AXISBRAIN: CROSS-AXIS CONSISTENCY FOR TOKEN-WISE RESIDUAL MODULATION IN EEG FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Electroencephalography (EEG) foundation models aim to learn transferable representations across tasks, subjects, and recording configurations. Recent approaches often pursue improvements through model scaling, complex pretraining pipelines, or anatomical priors. These strategies can introduce additional computational costs, training dependencies, or structural metadata requirements. We introduce AxisBrain, which uses cross-axis consistency to guide token-wise residual refinement while limiting these additional costs and requirements. At each layer, learned-query pooling across channels and temporal mean pooling within each channel produce complementary context descriptors. The sigmoid-transformed cosine similarity between their projected descriptors yields a bounded consistency score that modulates the recombined spatial and temporal attention response at each channel–time token. The 7.46M-parameter encoder is trained end to end using masked raw EEG patch reconstruction, without separately pretrained components or explicit anatomical metadata. Across ten datasets spanning seven task families, AxisBrain achieves a mean balanced accuracy of 75.67%, compared with literature-reported means of 73.25% for CBraMod and 75.42% for PATCHCODE-Large. Controlled ablations further indicate that both the learned modulation scores and their token-wise assignments contribute to performance. These results support cross-axis consistency as an effective basis for token-wise residual refinement in EEG foundation models.

1 INTRODUCTION

EEG non-invasively measures brain electrical activity with millisecond-scale temporal resolution, supporting applications ranging from neurological disorder detection and sleep staging to brain–computer interfaces. However, learning representations that transfer across datasets remains challenging because of subject variability, non-stationary noise, differences in electrode placement and channel configuration, and task-dependent responses. EEG foundation models address these challenges by pretraining on large unlabeled corpora and transferring the resulting representations to heterogeneous downstream tasks. Recent EEG foundation models have advanced rapidly along three broad axes.

First, model capacity has expanded from compact encoders with fewer than 5M parameters, such as CBraMod and CSBrain (Wang et al., 2025a; Zhou et al., 2025), to model variants exceeding 100M parameters in the scaling families of LaBraM (Jiang et al., 2024), REVE (El Ouahidi et al., 2025), LUNA (Döner et al., 2025), and Uni-NTFM (Chen et al., 2026).

Second, pretraining pipelines have become more elaborate, with CodeBrain learning a discrete tokenizer before sequence modeling (Ma et al., 2026), PATCHCODE separating discrete-token learning from representation learning through three complementary predictive pathways (Yu et al., 2026), and EmBrace fusing representations from multiple pretrained EEG models (Jia et al., 2026).

Third, models increasingly encode structural priors. REVE uses the actual 3D coordinates of the electrode associated with each EEG patch as positional information (El Ouahidi et al., 2025); THD-BAR constructs a brain-topology hierarchy by grouping electrodes according to standard placement systems, spatial proximity, and corresponding brain regions (Yang et al., 2025); NeurIPT uses physical electrode coordinates and lobe-level pooling (Fang et al., 2025); and PATCHCODE uses anatomical

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

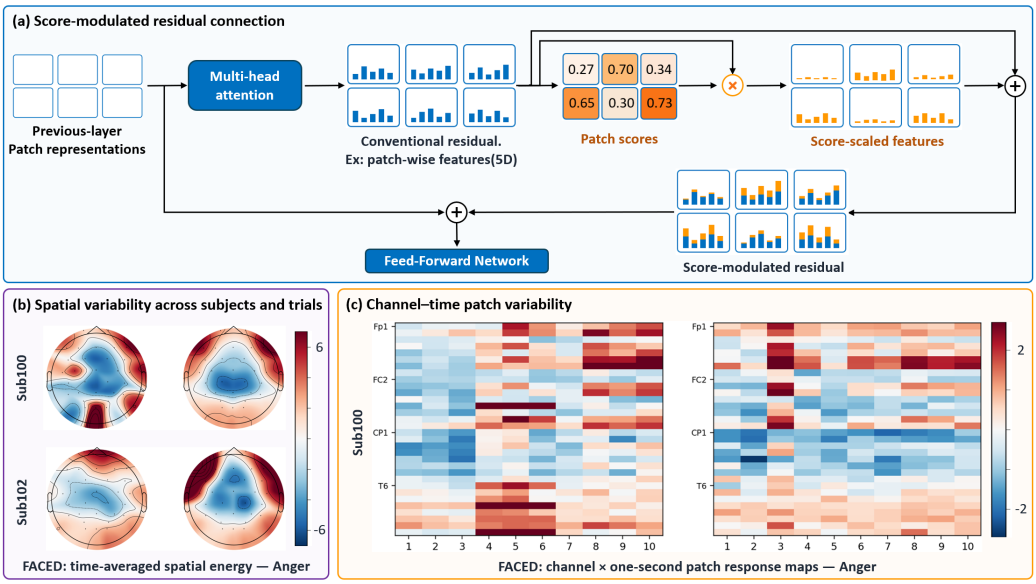


Figure 1: Motivation for data-dependent residual modulation. (a) Score-modulated residual updates. (b) Time-averaged relative channel-energy topographies. For each 10-s segment, channel energy is computed as the log-transformed mean squared amplitude and centered by the median across scalp channels. (c) Channel–time energy maps computed over non-overlapping 1-s patches and robustly standardized within each sample using the median and median absolute deviation. Panels (b) and (c) show the same segment position from two distinct Anger-eliciting videos for each of two FACED participants, illustrating subject- and trial-dependent spatial and temporal variability.

regions to guide both tokenization and interaction biases (Yu et al., 2026). Such priors inject useful domain knowledge into EEG representation learning.

These three directions, however, introduce practical trade-offs. Increased model capacity raises the cost of pretraining, hyperparameter exploration, downstream fine-tuning, and on-device deployment. Multi-stage pretraining pipelines require separately trained tokenizers, teachers, or auxiliary modules and are consequently more difficult to optimize. Structural priors can be sensitive to recording conditions because actual electrode positions may deviate from their nominal coordinates and are difficult to reproduce precisely across subjects and recording sessions. Moreover, assigning each electrode to a single region may oversimplify boundary electrodes whose signals can reflect activity associated with multiple neighboring regions, while spatial proximity does not necessarily imply functional similarity across tasks or individuals.

Although the preceding directions remain valuable, their limitations highlight the need to strengthen EEG representation learning while minimizing similar practical burdens and reliance on structural assumptions. Transformer encoders constitute the predominant backbone architecture in current EEG foundation models (Liu et al., 2026), making their residual paths a natural target for architectural improvement. Self-attention produces input-dependent token updates, but the standard residual connection adds them without a separate token-wise multiplier. Prior work has modulated information flow using input-dependent gates or masks (Srivastava et al., 2015; Hu et al., 2018; Woo et al., 2018; Wang et al., 2017), input-independent layer- or channel-wise residual scales (Bachlechner et al., 2021; Touvron et al., 2021), and condition-dependent but token-shared residual scales (Peebles & Xie, 2023). Coordinate Attention generates feature weights from direction-specific pooled descriptors (Hou et al., 2021), while CodeBrain applies a gated unit to concatenated global and local EEG features (Ma et al., 2026).

Against this background, the key question is not whether to scale a residual update, but what EEG-derived relation should determine its multiplier at each channel–time token. Raw magnitude, predefined electrodes, or fixed temporal intervals may not provide a reliable basis because informative patterns vary across tasks, subjects, sessions, and trials, as illustrated in Figure 1(b,c). Moreover,

token representations evolve across layers, making weights derived from fixed input-level properties potentially less suitable in deeper latent spaces. AxisBrain addresses this question through a relational construction: at each layer, a time-indexed cross-channel descriptor and a channel-indexed temporal descriptor are projected into a shared space, and their cosine agreement defines an input-dependent multiplier at their channel–time intersection. The novelty therefore lies not in residual scaling alone, but in deriving token-wise refinement strength from the compatibility of complementary EEG axis contexts. Under this inductive bias, stronger agreement indicates more compatible spatial and temporal organization and assigns a larger multiplier to the corresponding residual update. The multiplier is applied to the recombined attention response before residual addition, as illustrated in Figure 1(a). The mechanism is learned jointly with the backbone through masked raw EEG patch reconstruction and requires neither electrode coordinates nor predefined brain regions, preserving a compact, single-stage pretraining framework.

Our key contributions are summarized as follows:

- We introduce a relational construction of token-wise residual modulation for EEG. At each layer, agreement between complementary channel and temporal context descriptors determines an input-dependent multiplier for their intersecting channel–time token, without prespecifying task-responsive electrodes or brain regions.
- AxisBrain uses a 7.46M-parameter encoder and a single-stage, end-to-end masked raw EEG patch reconstruction objective, without separately pretrained tokenizers, teachers, auxiliary models, or additional pretraining objectives.
- Across ten datasets spanning seven task families, AxisBrain achieves higher mean balanced accuracy than the compared similarly compact EEG foundation models and remains competitive with a substantially larger recent model. The same five fine-tuning seeds are used across all datasets, with their values and seed-wise results fully disclosed.
- Controlled variants isolate response recombination, score construction, and token-wise assignment. Alternative mappings, non-adaptive controls, and disrupted assignments reduce performance, indicating that the gain cannot be explained by generic residual amplification alone and depends on both the learned scores and their input-dependent placement.

2 METHOD

2.1 OVERVIEW

Let an EEG segment be divided into one-second patches, $X \in \mathbb{R}^{B\times C\times T\times P}$, where B is the batch size, C is the number of channels, T is the number of temporal patches, and P is the number of samples per patch after resampling. For each patch, a temporal convolutional encoder and a linear projection of the real-FFT magnitude produce two 200-dimensional feature vectors. These vectors are added elementwise, and asymmetric conditional positional encoding (ACPE) is applied to the resulting channel–time grid. During pretraining, a reconstruction head predicts the masked raw EEG patches from the encoder outputs. We retain this CBraMod backbone and introduce two components within each Transformer layer: a within-token projection that recombines the spatial and temporal attention responses, and a cross-axis consistency mechanism that adaptively modulates the resulting residual update. The following sections describe these two components.

2.2 RECOMBINING SPATIAL AND TEMPORAL ATTENTION RESPONSES

We first briefly review the criss-cross attention mechanism introduced in CBraMod (Wang et al., 2025a), which provides the factorized attention response on which our recombination operates. The criss-cross attention mechanism itself is not a contribution of this work. Let $H_\ell \in \mathbb{R}^{B\times C\times T\times d}$ denote the input to the ℓ-th encoder layer. Attention is applied to the layer-normalized input, with its feature dimension divided into two halves. Spatial multi-head attention is applied across the C channels independently at each time index, whereas temporal multi-head attention is applied across the T patches independently for each channel:

$$X^S_{:,:,t,:} = \mathrm{MHA}_S\left((\mathrm{LN}(H_\ell))_{:,:,t,1:d/2}\right), \qquad X^T_{:,c,:,:} = \mathrm{MHA}_T\left((\mathrm{LN}(H_\ell))_{:,c,:,d/2+1:d}\right) \quad (1)$$

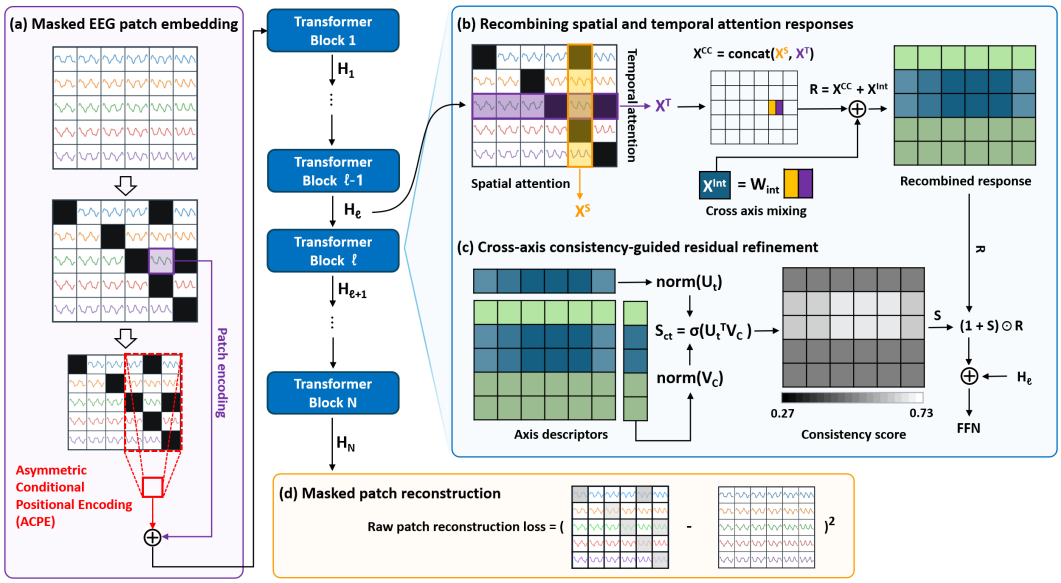


Figure 2: Overview of AxisBrain. (a) Masked EEG patch embedding with ACPE. (b) Criss-cross attention produces X^{cc}, which is recombined as $R = X^{\mathrm{cc}} + X^{\mathrm{int}}$. (c) Cross-axis descriptors yield the consistency score S, which modulates the response as $(1 + S) \odot R$ before the residual update and FFN. (d) Training uses masked raw-patch reconstruction.

$$X^{\mathrm{cc}} = \mathrm{Concat}\left[X^S, X^T\right] \in \mathbb{R}^{B\times C\times T\times d} \tag{2}$$

This factorization avoids constructing a global $CT \times CT$ attention matrix while retaining interactions along both axes. The conventional attention and FFN residual updates are

$$H_\ell^{\mathrm{attn}} = H_\ell + X^{\mathrm{cc}}, \qquad H_{\ell+1} = H_\ell^{\mathrm{attn}} + \mathrm{FFN}\left(\mathrm{LN}\left(H_\ell^{\mathrm{attn}}\right)\right) \tag{3}$$

Because X^{cc} is formed by feature-wise concatenation, its first half contains the spatial-attention response X^S, whereas its second half contains the temporal-attention response X^T. We introduce a full-dimensional affine projection to allow information from these two halves to interact before the attention response is added to H_ℓ:

$$X^{\mathrm{int}} = W_x X^{\mathrm{cc}} + b_x, \tag{4}$$

where $W_x \in \mathbb{R}^{d\times d}$ and $b_x \in \mathbb{R}^d$. We then retain the original criss-cross response as a direct term and add the projected response as a learned correction:

$$R = X^{\mathrm{cc}} + X^{\mathrm{int}} = (I_d + W_x)X^{\mathrm{cc}} + b_x. \tag{5}$$

This construction does not increase the set of affine transformations that can be represented. Because W_x is unconstrained, $I_d + W_x$ can represent any $d \times d$ matrix, while b_x can represent any bias. The additive formulation can therefore represent exactly the same affine mappings as an unconstrained projection $W'_x X^{\mathrm{cc}} + b'_x$. Accordingly, its potential advantage concerns the identity-offset parameterization and resulting optimization behavior rather than greater affine expressivity. We evaluate this design empirically in Table 3.

2.3 Cross-axis consistency-guided residual refinement

We now determine how strongly each recombined response contributes to the residual update. As motivated in the Introduction, informative EEG patterns can be characterized jointly by their distribution across channels and their evolution within individual channels. For a token at (c, t), we capture these complementary contexts using two descriptors computed from X^{cc}: one summarizes the channel tokens at time t, and the other summarizes the temporal tokens of channel c. Their agreement in a learned comparison space indicates how compatible these two contextual summaries are. We use

this agreement as a proxy for refinement strength, assigning a larger residual multiplier when the two contexts encode more compatible latent patterns. The descriptors and scores are computed at every encoder layer from its current attention responses, allowing the modulation to adapt to both the input and the evolving token representations.

Time-indexed cross-channel descriptor. At each time index t, the same learned query token $q^T \in \mathbb{R}^{1\times d}$ attends to the C channel tokens. The query is shared across samples and time indices within the layer:

$$A_t^T = \mathrm{LN}\big(\mathrm{MHA}_{\mathrm{pool}}\left(q^T, X^{\mathrm{cc}}_{:,:,t,:}, X^{\mathrm{cc}}_{:,:,t,:}\right)\big) \in \mathbb{R}^{B\times d} \tag{6}$$

Learned-query pooling can reduce the dilution of discriminative activity when it is concentrated in only a small subset of the available channels. Because the attention weights depend on the input tokens, different channels can contribute at different time indices and for different participants.

Channel-indexed temporal descriptor. For each channel c, we construct the complementary descriptor using a temporal mean:

$$A_c^C = \frac{1}{T}\sum_{t=1}^{T} X^{\mathrm{cc}}_{:,c,t,:} \tag{7}$$

Temporal mean pooling provides a simple, parameter-free summary of each channel over the analyzed interval. In our evaluated configurations, it outperformed learned-query temporal pooling, as shown in Table 5. This choice is well aligned with the datasets used for downstream evaluation in this study, where samples are generally constructed from intervals intended to capture a task response or a labeled physiological state over a substantial portion of the interval. For continuous EEG containing brief or sparsely occurring events with unknown boundaries, more temporally localized aggregation may be beneficial. Mean pooling may nevertheless remain effective when event boundaries are known or when a target interval can first be identified and pooling is restricted to that interval.

Cross-axis consistency score. We project the two descriptors into a common r-dimensional space:

$$U_t = W_T A_t^T + b_T, \qquad V_c = W_C A_c^C + b_C. \tag{8}$$

Their agreement in the learned latent space is measured using cosine similarity and transformed into a bounded consistency score:

$$G_{ct} = \frac{U_t^\top V_c}{\|U_t\|_2 \|V_c\|_2}, \qquad S_{ct} = \sigma(G_{ct}). \tag{9}$$

A high S_{ct} means that the cross-channel context at t and the temporal context of channel c have compatible projected feature directions. Under our inductive bias, this agreement is used as a proxy for refinement strength. The score requires neither electrode coordinates nor predefined brain-region assignments, and the same position can receive different scores across inputs and layers.

Since $G_{ct} \in [-1, 1]$, the consistency score satisfies

$$S_{ct} \in [\sigma(-1), \sigma(1)] \approx [0.2689, 0.7311]. \tag{10}$$

Score-guided residual modulation. Let S denote the score tensor formed by collecting S_{ct} over all channel and time indices. Using the recombined response $R = X^{\mathrm{cc}} + X^{\mathrm{int}}$ from equation 5, we define the score-modulated response as

$$\widetilde{X} = R + S \odot R = (1 + S) \odot R \tag{11}$$

where S is broadcast along the feature dimension. Because $1 + S \approx [1.2689,\ 1.7311]$, the score acts as a bounded additional refinement rather than a suppressive gate. The complete response R remains available at every channel–time position, while positions with greater cross-axis consistency receive stronger amplification. Using $\widetilde{X}$ as the attention-path response, the attention and FFN residual updates of the ℓ-th encoder layer are expressed as

$$\widetilde{H}_\ell^{\mathrm{attn}} = H_\ell + \widetilde{X}, \qquad H_{\ell+1} = \widetilde{H}_\ell^{\mathrm{attn}} + \mathrm{FFN}\left(\mathrm{LN}\left(\widetilde{H}_\ell^{\mathrm{attn}}\right)\right) \tag{12}$$

3 EXPERIMENTS

3.1 EXPERIMENTAL SETUP

Pretraining configuration. We pretrained on the Temple University EEG Corpus (TUEG) (Obeid & Picone, 2016), following the preprocessing protocol of CBraMod (Wang et al., 2025a). Preprocessing details, architecture specifications, and optimization hyperparameters are provided in Appendix B.

Downstream datasets. Experiments covered ten public EEG datasets and seven task families (Table 1). All signals were resampled to 200 Hz and divided into one-second patches. The downstream classifier architecture and fine-tuning hyperparameters are provided in Appendix B.4.

Table 1: Overview of downstream datasets.

Dataset	Task	# Channels	Duration (s)
MentalArithmetic (Zyma et al., 2019)	Rest vs. mental arithmetic classification	20	5
Mumtaz2016 (Mumtaz, 2016)	Mental Disorder Diagnosis	19	5
SHU-MI (Ma et al., 2022)	Motor Imagery Cls. (left/right hand)	32	4
PhysioNet-MI (Schalk et al., 2004)	Motor Imagery Cls. (left/right fist, both fists/feet)	64	4
ISRUC (Khalighi et al., 2016)	Sleep Staging (W, N1, N2, N3, REM)	6	30
HMC (Alvarez-Estevez & Rijsman, 2021)	Sleep Staging (W, N1, N2, N3, REM)	4	30
SIENA (Detti et al., 2020)	Seizure Detection	29	10
CHB-MIT (Shoeb, 2009)	Seizure Detection	16	10
FACED (Chen et al., 2023)	Emotion Recognition	32	10
TUAB (López et al., 2015)	Abnormal Detection	19	10

Evaluation metrics. Following the established downstream evaluation framework, binary tasks were evaluated using balanced accuracy, AUC-PR, and AUROC. Multi-class tasks used balanced accuracy, Cohen's kappa, and weighted F1. Preprocessing details, classifier specifications, and fine-tuning hyperparameters are provided in Appendix B. We report the mean $\pm$ sample standard deviation across five seeds. To facilitate reproducibility, the seed values and the corresponding results for each seed are provided in Appendix C.4.

Foundation-model baselines. We compare AxisBrain with LaBraM (Jiang et al., 2024), CBraMod (Wang et al., 2025a), CSBrain (Zhou et al., 2025), PATCHCODE (Yu et al., 2026), and CodeBrain (Ma et al., 2026) using published results for datasets in our evaluation suite. Values unavailable in the original model papers are taken from subsequent EEG foundation-model studies. The complete results, including comparisons with supervised models, are provided in Appendix C.1.

3.2 MAIN RESULTS

Table 2 presents the main results across the ten downstream datasets in terms of balanced accuracy. The corresponding AUC-PR and AUROC results for binary tasks and Cohen's kappa and weighted F1 results for multi-class tasks are provided in Tables 11 and 12. Averaged across these datasets, AxisBrain achieves a mean balanced accuracy of 75.67%, compared with 71.14% for LaBraM, 73.25% for CBraMod, and 74.04% for CSBrain. On the six datasets for which CodeBrain reports results, AxisBrain achieves a mean balanced accuracy of 73.95%, compared with 72.15% for CodeBrain. With 7.46M trainable parameters in its encoder backbone, AxisBrain has approximately one-fifth as many parameters as PATCHCODE-Large (37M), while achieving comparable ten-dataset mean balanced accuracy (75.67% versus 75.42%). Pretraining remains a single-stage masked raw-patch reconstruction procedure, without a separate tokenizer, momentum teacher, multiple pretrained encoders, or auxiliary objective. Moreover, AxisBrain requires neither electrode coordinates nor predefined brain regions.

Compared with its CBraMod base architecture, AxisBrain achieves higher balanced accuracy on eight of the ten datasets, with differences ranging from 0.01 to 7.58 percentage points. The only lower results occur on the two motor-imagery datasets, SHU-MI and PhysioNet-MI. One possible explanation for these two small gaps is the spatial localization of motor-imagery-related EEG activity. Because the time-indexed cross-channel descriptor aggregates representations across all channels at each time slot, discriminative evidence confined to a small subset of channels may contribute less to the pooled descriptor and therefore have less influence on the cross-axis agreement score.

Appendix D further examines this possibility by comparing the original dense pooling with sparse channel pooling on SHU-MI and PhysioNet-MI.

Table 2: Balanced accuracy on ten downstream datasets. Bold and underline indicate the best and second-best displayed means for each dataset, respectively. A dash denotes a result unavailable from the corresponding source paper. Parameter counts are given in parentheses.

Dataset	LaBraM (5.8M)	CBraMod (4.03M)	CSBrain (4.9M)	PATCHCODE (37M)	CodeBrain (15.17M)	**AxisBrain (7.46M)**
MentalArithmetic	0.6909±0.0125	0.7256±0.0132	0.7558±0.0106	0.7694±0.0091	0.7514±0.0203	**0.8014±0.0067**
Mumtaz2016	0.9409±0.0079	0.9560±0.0056	0.9643±0.0155	**0.9721±0.0083**	–	0.9616±0.0095
SHU-MI	0.6166±0.0192	0.6370±0.0151	0.6417±0.0037	**0.6593±0.0068**	0.6431±0.0066	0.6287±0.0101
PhysioNet-MI	0.6173±0.0122	**0.6417±0.0091**	0.6304±0.0090	0.6230±0.0068	–	0.6387±0.0046
ISRUC	0.7633±0.0102	0.7865±0.0110	0.7925±0.0030	**0.8120±0.0068**	0.7835±0.0033	0.8035±0.0032
HMC	0.7277±0.0101	0.7269±0.0041	0.7345±0.0047	**0.7556±0.0041**	–	0.7274±0.0051
SIENA	0.7082±0.0329	0.7317±0.0647	0.7662±0.0471	0.7685±0.0388	–	**0.8023±0.0065**
CHB-MIT	0.7075±0.0358	0.7398±0.0284	0.7262±0.0115	0.7620±0.0128	0.7273±0.0240	**0.7783±0.0246**
FACED	0.5273±0.0107	0.5509±0.0089	0.5752±0.0042	0.5950±0.0072	0.5941±0.0098	**0.5961±0.0039**
TUAB	0.8140±0.0019	0.8289±0.0022	0.8172±0.0043	0.8250±0.0035	**0.8294±0.0013**	0.8290±0.0019

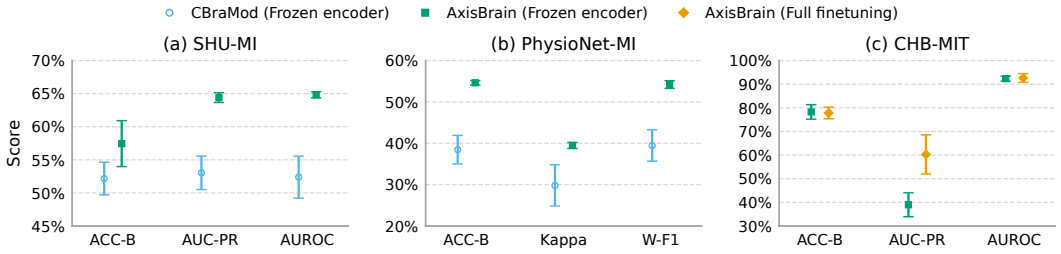


Figure 3: Frozen-encoder transfer performance. The CBraMod (Frozen encoder) results are taken from Table 18 of Wang et al. (2025a).

Frozen-encoder evaluation provides complementary evidence about representation learning by testing whether the representations learned during pretraining support downstream classification without updating the encoder. Figures 3(a,b) compare AxisBrain with the published CBraMod frozen-encoder results on SHU-MI and PhysioNet-MI, the two datasets on which AxisBrain is slightly lower after full fine-tuning. With both encoders frozen, AxisBrain achieves higher means on all three metrics for both datasets and lower standard deviations in five of the six dataset–metric comparisons. These results provide reference evidence that AxisBrain's pretrained representations support effective head-only transfer on the two motor-imagery datasets.

Figure 3(c) further shows that the representation learned by AxisBrain supports CHB-MIT classification without encoder adaptation. Frozen-encoder ACC-B slightly exceeds the full-fine-tuning result (78.27% versus 77.83%), while AUROC remains similar (0.9233 versus 0.9263). AUC-PR is nevertheless lower with the encoder frozen (0.3899 versus 0.6029), so the results do not indicate a uniform advantage for frozen-encoder evaluation. Complete results for all ten datasets and the detailed comparison with published CBraMod frozen-encoder results are provided in Appendix C.2.

3.3 Ablation Study

For each dataset, the mean and sample standard deviation are computed over five fine-tuning seeds. Aggregate entries in Tables 3–5 report the averages of these dataset-level means and standard deviations across the ten datasets.

Effect of spatial–temporal response recombination. We isolate the contribution of the response recombination introduced in Section 2.2. Because the proposed operation was implemented by extending the released CBraMod (Wang et al., 2025a) code, this ablation uses our direct CBraMod reproduction as the controlled baseline. Our reproduced mean ACC-B is lower than the value reported in the original CBraMod paper. This difference may reflect the finite set of fine-tuning hyperparameter configurations evaluated in our study, differences in hardware and software environments, and the use of a model pretrained by us rather than the released pretrained checkpoint. We therefore use the reproduced result only in this controlled ablation to quantify the incremental performance gain

obtained by adding the spatial–temporal response recombination to the reproduced CBraMod baseline, rather than to evaluate the complete proposed model. Dataset-level reproduction results and the fine-tuning hyperparameters used are provided in Appendix C.3.

Table 3: Effect of recombining spatial and temporal attention responses. † denotes our direct reproduction of CBraMod. For each dataset, the mean and sample standard deviation of balanced accuracy are computed across five fine-tuning seeds.

Variant	Attention-path response $\widetilde{X}$	Mean ACC-B
CBraMod† (Wang et al., 2025a)	X^{cc}	0.7134±0.0229
Cross-axis projection replacement	X^{int}	0.6936±0.0146
Additive cross-axis recombination	$X^{\mathrm{cc}} + X^{\mathrm{int}}$	**0.7312±0.0095**

As shown in Table 3, the comparison changes only the attention-path response $\widetilde{X}$ added to the layer input H_ℓ. The reproduced CBraMod baseline directly uses X^{cc} and achieves a mean balanced accuracy of 71.34%. Replacing this response with the projection-only formulation $X^{\mathrm{int}} = W_x X^{\mathrm{cc}} + b_x$ reduces performance to 69.36%. In contrast, the additive formulation

$$R = X^{\mathrm{cc}} + X^{\mathrm{int}} = (I_d + W_x)X^{\mathrm{cc}} + b_x \tag{13}$$

achieves 73.12%, exceeding the projection-only variant by 3.76 percentage points and the reproduced CBraMod baseline by 1.78 percentage points. This controlled comparison empirically supports the identity-offset parameterization, in which the projected response serves as a learned correction to the original criss-cross response.

Effectiveness of learned token-wise scores. The ablation in Table 4 addresses four questions about the design and necessity of the proposed score. First, the proposed bounded multiplier preserves the recombined response and adds only positive refinement. Variant [1] tests whether permitting both attenuation and amplification is preferable. Second, cosine normalization discards the descriptor norms, which may contain information in the learned embedding space; variants [2] and [3] therefore use unnormalized and dimension-scaled dot products. Third, variants [4]–[7] examine whether the bounded score range restricts expressivity by removing the sigmoid or changing its input scale. Finally, variants [8]–[14] test whether the improvement can be explained by generic residual amplification rather than input-dependent scores assigned to individual channel–time responses. All variants are pretrained from scratch using their respective score rules and fine-tuned under the same evaluation protocol.

The proposed score achieves 75.67%, outperforming every alternative mapping. Allowing suppression does not improve performance: the centered-sigmoid and pre-sigmoid variants obtain 71.85% and 69.36%, respectively. Both variants permit multipliers below one and can therefore attenuate the recombined response. In contrast, the proposed formulation preserves R at full strength and uses $S_{ct} \odot R$ as a bounded positive correction. These results do not imply that suppression is universally detrimental, but show that it is less effective than bounded positive refinement in the evaluated architecture and tasks.

Retaining descriptor norms also provides no benefit. The unnormalized and scaled dot-product scores reach 71.03% and 69.56%, supporting cosine normalization as a useful means of comparing descriptor directions while controlling score magnitude. Similarly, expanding, contracting, or learning the cosine-logit scale yields 70.90%, 72.27%, and 71.31%, respectively. The default scale therefore performs best among the tested ranges, indicating that its fixed interval does not underutilize the consistency signal in this setting.

The non-adaptive controls further distinguish token-wise scoring from generic amplification. The strongest fixed or random control is the fixed midpoint at 73.71%, leaving a 1.96-percentage-point gap to the proposed score; random bounded and layer-wise mean scores achieve 69.32% and 72.07%, respectively. Random permutation reassigns all CT scores to randomly selected token positions within each sample and layer. Partial high–low switching first divides the token positions into upper and lower halves according to their scores. From each half, it randomly selects an equal-sized subset containing 25% of all token positions and exchanges the scores between the two subsets; thus, only half of the token positions are modified, primarily by assigning relatively low scores to selected high-score positions and vice versa. Both interventions preserve the score distribution within each sample and layer while disrupting its correspondence with the responses being modulated. They

reduce mean balanced accuracy by 5.94 and 4.52 percentage points, respectively. Taken together, these results show that the gain depends on the combination of cosine-normalized bounded scoring, preservation of the original response, and correct input-dependent assignment of refinement strength to individual channel–time tokens.

Table 4: Ablation of token-wise score values, mappings, and assignments. Here, $R = X^{\mathrm{cc}} + X^{\mathrm{int}}$, and G_{ct} and S_{ct} are defined in equation 9.

Variant	Attention-path response $\widetilde{X}$	Mean ACC-B
Proposed learned score	$(1 + S_{ct}) \odot R$	**0.7567±0.0076**
[1] Centered sigmoid score	$R + (S_{ct} - 0.5) \odot R = (0.5 + S_{ct}) \odot R$	0.7185±0.0167
[2] Unnormalized dot-product score	$\left[1 + \sigma\left(U_t^\top V_c\right)\right] \odot R$	0.7103±0.0129
[3] Scaled dot-product score	$\left[1 + \sigma\left(U_t^\top V_c/\sqrt{r}\right)\right] \odot R$	0.6956±0.0126
[4] Pre-sigmoid score	$(1 + G_{ct}) \odot R$	0.6936±0.0129
[5] Expanded score range ($\alpha = 2$)	$[1 + \sigma(2G_{ct})] \odot R$	0.7090±0.0121
[6] Contracted score range ($\alpha = 0.5$)	$[1 + \sigma(0.5G_{ct})] \odot R$	0.7227±0.0161
[7] Learnable score range	$[1 + \sigma(\alpha G_{ct})] \odot R, \quad \alpha > 0$	0.7131±0.0221
[8] Random bounded score	$(1 + S_{\mathrm{rand}}) \odot R, \quad S_{\mathrm{rand}} \sim \mathcal{U}(0.27, 0.73)$	0.6932±0.0273
[9] Fixed score lower bound	$(1 + 0.27) \cdot R$	0.7181±0.0092
[10] Fixed score midpoint	$(1 + 0.50) \cdot R$	0.7371±0.0114
[11] Fixed score upper bound	$(1 + 0.73) \cdot R$	0.7264±0.0179
[12] Layer-wise mean score	$(1 + \bar{S}_\ell) \cdot R$	0.7207±0.0195
[13] Random score permutation	$(1 + S_{\pi(c,t)}) \odot R$	0.6973±0.0179
[14] Partial high–low score switching	$(1 + S_{ct}^{\mathrm{swap}}) \odot R$	0.7115±0.0141

Effect of axis-descriptor construction. We next compare how the time-indexed cross-channel descriptor A_t^T and the channel-indexed temporal descriptor A_c^C are constructed. This experiment tests both pooling choices and whether comparing two complementary axis descriptors is preferable to comparing one descriptor directly with each channel–time token.

Table 5: Effect of constructing the time-indexed cross-channel descriptor A_t^T and channel-indexed temporal descriptor A_c^C. Query denotes learned-query cross-attention in equation 6, and mean denotes feature-vector averaging as in equation 7. Token means that the corresponding axis descriptor is replaced by X_{ct}^{cc}, so the remaining descriptor is compared directly with each token.

A_t^T	A_c^C	Mean ACC-B	A_t^T	A_c^C	Mean ACC-B
Query	Mean	**0.7567±0.0076**	Mean	Mean	0.7269±0.0124
Query	Query	0.7106±0.0144	Query	Token	0.7105±0.0154
Mean	Query	0.7247±0.0197	Token	Mean	0.7257±0.0088

Learned-query pooling across channels combined with temporal mean pooling yields the highest mean balanced accuracy among the tested descriptor configurations. Replacing either context descriptor with the local token reduces performance. These comparisons support the chosen pair of complementary context summaries within the current architecture.

4 Conclusion

We introduced AxisBrain, an EEG foundation model that uses cross-axis context agreement to modulate attention-derived residual updates. Its 7.46M-parameter encoder is pretrained in a single stage with masked raw EEG patch reconstruction as its sole objective, without electrode coordinates or predefined brain-region assignments. Across ten downstream datasets, AxisBrain achieves competitive downstream performance, while controlled ablations support the proposed response recombination, consistency-score construction, and token-wise score assignment. Performance remains below the published CBraMod results on the two motor-imagery datasets, motivating further investigation of descriptors tailored to spatially localized EEG activity. Overall, the results show that cross-axis consistency-guided residual modulation can support competitive EEG representation learning without relying on large-scale architectures, multi-stage pretraining, multiple pretraining objectives, electrode coordinates, or predefined brain-region assignments.

ETHICS STATEMENT

This research adheres to the ICLR Code of Ethics. All experiments were conducted using publicly available, de-identified EEG datasets in accordance with their respective terms of use. No new human participants were recruited, and no personally identifiable information was used in this study.

REPRODUCIBILITY STATEMENT

We provide the following materials and experimental details to facilitate reproduction of our results:

- **Code and models:** The implementation code and model configurations are included in the supplementary materials submitted with this paper. To preserve anonymity during double-blind review, we withhold the public repository URL. Upon acceptance, we will release the source code and pretrained model checkpoints under an open-source license.
- **Datasets and preprocessing:** All datasets used for pretraining and downstream evaluation are publicly available. Table 1 summarizes the downstream datasets. Preprocessing procedures for both pretraining and downstream data are described in Appendix B.2.
- **Hyperparameters and random seeds:** Pretraining and downstream training settings are reported in Appendices B.3 and B.4, respectively. Because downstream performance can vary with the random seed, we explicitly report both the seed values and their use across datasets to support comparisons with reproduced results. We use the same five seeds $\{1000, 2000, 3000, 4000, 5000\}$ for fine-tuning on every downstream dataset. The results for each seed, together with their means and sample standard deviations, are provided in Appendix C.4.
- **Computing environment:** The software versions and hardware configuration used for the experiments are reported in Appendix B.1.

AI USE STATEMENT

We used generative AI tools to assist with the software implementation of author-designed methods, translation, manuscript drafting and editing, literature discovery and reference formatting, and feedback on the presentation and interpretation of experiments. The authors made all final methodological decisions, independently verified cited sources and numerical results, and reviewed all AI-assisted content for technical accuracy.

REFERENCES

Diego Alvarez-Estevez and Roselyne M. Rijsman. Inter-database validation of a deep learning approach for automatic sleep scoring. *PLOS ONE*, 16(8):e0256111, 2021.

Thomas Bachlechner, Bodhisattwa Prasad Majumder, Henry Mao, Gary Cottrell, and Julian McAuley. ReZero is all you need: Fast convergence at large depth. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, pp. 1352–1361. PMLR, 2021. URL https://proceedings.mlr.press/v161/bachlechner21a.html.

J. Chen, X. Wang, C. Huang, X. Hu, X. Shen, and D. Zhang. A large finer-grained affective computing EEG dataset. *Scientific Data*, 10(1):740, 2023. doi: 10.1038/s41597-023-02650-w.

Yuqi Chen, Kan Ren, Kaitao Song, Yansen Wang, Yifan Wang, Dongsheng Li, and Lili Qiu. EEGFormer: Towards transferable and interpretable large-scale EEG foundation model, 2024. URL https://arxiv.org/abs/2401.10278.

Zhisheng Chen, Yingwei Zhang, Qizhen Lan, Tianyu Liu, Huacan Wang, Yi Ding, Ziyu Jia, Ronghao Chen, Kun Wang, and Xinliang Zhou. Uni-NTFM: A unified neural time-frequency foundation model. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.24222.

Paolo Detti, Giampaolo Vatti, and Garazi Zabalo Manrique de Lara. EEG synchronization analysis for seizure prediction: A study on data of noninvasive recordings. *Processes*, 8(7):846, 2020.

Berkay Döner, Thorir Mar Ingolfsson, Luca Benini, and Yawei Li. LUNA: Efficient and topology-agnostic foundation model for EEG signal analysis. In *Advances in Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=uazfjnFL0G.

Yassine El Ouahidi, Jonathan Lys, Philipp Thölke, Nicolas Farrugia, Bastien Pasdeloup, Vincent Gripon, Karim Jerbi, and Giulia Lioi. REVE: A foundation model for eeg. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2510.21585.

Zitao Fang, Chenxuan Li, Hongting Zhou, Shuyang Yu, Guodong Du, Ashwaq Qasem, Yang Lu, Jing Li, Junsong Zhang, and Sim Kuan Goh. NeurIPT: A generalist neural signal foundation model. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2510.16548.

Qibin Hou, Daquan Zhou, and Jiashi Feng. Coordinate attention for efficient mobile network design. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13713–13722, 2021. doi: 10.1109/CVPR46437.2021.01350. URL https://openaccess.thecvf.com/content/CVPR2021/html/Hou_Coordinate_Attention_for_Efficient_Mobile_Network_Design_CVPR_2021_paper.html.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018. doi: 10.1109/CVPR.2018.00745. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Hu_Squeeze-and-Excitation_Networks_CVPR_2018_paper.html.

Ziyu Jia, Junyi Lin, Pu Wan, Jinxin Pi, Jingying Ma, Peiliang Gong, Xinliang Zhou, Yi Ding, and Chenyu Liu. EmBrace: A collective knowledge fusion framework toward unified EEG foundation models. In *International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=BJ5rYj8O8W.

Wei-Bang Jiang, Li-Ming Zhao, and Bao-Liang Lu. Large brain model for learning generic representations with tremendous EEG data in BCI. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=QzTpTRVtrP.

Wei-Bang Jiang, Yansen Wang, Bao-Liang Lu, and Dongsheng Li. NeuroLM: A universal multi-task foundation model for bridging the gap between language and EEG signals. In *International Conference on Learning Representations*, 2025.

Jin Jing, Wendong Ge, Shenda Hong, Marta Bento Fernandes, Zhen Lin, Chaoqi Yang, Sungtae An, Aaron F. Struck, Aline Herlopian, Ioannis Karakis, et al. Development of expert-level classification of seizures and rhythmic and periodic patterns during EEG interpretation. *Neurology*, 100(17): e1750–e1762, 2023. doi: 10.1212/WNL.0000000000207127. URL https://doi.org/10.1212/WNL.0000000000207127.

Sirvan Khalighi, Teresa Sousa, José Moutinho Santos, and Urbano Nunes. ISRUC-Sleep: A comprehensive public dataset for sleep researchers. *Computer Methods and Programs in Biomedicine*, 124:180–192, 2016.

Demetres Kostas, Stephane Aroca-Ouellette, and Frank Rudzicz. BENDR: Using transformers and a contrastive self-supervised learning task to learn from massive amounts of EEG data. *Frontiers in Human Neuroscience*, 15:653659, 2021. doi: 10.3389/fnhum.2021.653659. URL https://doi.org/10.3389/fnhum.2021.653659.

Vernon J. Lawhern, Amelia J. Solon, Nicholas R. Waytowich, Stephen M. Gordon, Chou P. Hung, and Brent J. Lance. EEGNet: A compact convolutional neural network for EEG-based brain–computer interfaces. *Journal of Neural Engineering*, 15(5):056013, 2018. doi: 10.1088/1741-2552/aace8c. URL https://doi.org/10.1088/1741-2552/aace8c.

Hongli Li, Man Ding, Ronghua Zhang, and Chunbo Xiu. Motor imagery EEG classification algorithm based on CNN–LSTM feature fusion network. *Biomedical Signal Processing and Control*, 72: 103342, 2022. URL https://www.sciencedirect.com/science/article/pii/S1746809421009392.

Jinhao Li, Zhiyuan Ma, Xueqiao Han, Zhongye Xia, Xinche Zhang, Shanghong Xie, Yixuan Liu, Yongjian Li, Runmin Gan, Tianlin Huo, and Sen Song. EEG-JEPA: Structured latent prediction for EEG foundation models. *arXiv preprint arXiv:2608.00114*, 2026. URL https://arxiv.org/abs/2608.00114.

Dingkun Liu, Yuheng Chen, Zhu Chen, Zhenyao Cui, Yaozhi Wen, Jiayu An, Jingwei Luo, and Dongrui Wu. EEG-FM-Compass: Progress, benchmarking, and future directions for EEG foundation models. *National Science Review*, pp. nwag466, 2026. doi: 10.1093/nsr/nwag466. URL https://doi.org/10.1093/nsr/nwag466.

S. López, G. Suarez, D. Jungreis, I. Obeid, and J. Picone. Automated identification of abnormal adult EEGs. In *2015 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pp. 1–5. IEEE, 2015. URL https://isip.piconepress.com/conferences/ieee_spmb/2015/papers/l01_06.pdf.

Jingying Ma, Feng Wu, Qika Lin, Yucheng Xing, Chenyu Liu, Ziyu Jia, and Mengling Feng. CodeBrain: Bridging decoupled tokenizer and multi-scale architecture for EEG foundation models. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=msJgEkjwh5.

Jun Ma, Banghua Yang, Wenzheng Qiu, Yunzhe Li, Shouwei Gao, and Xinxing Xia. A large EEG dataset for studying cross-session variability in motor imagery brain-computer interface. *Scientific Data*, 9(1):531, 2022.

Navid Mohammadi Foumani, Geoffrey Mackellar, Soheila Ghane, Saad Irtza, Nam Nguyen, and Mahsa Salehi. EEG2Rep: Enhancing self-supervised EEG representation through informative masked inputs. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 5544–5555, 2024. doi: 10.1145/3637528.3671600. URL https://doi.org/10.1145/3637528.3671600.

Wajid Mumtaz. MDD patients and healthy controls EEG data (new). Figshare, 2016. URL https://figshare.com/articles/dataset/EEG_Data_New/4244171.

Iyad Obeid and Joseph Picone. The Temple University Hospital EEG Data Corpus. In *Frontiers in Neuroscience*, volume 10, pp. 196, 2016. doi: 10.3389/fnins.2016.00196. URL https://doi.org/10.3389/fnins.2016.00196.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023. doi: 10.1109/ICCV51070.2023.00387. URL https://openaccess.thecvf.com/content/ICCV2023/html/Peebles_Scalable_Diffusion_Models_with_Transformers_ICCV_2023_paper.html.

Wei Yan Peh, Yuanyuan Yao, and Justin Dauwels. Transformer convolutional neural networks for automated artifact detection in scalp EEG. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society*, pp. 3599–3602, 2022. doi: 10.1109/EMBC48229.2022.9871916. URL https://doi.org/10.1109/EMBC48229.2022.9871916.

Ben Peters, Vlad Niculae, and André F. T. Martins. Sparse sequence-to-sequence models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1504–1519. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1146. URL https://doi.org/10.18653/v1/P19-1146.

Gerwin Schalk, Dennis J. McFarland, Thilo Hinterberger, Niels Birbaumer, and Jonathan R. Wolpaw. BCI2000: A general-purpose brain-computer interface (BCI) system. *IEEE Transactions on Biomedical Engineering*, 51(6):1034–1043, 2004.

Robin Tibor Schirrmeister, Lukas Gemein, Katharina Eggensperger, Frank Hutter, and Tonio Ball. Deep learning with convolutional neural networks for decoding and visualization of EEG pathology. In *2017 IEEE Signal Processing in Medicine and Biology Symposium*, pp. 1–7, 2017. doi: 10.1109/SPMB.2017.8257015. URL https://doi.org/10.1109/SPMB.2017.8257015.

Ali Hossam Shoeb. *Application of Machine Learning to Epileptic Seizure Onset Detection and Treatment*. PhD thesis, Massachusetts Institute of Technology, 2009.

Yonghao Song, Xueyu Jia, Lie Yang, and Longhan Xie. Transformer-based spatial-temporal feature learning for EEG decoding, 2021. URL https://arxiv.org/abs/2106.11170.

Yonghao Song, Qingqing Zheng, Bingchuan Liu, and Xiaorong Gao. EEG conformer: Convolutional transformer for EEG decoding and visualization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:710–719, 2023. doi: 10.1109/TNSRE.2022.3230250. URL https://doi.org/10.1109/TNSRE.2022.3230250.

Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Training very deep networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015. URL https://arxiv.org/abs/1507.06228.

Rahul Thapa, Magnus Ruud Kjaer, Bryan He, Ian Covert, Hyatt Moore, IV, Umaer Hanif, Gauri Ganjoo, M. Brandon Westover, Poul Jennum, Andreas Brink-Kjaer, Emmanuel Mignot, and James Zou. A multimodal sleep foundation model for disease prediction. *Nature Medicine*, 32(2): 752–762, 2026. doi: 10.1038/s41591-025-04133-4. URL https://doi.org/10.1038/s41591-025-04133-4.

Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 32–42, 2021. doi: 10.1109/ICCV48922.2021.00010. URL https://openaccess.thecvf.com/content/ICCV2021/html/Touvron_Going_Deeper_With_Image_Transformers_ICCV_2021_paper.html.

Christopher Wang, Vighnesh Subramaniam, Adam Uri Yaari, Gabriel Kreiman, Boris Katz, Ignacio Cases, and Andrei Barbu. BrainBERT: Self-supervised representation learning for intracranial recordings. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=xmcYx_reUn6.

Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3156–3164, 2017. doi: 10.1109/CVPR.2017.683. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Wang_Residual_Attention_Network_CVPR_2017_paper.html.

Guangyu Wang, Wenchao Liu, Yuhong He, Cong Xu, Lin Ma, and Haifeng Li. EEGPT: Pretrained transformer for universal and reliable representation of EEG signals. In *Advances in Neural Information Processing Systems*, volume 37, pp. 39249–39280, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/4540d267eeec4e5dbd9dae9448f0b739-Abstract-Conference.html.

Jiquan Wang, Sha Zhao, Zhiling Luo, Yangxuan Zhou, Haiteng Jiang, Shijian Li, Tao Li, and Gang Pan. CBraMod: A criss-cross brain foundation model for EEG decoding. In *International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=NPNUHgHF2w.

Jiquan Wang, Sha Zhao, Zhiling Luo, Yangxuan Zhou, Shijian Li, and Gang Pan. EEGMamba: An EEG foundation model with mamba. *Neural Networks*, 192:107816, 2025b. doi: 10.1016/j.neunet.2025.107816. URL https://doi.org/10.1016/j.neunet.2025.107816.

Yifan Wang, Der-Horng Lee, and Bruce X. B. Yu. BraSTORM: A dual-branch self-supervised framework for EEG representation learning via input-level spatio-temporal decomposition. In *AAAI Conference on Artificial Intelligence*, 2026. URL https://ojs.aaai.org/index.php/AAAI/article/view/38838.

Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision*, pp. 3–19, 2018. doi: 10.1007/978-3-030-01234-2_1. URL https://openaccess.thecvf.com/content_ECCV_2018/html/Sanghyun_Woo_Convolutional_Block_Attention_ECCV_2018_paper.html.

Qinfan Xiao, Ziyun Cui, Chi Zhang, Siqi Chen, Wen Wu, Andrew Thwaites, Alexandra Woolgar, Bowen Zhou, and Chao Zhang. BrainOmni: A brain foundation model for unified EEG and MEG signals. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1375. URL https://openreview.net/forum?id=cjHQj0tCy6.

Chaoqi Yang, M. Brandon Westover, and Jimeng Sun. BIOT: Biosignal transformer for cross-data learning in the wild. In *Advances in Neural Information Processing Systems*, volume 36, pp. 78240–78260, 2023a. URL https://openreview.net/forum?id=c2LZyTyddi.

Chaoqi Yang, Cao Xiao, M. Brandon Westover, and Jimeng Sun. Self-supervised electroencephalogram representation learning for automatic sleep staging: Model development and evaluation study. *JMIR AI*, 2:e46769, 2023b. doi: 10.2196/46769. URL https://ai.jmir.org/2023/1/e46769.

Wenchao Yang, Weidong Yan, Wenkang Liu, Yulan Ma, and Yang Li. THD-BAR: Brain topology hierarchy-driven autoregressive modeling for eeg foundation models. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2511.13733.

Kieren Yu, Ziyang Liu, Chang Huang, and Kaishun Wu. PATCHCODE: Discrete latent predictive learning for EEG foundation model. In *International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=NWcZ5vualM.

Daoze Zhang, Zhizhang Yuan, Yang Yang, Junru Chen, Jingjing Wang, and Yafeng Li. Brant: Foundation model for intracranial neural signal. In *Advances in Neural Information Processing Systems*, volume 36, pp. 26304–26321, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/535915d26859036410b0533804cee788-Abstract-Conference.html.

Yuchen Zhou, Jiamin Wu, Zichen Ren, Zhouheng Yao, Weiheng Lu, Kunyu Peng, Qihao Zheng, Chunfeng Song, Wanli Ouyang, and Chao Gou. CSBrain: A cross-scale spatiotemporal brain foundation model for EEG decoding. In *Advances in Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=agcXjEHmyW.

Igor Zyma, Sergii Tukaev, Ivan Seleznov, Ken Kiyono, Anton Popov, Mariia Chernykh, and Oleksii Shpenkov. Electroencephalograms during mental arithmetic task performance. *Data*, 4(1):14, 2019.

APPENDIX CONTENTS

A Related Work **16**

A.1 EEG foundation models 16

A.2 Trends in model scale, pipeline complexity, and structural priors 17

B Pretraining and fine-tuning Details **18**

B.1 Computing Environment 19

B.2 Data Preprocessing 19

B.3 Encoder Architecture and Hyperparameters for pretraining 20

B.4 Downstream Classifier and Fine-Tuning Details 20

C Detailed Experimental Results **22**

C.1 Complete Comparison with Foundation and Non-Foundation Models 22

C.2 Frozen-Encoder Evaluation 24

C.3 Reproduction of CBraMod Results 25

C.4 Full Five-Seed Results 27

D Sparse Channel Pooling for Motor-Imagery Tasks **28**

E Mask Ratio Sensitivity **29**

F Scaling Analysis **30**

F.1 Model Size Scaling 30

F.2 Pretraining Data Scaling 30

G Computational cost and model scale of AxisBrain **32**

H Qualitative Analysis of Channel–Time Consistency **33**

I Limitations and Future Directions **34**

A Related Work

A.1 EEG foundation models

EEG foundation models are pretrained on large, predominantly unlabeled recording collections and then adapted to heterogeneous downstream tasks. Most current models divide a multichannel recording into channel-specific temporal patches, embed the patches as tokens, add channel and temporal position information, and process the resulting channel–time grid with a Transformer or another sequence model. This token-based formulation provides a common interface for recordings with varying durations and electrode configurations. These properties facilitate transfer across heterogeneous tasks such as motor imagery, affective decoding, sleep staging, and clinical event detection.

EEG foundation models can be organized along several design axes, including backbone architecture, model scale, and self-supervised pretraining strategy. Here, we organize recent methodological developments by their self-supervised pretraining objectives. Among these objectives, masked prediction and reconstruction constitute the dominant family (Liu et al., 2026). We therefore further distinguish methods within this family by their prediction targets.

Self-supervised learning objectives. At a broad level, many current EEG foundation models can be organized into three recurring families of self-supervised objectives: contrastive learning, autoregressive prediction, and masked reconstruction, with some models combining multiple objectives.

Contrastive learning brings representations of related inputs closer while separating those of unrelated inputs. Contrastive objectives are used by BENDR (Kostas et al., 2021), BIOT (Yang et al., 2023a), and SleepFM (Thapa et al., 2026).

Autoregressive pretraining is adopted by NeuroLM (Jiang et al., 2025) and THD-BAR (Yang et al., 2025).

Masked prediction or reconstruction is used by CBraMod (Wang et al., 2025a), CSBrain (Zhou et al., 2025), Uni-NTFM (Chen et al., 2026), LaBraM (Jiang et al., 2024), CodeBrain (Ma et al., 2026), and EEG-JEPA (Li et al., 2026).

Hybrid approaches include EEGPT, which combines masked modeling with momentum-based representation alignment (Wang et al., 2024), and BraSTORM, which combines reconstruction and cross-branch contrastive learning (Wang et al., 2026).

Targets of masked prediction and reconstruction. Within the masked reconstruction family, methods can be further distinguished by the target on which the objective is defined.

Raw-signal reconstruction is used by Brant, CBraMod, EEGMamba, CSBrain, and REVE (Zhang et al., 2023; Wang et al., 2025a;b; Zhou et al., 2025; El Ouahidi et al., 2025). These targets retain detailed waveform information but may also include predictable background activity and acquisition noise.

Frequency-domain supervision includes spectrogram reconstruction in BrainBERT (Wang et al., 2023), spectral-amplitude and codebook-embedding reconstruction in EEGFormer (Chen et al., 2024), and joint time- and frequency-domain supervision through a dual-domain loss in Uni-NTFM (Chen et al., 2026). These targets emphasize spectral and oscillatory structure.

Discrete targets represent EEG patches using codebook indices or learned neural tokens. Representative approaches include LaBraM (Jiang et al., 2024), BrainOmni (Xiao et al., 2025), and CodeBrain (Ma et al., 2026).

Representation-space targets predict embedded or contextual representations rather than signal samples. EEG2Rep predicts masked latent representations (Mohammadi Foumani et al., 2024), PATCHCODE combines continuous representation and discrete-code targets (Yu et al., 2026), and EEG-JEPA predicts contextual latent representations (Li et al., 2026). Thus, each target space emphasizes different properties of EEG signals during representation learning.

A.2 TRENDS IN MODEL SCALE, PIPELINE COMPLEXITY, AND STRUCTURAL PRIORS

To highlight the practical advantages of our method, we compare it with recent EEG foundation models from ICLR, NeurIPS, and ICML. Table 6 summarizes the comparison along three dimensions: model scale, pretraining-pipeline complexity, and reliance on structural priors, including brain-region annotations and electrode-position metadata.

Table 6: Comparison of recent EEG foundation models in terms of model scale, pretraining stages and objectives, and the use of brain-region or electrode-layout information. Parameter counts follow the definitions reported by the respective papers, with total or active counts indicated where applicable. † For EmBrace, 15.6M denotes the active parameters reported for the SEED-V configuration with CodeBrain as the carrier, excluding the 25.9M frozen source-model pool. A dash indicates that no external brain-region annotation or electrode-position metadata is required.

Model	#Params	Pretraining stages and objectives	Brain-region or electrode-layout information
LaBraM (Jiang et al., 2024)	5.8/46/369M	Two stages: VQ-tokenizer pretraining with spectral reconstruction and quantization objectives, followed by masked discrete-token prediction	–
CBraMod (Wang et al., 2025a)	4.03M	Single stage: masked raw-patch MSE only	–
CSBrain (Zhou et al., 2025)	4.9M	Single stage: masked raw-signal reconstruction with cross-scale tokenization and structured sparse attention	Predefined anatomical regions and within-region spatial neighborhoods
LUNA (Döner et al., 2025)	7/43/311M	Single stage with a masked reconstruction loss and an auxiliary query specialization loss	–
REVE (El Ouahidi et al., 2025)	12/69/408M	Single stage with a masked reconstruction loss and a secondary global-token loss	Electrode coordinates within a 4-D electrode–time encoding and coordinate-radius masking
NeurIPT (Fang et al., 2025)	73.5M	Single stage with an amplitude-aware masked reconstruction loss and an MoE routing-balance loss	3-D electrode coordinates. Intra-lobe pooling followed by inter lobe concatenation.
THD-BAR (Yang et al., 2025)	124/354/1555M	Two stages: topology-hierarchical VQ tokenizer pretraining with compound reconstruction and quantization objectives, followed by next-scale-time autoregressive prediction	Multi-scale brain-topology hierarchy
CodeBrain (Ma et al., 2026)	15.17M	Two stages: dual-tokenizer pretraining with time-domain, spectral, and contrastive objectives, followed by masked discrete-token prediction	–
Uni-NTFM (Chen et al., 2026)	57M–1.9B	Single stage with multiple objectives: masked time-domain reconstruction, frequency-domain reconstruction, and MoE load-balancing loss	Five canonical brain regions, intra-region coordinates, and a unique global identifier to standard electrodes
PATCHCODE (Yu et al., 2026)	2/9/37M	Two stages: region-aware VQ-tokenizer pretraining with spectral, region-contrastive, and code-usage objectives, followed by EMA-teacher P2P, P2C, and C2P prediction	Five anatomical regions, Transformer with region-aware attention bias, and region-guided masking
EmBrace (Jia et al., 2026)	15.6M active†	Pretrained-source fusion: requires three separately pretrained EEG foundation models	The three models used in the paper—LaBraM, CBraMod, and CodeBrain—do not require structural priors.
AxisBrain	**7.46M**	**Single stage: masked raw-patch MSE only**	–

B PRETRAINING AND FINE-TUNING DETAILS

Table 7: Architecture specifications and representation flow for pretraining.

Component	Specification	Setting / input–output relationship
EEG patch construction	Input patches	$X \in \mathbb{R}^{B\times C\times T\times P}$
	Channels / time slots	$C = 19$ / $T = 30$
	Patch duration / samples	1 s / $P = 200$
	Mask ratio / replacement	0.5 / fixed all-zero vector
	Output	$X^{\text{mask}} \in \mathbb{R}^{B\times C\times T\times P}$
Time-domain patch encoder	Input	X^{mask}
	Convolutional layers	Three temporal convolutions, implemented with Conv2d
	Layer-wise convolution specifications	Layer 1: $1 \to 25$, $k = 49$, $s = 25$, $p = 24$; Layer 2: $25 \to 25$, $k = 3$, $s = 1$, $p = 1$; Layer 3: $25 \to 25$, $k = 3$, $s = 1$, $p = 1$
	Normalization / activation	GroupNorm (5 groups) and GELU after each convolution
	Output	Flatten 25×8 features per patch into $E^{\text{time}} \in \mathbb{R}^{B\times C\times T\times d}$
Frequency-domain patch encoder	Input	X^{mask}
	Spectral transform	Real FFT magnitude, `norm='forward'`
	Spectral coefficients	101 magnitude coefficients per patch
	Projection / dropout	Linear $101 \to 200$ / 0.1
	Output	$E^{\text{freq}} \in \mathbb{R}^{B\times C\times T\times d}$
Feature fusion and ACPE	Feature fusion	$E = E^{\text{time}} + E^{\text{freq}}$
	Positional operation	Depthwise Conv2d over the channel–time grid
	Input / output channels / groups	$200/200/200$
	Kernel / stride / padding	$(19, 7)/(1, 1)/(9, 3)$
	Encoder input	$H_1 = E + \text{ACPE}(E) \in \mathbb{R}^{B\times C\times T\times d}$
Criss-cross attention	Layers / input	$\ell = 1, \ldots, 12$ / $H_\ell \in \mathbb{R}^{B\times C\times T\times d}$
	Input normalization	LayerNorm before attention
	Spatial attention	First 100 features; 4 heads across channels at each time
	Temporal attention	Remaining 100 features; 4 heads across time per channel
	Attention dropout	0.1 in both attention branches
	Output	$X^{\text{cc}} = \text{Concat}[X^S, X^T] \in \mathbb{R}^{B\times C\times T\times d}$
Recombining spatial and temporal attention responses	Input	X^{cc}
	Within-token projection	Linear $200 \to 200$
	Projected response	$X^{\text{int}} = W_x X^{\text{cc}} + b_x$
	Output	$R = X^{\text{cc}} + X^{\text{int}} \in \mathbb{R}^{B\times C\times T\times d}$
Cross-axis consistency-guided residual refinement	Inputs	X^{cc} for scoring; R for modulation
	Learned query	One d-dimensional query per layer, initialized from $\mathcal{N}(0, 0.02^2)$
	Query sharing	Shared across samples and time indices
	Time-indexed descriptor	A^T: query pooling across channels, 8 heads, attention dropout 0.1, then LayerNorm
	Channel-indexed descriptor	$A^C_c = T^{-1} \sum_t X^{\text{cc}}_{:,c,t,:}$
	Descriptor projections	Separate linear maps $200 \to 32$: $A^T \to U$, $A^C \to V$
	Projected dimensions	$U \in \mathbb{R}^{B\times T\times 32}$, $V \in \mathbb{R}^{B\times C\times 32}$
	Score computation	$\bar{U}_t = U_t / \max(\|U_t\|_2, \epsilon)$, $\bar{V}_c = V_c / \max(\|V_c\|_2, \epsilon)$; $S_{ct} = \sigma(\bar{U}_t^\top \bar{V}_c)$, $\epsilon = 10^{-12}$
	Score tensor / range	$S \in \mathbb{R}^{B\times C\times T}$; $S_{ct} \in [\sigma(-1), \sigma(1)] \approx [0.2689, 0.7311]$
	Output	$\widetilde{X} = (1 + S) \odot R$, with S broadcast over features
Attention residual and FFN updates	Attention residual	$\widetilde{H}^{\text{attn}}_\ell = H_\ell + \text{Dropout}(\widetilde{X})$
	FFN input	$\text{LN}(\widetilde{H}^{\text{attn}}_\ell)$
	FFN layers	Linear $200 \to 800$, GELU, dropout, linear $800 \to 200$, dropout
	Dropout / LayerNorm ϵ	0.1 / 10^{-5}
	Layer output	$H_{\ell+1} = \widetilde{H}^{\text{attn}}_\ell + \text{FFN}(\text{LN}(\widetilde{H}^{\text{attn}}_\ell))$
Reconstruction head	Input	H_{13}, the output of the 12th encoder layer
	Output projection	Linear $200 \to 200$ per token
	Predicted patches	$\widehat{X} \in \mathbb{R}^{B\times C\times T\times P}$

B.1 Computing Environment

The software environment consisted of Ubuntu 22.04.5 LTS, Python 3.11.7, PyTorch 2.7.0, and CUDA 12.6. The hardware environment consisted of an AMD Ryzen Threadripper PRO 3975WX CPU and a single NVIDIA A6000 GPU. Pretraining for 20 epochs took approximately 40 hours on this setup.

B.2 Data Preprocessing

Pretraining data preprocessing. For pretraining, we used the publicly available preprocessing implementation released with CBraMod. We pretrained on the public Temple University EEG Corpus (TUEG) (Obeid & Picone, 2016). We selected 19 channels from the international 10–20 system and excluded recordings missing any of these channels. The signals were resampled to 200 Hz, followed by a 0.3–75-Hz band-pass filter and a 60-Hz notch filter. Recordings shorter than five minutes were discarded. We removed the first and last minute of each retained recording and divided the remaining signals into non-overlapping 30-s segments, discarding any trailing incomplete segment. Segments containing any amplitude with an absolute value of at least 100 μV were rejected.

Downstream data preprocessing. For MentalArithmetic, Mumtaz2016, SHU-MI, PhysioNet-MI, ISRUC, CHB-MIT, FACED, and TUAB, we used the publicly available preprocessing implementations released with CBraMod. For HMC and SIENA, we used the publicly available preprocessing implementations released with CSBrain. The principal preprocessing steps are summarized below.

We retained the dataset-specific channel configuration instead of restricting all datasets to the 19-channel montage used for pretraining. All signals were resampled to 200 Hz before being provided to the model and were subsequently organized into non-overlapping 1-s patches. Continuous recordings or provided trials were divided into fixed-duration samples according to each benchmark: 5 s for MentalArithmetic and Mumtaz2016, 4 s for SHU-MI and PhysioNet-MI, 30 s for ISRUC and HMC, and 10 s for SIENA, CHB-MIT, FACED, and TUAB. Trailing incomplete samples were discarded during regular non-overlapping segmentation.

Filtering was adapted to each dataset. Mumtaz2016 and TUAB were band-pass filtered at 0.3–75 Hz, with notch filters at 50 and 60 Hz, respectively. ISRUC was filtered at 0.3–35 Hz with a 50-Hz notch filter. PhysioNet-MI was high-pass filtered at 0.3 Hz and notch filtered at 60 Hz. HMC and SIENA were band-pass filtered at 0.1–75 Hz and notch filtered at 50 Hz. No additional filtering was introduced at this preprocessing stage for MentalArithmetic, SHU-MI, FACED, or CHB-MIT. For TUAB, we modified the released CBraMod preprocessing by retaining 19 referential scalp channels and applying common-average referencing over these channels; all other TUAB preprocessing procedures followed the released implementation.

TUAB is derived from TUEG. We report AxisBrain's TUAB performance using an encoder pretrained on TUEG without excluding TUAB. Accordingly, this result should be interpreted as an evaluation with corpus-level overlap rather than transfer to a fully unseen dataset. The reported CBraMod and CodeBrain results were obtained under the same type of overlap, while CSBrain reports pretraining on TUEG without specifying a TUAB exclusion. LaBraM states that its main pretraining setup excludes its downstream datasets, but reports little performance difference when TUAB is incorporated into pretraining. PATCHCODE also pretrains on the parent TUH-EEG corpus but reports subject-disjointness between pretraining and its TUAB benchmarks.

Downstream data partitions. For MentalArithmetic, SHU-MI, PhysioNet-MI, ISRUC, and FACED, the training/validation/test subject ranges were 1–28/29–32/33–36, 1–15/16–20/21–25, 1–70/71–89/90–109, 1–80/81–90/91–100, and 1–80/81–100/101–123, respectively. For SHU-MI, we followed the data-loading protocol implemented in the released CBraMod and CSBrain codebases. Specifically, the CBraMod version downloaded on March 26, 2025 and the released CSBrain implementation randomly retain 30% of the training examples for each fine-tuning run. The subset is determined by the corresponding run seed, whereas the validation and test sets are not subsampled. For CHB-MIT, we assigned chb01–chb20 to training, chb21–chb22 to validation, and chb23–chb24 to testing. For HMC, the first 100 subjects were used for training, the next 25 for validation, and the remaining 26 for testing. For SIENA, samples from subjects with IDs up to PN12 were divided into training and validation sets at an 8:2 ratio within each subject and class; subjects with IDs above PN12

were held out for testing. For TUAB, we used the official training and evaluation sets and divided the training subjects into training and validation sets at an 8:2 ratio within each class. Following the released CSBrain implementation, fine-tuning used a fixed subset of 50,000 10-s segments from the resulting training split; the validation and evaluation sets were not subsampled. For Mumtaz2016, sorted control recordings were divided into 40/8/remaining files and sorted MDD recordings into 42/10/remaining files for training/validation/testing, respectively.

B.3 Encoder Architecture and Hyperparameters for pretraining

Table 8: Optimization hyperparameters for pretraining.

Parameter	Value	Parameter	Value
Optimizer	AdamW	Learning rate	5×10^{-4}
Scheduler	cosine annealing	Cosine-schedule horizon	40 epochs
Minimum LR	10^{-5}	Pretraining seed	42
Weight decay	0.05	Gradient clipping	1
Epochs	20	Batch size	128
Dropout	0.1	Mask ratio	0.5

Table 7 summarizes the architecture and the input–output relationships between components. The recombination of spatial and temporal attention responses and cross-axis consistency-guided residual refinement are applied in all 12 Transformer layers. The table lists modules used in the forward computation.

The optimization hyperparameters used for pretraining are summarized in Table 8. The pretraining objective is the mean squared error between the reconstructed and original signal values, computed exclusively over masked patches.

B.4 Downstream Classifier and Fine-Tuning Details

For downstream fine-tuning, the backbone reconstruction projection is replaced by an identity mapping. Except for ISRUC, all $C \times T$ patch representations are flattened and passed to the following three-layer classifier:

$$\mathbb{R}^{CTd} \to \mathbb{R}^{Td} \to \mathbb{R}^{d} \to \mathbb{R}^{K}, \qquad d = 200, \tag{14}$$

where $K = 1$ for binary classification and equals the number of classes for multi-class classification. ELU activation and dropout of 0.1 follow the first two linear layers. ISRUC is processed as a sequence of 30-second sleep epochs: each epoch's $6 \times 30 \times 200$ backbone output is projected to 512 dimensions, followed by GELU, a one-layer Transformer encoder with four heads and a 2,048-dimensional feed-forward block, and a linear classifier that predicts one of the five sleep stages for each epoch.

AxisBrain downstream runs use the dataset's original channel configuration, a 200-Hz sampling rate, one-second patches, and the class-count-specific metrics defined in Section 3.1. The common fine-tuning hyperparameters are summarized in Table 9. Only the learning rate, number of training epochs, and batch size are adjusted for each dataset, as reported in Table 10. For each dataset, we selected the checkpoint using the validation set. The selection metric was AUROC for binary classification, except for CHB-MIT, where we used balanced accuracy, and Cohen's kappa for multi-class classification. All reported test metrics were computed using the selected checkpoint. All reported results are based on five runs using the same five seeds—1000, 2000, 3000, 4000, and 5000—across all compared variants.

Table 9: Common optimization hyperparameters for downstream fine-tuning.

Parameter	Value	Parameter	Value
Optimizer	AdamW	Weight decay	0.05
Scheduler	cosine annealing	Minimum LR	10^{-6}
Gradient clipping	1	Dropout	0.1
Label smoothing	0.1	LR parameter groups	shared
Backbone initialization	pretrained	Backbone updating	enabled

Table 10: Dataset-specific downstream settings.

Dataset	Learning rate	Epochs	Batch size
MentalArithmetic	7.5×10^{-6}	50	64
Mumtaz2016	2.5×10^{-6}	50	64
SHU-MI	5×10^{-5}	50	64
PhysioNet-MI	2.5×10^{-4}	50	64
ISRUC	7.5×10^{-5}	10	8
HMC	5×10^{-4}	30	64
SIENA	2.5×10^{-6}	10	64
CHB-MIT	1×10^{-4}	10	64
FACED	6×10^{-4}	50	64
TUAB	5×10^{-5}	5	64

C Detailed Experimental Results

C.1 Complete Comparison with Foundation and Non-Foundation Models

Table 11: Comparison with EEG foundation models and task-specific supervised models on MentalArithmetic, Mumtaz2016, SHU-MI, PhysioNet-MI, ISRUC, and HMC. Results are reported as mean ± standard deviation over five runs. Bold and underline mark the best and second-best results respectively.

	MentalArithmetic			Mumtaz2016		
Method	ACC-B	AUC-PR	AUROC	ACC-B	AUC-PR	AUROC
Task-specific models						
EEGNet (0.003M)	0.6770±0.0116	0.5763±0.0102	0.7321±0.0108	0.9232±0.0104	0.9626±0.0095	0.9639±0.0093
EEGConformer (0.55M)	0.6805±0.0123	0.5829±0.0134	0.7424±0.0128	0.9308±0.0117	0.9684±0.0105	0.9702±0.0101
SPaRCNet (0.79M)	0.6879±0.0107	0.5825±0.0193	0.7418±0.0132	0.9316±0.0095	0.9754±0.0065	0.9781±0.0063
ContraWR (1.6M)	0.6631±0.0097	0.5787±0.0164	0.7332±0.0082	0.9195±0.0115	0.9589±0.0102	0.9621±0.0092
CNN-Transformer (3.2M)	0.6779±0.0268	0.5777±0.0285	0.7258±0.0336	0.9305±0.0068	0.9757±0.0074	0.9742±0.0059
FFCL (2.4M)	0.6798±0.0142	0.5786±0.0266	0.7330±0.0198	0.9314±0.0083	0.9717±0.0021	0.9753±0.0033
ST-Transformer (3.5M)	0.6631±0.0173	0.5672±0.0259	0.7132±0.0174	0.9135±0.0103	0.9578±0.0086	0.9594±0.0059
EEG foundation models						
LaBraM (5.8M)	0.6909±0.0125	0.5999±0.0155	0.7721±0.0093	0.9409±0.0079	0.9798±0.0093	0.9782±0.0057
CBraMod (4.03M)	0.7256±0.0132	0.6267±0.0099	0.7905±0.0073	0.9560±0.0056	0.9923±0.0032	0.9921±0.0025
CSBrain (4.9M)	0.7558±0.0106	0.6696±0.0221	0.8478±0.0297	0.9643±0.0155	0.9936±0.0034	0.9956±0.0020
PATCHCODE (37M)	0.7694±0.0091	0.6915±0.0184	0.8632±0.0217	**0.9721±0.0083**	0.9952±0.0028	**0.9971±0.0017**
CodeBrain (15.17M)	0.7514±0.0203	0.7177±0.0421	**0.8707±0.0209**	–	–	–
AxisBrain (7.46M)	**0.8014±0.0067**	**0.7758±0.0108**	0.8243±0.0070	0.9616±0.0095	**0.9959±0.0005**	0.9957±0.0005

	SHU-MI			PhysioNet-MI		
Method	ACC-B	AUC-PR	AUROC	ACC-B	Kappa	W-F1
Task-specific supervised models						
EEGNet (0.003M)	0.5889±0.0177	0.6311±0.0142	0.6283±0.0152	0.5814±0.0125	0.4468±0.0199	0.5796±0.0115
EEGConformer (0.55M)	0.5900±0.0107	0.6370±0.0093	0.6351±0.0101	0.6049±0.0104	0.4736±0.0171	0.6062±0.0095
SPaRCNet (0.79M)	0.5978±0.0097	0.6510±0.0062	0.6431±0.0117	0.5932±0.0152	0.4564±0.0234	0.5937±0.0147
ContraWR (1.6M)	0.5873±0.0128	0.6315±0.0105	0.6273±0.0113	0.5892±0.0133	0.4527±0.0248	0.5918±0.0116
CNN-Transformer (3.2M)	0.5975±0.0169	0.6412±0.0076	0.6343±0.0082	0.6053±0.0118	0.4725±0.0223	0.6041±0.0105
FFCL (2.4M)	0.5692±0.0252	0.5943±0.0171	0.6326±0.0082	0.5726±0.0092	0.4323±0.0182	0.5701±0.0079
ST-Transformer (3.5M)	0.5992±0.0206	0.6394±0.0122	0.6431±0.0111	0.6035±0.0081	0.4712±0.0199	0.6053±0.0075
EEG foundation models						
LaBraM (5.8M)	0.6166±0.0192	0.6761±0.0083	0.6604±0.0091	0.6173±0.0122	0.4912±0.0192	0.6177±0.0141
CBraMod (4.03M)	0.6370±0.0151	0.7139±0.0088	0.6988±0.0068	**0.6417±0.0091**	**0.5222±0.0169**	**0.6427±0.0100**
CSBrain (4.9M)	0.6417±0.0037	0.7230±0.0079	0.7200±0.0058	0.6304±0.0090	0.5071±0.0120	0.6308±0.0095
PATCHCODE (37M)	**0.6593±0.0068**	**0.7382±0.0065**	**0.7351±0.0059**	0.6230±0.0068	0.5085±0.0088	0.6195±0.0075
CodeBrain (15.17M)	0.6431±0.0066	0.7166±0.0106	0.7124±0.0050	–	–	–
AxisBrain (7.46M)	0.6287±0.0101	0.6791±0.0146	0.6840±0.0101	0.6387±0.0046	0.5182±0.0061	0.6395±0.0047

	ISRUC			HMC		
Method	ACC-B	Kappa	W-F1	ACC-B	Kappa	W-F1
Task-specific supervised models						
EEGNet (0.003M)	0.7154±0.0121	0.7040±0.0173	0.7513±0.0124	0.6534±0.0122	0.5886±0.0201	0.6536±0.0168
EEGConformer (0.55M)	0.7400±0.0133	0.7143±0.0162	0.7634±0.0151	0.7149±0.0086	0.6432±0.0055	0.7080±0.0039
SPaRCNet (0.79M)	0.7487±0.0075	0.7097±0.0132	0.7624±0.0092	0.4756±0.1109	0.3147±0.1315	0.4108±0.1310
ContraWR (1.6M)	0.7402±0.0126	0.7178±0.0156	0.7610±0.0137	0.4242±0.0541	0.2340±0.0554	0.2987±0.0288
CNN-Transformer (3.2M)	0.7363±0.0087	0.7129±0.0121	0.7719±0.0153	0.6573±0.0141	0.5961±0.0105	0.6896±0.0065
FFCL (2.4M)	0.7277±0.0182	0.7016±0.0292	0.7614±0.0197	0.4427±0.0702	0.2542±0.0654	0.2902±0.0485
ST-Transformer (3.5M)	0.7381±0.0205	0.7013±0.0352	0.7681±0.0175	0.2559±0.0141	0.0503±0.0183	0.1428±0.0122
EEG foundation models						
LaBraM (5.8M)	0.7633±0.0102	0.7231±0.0182	0.7810±0.0133	0.7277±0.0101	0.6813±0.0053	0.7454±0.0027
CBraMod (4.03M)	0.7865±0.0110	0.7442±0.0152	0.8011±0.0099	0.7269±0.0041	0.6685±0.0104	0.7395±0.0089
CSBrain (4.9M)	0.7925±0.0030	0.7406±0.0102	0.7990±0.0091	0.7345±0.0047	0.6818±0.0046	0.7506±0.0042
PATCHCODE (37M)	**0.8120±0.0068**	0.7520±0.0085	0.8095±0.0075	**0.7556±0.0041**	**0.7052±0.0045**	**0.7715±0.0042**
CodeBrain (15.17M)	0.7835±0.0033	0.7476±0.0040	0.8020±0.0018	–	–	–
AxisBrain (7.46M)	0.8035±0.0032	**0.7636±0.0079**	**0.8186±0.0060**	0.7274±0.0051	0.6792±0.0038	0.7480±0.0029

Table 12: Comparison with EEG foundation models and task-specific supervised models on SIENA, CHB-MIT, FACED, and TUAB.

Method	SIENA			CHB-MIT		
	ACC-B	AUC-PR	AUROC	ACC-B	AUC-PR	AUROC
Task-specific supervised models						
EEGNet (0.003M)	0.7487±0.0521	0.3753±0.0867	0.8687±0.0527	0.5658±0.0106	0.1914±0.0182	0.8048±0.0136
EEGConformer (0.55M)	0.7556±0.0210	0.2091±0.0786	0.8159±0.0261	0.5976±0.0141	0.2209±0.0215	0.8226±0.0170
SPaRCNet (0.79M)	0.6572±0.0381	0.3164±0.0659	0.7334±0.0857	0.5876±0.0191	0.1247±0.0119	0.8143±0.0148
ContraWR (1.6M)	0.6546±0.0311	0.3711±0.0405	0.7819±0.0596	0.6344±0.0002	0.2264±0.0174	0.8097±0.0114
CNN-Transformer (3.2M)	0.6982±0.0560	0.3835±0.0709	0.8719±0.0403	0.6389±0.0067	0.2479±0.0227	0.8662±0.0082
FFCL (2.4M)	0.6616±0.0391	0.3938±0.0903	0.8154±0.1155	0.6262±0.0104	0.2049±0.0346	0.8271±0.0051
ST-Transformer (3.5M)	0.7527±0.0381	0.3636±0.0252	0.8884±0.0091	0.5915±0.0195	0.1422±0.0094	0.8237±0.0491
EEG foundation models						
LaBraM (5.8M)	0.7082±0.0329	0.3122±0.0976	0.8814±0.0328	0.7075±0.0358	0.3287±0.0402	0.8679±0.0199
CBraMod (4.03M)	0.7317±0.0647	0.4107±0.0720	0.9038±0.0218	0.7398±0.0284	0.3689±0.0382	0.8892±0.0154
CSBrain (4.9M)	0.7662±0.0471	0.4871±0.0343	0.9076±0.0119	0.7262±0.0115	0.5164±0.0449	0.8915±0.0321
PATCHCODE (37M)	0.7685±0.0388	**0.5020±0.0428**	**0.9102±0.0202**	0.7620±0.0128	0.3910±0.0152	0.9085±0.0095
CodeBrain (15.17M)	–	–	–	0.7273±0.0240	0.4377±0.0288	0.8961±0.0174
AxisBrain (7.46M)	**0.8023±0.0065**	0.3478±0.0099	0.9007±0.0023	**0.7783±0.0246**	**0.6029±0.0828**	**0.9263±0.0183**

Method	FACED			TUAB		
	ACC-B	Kappa	W-F1	ACC-B	AUC-PR	AUROC
Task-specific supervised models						
EEGNet (0.003M)	0.4090±0.0122	0.3342±0.0251	0.4124±0.0141	0.7642±0.0036	0.8299±0.0043	0.8412±0.0031
EEGConformer (0.55M)	0.4559±0.0125	0.3858±0.0186	0.4514±0.0107	0.7758±0.0049	0.8427±0.0054	0.8445±0.0038
SPaRCNet (0.79M)	0.4673±0.0155	0.3978±0.0289	0.4729±0.0133	0.7896±0.0018	0.8414±0.0018	0.8676±0.0012
ContraWR (1.6M)	0.4887±0.0078	0.4231±0.0151	0.4884±0.0074	0.7746±0.0041	0.8421±0.0104	0.8456±0.0074
CNN-Transformer (3.2M)	0.4697±0.0132	0.4017±0.0166	0.4702±0.0125	0.7777±0.0022	0.8433±0.0039	0.8461±0.0013
FFCL (2.4M)	0.4673±0.0158	0.3987±0.0338	0.4699±0.0145	0.7848±0.0038	0.8448±0.0065	0.8569±0.0051
ST-Transformer (3.5M)	0.4810±0.0079	0.4137±0.0133	0.4795±0.0096	0.7966±0.0023	0.8521±0.0026	0.8707±0.0019
EEG foundation models						
LaBraM (5.8M)	0.5273±0.0107	0.4698±0.0188	0.5288±0.0102	0.8140±0.0019	0.8965±0.0016	0.9022±0.0009
CBraMod (4.03M)	0.5509±0.0089	0.5041±0.0122	0.5618±0.0093	0.8289±0.0022	**0.9258±0.0008**	**0.9227±0.0011**
CSBrain (4.9M)	0.5752±0.0042	0.5204±0.0036	0.5796±0.0031	0.8172±0.0043	0.9005±0.0066	0.8957±0.0046
PATCHCODE (37M)	0.5950±0.0072	0.5124±0.0089	0.5888±0.0078	0.8250±0.0035	0.9125±0.0052	0.9102±0.0038
CodeBrain (15.17M)	0.5941±0.0098	0.5406±0.0084	0.5953±0.0113	**0.8294±0.0013**	0.9100±0.0006	0.9030±0.0009
AxisBrain (7.46M)	**0.5961±0.0039**	**0.5438±0.0049**	**0.6043±0.0041**	0.8290±0.0019	0.9084±0.0026	0.9027±0.0024

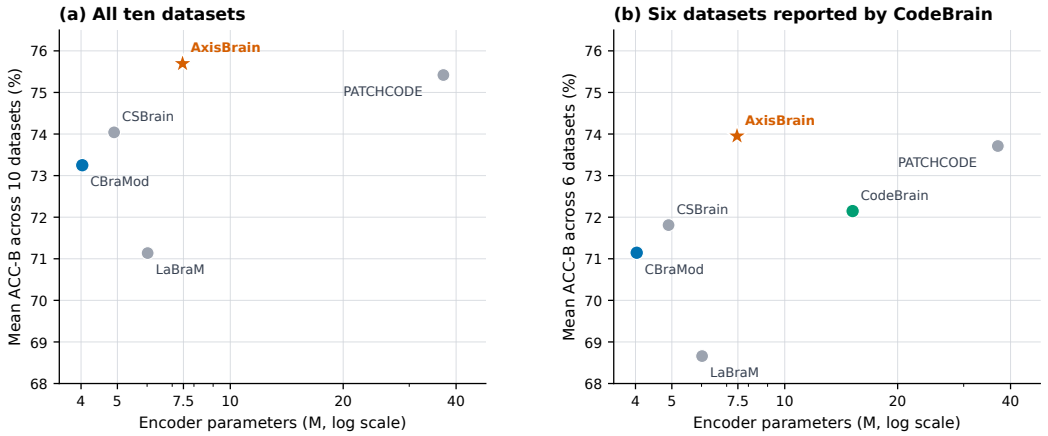


Figure 4: Mean balanced accuracy versus encoder size. (a) Mean balanced accuracy across all ten downstream datasets. CodeBrain is omitted because it does not report results for all ten datasets. (b) Mean balanced accuracy across the six datasets for which CodeBrain reports results: MentalArithmetic, SHU-MI, ISRUC, CHB-MIT, FACED, and TUAB. Parameter counts are shown on a logarithmic scale.

To complement the main results, Tables 11 and 12 provide an expanded comparison that includes both EEG foundation models and representative task-specific supervised EEG models. The complete results across all ten downstream datasets enable a unified comparison across these two model families. Results for the supervised baselines are reproduced from PATCHCODE (Yu et al., 2026), including EEGNet (Lawhern et al., 2018), EEGConformer (Song et al., 2023), SPaRCNet (Jing et al., 2023), ContraWR (Yang et al., 2023b), CNN–Transformer (Peh et al., 2022), FFCL (Li et al., 2022), and ST–Transformer (Song et al., 2021).

Figure 4 compares mean balanced accuracy with encoder size. AxisBrain achieves competitive mean balanced accuracy in both the ten-dataset comparison and the six-dataset comparison that includes CodeBrain, while using substantially fewer parameters than PATCHCODE. This demonstrates a favorable balance between model size and downstream performance.

C.2 Frozen-Encoder Evaluation

To examine the transferability of the pretrained representations without task-specific encoder adaptation, we evaluate AxisBrain under a frozen-encoder setting. The pretrained encoder is kept fixed, and only the dataset-specific classification head is trained. For each dataset, frozen-encoder evaluation and full fine-tuning use the same pretrained checkpoint, head architecture, data split, evaluation metrics, and five random seeds.

Table 13: Frozen-encoder evaluation and full fine-tuning of AxisBrain across ten downstream datasets. Entries report mean ± sample standard deviation over five seeds. For binary tasks, the second and third metric columns denote AUC-PR and AUROC; for multi-class tasks, they denote Cohen's kappa and weighted F1. Bold indicates the higher mean between the two adaptation settings for each metric.

	Frozen encoder			Full fine-tuning		
Dataset	ACC-B	AUC-PR / κ	AUROC / W-F1	ACC-B	AUC-PR / κ	AUROC / W-F1
MentalArithmetic	0.5167±0.0119	0.6077±0.0096	0.7731±0.0051	**0.8014±0.0067**	**0.7758±0.0108**	**0.8243±0.0070**
Mumtaz2016	0.9038±0.0086	0.9759±0.0013	0.9817±0.0010	**0.9616±0.0095**	**0.9959±0.0005**	**0.9957±0.0005**
SHU-MI	0.5745±0.0347	0.6440±0.0074	0.6483±0.0046	**0.6287±0.0101**	**0.6791±0.0146**	**0.6840±0.0101**
PhysioNet-MI	0.5462±0.0054	0.3947±0.0072	0.5420±0.0093	**0.6387±0.0046**	**0.5182±0.0061**	**0.6395±0.0047**
ISRUC	0.3985±0.0313	0.3136±0.0360	0.4478±0.0281	**0.8035±0.0032**	**0.7636±0.0079**	**0.8186±0.0060**
HMC	0.3960±0.0097	0.2838±0.0154	0.4476±0.0179	**0.7274±0.0051**	**0.6792±0.0038**	**0.7480±0.0029**
SIENA	0.6328±0.0744	0.0377±0.0124	0.7292±0.0264	**0.8023±0.0065**	**0.3478±0.0099**	**0.9007±0.0023**
CHB-MIT	**0.7827±0.0308**	0.3899±0.0507	0.9233±0.0105	0.7783±0.0246	**0.6029±0.0828**	**0.9263±0.0183**
FACED	0.3519±0.0172	0.2631±0.0197	0.3239±0.0236	**0.5961±0.0039**	**0.5438±0.0049**	**0.6043±0.0041**
TUAB	0.6793±0.0075	0.8593±0.0009	0.8569±0.0011	**0.8290±0.0019**	**0.9084±0.0026**	**0.9027±0.0024**

Table 13 shows that full fine-tuning increases the ten-dataset mean balanced accuracy from 57.82% to 75.67%, a gain of 17.85 percentage points. The largest gains occur on ISRUC and HMC, where balanced accuracy increases by 40.50 and 33.14 percentage points, respectively. Both are sleep-staging datasets evaluated on 30-s segments, matching the pretraining segment duration. Full fine-tuning is therefore especially beneficial on these two datasets, although the present experiments do not establish why their frozen-to-fine-tuned gaps are larger.

FACED and MentalArithmetic also show relatively low frozen-encoder balanced accuracy. FACED contains nine emotion classes, and its 24.42-point improvement with full fine-tuning suggests that separating these fine-grained classes may benefit from task-specific adaptation throughout the encoder. MentalArithmetic has only 1,343 training samples, the fewest among the downstream datasets; Mumtaz2016 has the next-smallest training set with 4,891 samples. The limited size of MentalArithmetic may make it difficult to estimate an effective decision boundary using only the classification head. However, the substantially stronger frozen-encoder performance on Mumtaz2016 indicates that training-set size alone does not explain the result, and differences in the tasks and their class-discriminative patterns are also likely relevant.

CHB-MIT is the only dataset on which frozen-encoder balanced accuracy is slightly higher than full fine-tuning, at 78.27% versus 77.83%. Its frozen-encoder AUROC is also similar to that obtained with full fine-tuning (0.9233 versus 0.9263), although its AUC-PR is substantially lower (0.3899 versus 0.6029). This mixed metric-level result does not establish an overall advantage for freezing the encoder, but it indicates that the pretrained representation already supports strong seizure discrimination without encoder adaptation.

The reduction under frozen-encoder evaluation is concentrated in a subset of datasets. ISRUC, HMC, MentalArithmetic, and FACED show the four largest decreases in balanced accuracy, ranging from 24.42 to 40.50 percentage points. Across the remaining six datasets, mean balanced accuracy decreases from 77.31% with full fine-tuning to 68.66% with the frozen encoder, a difference of 8.65 points.

Table 14: Frozen-encoder results on three overlapping datasets. CBraMod (Fixed) values are reproduced from Table 18 of Wang et al. (2025a), while AxisBrain (Fixed) results are obtained in this work over five seeds.

Dataset	Metric	CBraMod (Fixed)	AxisBrain (Fixed)
FACED	Balanced accuracy	0.3146 ± 0.0346	$\mathbf{0.3519 \pm 0.0172}$
	Cohen's kappa	0.2579 ± 0.0542	$\mathbf{0.2631 \pm 0.0197}$
	Weighted F1	0.3077 ± 0.0298	$\mathbf{0.3239 \pm 0.0236}$
PhysioNet-MI	Balanced accuracy	0.3845 ± 0.0345	$\mathbf{0.5462 \pm 0.0054}$
	Cohen's kappa	0.2983 ± 0.0498	$\mathbf{0.3947 \pm 0.0072}$
	Weighted F1	0.3946 ± 0.0378	$\mathbf{0.5420 \pm 0.0093}$
SHU-MI	Balanced accuracy	0.5217 ± 0.0247	$\mathbf{0.5745 \pm 0.0347}$
	AUC-PR	0.5304 ± 0.0253	$\mathbf{0.6440 \pm 0.0074}$
	AUROC	0.5238 ± 0.0317	$\mathbf{0.6483 \pm 0.0046}$

We additionally compare AxisBrain with the CBraMod (Fixed) results reported in Table 18 of Wang et al. (2025a). As shown in Table 14, AxisBrain obtains higher means on all nine dataset–metric combinations. In particular, its balanced accuracy exceeds the published CBraMod (Fixed) result by 16.17 percentage points on PhysioNet-MI, 5.28 points on SHU-MI, and 3.73 points on FACED. These results suggest that the AxisBrain representation retains more directly usable information for head-only downstream adaptation on the three overlapping datasets. Both models use fixed pretrained encoders and trained task-specific heads. We expect their head architectures to be similar because our evaluation uses the default task-specific heads provided in the public CBraMod implementation. However, the CBraMod paper describes its heads only as multilayer perceptrons, so exact architectural correspondence and identical optimization conditions for the published results cannot be confirmed. The comparison should therefore be interpreted as a reference comparison rather than a fully controlled ablation. A systematic frozen-encoder comparison across all foundation-model baselines would require a common evaluation implementation and comparable hyperparameter-search budgets across models and datasets, which are beyond the scope of this evaluation.

C.3 Reproduction of CBraMod Results

We reproduced CBraMod using its released implementation and evaluated it on the same ten downstream datasets. Table 15 compares our reproduction with the reference CBraMod results used in the main comparison. For eight datasets, the reference values are those reported in the original CBraMod paper (Wang et al., 2025a). Because HMC and SIENA were not evaluated in that paper, their reference values are the CBraMod reproduction results reported by CSBrain (Zhou et al., 2025). The ten-dataset mean ACC-B is 73.25% for these reference results and 71.34% for our reproduction, although our reproduced ACC-B is higher on ISRUC and FACED. Because the proposed operation was implemented by extending the released CBraMod code, we use our direct reproduction as the controlled baseline for the ablation in Section 3.3. Differences from the reference results may reflect sensitivity to dataset-specific fine-tuning hyperparameters and the finite set of configurations evaluated in this study. Although Table 6 of the original CBraMod paper (Wang et al., 2025a) provides dataset-specific fine-tuning hyperparameters, our empirically selected configurations yielded better reproduction performance in our experimental environment. We therefore report the reproduction results obtained using these configurations. The common fine-tuning hyperparameters are provided in Table 9, while the learning rate, number of epochs, and batch size selected for each dataset are listed in Table 16. Accordingly, our reproduction should be interpreted only as a within-implementation reference for quantifying the incremental gain from spatial–temporal response recombination, rather than as a comprehensive reassessment of CBraMod. All cross-paper comparisons use the reference results reported by the corresponding source papers.

Table 15: Reference CBraMod results and our direct reproduction using the released implementation. The reference results for HMC and SIENA are CBraMod reproductions reported by CSBrain (Zhou et al., 2025); the remaining reference results are reported in the original CBraMod paper (Wang et al., 2025a). Values are mean ± sample standard deviation over five seeds. Binary tasks report ACC-B, AUC-PR, and AUROC; multi-class tasks report ACC-B, Cohen's kappa, and W-F1.

	MentalArithmetic			Mumtaz2016		
Result source	ACC-B	AUC-PR	AUROC	ACC-B	AUC-PR	AUROC
Reported	0.7256±0.0132	0.6267±0.0099	0.7905±0.0073	0.9560±0.0056	0.9923±0.0032	0.9921±0.0025
Reproduced	0.7069±0.0386	0.6429±0.0479	0.8051±0.0196	0.9209±0.0102	0.9864±0.0017	0.9864±0.0015
	SHU-MI			PhysioNet-MI		
Result source	ACC-B	AUC-PR	AUROC	ACC-B	Kappa	W-F1
Reported	0.6370±0.0151	0.7139±0.0088	0.6988±0.0068	0.6417±0.0091	0.5222±0.0169	0.6427±0.0100
Reproduced	0.6163±0.0080	0.6819±0.0130	0.6805±0.0137	0.6321±0.0093	0.5094±0.0123	0.6325±0.0094
	ISRUC			HMC		
Result source	ACC-B	Kappa	W-F1	ACC-B	Kappa	W-F1
Reported	0.7865±0.0110	0.7442±0.0152	0.8011±0.0099	0.7269±0.0041	0.6685±0.0104	0.7395±0.0089
Reproduced	0.7930±0.0063	0.7608±0.0058	0.8155±0.0047	0.7186±0.0035	0.6645±0.0049	0.7362±0.0056
	SIENA			CHB-MIT		
Result source	ACC-B	AUC-PR	AUROC	ACC-B	AUC-PR	AUROC
Reported	0.7317±0.0647	0.4107±0.0720	0.9038±0.0218	0.7398±0.0284	0.3689±0.0382	0.8892±0.0154
Reproduced	0.6699±0.0153	0.3475±0.0320	0.8785±0.0062	0.6780±0.1284	0.4160±0.2218	0.8773±0.0843
	FACED			TUAB		
Result source	ACC-B	Kappa	W-F1	ACC-B	AUC-PR	AUROC
Reported	0.5509±0.0089	0.5041±0.0122	0.5618±0.0093	0.8289±0.0022	0.9258±0.0008	0.9227±0.0011
Reproduced	0.5929±0.0045	0.5388±0.0053	0.5972±0.0037	0.8055±0.0046	0.8890±0.0019	0.8864±0.0016

Table 16: Fine-tuning hyperparameters for our CBraMod reproduction.

Dataset	Learning rate	Epochs	Batch size
MentalArithmetic	2.5×10^{-5}	50	64
Mumtaz2016	5×10^{-6}	50	64
SHU-MI	5×10^{-5}	50	64
PhysioNet-MI	2.5×10^{-4}	50	64
ISRUC	7.5×10^{-5}	10	8
HMC	1×10^{-4}	30	64
SIENA	1×10^{-5}	50	64
CHB-MIT	5×10^{-5}	10	64
FACED	4×10^{-4}	50	64
TUAB	5×10^{-5}	5	64

C.4 FULL FIVE-SEED RESULTS

Because performance on EEG datasets can vary across random seeds, we report the result of each seed to facilitate reproduction of our findings using the released code. The complete five-seed results are provided in Table 17.

Table 17: Detailed downstream results of AxisBrain for five random seeds.

	MentalArithmetic			Mumtaz2016		
Seed	ACC-B	AUC-PR	AUROC	ACC-B	AUC-PR	AUROC
1000	0.8021	0.7808	0.8236	0.9623	0.9959	0.9956
2000	0.8056	0.7846	0.8262	0.9565	0.9965	0.9963
3000	0.7917	0.7575	0.8148	0.9688	0.9956	0.9955
4000	0.7986	0.7809	0.8342	0.9484	0.9953	0.9950
5000	0.8090	0.7750	0.8225	0.9720	0.9963	0.9961
Mean ± SD	**0.8014±0.0067**	**0.7758±0.0108**	**0.8243±0.0070**	**0.9616±0.0095**	**0.9959±0.0005**	**0.9957±0.0005**
	SHU-MI			PhysioNet-MI		
Seed	ACC-B	AUC-PR	AUROC	ACC-B	Kappa	W-F1
1000	0.6385	0.6962	0.6960	0.6379	0.5171	0.6390
2000	0.6332	0.6819	0.6838	0.6456	0.5275	0.6467
3000	0.6302	0.6859	0.6890	0.6335	0.5112	0.6342
4000	0.6118	0.6573	0.6685	0.6362	0.5149	0.6370
5000	0.6298	0.6740	0.6826	0.6401	0.5201	0.6404
Mean ± SD	**0.6287±0.0101**	**0.6791±0.0146**	**0.6840±0.0101**	**0.6387±0.0046**	**0.5182±0.0061**	**0.6395±0.0047**
	ISRUC			HMC		
Seed	ACC-B	Kappa	W-F1	ACC-B	Kappa	W-F1
1000	0.8074	0.7676	0.8213	0.7337	0.6835	0.7522
2000	0.8024	0.7650	0.8213	0.7271	0.6805	0.7481
3000	0.8034	0.7497	0.8080	0.7272	0.6778	0.7470
4000	0.8055	0.7666	0.8204	0.7292	0.6807	0.7487
5000	0.7990	0.7690	0.8221	0.7196	0.6734	0.7442
Mean ± SD	**0.8035±0.0032**	**0.7636±0.0079**	**0.8186±0.0060**	**0.7274±0.0051**	**0.6792±0.0038**	**0.7480±0.0029**
	SIENA			CHB-MIT		
Seed	ACC-B	AUC-PR	AUROC	ACC-B	AUC-PR	AUROC
1000	0.8124	0.3565	0.9009	0.7606	0.6859	0.9421
2000	0.7979	0.3383	0.8979	0.7551	0.5690	0.9082
3000	0.8036	0.3603	0.9039	0.7940	0.5110	0.9084
4000	0.8019	0.3422	0.9016	0.7687	0.5536	0.9255
5000	0.7955	0.3417	0.8990	0.8133	0.6951	0.9474
Mean ± SD	**0.8023±0.0065**	**0.3478±0.0099**	**0.9007±0.0023**	**0.7783±0.0246**	**0.6029±0.0828**	**0.9263±0.0183**
	FACED			TUAB		
Seed	ACC-B	Kappa	W-F1	ACC-B	AUC-PR	AUROC
1000	0.5942	0.5427	0.6064	0.8320	0.9068	0.9005
2000	0.5926	0.5391	0.5993	0.8278	0.9121	0.9060
3000	0.5950	0.5427	0.6041	0.8278	0.9093	0.9041
4000	0.5962	0.5424	0.6019	0.8276	0.9053	0.9004
5000	0.6027	0.5521	0.6100	0.8299	0.9086	0.9024
Mean ± SD	**0.5961±0.0039**	**0.5438±0.0049**	**0.6043±0.0041**	**0.8290±0.0019**	**0.9084±0.0026**	**0.9027±0.0024**

D SPARSE CHANNEL POOLING FOR MOTOR-IMAGERY TASKS

The relatively weaker results on SHU-MI and PhysioNet-MI may arise because dense pooling of the cross-channel context mixes spatially localized motor-related evidence with less informative channels. To examine this possibility, we replace the original softmax pooling used to construct the time-indexed cross-channel descriptor with entmax pooling (Peters et al., 2019). Entmax can assign exactly zero weight to channels while retaining a differentiable learned mixture over its selected support. We compare the two pooling schemes on SHU-MI and PhysioNet-MI using the same fine-tuning protocol and five matched seeds.

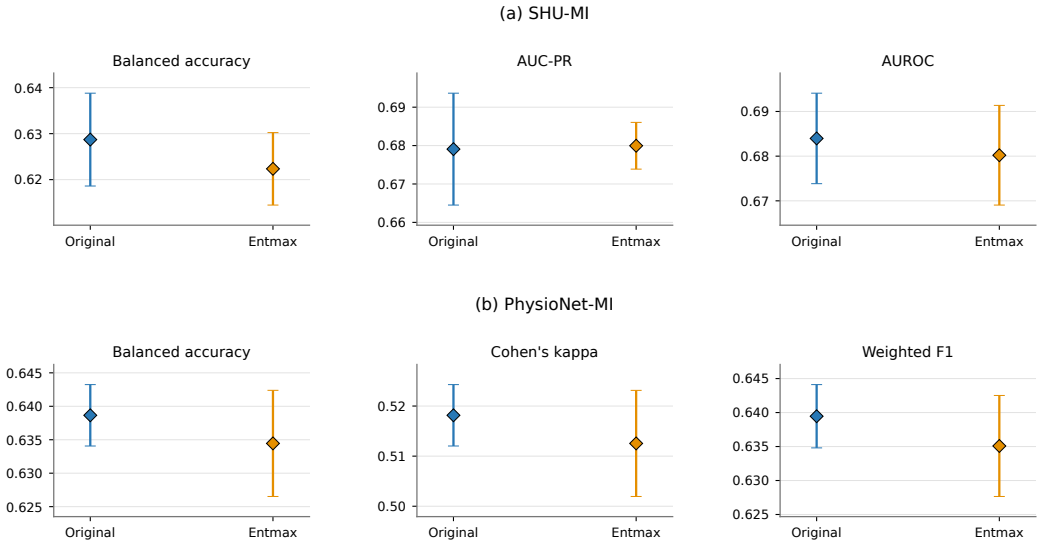


Figure 5: Original and entmax channel pooling on two motor-imagery datasets. Diamonds show the mean over five matched fine-tuning seeds, and error bars show one sample standard deviation.

As shown in Figure 5, the original pooling performs slightly better on five of the six reported metrics. On SHU-MI, it exceeds entmax by 0.64 percentage points in balanced accuracy and 0.38 points in AUROC, whereas entmax improves AUC-PR by only 0.09 points. On PhysioNet-MI, the original pooling is higher by 0.42 points in balanced accuracy, 0.56 points in Cohen's kappa, and 0.44 points in weighted F1. Thus, directly imposing sparse channel support does not recover the comparatively weaker motor-imagery performance.

One possible explanation is that exact sparsity makes channel-selection errors more costly. If the learned pooling logits exclude a channel carrying relevant motor-related evidence, its contribution becomes exactly zero rather than remaining available with a small dense weight. This risk is amplified when informative channel sets vary across subjects or trials, or when evidence is distributed across several channels instead of confined to a fixed sparse support. The results therefore do not rule out the value of spatially selective pooling, but indicate that applying a generic sparse mapping is insufficient. Future work should develop EEG-specific descriptors and selection mechanisms that preserve weak but complementary channel evidence while still emphasizing spatially localized responses.

E MASK RATIO SENSITIVITY

We vary the pretraining mask ratio from 0.3 to 0.7 and evaluate each corresponding pretrained model on TUAB and ISRUC, selected to represent binary and multi-class downstream classification, respectively. ISRUC reports balanced accuracy (ACC-B), Cohen's kappa, and weighted F1 (W-F1), whereas TUAB reports ACC-B, AUC-PR, and AUROC. Figure 6 presents the mean and sample standard deviation over five fine-tuning runs.

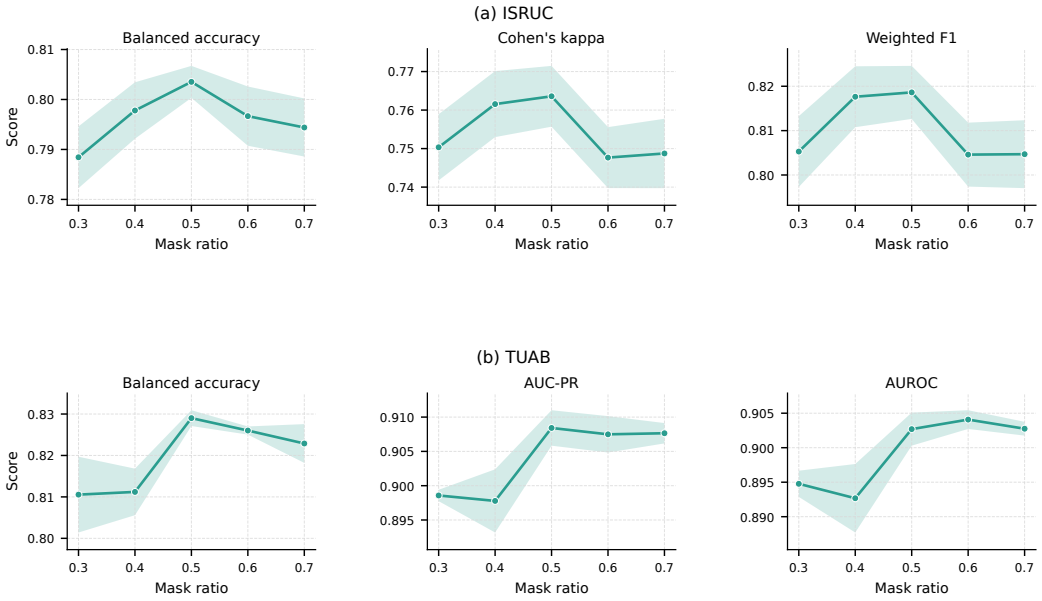


Figure 6: Downstream performance on ISRUC and TUAB across pretraining mask ratios. Each point shows the mean over five fine-tuning runs, and the shaded region shows one sample standard deviation.

The two-dataset mean ACC-B increases from 79.95% at a mask ratio of 0.3 to 80.45% at 0.4 and reaches its maximum of 81.63% at 0.5. It then decreases to 81.13% and 80.86% at ratios of 0.6 and 0.7, respectively. On ISRUC, all three metrics reach their highest values at a mask ratio of 0.5. On TUAB, ACC-B and AUC-PR peak at 0.5, while AUROC is 0.14 percentage points higher at 0.6. These results indicate that a moderate masking level provides the most effective aggregate trade-off for the two tasks. We therefore retain 0.5 as the default mask ratio, while the metric-specific results show that the preferred masking strength can vary slightly across datasets and evaluation criteria.

F Scaling Analysis

F.1 Model Size Scaling

We vary the Transformer depth over 3, 6, 9, 12, and 15 layers, which correspond to approximately 1.9M, 3.8M, 5.6M, 7.5M, and 9.3M trainable parameters, respectively, excluding the reconstruction head. Each pretrained model is evaluated on ISRUC and TUAB, selected to represent multi-class and binary downstream classification. Figure 7 reports the mean and sample standard deviation over five fine-tuning runs.

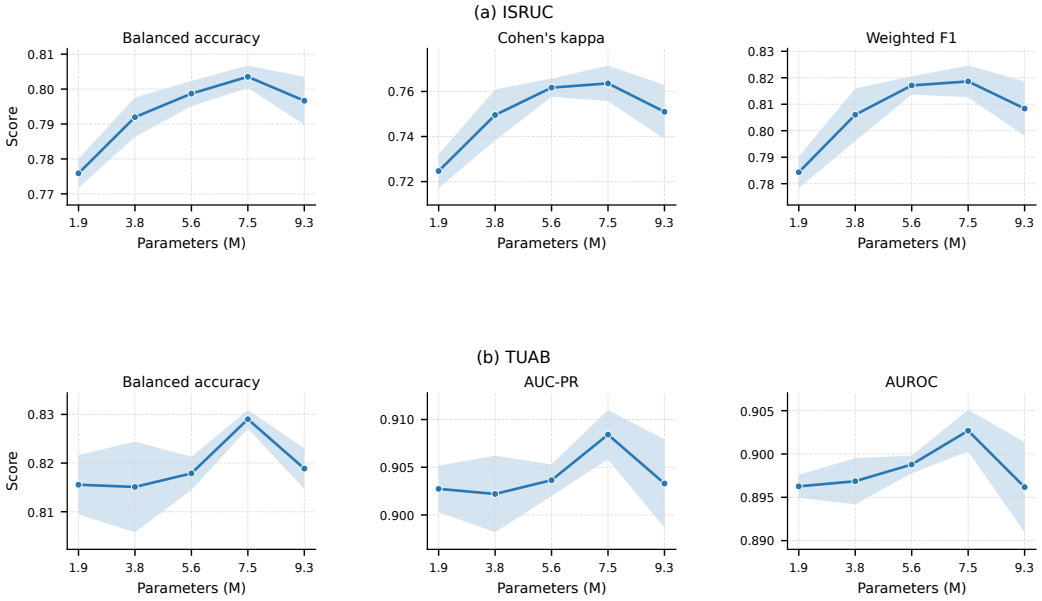


Figure 7: Downstream performance on ISRUC and TUAB as a function of model size. Each point shows the mean over five fine-tuning runs, and the shaded region shows one sample standard deviation. Parameter counts exclude the reconstruction head.

The mean ACC-B across ISRUC and TUAB increases from 79.57% with 1.9M parameters to 81.63% with 7.5M parameters, an improvement of 2.06 percentage points. Increasing the model further to 9.3M parameters reduces the mean to 80.78%. All three metrics on both ISRUC and TUAB reach their highest values at 7.5M parameters. Thus, performance generally improves with model capacity up to 12 layers, although the trends are not strictly monotonic across all smaller configurations. Increasing the depth from 12 to 15 layers does not provide further improvement; instead, the mean ACC-B decreases from 81.63% to 80.78%. We therefore use the 12-layer, 7.46M-parameter model as the default setting.

F.2 Pretraining Data Scaling

We next fix the proposed 12-layer architecture and use 10%, 25%, 50%, 75%, and 100% of the 9,000-hour pretraining corpus, corresponding to 900, 2,250, 4,500, 6,750, and 9,000 hours. The downstream fine-tuning protocol is held constant across all settings.

Figure 8 shows a clearer and more consistent benefit from increasing pretraining data than from increasing model size. Mean balanced accuracy across ISRUC and TUAB rises successively from 78.06% at 900 hours to 79.88%, 80.56%, 81.01%, and 81.63% at 2,250, 4,500, 6,750, and 9,000 hours, respectively. All three ISRUC metrics improve as the pretraining corpus grows, although the increments become smaller toward the full-data setting. On TUAB, AUC-PR and AUROC also improve at every step. Balanced accuracy temporarily decreases from 82.43% at 4,500 hours to 82.05% at 6,750 hours, but reaches its highest value of 82.90% with 9,000 hours. Thus, the full corpus provides the strongest transfer performance on both datasets, while the small TUAB fluctuation indicates that individual metrics need not improve monotonically at every intermediate data volume.

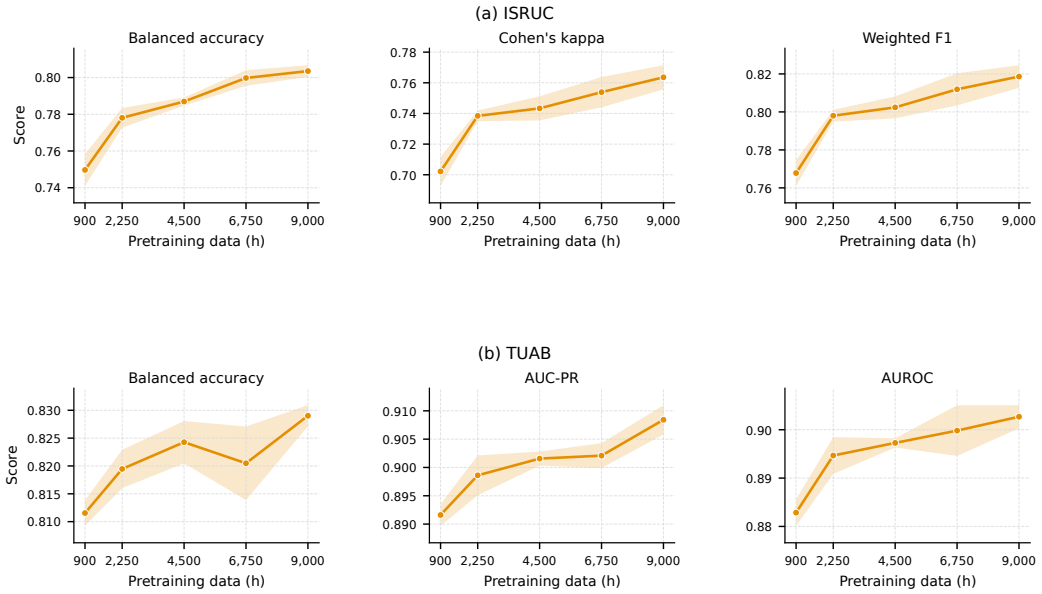


Figure 8: Effect of pretraining data volume on ISRUC and TUAB. Each point shows the mean over five fine-tuning runs, and the shaded region shows one sample standard deviation.

G COMPUTATIONAL COST AND MODEL SCALE OF AXISBRAIN

Parameter counts and floating-point operation (FLOP) estimates provide hardware-independent proxies for model scale and arithmetic workload. Table 6 summarizes the reported parameter counts of recent EEG foundation models. Among the models listed there, CBraMod (Wang et al., 2025a) has the fewest reported parameters.

Published FLOP values are not directly comparable unless the model boundary, input shape, implementation, operation coverage, and MAC-to-FLOP convention are aligned. For example, Table 22 of the CBraMod paper reports 318.9M FLOPs for its criss-cross variant using a 16-channel, 10-second CHB-MIT input, whereas Table 19 of CodeBrain (Ma et al., 2026) reports 1.29G FLOPs for CBraMod. These values illustrate the sensitivity of FLOP estimates to the implementation and counting protocol. We therefore restrict our FLOP analysis to a controlled comparison between AxisBrain and its CBraMod base architecture.

Both variants are evaluated for a single encoder-side forward pass with batch size one and a 10-second, 19-channel EEG input sampled at 200 Hz, corresponding to 2,000 samples per channel. The counted model boundary extends from the preprocessed EEG input through the encoder output and includes the patch encoder and representation backbone. Downstream classification heads and pretraining reconstruction heads are excluded. One multiply–accumulate operation (MAC) is counted as two FLOPs.

The operation-count calculation includes convolutional and linear projections, the query–key and attention–value matrix multiplications, feed-forward layers, descriptor projections, and cross-axis dot products. Bias addition, normalization, nonlinear activations, softmax and sigmoid operations, pooling reductions, dropout, low-order element-wise operations, and data movement are not included. The real FFT used by the shared patch encoder is estimated separately rather than included in the headline MAC-based totals. Adding its leading-order estimate does not change either displayed result at two decimal places.

Table 18: Controlled encoder-side arithmetic cost for CBraMod and AxisBrain under the same implementation, input, and counting convention. One MAC is counted as two FLOPs. Values are rounded to two decimal places; pretraining reconstruction and downstream classification heads are excluded.

Model	CBraMod	AxisBrain
GFLOPs	1.88	2.46
Relative cost	1.00×	1.30×

The controlled calculation gives 0.94 GMACs, or 1.88 GFLOPs, for the CBraMod path and 1.23 GMACs, or 2.46 GFLOPs, for AxisBrain. Based on the unrounded estimates, the proposed operations add 0.29 GMACs, equivalent to 0.57 GFLOPs, and increase the counted arithmetic workload by 30.42%. This increase primarily arises from learned-query descriptor attention, the full-dimensional projection of X^{cc}, projection of the two axis descriptors, and token-wise cross-axis dot products at each encoder layer.

AxisBrain also maintains a relatively compact model scale. Its 7.46M-parameter encoder is smaller than CodeBrain and PATCHCODE-Large, which have 15.17M and 37M reported parameters, respectively. Although AxisBrain is larger than the 4.03M parameters reported for CBraMod, it achieves a higher literature-reported ten-dataset mean balanced accuracy while remaining substantially smaller than these larger models.

Overall, AxisBrain provides a favorable balance between downstream performance, model scale, and counted arithmetic workload. Under the controlled calculation, AxisBrain requires 30.42% more encoder-side FLOPs than CBraMod while keeping the encoder below 7.5M parameters and achieving competitive downstream performance.

H Qualitative Analysis of Channel–Time Consistency

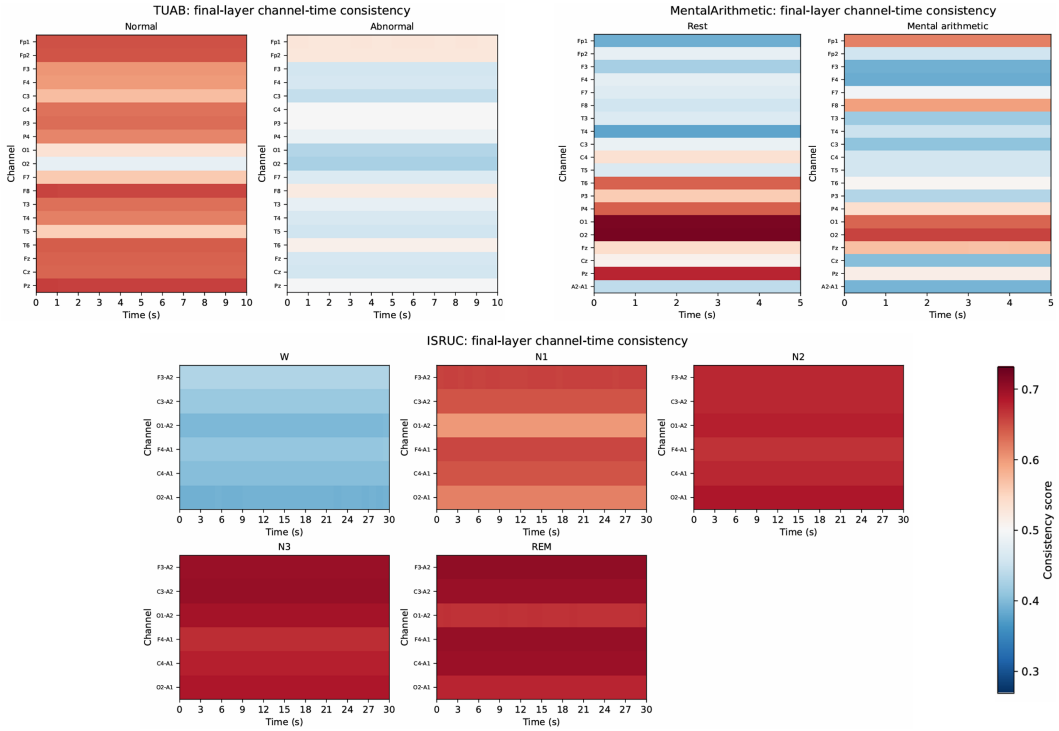


Figure 9: Class-averaged final-layer channel–time consistency scores. Rows denote EEG channels or derivations, and columns denote one-second temporal patches. (a) TUAB shows a broad difference between normal and the heterogeneous abnormal class. (b) MentalArithmetic shows a channel-selective redistribution between rest and mental arithmetic. (c) ISRUC shows low consistency during wake and broadly elevated consistency across the analyzed channels during sleep stages. Warmer colors indicate stronger agreement between the projected temporal and channel descriptors and therefore a larger residual multiplier.

To examine how the learned modulation varies across tasks, we visualize the class-averaged final-layer consistency scores for three datasets in Figure 9. Each cell corresponds to the agreement between the temporal descriptor at a given time position and the descriptor of a given channel after their projection into the shared latent space. The sigmoid-bounded score directly determines the residual multiplier $1 + S_{ct}$. Because the scores are averaged over test samples within each class, these maps emphasize systematic class-level patterns and may smooth sample-specific temporal variation. A high score indicates strong cross-axis descriptor agreement, not high EEG amplitude, attention weight, or feature importance.

TUAB shows a broad class-level difference across the montage. Normal recordings produce relatively high consistency scores over most channels, whereas abnormal recordings remain closer to or below the midpoint of the score range. Importantly, the abnormal label in TUAB aggregates heterogeneous pathological findings rather than representing a single controlled condition or pathology. Clinical reports associated with pathological recordings describe findings such as temporal or frontal slowing, excess theta activity, background disorganization, asymmetry, and epileptiform discharges (López et al., 2015; Schirrmeister et al., 2017). These abnormalities can differ in their spatial distribution, temporal persistence, and severity, and their interpretation may also depend on contextual factors such as age and vigilance state. Moreover, in our evaluation protocol, recording-level labels are inherited by individual 10-s segments, which need not contain the same abnormal pattern or an overt pathological event.

One possible interpretation is that this heterogeneity produces less stable or less mutually compatible cross-channel and temporal summaries across abnormal segments. Conversely, segments from recordings labeled normal may more often preserve broadly organized background activity, allowing

the two axis-specific descriptors to occupy more compatible directions in the learned comparison space and thereby yielding higher consistency scores. This interpretation concerns within-segment agreement between the two descriptors rather than implying that all normal recordings form a uniformly compact class in the overall latent space. Taken together, the observed pattern suggests that the greater morphological, spatial, and temporal heterogeneity of abnormal EEG may reduce cross-axis agreement, whereas the more consistently organized background activity of normal EEG may promote alignment between the two axis-specific descriptors.

MentalArithmetic instead exhibits a channel-selective redistribution rather than a uniform class-wise shift. The resting condition has high consistency over several posterior channels, most prominently O1, O2, and Pz, whereas mental arithmetic increases consistency at selected frontal and frontotemporal channels, including Fp1, F8, and Fz, while retaining high values at O1 and O2. Other channels remain comparatively low. This pattern is compatible with the distributed frontal and posterior involvement expected during mental calculation, although the consistency score should not be interpreted as a direct measure of frequency-specific EEG power.

ISRUC presents a second form of broad modulation. Wake epochs have low scores throughout the six analyzed derivations. Scores rise markedly in N1 and remain high and spatially widespread in N2, N3, and REM. Since ISRUC already uses a restricted set of sleep-relevant derivations, this result is better understood as a state-dependent shift across the available montage than as the discovery of a single isolated channel. The map supports a clear wake-to-sleep change, but it should not be described as a strictly monotonic increase with sleep depth because REM also exhibits high consistency.

Together, the three datasets illustrate two complementary behaviors of the proposed mechanism. ISRUC and TUAB are dominated by broad class-dependent shifts across channels, whereas MentalArithmetic exhibits a more selective redistribution among channels. Thus, the same learned scoring mechanism can express either global or localized cross-axis agreement without requiring predefined brain regions or electrode-coordinate priors. These visualizations are descriptive and are not intended as tests of statistical significance or as causal neurophysiological explanations.

I Limitations and Future Directions

AxisBrain remains weaker than CBraMod on the two motor-imagery benchmarks. As examined in Appendix D, replacing dense channel pooling with entmax does not improve performance, indicating that generic sparsification is insufficient. Future work should instead investigate EEG-specific, input-adaptive channel descriptors that emphasize spatially localized motor activity without discarding weak but complementary evidence from other channels.

Temporal mean pooling performed best among the descriptor variants tested here, but a single summary over the entire interval may miss task-relevant temporal variation. Future work could evaluate multi-scale or localized temporal aggregation and, where event boundaries can be identified, boundary-aware aggregation. Comparing these alternatives with temporal mean pooling under matched protocols would clarify whether more flexible descriptors improve transfer across tasks with different temporal structures.