and Evaluation

With the aim of training and performance evaluation of the kinematics decoding models, the dataset was partitioned into separate

training, validation, and test sets. The training set was used to train the models, while the validation set was used to tune the model hyper-parameters. The test set was used for the performance evaluation of the trained models. The adaptive moment estimation (Adam) optimization algorithm [15] was employed with a mean squared error (MSE) loss function for training the deep learning neural decoders. The early stopping technique was adopted to avoid over-fitting of the neural decoders. Total 294 trials were utilised for each participant from the WAY-EEG-GAL dataset. Total trials were divided into three separate subsets for each subject: (a) training data consisting of 234 trials data samples; (b) validation data with 30 trials data samples; and (c) test data with 30 trials data samples. The performance of the trained models is evaluated using Pearson's correlation coefficient (PCC) between the predicted hand kinematics and measured kinematics data. It may be noted that the training step is computationally strenuous, while the estimation is rapid. Hence, the decoding model can be implemented to control external prostheses or exoskeletons/exosuits.

## III. RESULTS

PCC is utilised herein as a performance metric for measuring the efficiency of the neural decoder. In particular, PCC is evaluated at different time lags and compared the proposed neural decoders with the state-of-the-art mLR neural decoder.

The PCC analysis for the various neural decoders is presented in Table-I with varying time lags. The analysis is done for twelve subjects from the WAY-EEG-GAL dataset in all the x, y, and z directions. It may be observed that the proposed PreMovNet-I and PreMovNet-II perform reasonably better than the state-of-the-art mLR decoder. Additionally, the CNN-LSTM based PreMovNet-II performs better than MLP-based PreMovNet-I in the x and y-directions, while in the z-direction, the two proposed decoders have similar performance. In particular, CNN-LSTM based PreMovNet-II performs best with 250 ms time lag in x and y directions, while in the z-direction, the best performance is achieved by MLP-based PreMovNet-I at time lag of 350 ms. The average PCC of the CNN-LSTM based PreMovNet with 250 ms time lag is  $0.7908 \pm 0.043$ ,  $0.7990 \pm 0.042$  and  $0.6005 \pm 0.090$  in the x, y, and z-directions, respectively. The poor correlation in

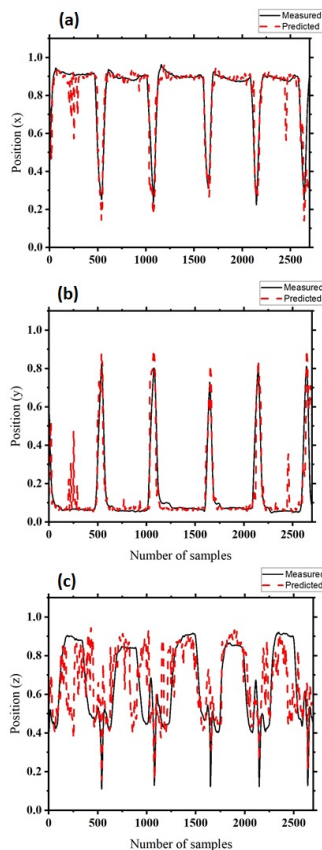


Fig. 4. Trajectory estimation in x, y and z direction using the PreMovNet neural decoder is presented in (a)-(c) respectively.

z-direction may be because of transient movement along it. The two proposed neural decoder for MTD can be selected based on desired performance and available processing capability.

The predicted hand trajectory is additionally compared herein with measured trajectory. Fig. 4 (a)-(c) shows the measured and predicted hand trajectories of participant 04 for the CNN-LSTM based PreMovNet-II neural decoder in x, y, and z-directions, respectively with 200 ms lag. It may be noted that the estimated trajectories in x and y directions closely follow the measured ones. Poor correlation in z-direction results in less accurate estimated trajectory.

#### IV. CONCLUSIONS AND FUTURE DIRECTIONS

In this letter, two deep learning-based neural decoders are proposed for motion trajectory decoding (MTD) that make use of pre-movement EEG for efficient hand kinematics decoding. The inclusion of the neural motor information before the movement onset as an input feature improves the MTD ability of the proposed neural decoders. The proposed neural decoders, PreMovNet-I and PreMovNet-II, are compared with state-of-the-art mLR decoder on WAY-EEG-GAL dataset. The performance evaluation of the proposed decoders is presented using the Pearson correlation coefficient analysis. Additionally, the decoded hand trajectory is compared with the measured hand trajectory in x, y, and z-directions. A significant improvement in correlation and estimated trajectory is observed using deep-learning based PreMovNet when compared with traditional mLR based approach. The hand kinematics decoding may be employed

for motor neurorehabilitation and power augmentation by controlling the external BCI devices such as exoskeletons/exosuits.

#### ACKNOWLEDGMENT

The authors would like to thank Prof. Sitikantha Roy, Prof. Shubhendu Bhasin, and Prof. Sushma Santapuri from Indian Institute of Technology Delhi (IITD), and Dr. Suriya Prakash from All India Institute of Medical Sciences (AIIMS) Delhi for their discussion and constructive comments during the preparation of the manuscript.

#### REFERENCES

- [1] G. Zhang, V. Davoodnia, A. Sepas-Moghaddam, Y. Zhang, and A. Etemad, "Classification of hand movements from EEG using a deep attention-based LSTM network," *IEEE Sensors Journal*, vol. 20, no. 6, pp. 3113–3122, 2019.
- [2] R. Chaisaen, P. Autthasan, N. Mingchinda, P. Leelaarporn, N. Kunaseth, S. Tammajarung, P. Manoonpong, S. C. Mukhopadhyay, and T. Wilaiprasitporn, "Decoding EEG rhythms during action observation, motor imagery, and execution for standing and sitting," *IEEE Sensors Journal*, vol. 20, no. 22, pp. 13 776–13 786, 2020.
- [3] Y. Mishchenko, M. Kaya, E. Ozbay, and H. Yanar, "Developing a three-to six-state EEG-based brain–computer interface for a virtual robotic manipulator control," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 4, pp. 977–987, 2018.
- [4] H. Gao, L. Luo, M. Pi, Z. Li, Q. Li, K. Zhao, and J. Huang, "EEG-based volitional control of prosthetic legs for walking in different terrains," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 2, pp. 530–540, 2019.
- [5] T. J. Bradberry, R. J. Gentili, and J. L. Contreras-Vidal, "Reconstructing three-dimensional hand movements from noninvasive electroencephalographic signals," *Journal of Neuroscience*, vol. 30, no. 9, pp. 3432–3437, 2010.
- [6] R. Sosnik and O. B. Zur, "Reconstruction of hand, elbow and shoulder actual and imagined trajectories in 3D space using EEG slow cortical potentials," *Journal of Neural Engineering*, vol. 17, no. 1, p. 016065, 2020.
- [7] S. Pancholi, A. Giri, A. Jain, L. Kumar, and S. Roy, "Source aware deep learning framework for hand kinematic reconstruction using eeg signal," *IEEE Transactions on Cybernetics*, pp. 1–13, 2022.
- [8] J.-H. Jeong, K.-H. Shim, D.-J. Kim, and S.-W. Lee, "Trajectory decoding of arm reaching movement imageries for brain-controlled robot arm system," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 5544–5547.
- [9] —, "Brain-controlled robotic arm system based on multi-directional CNN-BiLSTM network using EEG signals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 28, no. 5, pp. 1226–1238, 2020.
- [10] P. Ofner and G. R. Müller-Putz, "Decoding of velocities and positions of 3D arm movement from EEG," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2012, pp. 6406–6409.
- [11] A. Úbeda, E. Hortal, E. Iáñez, C. Perez-Vidal, and J. M. Azorín, "Assessing movement factors in upper limb kinematics decoding from EEG signals," *PLoS One*, vol. 10, no. 5, p. e0128456, 2015.
- [12] M. D. Luciw, E. Jarocka, and B. B. Edin, "Multi-channel EEG recordings during 3,936 grasp and lift trials with varying weight and friction," *Scientific data*, vol. 1, no. 1, pp. 1–11, 2014.
- [13] A. Delorme and S. Makeig, "EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis," *Journal of neuroscience methods*, vol. 134, no. 1, pp. 9–21, 2004.
- [14] N. Robinson, C. Guan, and A. Vinod, "Adaptive estimation of hand movement trajectory in an EEG based brain-computer interface system," *Journal of neural engineering*, vol. 12, no. 6, p. 066019, 2015.
- [15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.