OPEN ACCESS
International Journal of Neural Systems, Vol. 35, No. 13 (2025) 2550073 (18 pages)

DOI: 10.1142/S012906572550073X

World Scientific
www.worldscientific.com

An Explainable 3D-Deep Learning Model for EEG Decoding in Brain–Computer Interface Applications

Muhammad Suffian
DIIES, University Mediterranea of Reggio Calabria
Via Zehender, Loc. Feo di Vito, Reggio Calabria 89122, Italy
m.suffian@unirc.it

Cosimo Ieracitano
DICMAPI, University of Naples "Federico II"
Piazzale Vincenzo Tecchio, 80, Napoli 80125, Italy
cosimo.ieracitano@unina.it

Francesco C. Morabito* and Nadia Mammone†
DICEAM, Mediterranea University of Reggio Calabria
Via Zehender, Loc. Feo di Vito, Reggio Calabria 89122, Italy
**morabito@unirc.it*
†nadia.mammone@unirc.it

Received 24 April 2025
Accepted 19 September 2025
Published Online 18 October 2025

Decoding electroencephalographic (EEG) signals is of key importance in the development of brain–computer interface (BCI) systems. However, high inter-subject variability in EEG signals requires user-specific calibration, which can be time-consuming and limit the application of deep learning approaches, due to general need of large amount of data to properly train these models. In this context, this paper proposes a multidimensional and explainable deep learning framework for fast and interpretable EEG decoding. In particular, EEG signals are projected into the spatial–spectral–temporal domain and processed using a custom three-dimensional (3D) Convolutional Neural Network, here referred to as *EEGCubeNet*. In this work, the method has been validated on EEGs recorded during motor BCI experiments. Namely, hand open (HO) and hand close (HC) movement planning was investigated by discriminating them from the absence of movement preparation (resting state, RE). The proposed method is based on a global- to subject-specific fine-tuning. The model is globally trained on a population of subjects and then fine-tuned on the final user, significantly reducing adaptation time. Experimental results demonstrate that *EEGCubeNet* achieves state-of-the-art performance (accuracy of 89.56 ± 4.29 and 89.06 ± 4.86 for HC versus RE and HO versus RE, binary classification tasks, respectively) with reduced framework complexity and with a reduced training time. In addition, to enhance transparency, a 3D occlusion sensitivity analysis-based explainability method (here named *3D xAI-OSA*) that generates relevance maps revealing the most significant features to each prediction, was introduced. The data and source code are available at the following link: https://github.com/AI-Lab-UniRC/EEGCubeNet

Keywords: Electroencephalography; 3D convolutional neural networks; explainable artificial intelligence; brain–computer interfaces.

†Corresponding author.

1. Introduction

In the field of brain–computer interface (BCI), a direct communication is established between the brain and the machines. Brain signals are collected, decoded, and used to communicate with an interactive application.[1] Motor BCIs represent a promising perspective for patients in active neurorehabilitation and neuroprosthetic control. Electroencephalography (EEG)-based motor BCIs are widely studied due to their noninvasiveness, portability, and high temporal resolution, making them suitable for a wide range of applications, including neurorehabilitation, assistive technologies, and cognitive monitoring.[2] In the context of motor BCIs, this study specifically targets motor preparation: the early, transient phase that precedes the initiation of either imagined or executed movement. This phase is characterized by brief planning-related activity, typically occurring 1–1.5 s before movement execution. Neurophysiologically, motor preparation involves early components of the movement-related cortical potential (MRCP),[3] and is associated with rapid shifts in neural activity over premotor and motor areas. Thus, while the experimental task falls within the broader category of motor BCI paradigms, the neural features here investigated are related to the anticipatory planning activity (motor preparation), which may contain discriminative information valuable for low-latency BCI applications. In BCI systems, user-specific calibration is essential to account for the high variability of EEG signals across individuals. Long calibration sessions are indeed impractical and limit the usability of BCIs in real-world scenarios. Therefore, fast subject-specific fine-tuning (i.e. subject-specific short calibration) is a critical requirement for future applications. This remains a major challenge when using complex deep learning (DL) models, which typically require large datasets and extensive training to adapt to the user. In this context, DL models that can be quickly adapted to the final user are especially desirable, and this is the reason why the proposed model was trained globally on a population of subjects and then fine-tuned over the final user (unseen subject). To address the aforementioned issues, a novel multidimensional and explainable DL network, *EEGCubeNet*, designed for efficient EEG decoding in motor BCI tasks, was introduced. *EEGCubeNet* aims to significantly reduce calibration time while ensuring high classification performance. The model is first trained on a population-level dataset and then fine-tuned on previously unseen subjects, enabling rapid adaptation to individual users with minimal additional data. *EEGCubeNet* processes EEG data in the form of three-dimensional (3D) matrices that represent projections into the space–frequency–time domain, allowing it to capture the full complexity of spatio-temporal neural dynamics associated with motor intention. To assess the performance of the proposed architecture, an extensive analysis was carried out using EEG recordings collected during motor BCI experiments. In particular, the attention was focused on motor BCIs that decode the preparatory phases of movements (i.e. the phase preceding the onset of movements when the brain is preparing the movement execution).[4] The ability to decode EEG signals during the motor preparation phase offers a promising, and still mostly unexplored, opportunity to better understand the early neural mechanisms underlying movement generation, from initial intention and planning to execution.[5,6] By targeting this early stage, we can gain novel insights into the temporal evolution of motor commands and how the brain transitions from planning to action. From a practical perspective, decoding motor intention at the preparatory level may enable predictive BCIs, capable of predicting the intended movement before it is executed, and possibly implementing it through a device when execution is not possible. BCIs based on motor preparation could also complement and enhance motor imagery (MI)-based systems, which have already demonstrated a remarkable impact in stroke neurorehabilitation.[7,8] By integrating motor preparation signals, MI-based BCIs could benefit from faster response times and increased robustness, particularly in cases where sustained imagery is difficult to maintain. The dataset used for this study originates from a publicly available repository curated by Ofner *et al.*,[15] which features EEG signals captured while subjects performed a series of complex motor tasks involving the same upper limb. Several DL models,[9,10] have been proposed to analyze the aforementioned dataset. The common objective across these studies was to identify the specific movement being executed, accounting for its entire temporal evolution from preparation to initiation

and execution. This work focuses on the preparation phase alone, which is inherently more challenging as neural correlates of motion initiation are missing. Specifically, upcoming sub-movements are predicted from EEG data as early as their preparatory stage. In particular, hand opening (HO) and hand closing (HC) movement initiation are investigated, discriminating them for the neutral resting (RE) state condition. Past approaches that focused on sub-movement prediction relied on spatial and spectral features,[4] or spatial and temporal ones,[11,12] not capturing the overall spatial-, spectral-, temporal-evolution of the dynamics of EEG signals. Ieracitano *et al.*[11,12] required computationally intensive preprocessing steps such as inverse problem solution to reconstruct cortical sources, which can introduce biases as head models averaged over multiple subjects are generally adopted.[13] Beyond performance in EEG decoding, explainability could provide meaningful insights into BCI systems' behavior, particularly as these systems move from research settings to real-world applications. EEG signals are indeed inherently noisy, nonstationary, and subject to inter- and intra-subject variability, making the underlying neural decoding processes complex and often opaque when machine learning models are used. In this context, explainable AI (xAI) methods may help to investigate the internal mechanisms by which BCI systems interpret neural data, thereby enhancing model transparency and trustworthiness.[14] There is a lack of dedicated methods to interpret or explain the decoding mechanism of EEG signals for BCI due to the lack of sufficient literature on BCI systems in the field of xAI.[14]

1.1. *Main contributions*

To fill this gap, a novel 3D xAI Occlusion Sensitivity Analysis, here referred to as *3D xAI-OSA*, is proposed. The outcomes of the proposed classification model, *EEGCubeNet*, are then explained using the *3D xAI-OSA* framework. *3D xAI-OSA* provides 3D relevance maps by simultaneously evaluating the relevance of channels as the considered frequency varies over time. In summary, the following are the main contributions of this paper:

- Development of a novel DL-based EEG decoding model that simultaneously captures the spatial, spectral, and temporal characteristics of EEG signals;
- Achievement of state-of-the-art performance without the need to solve the inverse problem for cortical source reconstruction, thereby avoiding the computational burden and dependence on a head model;
- Comprehensive validation of the proposed model's ability to transfer knowledge acquired during training on multiple subjects (global training) to the adaptation to the unseen final subject (subject-wise fine-tuning), enabling high individual performance with significantly reduced training time;
- Introduction of a novel *3D xAI-OSA* framework that provides simultaneous interpretability of EEG activity across spatial, spectral, and temporal domains.

The rest of this paper is organized as follows. Section 2 describes the current state-of-the-art in the field. Section 3 presents the proposed *EEGCubeNet* and the *3D xAI-OSA* method. Section 4 presents the results. Sections 5 and 6 discuss limitations, potential future extensions, and conclude the paper.

2. Related Work

The dataset here used (Ofner *et al.*[15]) has been employed also in other studies. In particular, Ofner *et al.*[15] assessed classifier performance across different input EEG segment lengths. Their highest accuracy for sub-movement execution versus RE was 80%, while classification between distinct sub-movement executions reached a maximum of 40%. Namazi *et al.*[9] found out that EEG signals exhibited higher complexity during elbow flexion and HC movements in motor execution (ME), while lower complexity was observed during HO and RE conditions in ME. Jeong *et al.*[10] introduced a subject-dependent, section-wise spectral filtering (SSSF) method for decoding MRCP. Using this approach, they performed binary classifications of ME, achieving an average accuracy of 0.72 ± 0.09 for HC versus RE, and 0.76 ± 0.06 for HO versus RE tasks. In a related study, Jeong *et al.*[16] proposed a Hierarchical Flow Convolutional Neural Network (CNN) for a three-class classification task distinguishing between right forearm supination, pronation, and RE. This approach yielded an average classification accuracy of 0.52 ± 0.03. The common objective across these studies was to identify the specific movement being executed, however, the focus of this work is on the preparation phase alone, which is

inherently more challenging as neural correlates of motion initiation are missing. Another work based on support vector machines-based[17] performed the classification of motor EEG and RE states using Fourier-based synchrosqueezing transform (FSST) as a feature extractor and singular value decomposition (SVD) as a feature selection method. Another work[18] proposed graph convolutional network (GCN) as compared to CNN for the classification of four movements of MI. Their model was built on functional connectivity of topological structures and time points.

3. Methodology

The data and source code are available at the following link: https://github.com/AI-Lab-UniRC/EEGCubeNet. The proposed approach involves the following key steps: (1) *Data preprocessing*: EEG signals are projected into the spatial–spectral–temporal domain, forming 3D volumes of shape channels× frequency × time to capture the multidimensional dynamics of brain activity; (2) *Classification*: the 3D volumes are fed into the proposed DL model, here-in referred to as *EEGCubeNet*, for feature extraction and classification (i.e. HC versus RE and HO versus RE); (3) *Fine-tuning*: a global training strategy followed by subject-specific fine-tuning of *EEGCubeNet* is employed; (4) *xAI*: the output of *EEGCubeNet* is interpreted using the proposed *3D xAI-OSA* framework, which provides insights into the spatial, spectral, and temporal relevance of EEG features.

3.1. *EEG preprocessing and dataset construction*

3.1.1. *EEG preprocessing*

In order to evaluate the effectiveness of the proposed method, an in-depth analysis was carried out utilizing EEG data obtained from motor-based BCI experiments. The dataset employed in this investigation is sourced from a publicly accessible repository compiled by Ofner *et al.*[15] The study included a total of 15 healthy participants (mean age 27 ± 5 years), nine female and six male. One participant was excluded from the analysis due to low-quality EEG recordings. Since the specific aim of this work is to discriminate the preparatory phases of sub-movements HC/HO from the absence of movement preparation (i.e. the RE state), a dataset of EEG segments preceding HO/HC movement onset was created, together with segments of EEG signals acquired in the RE state condition. In the work of Ofner *et al.*,[15] EEG data were recorded using 61 active electrodes in conjunction with four 16-channel amplifiers (g.tec medical engineering GmbH, Austria). The right mastoid was used as the reference electrode, and AFz was designated as the ground. The signals underwent band-pass filtering between 0.01 Hz and 200 Hz using an eighth-order Chebyshev filter, followed by a 50 Hz notch filter to suppress power line interference. Data were sampled at a rate of 512 Hz, further details can be found in Ref. 15. During the experiment, participants were seating comfortably with their right arm supported by an anti-gravity exoskeleton (Hocoma, Switzerland). A cue-based movement paradigm was employed, in which subjects performed specific right upper limb actions from a standardized neutral position — defined as the arm extended at 120° with neutral rotation and the hand partially open.[15] The experimental protocol consisted of 10 runs, each comprising six trials. In every trial, participants responded to the requested movement cues. After each movement, the hand was returned to the neutral position before the next cue was presented. Movement onset was determined using motion sensor data embedded in the glove, following the approach described in Ref. 15. To ensure precision, the automatically detected onset times were visually verified across all selected pre-movement intervals. For each detected movement, a 1 s (corresponding to 512 samples) EEG segment was extracted, preceding the initiation of movement. This process yielded 840 EEG segments per movement class (i.e. HO and HC) across 10 runs, 6 trials per run, and 14 subjects. To maintain class balance, an equal number of RE segments was included, resulting in a total of 2520 EEG segments encompassing all three classes: HO, HC, and RE. In summary, the dataset contained 120 EEG trials (thus 120 EEG chans × freq × time 3D matrices), 60 per class (HC or HO) and 60 for RE, per subject.

3.1.2. *Time–frequency analysis of spatially filtered EEGs*

For every EEG segment, to mitigate volume conduction effects and reduce signal correlations between adjacent electrodes, EEG signals are initially spatially filtered. Specifically, a small Laplacian filter

is applied to enhance the local activity at each electrode while attenuating the influence of distant sources, thereby improving the spatial resolution of the EEG signals.[19] After that, spatially filtered EEGs are projected into the time–frequency (TF) domain. This process is schematized in Fig. 1 Spatially filtered EEG signals were then projected into the TF domain using the Continuous Wavelet Transform (CWT).[20] Specifically, for each spatially filtered EEG segment EEG_s, the TF representation was computed for each individual channel ($i = 1, \ldots, 59$), resulting in 59 distinct TF maps. These maps were then stacked to form a 3D matrix with dimensions: channel × frequency × time, capturing EEG dynamics across spatial, spectral, and temporal dimensions. To investigate frequency bands relevant to motor planning and preparation — namely, MRCP (< 5 Hz),[3] sensorimotor rhythms (13–15 Hz),[21] and the β-band (approximately 13–40 Hz), the analysis was focused on the 0.5–40 Hz range. A custom set of 59 pseudo-frequencies spanning this range was generated using the *scal2freq* function in *MATLAB2024b*, matching the number of scalp EEG channels. In the end, 3D maps sized $59 \times 59 \times 512$ were created, and then downsampled to $59 \times 59 \times 128$, representing the spatial–spectral–temporal dynamics of the EEG segment under analysis. Downsampling allowed for a faster computation while not affecting EEG dynamics representation as a sampling rate of 128 Hz is able to follow the variations in the range under analysis (0.5–40 Hz).

3.2. *EEGCubeNet*

The 3D CNNs extend traditional two-dimensional (2D) CNNs by incorporating an additional temporal dimension, making them particularly effective for processing volumetric data such as videos or medical imaging. Unlike 2D CNNs, which apply convolutions only across spatial dimensions (height and width), 3D CNNs perform indeed convolutions over height, width, and time, enabling them to capture both spatial and temporal dependencies simultaneously.[22] In this paper, different 3D CNNs were developed according to a *trial-and-error* approach, as reported in Table 1. Specifically, kernel sizes and strides, along

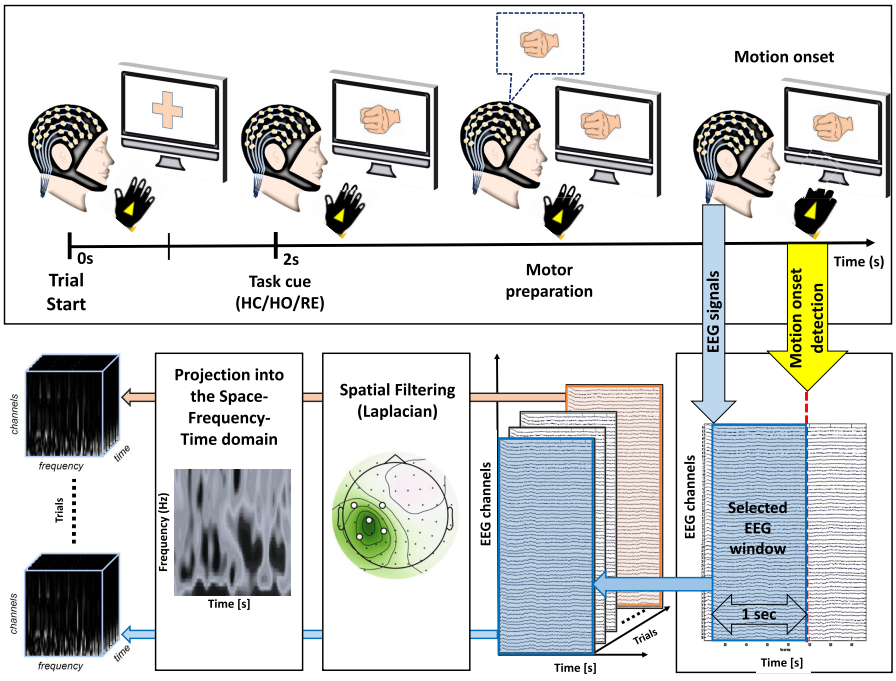

Fig. 1. EEG acquisition, preprocessing, and dataset creation. The figure illustrates the data acquisition paradigm, which is described in detail in Sec. 3.1. EEG segments of 1 s preceding the onset of motion — referred to as pre-motion EEG segments — are extracted, labeled (as HC, HO, or RE), and stored in a dataset. These EEG signals undergo spatial filtering using the Laplacian method, followed by TF analysis via the CWT. The resulting TF representations are structured into volumes organized by channel, frequency, and time. These volumes are then labeled accordingly and stored for further analysis.

Table 1. Architectures of different 3D-CNNs. It is worth noting that *Model 7* refers to the configuration providing the best performance (on average, across the k-fold cross-validation runs), which was adopted in this study and herein referred to as *EEGCubeNet*.

Layer	Parameter	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8	Model 9	Model 10	Model 11	Model 12
Conv1	Kernels	32	16	8	8	8	8	**16**	16	32	32	32	32
	Kernel size	$3\times3\times3$	$3\times3\times3$	$3\times3\times3$	$3\times3\times3$	$3\times5\times5$	$5\times5\times5$	$3\times3\times3$	$5\times5\times5$	$3\times3\times3$	$3\times3\times3$	$5\times5\times3$	$7\times7\times7$
	Stride	$2\times2\times2$	$2\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$2\times2\times2$	$2\times2\times2$	$1\times2\times2$	$2\times2\times2$
	Padding	$2\times2\times2$	$2\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$2\times2\times2$	$1\times1\times1$	$2\times2\times2$	$3\times3\times3$
Conv2	Kernels	—	32	16	16	16	16	**32**	32	64	64	64	64
	Kernel size	—	$3\times3\times3$	$3\times3\times3$	$3\times3\times3$	$3\times5\times5$	$5\times5\times5$	$3\times3\times3$	$5\times5\times5$	$3\times3\times3$	$3\times3\times3$	$5\times5\times3$	$7\times7\times7$
	Stride	—	$2\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$2\times2\times2$	$2\times2\times2$	$1\times2\times2$	$2\times2\times2$
	Padding	—	$2\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$1\times2\times2$	$2\times2\times2$	$2\times2\times2$	$2\times2\times2$	$3\times3\times3$
Conv3	Kernels	—	—	32	32	32	32	**64**	64	128	128	128	128
	Kernel size	—	—	$3\times3\times3$	$3\times3\times3$	$3\times5\times3$	$5\times5\times5$	$3\times3\times3$	$3\times3\times3$	$3\times3\times3$	$3\times3\times3$	$5\times5\times3$	$7\times7\times7$
	Stride	—	—	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$2\times2\times2$	$2\times2\times2$	$1\times2\times2$	$2\times2\times2$
	Padding	—	—	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$2\times2\times2$	$3\times3\times3$
Conv4	Kernels	—	—	—	64	64	64	**128**	128	256	—	—	
	Kernel size	—	—	—	$3\times3\times3$	$3\times3\times3$	$3\times3\times3$	$3\times3\times3$	$3\times3\times3$	$3\times3\times3$	—	—	
	Stride	—	—	—	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$2\times2\times2$	—	—	
	Padding	—	—	—	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	$1\times1\times1$	—	—	
Performance	Accuracy	59.40 ± 7.43	61.33 ± 6.66	70.00 ± 7.33	71.66 ± 7.20	71.33 ± 6.70	65.83 ± 8.45	$\mathbf{75.00\pm5.10}$	73.33 ± 6.33	59.16 ± 6.73	71.66 ± 6.44	69.16 ± 7.33	71.66 ± 6.77
	K-Cohen	28.33 ± 6.33	35.70 ± 5.33	41.66 ± 6.40	43.33 ± 6.20	46.33 ± 5.80	31.66 ± 5.90	$\mathbf{50.00\pm5.50}$	46.46 ± 5.88	18.33 ± 9.33	43.33 ± 6.12	38.33 ± 6.33	43.33 ± 7.29

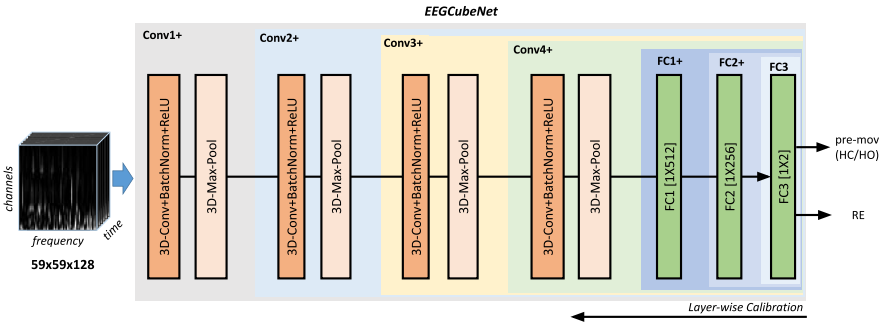

Fig. 2. The architecture of *EEGCubeNet* model and its fine-tuning mechanism, where layer-wise fine-tuning is illustrated using different colors. The shaded area with different colors represents the progressive fine-tuning process, starting from the last layer and extending to the entire model.

with the number of layers, were systematically varied to explore different architectural configurations. In this study, *Model 7* is employed for the binary classification tasks (HC versus RE and HO versus RE) and it is here referred to as *EEGCubeNet*. It is to be noted that Model 7 was selected because it achieved the highest average performance across multiple runs of the k-fold validation procedure, specifically in terms of both accuracy and Cohen's Kappa. Importantly, this selection was not based on a single metric or run but reflects superior performance across multiple folds and metrics. The architecture is shown in Fig. 2 and it consists of four 3D convolutional layers, each followed by batch normalization, ReLU activation, and max pooling. The input to the model is the 3D volume channels $\times$ frequency $\times$ time sized $59 \times 59 \times 128$. The first convolutional layer (Conv1) applies 16 filters, producing 16 feature maps of size $31 \times 31 \times 128$. The 3D max-pooling layer (Max-Pool1) downsamples the 16 features maps to $15 \times 15 \times 64$. The second convolutional layer (Conv2) has 32 filters and produces 32 feature maps of size $9 \times 9 \times 64$, that are further downsampled to $4 \times 4 \times 32$ by means of the max-pooling operation (Max-Pool2). The third convolutional layer (Conv3) consists of 64 filters with output dimensions of $4 \times 4 \times 32$. The third max-pooling layer (Max-Pool3) generates 64 features maps sized $2 \times 2 \times 16$; while the final convolutional layer (Conv4) contains 128 filters, resulting in 128 feature maps of the same size ($2 \times 2 \times 16$). The last max-pooling layer (Max-Pool4) outputs 128 features vectors of size $1 \times 1 \times 8$. These are flattened and passed through three fully connected layers. The first fully connected layer (FC1) consists of 512 neurons, followed by FC2 with 256 neurons, and finally, FC3 with 2 neurons for binary classification (i.e. HC versus RE and HO versus RE). It is to be noted that for Conv1 and Conv2 the kernel size is $3 \times 3 \times 3$, strides $1 \times 2 \times 2$ and padding is $1 \times 2 \times 2$; while, for Conv3 and Conv4 the kernel size is $3 \times 3 \times 3$, stride is $1 \times 1 \times 1$ and padding is $1 \times 1 \times 1$. In addition, the four max-pooling layers have kernel size of $2 \times 2 \times 2$ and stride of $2 \times 2 \times 2$. The model was implemented in PyTorch[23] and trained with the Adaptive Moment Estimation (Adam) optimizer,[24] using learning rate of 0.001, weight decay of 0.9, and a batch size of 8, for up to 50 epochs and adopting the early stopping strategy on an NVIDIA RTX 4000 Ada Generation installed on a processor with an Intel Xeon (R) CPU @2.30 GHz and a RAM of 125 GB. To mitigate overfitting and enhance robustness, a dropout rate of 25% was also applied after the pooling layers.

3.3. *From global training to subject-specific fine-tuning*

In order to develop a model that requires a reduced training time on the subject of interest, thus enabling a short calibration, an in-depth evaluation of the proposed architecture, *EEGCubeNet*, was carried out. To ensure robust evaluation, a Leave-One-Subject-Out Cross-Validation (LOSO-CV) approach was employed. For this purpose, a two-stage approach was adopted: initially, the model is trained on $N-1$ subjects (global training), followed by fine-tuning on the excluded subject (subject-specific fine-tuning). In the global training phase, the training was conducted using the data from the $N-1$ subjects. The trained model was then adapted to the

left-out subject in the subject-specific fine-tuning stage. A k-fold cross-validation was carried out which was crucial to evaluate the adaptation performance on the target subject more reliably and to minimize the risk of overfitting due to the limited amount of subject-specific data. This two-stage strategy with global training and subsequent subject-specific fine-tuning, was explicitly designed to enhance the robustness and generalizability of our results, despite the constraints imposed by the dataset size. As illustrated in Fig. 2, the model architecture is shaded with different colors, and each colored box decodes the fine-tuning process of the boxed layers. The evaluation of the fine-tuning process was performed incrementally, starting from the last fully connected layer (FC3) and extending back to the first convolutional layer (Conv1). In the first step of the evaluation, only the FC3 layer was involved in the fine-tuning. In the second step, both FC2 and FC3 layers were adapted. These two layers are grouped within a box labeled *FC2+*, indicating that FC2 and all subsequent layers were fine-tuned, and so on, progressively including more layers until the entire model (starting from the last fully connected layer (FC3) and extending back to the first convolutional layer (Conv1+) was adapted. The goal of this layer-by-layer performance evaluation was to analyze the trade-off between the number of model parameters required for fine-tuning and the accuracy achieved within a given time. This approach allows for estimating the necessary computational resources to fine-tune the model efficiently, enabling the decoding and classification of new subject data with minimal resource requirements. In the context of EEG classification, this results in minimizing the calibration time required for the model to accurately classify signals from a new subject.

3.4. *Explainability analysis: 3D xAI-OSA*

In the proposed *3D xAI-OSA* approach, the 3D maps (channels × frequency × time) extracted from EEG signals are occluded to compute the spatial, spectral, and temporal relevance of specific regions over time. Input volumes are systematically occluded across spatial, spectral, and temporal dimensions using varying sizes of occlusion 3D masks. Figure 3 illustrates the schematic of the proposed *3D xAI-OSA* approach. The masking process, illustrated in Fig. 3,

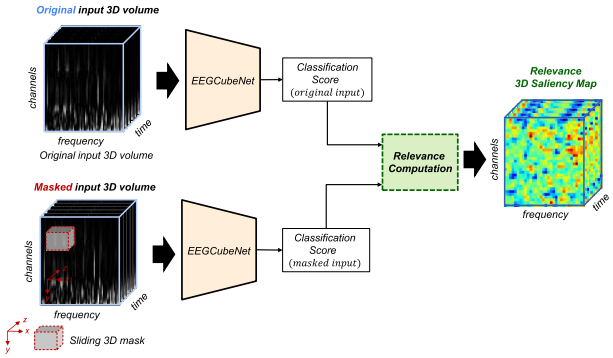

Fig. 3. Proposed *3D xAI-OSA* approach. The EEG volumetric data (channels × frequency × time) are systematically occluded using a 3D sliding mask across all three dimensions and fed into the trained *EEGCubeNet* model. For each occluded mask, the classification score is compared with that of the original input. The relevance is then quantified using the SAD between the two classification scores. By iteratively scanning the entire input volume, a 3D saliency map is generated, highlighting the relevance of each region.

generates occluded 3D regions by assigning zero values to the masked areas within the 3D input matrix (channels × frequency × time). Given a predefined mask size, a sliding 3D mask is applied across the input volume, systematically scanning it along the spatial (channels), spectral (frequency), and temporal (time samples) dimensions. A mask is defined as follows:

$$M_k = \{x^k_{\text{start}}, x^k_{\text{end}}, y^k_{\text{start}}, y^k_{\text{end}}, t^k_{\text{start}}, t^k_{\text{end}}, R_k\}, \quad (1)$$

where M_k represents the kth mask, $x^k_{\text{start}}, x^k_{\text{end}}$ define the horizontal range (frequency) of the mask, y^k_{start}, y^k_{end} define the vertical range of the mask (channel), $t^k_{\text{start}}, t^k_{\text{end}}$ define the temporal range (number of frames) of the mask. R_k is the relevance score assigned to the masked region based on the predictions of the model and the change of degree in the outcome. Once the masked input is processed by *EEGCubeNet*, class probability scores are obtained via the softmax function for both the original input and the masked input. To assign the relevance to the masked pixels/regions of the input, Sum of Absolute Difference (SAD) was used.[25] SAD is an appropriate metric for quantifying changes in the degree of outcome in model predictions, especially when the focus is on the overall magnitude of difference rather than subtle shape variations between distributions. SAD provides a simple, easily interpretable measure of the total difference. The input volumes (channels×

frequency × time) represent EEG signal dynamics across the spatial, spectral, and temporal domains. Masking a region of the input volume means occluding specific channels within a given frequency and time range. Understanding the impact of masking that region on classification allows to assess the relevance of those channels, in that frequency band, during that time interval for the classification task (premovement versus RE). To ensure a smooth and continuous relevance evaluation, an overlapping sliding window approach was adopted with a stride equal to $T/2$. In this way, the relevance of a given channel (c), frequency (f), and time point (t) is assessed multiple times, depending on how often it is occluded by the sliding mask. The average of these relevance values is then computed to obtain a mean relevance score for that specific element (c,f,t) of the input volume. Specifically, when a (c,f,t) element is occluded by a mask and the relevance of the occluded region is evaluated, the resulting relevance score is assigned to all (c,f,t) elements affected by the occlusion. During the full input volume scanning procedure, each (c,f,t) element will be occluded a total of n times, and n corresponding relevance scores will be computed for it. At the end of the procedure, these scores are averaged (i.e. summed and divided by n), yielding a single mean relevance value for that element. Finally, a 3D matrix representing the relevance of each channel and frequency over time is generated. To ensure a robust relevance evaluation, an analysis across different mask sizes was performed. Masks sized $2 \times 2 \times T$, $4 \times 4 \times T$, $8 \times 8 \times T$, $16 \times 16 \times T$, $24 \times 24 \times T$, with $T = 32$ and $T = 64$ were used and their impact on the model's decisions was evaluated. Given a mask size, the mask slides over the entire input volume (in an overlapping way), occluding a region of the volume at each step. To complete the scan of the entire volume, the mask moves through M different positions. For each of the M occluded positions, the probability score of both the original input volume and the occluded one, is evaluated. The mask size that maximizes the difference between the probability score of the original input volume and that of the occluded input volume will be selected for the explainable analysis. Given a relevance 3D saliency map representing the relevance of channels over the frequencies and over the time, in order to evaluate the relevance within the main EEG frequency bands (delta (0–4 Hz); theta (4–8 Hz); alpha (8–13 Hz); and beta (13–40 Hz)), the relevance was averaged over the frequencies (of a given sub-band) and over the time. Namely, the relevance map is 3D channel × frequency × time matrix (sized $59 \times 59 \times 128$). An average is computed over the time dimension, resulting in a channel × frequency map (with dimensions 59×59). Next, to quantify the relevance within each band (delta, theta, alpha, and beta), the frequencies corresponding to each of the four bands are identified, and then the columns of the matrix associated with the same band are averaged. This yields a channel × band matrix (size 59×4), which represents the relevance of each channel in each band. Each column in this matrix represents the relevance of channels in a specific sub-band, and will be represented as a topographical map in Sec. 4.2.

3.5. *Performance metrics*

The performance of the proposed *EEGCubeNet* was evaluated using standard metrics, namely: Accuracy $= \frac{TP+TN}{TP+TN+FP+FN}$; Precision $= \frac{TP}{TP+FP}$; Recall $= \frac{TP}{TP+FN}$; $F1\text{-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision}+\text{Recall}}$; and Cohen's Kappa coefficient (K-Cohen) $\kappa = \frac{p_o - p_e}{1-p_e}$ (where p_o is the observed agreement and p_e is the expected agreement by chance) which quantifies the agreement between predictions and true labels. While accuracy only measures the proportion of correct predictions, K-Cohen accounts for the level of agreement that could occur by chance. It is to be noted that TP and TN represent true positives and true negatives, and FP and FN represent false positives and false negatives. Specifically, TP denotes EEG trials from the premovement (HC or HO) category correctly classified as premovement, whereas TN refers to RE trials correctly classified as RE. Conversely, FP represents RE trials misclassified as premovement, and FN refers to premovement trials misclassified as RE.

4. Results

In this section, the classification performance of *EEGCubeNet* is reported. The proposed method, EEGCubeNet, was compared with 3D-EEGNet.[26] The 3D-EEGNet was implemented from scratch, following the structure described in Ref. 26. Both

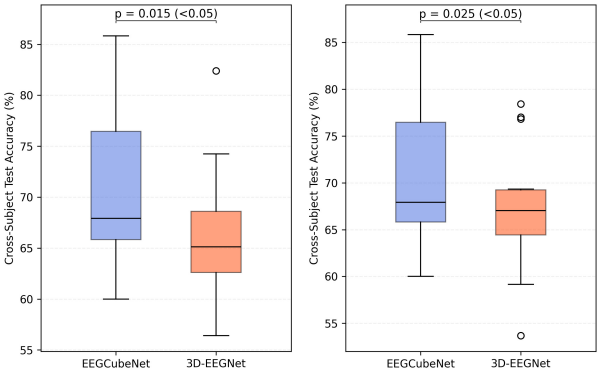

Fig. 4. Boxplots of the test accuracies achieved by EEGCubeNet and 3D-EEGNet in the HC versus RE (left subplot) and in the HO versus RE (right boxplot) classification.

methods were assessed using a LOSO-CV strategy: each model was trained on data from $N-1$ subjects and tested on the remaining subject. The comparative results are presented in Fig. 4, which displays boxplots of the test accuracies obtained by both methods. To assess the statistical significance, a Wilcoxon signed-rank test was performed, revealing that EEGNet significantly outperformed 3D-EEGNet ($p < 0.05$).

4.1. *Analysis of global training to subject-specific fine-tuning*

This section describes the performance of the *EEGCubeNet* model when globally trained on $N-1$ subjects and then fine-tuned on the specific remaining subject (LOSO-CV approach) for the two tasks, HC versus RE and HO versus RE. The results are reported in Tables 2 and 3, respectively. Each column in Tables 2 and 3, represents a subject and reports the performance of the globally trained model fine-tuned on that subject. Each row corresponds to a specific fine-tuned model, as described in Sec. 3.3. As detailed in the same section, a 10-fold cross-validation was applied during both the global training and the subject-specific fine-tuning stages. Accuracy, Precision, Recall, $F1$-score, detailed in Sec. 3.5, are reported as mean $\pm$ standard deviation.

For HC versus RE task (Table 2), the best average performance was achieved in the Conv3+ setting, while the worst performance was observed for FC3. The average accuracy for Conv3+ was 89.56 ± 4.29 across subjects. Additionally, the best individual subject performance was recorded for subject S02, achieving an accuracy 97.78 ± 2.72, with the lowest standard deviation, reflecting the robustness of the model across the k-fold runs. Similarly, Table 3 presents the model performance for HO versus RE task. The best performance was observed for Conv2+ layer and the worst for FC3. The average accuracy of Conv2+ for this task was 89.06 ± 4.86. A gradually increasing trend of performance can be observed when fine-tuning was performed gradually on more layers, resulting in enhanced performance from FC3 to Conv2+. From Tables 2 and 3, one can observe that the average results across all subjects indicate an improvement in performance as more layers of the model are fine-tuned. The advantages of fine-tuning process are analyzed when fewer or more layers are fine-tuned, and the corresponding insights in terms of training time and number or trained parameters are presented in Sec. 4.1.2.

Further insights on models' performance are illustrated in Figs. 5 and 6 in terms of K-Cohen, as described in Sec. 3.5. Figures 5 and 6 illustrate the boxplots for the HC versus RE task and HO versus RE task, respectively, representing the K-Cohen coefficient for all the fine-tuned models under analysis. The y-axes presents the K-Cohen values and x-axes presents the subjects on which the models were fine-tuned. On the right, the K-Cohen ranges of significance are reported, from "no agreement" to "very high agreement". The results provided by the different fine-tuned models are represented with different colors. Given a subject, the K-Cohen produced by the different fine-tuned models cases are plotted adjacent to each other. Further, horizontal lines are drawn to show the average K-Cohen score for every specific fine-tuned model using the same color of the boxplot. It can be observed that the average K-Cohen falls in the range "substantial agreement" for all the Conv4+ to Conv1+ (all layers) models, with average K-Cohen of models Conv3+, Conv2+, Conv1+ tending towards to the "very high agreement" range. When only the fully connected layers were fine-tuned (models FC3, FC2+, FC1+), a "moderate agreement" was achieved, on average.

4.1.1. *Analysis of the extracted features*

To further investigate the performance of *EEGCubeNet* model, t-distributed stochastic neighbor embedding (t-SNE) analysis was performed to visualize the separability of the feature maps and assess

Table 2. Results of global training to subject-specific fine-tuning for the HC versus RE classification task.

Tasks	Metric	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10	S11	S12	S13	S14	Average
FC3	Accuracy	60.0 ± 5.0	75.8 ± 6.9	84.1 ± 2.5	73.3 ± 7.2	53.3 ± 11.3	70.8 ± 8.5	68.3 ± 6.2	63.3 ± 7.6	69.1 ± 3.8	85.0 ± 6.2	78.3 ± 5.5	69.1 ± 5.3	62.5 ± 8.5	91.6 ± 8.3	71.7 ± 6.7
	Precision	61.0 ± 5.7	76.6 ± 6.0	84.2 ± 2.8	77.3 ± 9.0	54.8 ± 18.3	71.5 ± 8.1	72.5 ± 9.5	66.3 ± 8.3	76.8 ± 5.6	85.6 ± 6.2	80.4 ± 5.8	71.3 ± 5.7	62.7 ± 8.7	92.5 ± 7.8	74.1 ± 8.0
	Recall	60.0 ± 5.0	75.8 ± 6.9	84.1 ± 2.5	73.3 ± 7.2	53.3 ± 11.3	70.8 ± 8.5	68.3 ± 6.2	63.3 ± 7.6	69.1 ± 3.8	85.0 ± 6.2	78.3 ± 5.5	69.1 ± 5.3	62.5 ± 8.5	91.6 ± 8.3	71.7 ± 6.7
	$F1$-score	59.1 ± 5.1	75.5 ± 7.3	84.1 ± 2.4	72.4 ± 7.2	50.0 ± 12.5	70.3 ± 9.0	67.2 ± 5.7	61.5 ± 9.0	66.8 ± 4.7	84.9 ± 6.2	77.9 ± 5.7	68.3 ± 5.5	62.2 ± 8.7	91.5 ± 8.4	70.8 ± 6.9
FC2+	Accuracy	70.5 ± 3.5	85.5 ± 2.9	85.0 ± 3.5	65.0 ± 1.8	68.6 ± 7.0	83.0 ± 3.8	75.5 ± 1.6	60.5 ± 6.5	87.7 ± 2.8	85.5 ± 3.4	61.6 ± 4.6	64.4 ± 3.2	75.8 ± 3.9	73.6 ± 3.3	74.2 ± 3.6
	Precision	73.7 ± 4.6	85.7 ± 2.9	86.4 ± 2.6	65.4 ± 1.9	69.2 ± 7.4	83.6 ± 4.1	77.6 ± 2.2	61.5 ± 6.7	87.9 ± 2.8	86.2 ± 3.1	61.9 ± 4.5	67.3 ± 4.3	76.2 ± 3.8	73.8 ± 3.1	75.0 ± 3.9
	Recall	70.5 ± 3.5	85.5 ± 2.9	85.0 ± 3.5	65.0 ± 1.8	68.6 ± 7.0	83.0 ± 3.8	75.5 ± 1.6	60.5 ± 6.5	87.7 ± 2.8	85.5 ± 3.4	61.6 ± 4.6	64.4 ± 3.2	75.8 ± 3.9	73.6 ± 3.3	74.2 ± 3.6
	$F1$-score	69.6 ± 3.5	85.5 ± 3.0	84.8 ± 3.7	64.7 ± 1.8	68.3 ± 7.0	82.9 ± 3.8	75.1 ± 1.6	58.9 ± 8.2	87.7 ± 2.8	85.4 ± 3.5	61.4 ± 4.7	63.0 ± 2.9	75.7 ± 4.0	73.5 ± 3.4	73.8 ± 3.9
FC1+	Accuracy	73.6 ± 3.9	85.5 ± 2.0	85.5 ± 2.9	76.3 ± 8.7	70.5 ± 2.8	84.7 ± 3.1	74.7 ± 2.3	59.7 ± 4.1	89.7 ± 2.5	90.5 ± 4.3	63.0 ± 4.1	65.8 ± 4.4	80.8 ± 3.6	75.0 ± 3.9	77.8 ± 2.9
	Precision	73.8 ± 3.8	85.7 ± 2.0	86.0 ± 2.5	77.5 ± 9.2	70.6 ± 2.8	85.9 ± 3.1	76.3 ± 2.6	62.2 ± 6.4	90.2 ± 2.7	90.9 ± 4.2	64.4 ± 3.7	67.7 ± 5.3	81.1 ± 3.3	75.8 ± 3.6	77.6 ± 2.5
	Recall	73.6 ± 3.9	85.5 ± 2.0	85.5 ± 2.9	76.3 ± 8.7	70.5 ± 2.8	84.7 ± 3.1	74.7 ± 2.3	59.7 ± 4.1	89.7 ± 2.5	90.5 ± 4.3	63.0 ± 4.1	65.8 ± 4.4	80.8 ± 3.6	75.0 ± 3.9	77.8 ± 2.9
	$F1$-score	73.5 ± 4.0	85.5 ± 2.0	85.4 ± 3.0	76.1 ± 8.7	70.5 ± 2.8	84.5 ± 3.1	74.3 ± 2.3	57.7 ± 5.7	89.6 ± 2.4	90.5 ± 4.3	61.9 ± 5.0	65.0 ± 4.4	80.7 ± 3.6	74.7 ± 4.0	77.6 ± 2.9
Conv4+	Accuracy	72.5 ± 6.5	91.6 ± 6.4	93.3 ± 6.2	83.3 ± 3.7	85.8 ± 7.5	91.6 ± 8.3	80.8 ± 11.2	85.0 ± 8.9	91.6 ± 7.4	90.8 ± 5.8	83.3 ± 3.7	95.8 ± 5.5	76.6 ± 12.8	92.3 ± 6.3	86.4 ± 7.3
	Precision	77.4 ± 7.5	92.9 ± 5.4	94.6 ± 4.7	84.7 ± 4.2	88.7 ± 6.1	92.8 ± 6.9	82.0 ± 11.2	86.6 ± 9.0	93.5 ± 5.1	92.2 ± 4.9	86.4 ± 4.1	96.6 ± 4.4	80.7 ± 12.7	91.5 ± 5.8	88.4 ± 6.7
	Recall	72.5 ± 6.5	91.6 ± 6.4	93.3 ± 6.2	83.3 ± 3.7	85.8 ± 7.5	91.6 ± 8.3	80.8 ± 11.2	85.0 ± 8.9	91.6 ± 7.4	90.8 ± 5.8	83.3 ± 3.7	95.8 ± 5.5	76.6 ± 12.8	92.6 ± 6.3	86.3 ± 7.3
	$F1$-score	71.1 ± 7.3	91.5 ± 6.5	93.2 ± 6.4	83.1 ± 3.7	85.4 ± 7.8	91.4 ± 8.6	80.5 ± 11.3	84.8 ± 9.0	91.4 ± 7.8	90.7 ± 5.9	82.9 ± 3.7	95.7 ± 5.7	75.7 ± 13.2	91.5 ± 6.4	85.7 ± 7.6
Conv3+	Accuracy	91.6 ± 3.5	97.7 ± 2.7	95.8 ± 1.8	88.8 ± 3.2	78.8 ± 6.2	88.3 ± 3.4	88.3 ± 4.4	76.1 ± 10.1	92.7 ± 2.2	94.4 ± 2.7	82.5 ± 5.2	83.8 ± 3.2	91.3 ± 4.0	84.4 ± 4.6	**89.5 ± 4.2**
	Precision	92.1 ± 3.4	98.0 ± 2.3	96.1 ± 1.7	90.4 ± 2.9	83.9 ± 4.3	89.8 ± 2.2	89.5 ± 3.1	80.3 ± 8.3	93.2 ± 2.2	95.0 ± 2.2	83.6 ± 6.1	85.7 ± 3.1	92.4 ± 3.6	85.2 ± 4.7	**89.6 ± 3.1**
	Recall	91.6 ± 3.5	97.7 ± 2.7	95.8 ± 1.8	88.8 ± 3.2	78.8 ± 6.2	88.3 ± 3.4	88.3 ± 4.4	76.1 ± 10.1	92.7 ± 2.2	94.4 ± 2.7	82.5 ± 5.2	83.8 ± 3.2	91.3 ± 4.0	84.4 ± 4.6	**89.4 ± 4.1**
	$F1$-score	91.6 ± 3.5	97.7 ± 2.7	95.8 ± 1.8	88.7 ± 3.3	77.8 ± 7.2	88.1 ± 3.6	88.1 ± 4.6	74.7 ± 12.1	92.7 ± 2.2	94.4 ± 2.8	82.4 ± 5.2	83.6 ± 3.3	91.3 ± 4.0	84.3 ± 4.1	**89.4 ± 4.1**
Conv2+	Accuracy	90.8 ± 4.6	95.2 ± 2.7	94.4 ± 3.9	87.2 ± 1.8	79.1 ± 4.1	85.8 ± 5.6	90.0 ± 5.8	71.1 ± 11.8	92.2 ± 2.7	95.0 ± 2.4	88.0 ± 4.3	84.4 ± 6.9	85.5 ± 7.3	85.5 ± 3.8	87.0 ± 4.9
	Precision	91.4 ± 4.1	95.8 ± 2.2	95.1 ± 3.1	89.1 ± 2.3	84.2 ± 3.7	87.7 ± 4.4	91.6 ± 4.5	79.2 ± 8.6	92.8 ± 2.7	95.4 ± 2.1	89.5 ± 4.1	87.4 ± 4.9	89.1 ± 4.2	86.3 ± 3.6	89.0 ± 4.2
	Recall	90.8 ± 4.6	95.2 ± 2.7	94.4 ± 3.9	87.2 ± 1.8	79.1 ± 4.1	85.8 ± 5.6	90.0 ± 5.8	71.1 ± 11.8	92.2 ± 2.7	95.0 ± 2.4	88.0 ± 4.3	84.4 ± 6.9	85.5 ± 7.3	85.5 ± 3.8	87.0 ± 4.9
	$F1$-score	90.7 ± 4.7	95.2 ± 2.8	94.4 ± 3.9	87.0 ± 1.8	78.3 ± 4.5	85.5 ± 5.9	89.8 ± 6.0	67.8 ± 14.8	92.1 ± 2.7	94.9 ± 2.4	87.9 ± 4.3	83.9 ± 7.3	84.9 ± 8.0	85.4 ± 3.9	86.5 ± 5.3
Conv1+ (all layers)	Accuracy	90.5 ± 3.7	96.1 ± 1.8	93.0 ± 6.4	88.0 ± 3.3	82.7 ± 5.9	83.8 ± 4.7	93.3 ± 4.8	65.5 ± 10.3	90.2 ± 4.5	93.3 ± 5.1	87.2 ± 3.9	88.6 ± 3.8	88.8 ± 5.5	85.0 ± 4.1	87.9 ± 4.9
	Precision	91.0 ± 3.5	96.3 ± 1.7	94.2 ± 4.7	89.8 ± 3.2	86.8 ± 3.1	86.5 ± 4.3	93.8 ± 4.4	77.4 ± 5.9	91.1 ± 3.6	94.3 ± 3.7	89.4 ± 3.2	90.2 ± 3.1	91.1 ± 3.4	86.0 ± 3.8	90.0 ± 3.7
	Recall	90.5 ± 3.7	96.1 ± 1.8	93.0 ± 6.4	88.0 ± 3.3	82.7 ± 5.9	83.8 ± 4.7	93.3 ± 4.8	65.5 ± 10.3	90.2 ± 4.5	93.3 ± 5.1	87.2 ± 3.9	88.6 ± 3.8	88.8 ± 5.5	85.0 ± 4.1	87.9 ± 4.9
	$F1$-score	90.5 ± 3.8	96.1 ± 1.8	92.9 ± 6.7	87.9 ± 3.3	82.0 ± 6.6	83.5 ± 4.9	93.3 ± 4.8	60.0 ± 14.5	90.1 ± 4.6	93.2 ± 5.3	87.0 ± 4.1	88.4 ± 3.9	88.6 ± 5.9	84.8 ± 4.2	87.3 ± 5.5

Table 3. Results of global training to subject-specific fine-tuning for the HO versus RE classification task.

Tasks	Metric	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10	S11	S12	S13	S14	Average
FC3	Accuracy	82.5 ± 2.5	73.3 ± 5.0	59.1 ± 5.8	50.8 ± 8.7	50.0 ± 12.3	69.1 ± 8.3	68.3 ± 3.3	70.0 ± 8.5	61.6 ± 5.5	85.8 ± 8.3	67.5 ± 16.0	70.0 ± 9.2	62.5 ± 12.5	89.1 ± 7.5	69.8 ± 6.5
	Precision	82.5 ± 2.2	73.9 ± 4.9	60.7 ± 5.9	50.0 ± 9.7	52.3 ± 15.9	70.5 ± 9.2	71.7 ± 5.8	71.4 ± 9.2	63.8 ± 5.8	89.5 ± 4.5	71.1 ± 17.7	75.0 ± 12.1	62.6 ± 13.1	90.9 ± 6.3	72.4 ± 5.8
	Recall	82.5 ± 2.5	73.3 ± 5.0	59.1 ± 5.8	50.8 ± 8.7	50.0 ± 12.3	69.1 ± 8.3	68.3 ± 3.3	70.0 ± 8.5	61.6 ± 5.5	85.8 ± 8.3	67.5 ± 16.0	70.0 ± 9.2	62.5 ± 12.5	89.1 ± 7.5	69.2 ± 4.5
	$F1$-score	82.4 ± 2.5	73.1 ± 5.1	57.6 ± 6.1	49.5 ± 9.9	47.7 ± 11.9	68.7 ± 8.3	67.2 ± 3.0	68.9 ± 9.9	60.0 ± 5.8	85.1 ± 9.5	65.7 ± 17.2	68.9 ± 8.8	62.0 ± 12.9	88.9 ± 7.8	69.8 ± 5.6
FC2+	Accuracy	68.6 ± 3.3	83.3 ± 2.5	81.9 ± 3.8	46.4 ± 6.8	62.2 ± 3.8	77.2 ± 4.1	73.3 ± 2.8	76.4 ± 5.5	80.8 ± 2.3	84.7 ± 1.4	56.7 ± 6.1	74.4 ± 2.7	73.9 ± 4.5	69.7 ± 7.1	73.0 ± 4.1
	Precision	69.1 ± 3.4	83.6 ± 2.6	82.6 ± 3.7	46.0 ± 7.4	62.5 ± 3.7	79.0 ± 4.9	73.5 ± 2.9	77.4 ± 5.0	81.6 ± 2.3	85.3 ± 1.6	56.7 ± 6.3	76.1 ± 2.8	74.4 ± 4.5	70.1 ± 7.1	73.6 ± 4.2
	Recall	68.6 ± 3.3	83.3 ± 2.5	81.9 ± 3.8	46.4 ± 6.8	62.2 ± 3.8	77.2 ± 4.1	73.3 ± 2.8	76.4 ± 5.5	80.8 ± 2.3	84.7 ± 1.4	56.7 ± 6.1	74.4 ± 2.7	73.9 ± 4.5	69.7 ± 7.1	73.0 ± 4.1
	$F1$-score	68.4 ± 3.3	83.3 ± 2.5	81.9 ± 3.9	45.4 ± 7.1	62.0 ± 4.0	76.9 ± 4.0	73.3 ± 2.9	76.1 ± 5.8	80.7 ± 2.3	84.7 ± 1.4	55.9 ± 7.1	74.0 ± 2.9	73.8 ± 4.6	69.5 ± 7.2	72.7 ± 4.2
FC1+	Accuracy	71.4 ± 2.8	80.8 ± 2.3	88.1 ± 4.1	53.9 ± 7.5	64.2 ± 3.8	75.0 ± 3.3	76.4 ± 5.6	79.2 ± 5.6	86.4 ± 5.0	89.2 ± 2.3	56.9 ± 6.4	75.0 ± 2.2	70.6 ± 3.1	68.9 ± 3.0	74.9 ± 3.9
	Precision	72.6 ± 3.3	81.0 ± 2.4	88.1 ± 4.1	54.1 ± 7.7	64.7 ± 4.4	76.7 ± 3.4	76.7 ± 5.7	80.3 ± 5.1	87.3 ± 4.9	89.7 ± 2.4	57.2 ± 6.9	76.1 ± 2.6	71.3 ± 3.8	69.5 ± 3.4	75.2 ± 4.0
	Recall	71.4 ± 2.8	80.8 ± 2.3	88.1 ± 4.1	53.9 ± 7.5	64.2 ± 3.8	75.0 ± 3.3	76.4 ± 5.6	79.2 ± 5.6	86.4 ± 5.0	89.2 ± 2.3	56.9 ± 6.4	75.0 ± 2.2	70.6 ± 3.1	68.9 ± 3.0	74.9 ± 3.9
	$F1$-score	71.0 ± 2.8	80.8 ± 2.3	88.1 ± 4.1	53.5 ± 7.5	63.9 ± 3.7	74.6 ± 3.4	76.3 ± 5.6	78.9 ± 5.8	86.3 ± 5.1	89.1 ± 2.3	56.2 ± 7.0	74.7 ± 2.2	70.3 ± 3.0	68.7 ± 3.0	74.7 ± 3.9
Conv4+	Accuracy	82.5 ± 2.5	86.7 ± 5.5	94.2 ± 7.5	87.5 ± 7.7	59.2 ± 7.9	85.8 ± 3.8	90.8 ± 6.9	82.5 ± 2.5	80.8 ± 7.5	90.8 ± 2.5	89.2 ± 5.3	96.7 ± 5.5	95.0 ± 5.5	65.8 ± 5.8	83.8 ± 4.5
	Precision	83.0 ± 2.7	88.2 ± 5.6	94.7 ± 7.2	88.6 ± 7.4	63.7 ± 12.6	89.1 ± 2.5	91.6 ± 6.9	86.7 ± 1.7	81.6 ± 7.7	92.3 ± 1.6	91.4 ± 3.1	97.3 ± 4.3	95.5 ± 5.3	66.9 ± 6.1	86.2 ± 5.0
	Recall	82.5 ± 2.5	86.7 ± 5.5	94.2 ± 7.5	87.5 ± 7.7	59.2 ± 7.9	85.8 ± 3.8	90.8 ± 6.9	82.5 ± 2.5	80.8 ± 7.5	90.8 ± 2.5	89.2 ± 5.3	96.7 ± 5.5	95.0 ± 5.5	65.8 ± 5.8	85.4 ± 5.2
	$F1$-score	82.4 ± 2.5	86.5 ± 5.6	94.1 ± 7.6	87.4 ± 7.8	57.4 ± 6.8	85.5 ± 4.0	90.8 ± 7.0	82.0 ± 2.9	80.7 ± 7.6	90.7 ± 2.6	88.9 ± 5.8	96.6 ± 5.7	95.0 ± 5.5	65.3 ± 5.9	84.2 ± 4.8
Conv3+	Accuracy	78.9 ± 4.3	89.7 ± 3.7	95.3 ± 2.5	95.6 ± 2.8	82.5 ± 7.2	89.4 ± 4.8	93.6 ± 3.3	88.3 ± 4.3	96.1 ± 2.8	92.5 ± 2.2	76.1 ± 5.6	85.3 ± 3.5	80.0 ± 5.2	85.3 ± 5.8	87.2 ± 4.5
	Precision	82.1 ± 3.7	90.3 ± 3.9	95.7 ± 2.0	95.8 ± 2.7	84.2 ± 7.5	90.1 ± 4.6	93.8 ± 3.3	89.3 ± 3.5	96.5 ± 2.5	93.5 ± 1.6	80.9 ± 5.7	86.9 ± 4.0	80.3 ± 5.4	85.8 ± 5.6	88.4 ± 4.1
	Recall	78.9 ± 4.3	89.7 ± 3.7	95.3 ± 2.5	95.6 ± 2.8	82.5 ± 7.2	89.4 ± 4.8	93.6 ± 3.3	88.3 ± 4.3	96.1 ± 2.8	92.5 ± 2.2	76.1 ± 5.6	85.3 ± 3.5	80.0 ± 5.2	85.3 ± 5.8	87.2 ± 4.5
	$F1$-score	78.3 ± 4.8	89.7 ± 3.7	95.3 ± 2.5	95.6 ± 2.8	82.3 ± 7.4	89.4 ± 4.8	93.6 ± 3.3	88.2 ± 4.4	96.1 ± 2.9	92.5 ± 2.2	75.1 ± 6.4	85.1 ± 3.5	80.0 ± 5.3	85.2 ± 5.9	87.0 ± 4.6
Conv2+	Accuracy	77.5 ± 5.0	90.8 ± 3.3	93.9 ± 2.4	96.7 ± 2.4	87.2 ± 6.8	90.8 ± 8.6	92.2 ± 3.5	88.3 ± 6.2	95.6 ± 3.3	90.6 ± 2.8	80.0 ± 3.9	85.6 ± 4.9	82.8 ± 9.4	85.0 ± 5.2	**89.1 ± 4.9**
	Precision	82.6 ± 4.6	91.5 ± 2.8	94.6 ± 1.9	96.9 ± 2.3	87.9 ± 7.0	92.8 ± 4.9	92.7 ± 3.2	90.1 ± 4.7	95.7 ± 3.2	92.2 ± 2.0	83.9 ± 3.2	87.5 ± 4.6	86.2 ± 7.4	86.5 ± 4.0	**90.2 ± 3.9**
	Recall	77.5 ± 5.0	90.8 ± 3.3	93.9 ± 2.4	96.7 ± 2.4	87.2 ± 6.8	90.8 ± 8.6	92.2 ± 3.5	88.3 ± 6.2	95.6 ± 3.3	90.6 ± 2.8	80.0 ± 3.9	85.6 ± 4.9	82.8 ± 9.4	85.0 ± 5.2	**89.1 ± 4.9**
	$F1$-score	76.5 ± 5.5	90.8 ± 3.4	93.9 ± 2.5	96.7 ± 2.4	87.2 ± 6.8	90.4 ± 9.8	92.2 ± 3.5	88.1 ± 6.4	95.6 ± 3.3	90.5 ± 2.9	79.3 ± 4.3	85.3 ± 5.1	82.1 ± 10.3	84.8 ± 5.4	**88.9 ± 5.2**
Conv1+ (all layers)	Accuracy	80.6 ± 5.1	90.0 ± 4.0	93.3 ± 1.4	95.6 ± 2.6	88.6 ± 4.9	86.7 ± 4.6	93.6 ± 4.1	88.6 ± 5.9	95.8 ± 3.3	90.8 ± 2.2	81.1 ± 4.3	83.9 ± 6.7	77.2 ± 12.5	88.9 ± 4.7	88.1 ± 4.6
	Precision	83.7 ± 5.1	90.7 ± 3.5	94.1 ± 1.0	95.8 ± 2.5	89.7 ± 4.8	89.0 ± 3.0	94.2 ± 3.2	90.4 ± 4.5	96.0 ± 3.2	92.3 ± 1.5	84.6 ± 4.4	87.9 ± 3.9	83.5 ± 7.9	89.7 ± 3.8	90.3 ± 3.6
	Recall	80.6 ± 5.1	90.0 ± 4.0	93.3 ± 1.4	95.6 ± 2.6	88.6 ± 4.9	86.7 ± 4.6	93.6 ± 4.1	88.6 ± 5.9	95.8 ± 3.3	90.8 ± 2.2	81.1 ± 4.3	83.9 ± 6.7	77.2 ± 12.5	88.9 ± 4.7	88.1 ± 4.6
	$F1$-score	80.1 ± 5.4	89.9 ± 4.0	93.3 ± 1.4	95.6 ± 2.6	88.5 ± 4.9	86.4 ± 4.9	93.6 ± 4.3	88.4 ± 6.1	95.8 ± 3.4	90.7 ± 2.2	80.6 ± 4.4	83.2 ± 7.3	75.1 ± 14.6	88.8 ± 4.8	88.0 ± 4.9

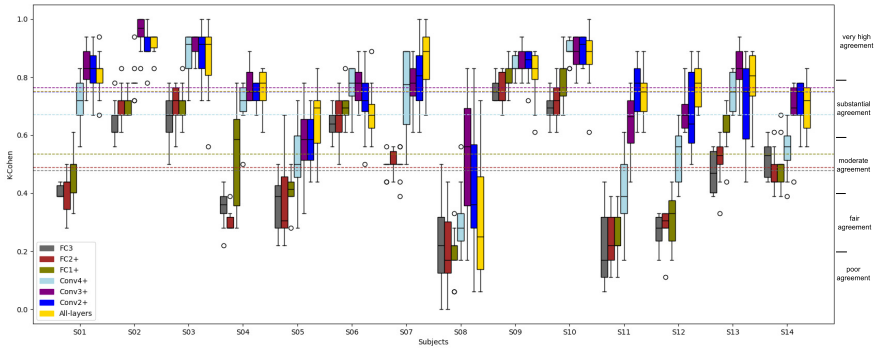

Fig. 5. Cohen Kappa performance of *EEGCubeNet* model for HC versus RE task when different layers are fine-tuned, and plotted with boxplots.

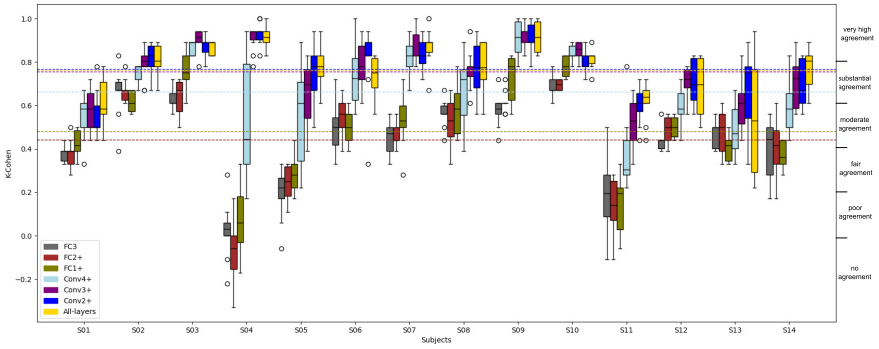

Fig. 6. Cohen Kappa performance of *EEGCubeNet* model for HO versus RE task when different layers are fine-tuned, and plotted with boxplots.

the effectiveness of different fine-tuning strategies in distinguishing between classes. Specifically, t-SNE projects high-dimensional data into a lower-dimensional feature space by converting pairwise distances — using the Chebyshev distance in this case — into joint probability distributions. It then minimizes the Kullback–Leibler divergence between the joint distributions of the high-dimensional and low-dimensional representations to preserve local structure.[27] The visualizations of t-SNE plots are presented in Figs. 7 and 8 for the tasks HC versus RE and HO versus RE, respectively, for a sample subject (S10). The separability of the features projections can be visually evaluated for the different *EEGCubeNet* fine-tuned models (from FC3 to Conv1+). A clear trend can be observed in the figures, progressing from right to left, where the separation between classes improves as fine-tuning is extended to deep layers, endorsing the positive impact of progressive fine-tuning. This pattern is consistent for both HC versus RE and HO versus RE tasks. A peak of average separability is detectable for Conv2+ and Conv3+ fine-tuned versions of *EEGCubeNet*, in agreement with classification performances reported in Tables 2 and 3, validating the effectiveness of the proposed global to subject-specific approach exploiting the knowledge acquired in the global training stage to the subject-specific fine-tuning stage. To further analyze the t-SNE results of other subjects, we provided plots for subject S02 in the *Supplementary Materials*. Supplementary material is provided at the following link https://github.com/AI-Lab-UniRC/EEGCubeNet

4.1.2. *Performance versus model complexity*

The choice of the model and to what extent it should be fine-tuned for the new subjects depends not only on the classification performance but also on some other key aspects such as the model complexity in terms of both size and training time. To provide a comparison (tradeoff) between performance (Accuracy) and model complexity (training time and number of parameters),

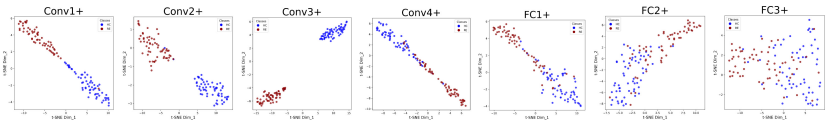

Fig. 7. The high-dimensional projections of feature maps of *EEGCubeNet* model are visualized with t-SNE for the task of HC versus RE for subject S10. In each plot, the feature maps from the last fully connected layer are visualized, when fine-tuning was performed for different layers of the trained model for left out subject. Each plot represents to specific layer, Conv1+ (all layers) to FC3 can be read from left to right.

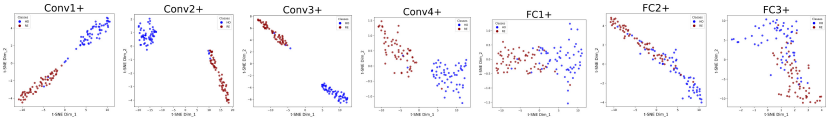

Fig. 8. The high-dimensional projections of feature maps of *EEGCubeNet* model are visualized with t-SNE for the task of HO versus RE for subject S10. In each plot, the feature maps from the last fully connected layer are visualized, when fine-tuning was performed for different layers of the trained model for left out subject. Each plot represents to specific layer, Conv1+ (all layers) to FC3 can be read from left to right.

in Fig. 9 every model was represented as a circle with a different color, allowing for a clear and intuitive interpretation of the trends. The center of the circle corresponds to the number of parameters in log scale (x-axis) and to the average accuracy provided by that model (y-axis). The radius of the circle is proportional to the training time necessary to complete one run of the 10-fold training process. Fine-tuning the complete model on the whole dataset of a single subject, over 10 folds, took around 2.5 min on average. This analysis highlights a consistent pattern: as more layers of the model were fine-tuned, and the number of trainable parameters increased, the classification accuracy also increased, from FC3 to Conv3+. From Conv3+ onwards, the trend slows down and reverses. The Conv+ models, in which the convolutional layers were fine-tuned, achieved better performance. This improvement comes with an increase in the number of parameters, nevertheless, the parameter count remains within a range suitable for potential future deployment. These characteristics make the model particularly suitable for real BCI applications, where a short model fine-tuning time enables a short calibration, which is crucial since calibration requires the active cooperation of the subject, who must undergo data acquisition sessions aimed at tailoring the model to his/her own brain waves. Furthermore, the results highlight that a single, well-calibrated model could be effectively used to decode EEG activity for new incoming subjects without requiring extensive retraining. This endorses the model's practicality and adaptability in dynamic and resource-constrained environments.

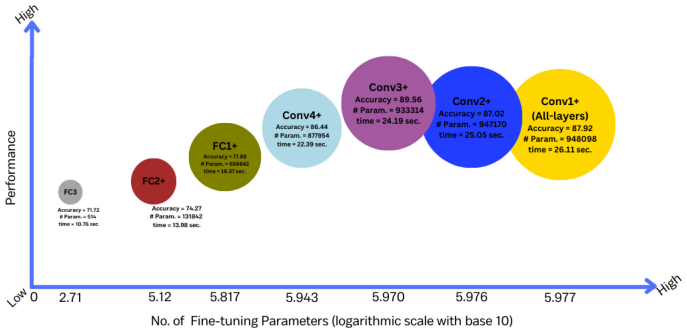

Fig. 9. Performance versus calibration time versus model size (number of parameters) of *EEGCubeNet* model. The number of parameters are mentioned as #Param. in the circles of the specific models (the different fine-tuned versions of *EEGCubeNet* from FC3 to Conv1+, for details, refer to Fig. 2). The number of parameters are reported in log scale with base 10 on x-axes to improve the figure readability.

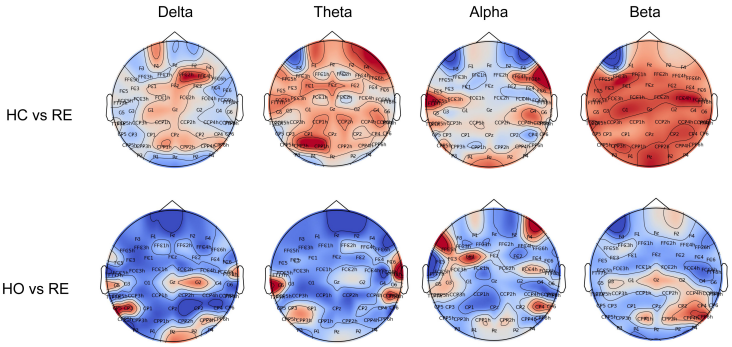

Fig. 10. Topographical maps showing the average relevance of channels across the frequency bands (delta, theta, alpha, and beta).

4.2. *Evaluation of 3D xAI-OSA*

This section presents the results of the explanation of the outcomes of *EEGCubeNet* model in classifying EEG data for the tasks of HC versus RE and HO versus RE. The proposed explainable analysis, *3D xAI-OSA*, was described in Sec. 3.4. Specifically, this approach generates and applies multiple masks of different dimensions to each EEG trial (channels × frequency × time 3D matrices), and, in return, a 3D relevance matrix was obtained containing the relevance of the input volume along the spatial- spectral- and temporal dimensions. As explained in Sec. 3.4, to assess the relevance within key EEG frequency bands (delta: 0–4 Hz, theta: 4–8 Hz, alpha: 8–13 Hz, and beta: 13–40 Hz), the 3D saliency map, representing channel relevance across frequency and time, was utilized to observe the average behavior of channels within each specific frequency sub-band. Specifically, the relevance of EEG channels within the four primary frequency bands (delta, theta, alpha, and beta) was estimated across trials for each subject. The average relevance values were then visualized using topographical plots, as explained in Sec. 3.4, using the open-source Python package MNE, to facilitate the interpretation of spatial patterns. Figure 10 presents examples of the topographical maps achieved for subject 01, while the *Supplementary Material* reports the corresponding data for all subjects. The red region in the topographical maps shows higher relevance/activation and blue shows lower relevance. In addition, the most relevant channels are identified for each trial, and the average relevance ± standard deviation across trials is displayed in the form of histograms. Figure 11 shows the histograms for subject 01, with corresponding data for all subjects provided in the *Supplementary Material*.

5. Discussion

In this study, *EEGCubeNet*, a novel multidimensional and explainable DL framework for decoding EEG signals was introduced and validated on EEGs recorded during motor BCI experiments. The key motivation behind this work was to address one of the central challenges in applying DL models to EEG-based BCI: the substantial inter-subject variability in EEG signals, which necessitates user-specific calibration. Calibration procedures are often time consuming, especially when DL models are involved, as they typically require a large amount of data for training, which results in the need of long EEG recordings to fine-tune the model on the final user. In this context, developing models that support rapid subject-specific fine-tuning is a critical step toward more deployable and user-friendly DL-based BCI systems. To meet this need, *EEGCubeNet* was first trained on a population-level dataset and subsequently fine-tuned on unseen subject. In particu-

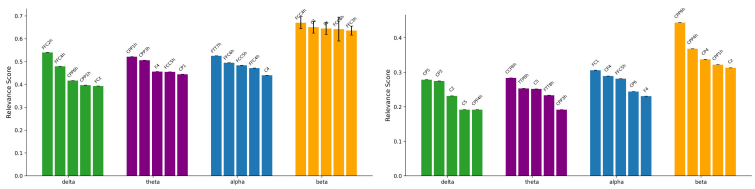

Fig. 11. Average relevance of the top-5 relevant channels across the frequency bands in the HC versus RE classification (left subplot) and HO versus RE classification (right subplot). Each bar represents the mean relevance of a given channel in a specific sub-band.

lar, EEG derived from a public dataset acquired during motor BCI experiments were analyzed, with the aim of characterizing the HO motor initiation (pre-HO versus RE classification) and the HC motor initiation (pre-HC versus RE classification). *EEGCubeNet* was fed with 3D matrices representing the projections of EEG signals into the space–frequency–time domain. This strategy demonstrated the feasibility of efficiently adapting a complex DL model to new users, thereby reducing the training burden while preserving classification performance. Beyond performance, model explainability can play an important role in ensuring transparency and interpretability of neural decoding processes, especially in clinical and assistive settings. In Ref. 11, the authors interpreted CNN-based EEG decoding through OSA at source level, revealing that the central region, the right temporal zone of the premotor cortex, and the primary motor cortex played a crucial role in EEG sources classification. It is worth mentioning that in OSA, the sensitivity masks are typically applied to 2D images to interpret classification decisions. A recent study extended this approach to visually explain the decision-making process of 3D CNNs by incorporating a temporal dimension into occlusion sensitivity analysis.[28] However, applying this strategy to 3D volumetric EEG data poses challenges due to the continuous nature of non-stationary and noisy EEG signals distributed across multiple frames with a variability in and between the subjects. Furthermore, using optical flow masking would likely result in occluding large portions of each channel × frequency frame, leading to the loss of information necessary for identifying highly relevant channels and their frequency variations over time. To this end, a novel 3D explainability framework was also introduced, referred to as *3D xAI-OSA*, based on the Occlusion Sensitivity Analysis. This technique enabled a deeper understanding of the spatial–spectral–temporal patterns contributing to the network's decisions, offering insights into how preparatory brain activity associated with hand movements is represented in EEG data.

5.1. *Impact of the proposed approach*

Overall, *EEGCubeNet*, coupled with the *3D xAI-OSA* framework, represents a promising direction for creating robust, adaptable, and interpretable BCI systems. The advantages of *EEGCubeNet* over previous models[4,11,12] are manifold:

- It processes EEG signals at the scalp level without reconstructing cortical sources, which provides an advantage in terms of computational complexity. Furthermore, it does not require the use of a head model, which might not accurately reflect the brain structure of the subject under examination, potentially introducing bias[11];
- Thanks to the global-to-subject-specific approach, it leverages knowledge acquired from the subject population during the training phase to achieve comparable performance with reduced training time on the final user;
- It decodes EEG signals with comparable performance while simultaneously taking into account the evolution of their dynamics in space, frequency, and time, not just space and frequency[4] or space and time[11];
- *EEGCubeNet*, when coupled with the *xAI-3D-OSA* method, enabled explanation of the model's outcomes, allowing the detection of a subject-specific trend among the most relevant channels as frequency varies during motion initiation of HC and HO.

As regards the training time, in Ref. 12 which is a double branch DL model, the training time was of about 30 min for one branch and 1.5 h for the other branch, per subject for 10 folds. In Ref. 11, the training time was of 10 min per subject for 10 folds, excluding the time necessary to perform the inverse problem solution to reconstruct cortical sources. In Ref. 4, the average training time was of approximately 9 min per subject, for 10 folds, but it is worth emphasizing that the approach adopted in this paper is not comparable as classification was carried out at frame level (one frame is a channel × frequency map and one EEG trial of 1 s includes 512 frames), and not at trial level, like in this work. The average training time of *EEGCubeNet* for one subject was of 4.35 min for 10 folds and inference time was 23 s; and the average training time of 3D EEGNet for 10 folds was 65 min and inference time was 33 s.

5.2. *Limitations and future works*

Among the limitations of this work, the small number of available trials per class per subject should be noted. Future studies should consider larger datasets, also for the purpose of validating the model in decoding more complex tasks. This will also enable the extension of the explainability analysis, whose effectiveness depends on the classifier's ability to generalize.

Second, movement's onset were marked by processing the data collected by the motion sensors embedded in the glove that the participant used to wear during the experiment. Motion data collected through the glove are smooth and do not allow one to detect onset instantaneously, which means the epochs used for training may have captured the early milliseconds of motion implementation, which may have caused, in principle, the similar activation patterns. Among the future potentials of the proposed method is its applicability to online frameworks, given the small number of parameters that need to be updated to adapt the model to the final user. Since *EEGCubeNet* is designed to analyze 3D matrices representing EEG signals in the space–frequency–time domain, it is capable of effectively capturing the complex mutual interactions among the temporal evolution of EEG signal features across both the spatial domain (i.e. across different channels) and the spectral domain (i.e. across varying frequencies). This inherent capability makes it reasonable to assume that extending the model to other applications, such as attention detection, emotion recognition, sleep monitoring, neurological disorders diagnosis, could be relatively straightforward. However, in order to validate this assumption, it is essential to adapt and rigorously test the model on appropriate datasets that are specific to each application domain. This direction will be one of the primary goals of future research, especially considering the growing importance of these topics in the fields of computational neuroscience and neurotechnology.

6. Conclusions

This study presented *EEGCubeNet*, a novel explainable DL framework designed to decode EEG signals with a particular focus on minimizing subject-specific adaptation. By projecting EEG data into the space–frequency–time domain and adopting a global-to-subject-specific training approach, *EEGCubeNet* demonstrated strong decoding performance while significantly reducing the training time required for new users. The model's architecture, which operates directly on scalp-level EEGs without source reconstruction, offers computational efficiency and avoids potential bias introduced by head models. Coupled with the newly proposed *xAI-3D-OSA* explainability framework, *EEGCubeNet* also provides transparent insights into the neural patterns driving its predictions, making it especially suitable for clinical and assistive applications. In addition, *EEGCubeNet*'s adaptability, efficiency, and interpretability make it a good candidate for integration into real-time BCI systems.

Acknowledgments

This work was supported in part by the European Union — Next Generation EU — PRIN 2022 program, Italian Ministry of University and Research (MUR), project title: "EXEGETE: Explainable Generative Deep Learning Methods for Medical Signal and Image Processing" (project code: 2022ENK9LS, CUP: C53D23003650001); in part by the European Union — Next Generation EU, PRIN 2022 PNRR call, under the project "Interactive digital twin solutions for cardiovascular disease Management, PReventiOn and treatment leVeraging the internet of things and Edge intelligence paradigms — IMPROVE" (project code: P2022NH2CK, CUP: C53D23007960001); in part by the Fa.Per.M.E project (project code: T3-AN-15, CUP C33C22000390006, CUP MASTER H53C22000640006) funded by the Italian Ministry of Health; in part by the Next Generation EU — Italian NRRP, Mission 4, Component 2, Investment 1.5, call for the creation and strengthening of "Innovation Ecosystems", building "Territorial R&D Leaders" (Directorial Decree No. 2021/3277) — project Tech4-You — Technologies for climate change adaptation and quality of life improvement, No. ECS0000009 (CUP: C33C22000290006). This work reflects only the authors' views and opinions, neither the Ministry for University and Research nor the European Commission can be considered responsible for them.

ORCID

Muhammad Suffian https://orcid.org/0000-0002-1946-285X
Cosimo Ieracitano https://orcid.org/0000-0001-7890-2897
Francesco C. Morabito https://orcid.org/0000-0003-0734-9136
Nadia Mammone https://orcid.org/0000-0003-4962-3500

References

1. F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy and F. Yger, A review of classification algorithms for EEG-based brain–computer interfaces: A 10 year update, *J. Neural Eng.* **15**(3) (2018) 031005.
2. P. L. Nunez and R. Srinivasan, *Electric Fields of the Brain: The Neurophysics of EEG* (Oxford University Press, New York, 2006).
3. J. N. Spring, N. Place, F. Borrani, B. Kayser and J. Barral, Movement-related cortical potential amplitude reduction after cycling exercise relates to the extent of neuromuscular fatigue, *Front. Hum. Neurosci.* **10** (2016) 257.
4. N. Mammone, C. Ieracitano, R. Spataro, C. Guger, W. Cho and F. C. Morabito, A few-shot transfer learning approach for motion intention decoding from electroencephalographic signals, *Int. J. Neural Syst.* **34**(2) (2024) 2350068.
5. S. T. Grafton and C. M. Tipper, Decoding intention: A neuroergonomic perspective, *NeuroImage* **59**(1) (2012) 14–24.
6. G. Pfurtscheller, C. Neuper, C. Andrew and G. Edlinger, Foot and hand area mu rhythms, *Int. J. Psychophysiol.* **26**(1–3) (1997) 121–135.
7. M. S. Kim, H. Park, I. Kwon, K. O. An, H. Kim, G. Park, W. Hyung, C. H. Im and J. H. Shin, Efficacy of brain-computer interface training with motor imagery-contingent feedback in improving upper limb function and neuroplasticity among persons with chronic stroke: A double-blinded, parallel-group, randomized controlled trial, *J. NeuroEng. Rehabil.* **22**(1) (2025) 1.
8. S. Sieghartsleitner, M. Sebastián-Romagosa, R. Ortner, W. Cho and C. Guger, BCIs for stroke rehabilitation, in *Brain-Computer Interfaces* (Elsevier, 2025), pp. 131–150.
9. H. Namazi, T. S. Ala and V. Kulish, Decoding of upper limb movement by fractal analysis of electroencephalogram (EEG) signal, *Fractals* **26**(05) (2018) 1850081.
10. J. H. Jeong, N. S. Kwak, C. Guan and S. W. Lee, Decoding movement-related cortical potentials based on subject-dependent and section-wise spectral filtering, *IEEE Trans. Neural Syst. Rehabil. Eng.* **28**(3) (2020) 687–698.
11. C. Ieracitano, N. Mammone, A. Hussain and F. C. Morabito, A novel explainable machine learning approach for EEG-based brain-computer interface systems, *Neural Comput. Appl.* **34**(14) (2022) 11347–11360.
12. C. Ieracitano, F. C. Morabito, A. Hussain and N. Mammone, A hybrid-domain deep learning-based BCI for discriminating hand motion planning from EEG sources, *Int. J. Neural Syst.* **31**(09) (2021) 2150038.
13. S. Haufe and A. Ewald, A simulation framework for benchmarking EEG-based brain connectivity estimation methodologies, *Brain Topogr.* **32** (2019) 625–642.
14. P. Rajpura, H. Cecotti and Y. K. Meena, Explainable artificial intelligence approaches for brain–computer interfaces: A review and design space, *J. Neural Eng.* **21** (2024) 041003.
15. P. Ofner, A. Schwarz, J. Pereira and G. R. Müller-Putz, Upper limb movements can be decoded from the time-domain of low-frequency EEG, *PLoS One* **12**(8) (2017) e0182578.
16. J. H. Jeong, B. H. Lee, D. H. Lee, Y. D. Yun and S. W. Lee, EEG classification of forearm movement imagery using a hierarchical flow convolutional neural network, *IEEE Access* **8** (2020) 66941–66950.
17. N. Karakullukcu and B. Yilmaz, Detection of movement intention in EEG-based brain-computer interfaces using Fourier-based synchrosqueezing transform, *Int. J. Neural Syst.* **32**(01) (2022) 2150059.
18. N. Feng, F. Hu, H. Wang and B. Zhou, Motor intention decoding from the upper limb by graph convolutional network based on functional connectivity, *Int. J. Neural Syst.* **31**(12) (2021) 2150047.
19. D. J. McFarland, L. M. McCane, S. V. David and J. R. Wolpaw, Spatial filter selection for EEG-based communication, *Electroencephalogr. Clin. Neurophysiol.* **103**(3) (1997) 386–394.
20. I. Daubechies, *Ten Lectures on Wavelets* (Society for Industrial and Applied Mathematics, 1992).
21. C. Neuper, M. Wörtz and G. Pfurtscheller, Erd/ers patterns reflecting sensorimotor activation and deactivation, *Prog. Brain Res.* **159** (2006) 211–222.
22. D. Tran, L. Bourdev, R. Fergus, L. Torresani and M. Paluri, Learning spatiotemporal features with 3D convolutional networks, in *Proc. IEEE Int. Conf. Computer Vision* (IEEE, 2015), pp. 4489–4497.
23. A. Paszke *et al.*, PyTorch: An imperative style, high-performance deep learning library, in *Proc. 33rd Int. Conf. Neural Information Processing Systems* (Curran Associates, 2019), pp. 8026–8037.
24. D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, preprint (2014), arXiv:1412.6980.
25. M. C. López-De-Los-Mozos and J. A. Mesa, The sum of absolute differences on a network: Algorithm and comparison with other equality measures, *INFOR: Inf. Syst. Oper. Res.* **41**(2) (2003) 195–210.
26. D. Park, H. Park, S. Kim, S. Choo, S. Lee, C. S. Nam and J. Y. Jung, Spatio-temporal explanation of 3D-EEGNet for motor imagery EEG classification using permutation and saliency, *IEEE Trans. Neural Syst. Rehabil. Eng.* **31** (2023) 4504–4513.
27. G. E. Hinton and S. Roweis, Stochastic neighbor embedding, in *Proc. 16th Int. Conf. Neural Information Processing Systems* (MIT Press, 2002), pp. 857–864.
28. T. Uchiyama, N. Sogi, K. Niinuma and K. Fukui, Visually explaining 3D-CNN predictions for video classification with an adaptive occlusion sensitivity analysis, in *Proc. IEEE/CVF Winter Conf. Applications of Computer Vision* (IEEE, 2023), pp. 1513–1522.