B[FM]^2: Brain Foundation Model via Flow Matching with SplitUNet

Jaedong Hwang[1] **Kathleen Zhang**[1] **Wei Dai**[1] **Konstantinos Kontras**[1,2]
Maarten De Vos[2] **Ila Fiete**[1] **Paul Pu Liang**[1]
[1]Massachusetts Institute of Technology [2]KU Leuven

Abstract

EEG foundation models can learn generalizable representations from large-scale EEG corpora to enable single-backbone transfer across diverse clinical and brain-computer interface tasks. Existing models typically discretize the continuous multi-channel EEG waveform into patches or codebook tokens and train a transformer with masked self-supervision. Recognizing that this discretization fragments continuous brain rhythms and obscures fine-grained temporal dynamics, we present B[FM]^2 (Brain Foundation Model via Flow Matching), whose inductive bias aligns with the data by pretraining directly on the raw signal using continuous-time flow matching without patches, tokenization, or masking. However, multi-channel EEG signals pose an architectural challenge for flow matching: time is densely sampled and highly autocorrelated (thousands of timepoints), while the electrode axis is short (tens of channels) at distinct scalp positions. To address this time-electrode asymmetry, we introduce SplitUNet, a velocity network that factorizes each block into separate 1D temporal and 1D electrode convolutions and downsamples only along time, preserving electrode topology throughout the hierarchy. B[FM]^2 sets a new state of the art on 7 of 9 standard downstream EEG classification tasks, using a pretraining budget of only $36{,}895$ segments (≈ 307 h), 1–2 orders of magnitude ($\approx 30\times$) less than required by existing EEG foundation models. Further, it generates synthetic EEGs that two board-certified neurologists cannot distinguish from brain data (Cohen's $\kappa = -0.096$)[1].

1 Introduction

Brain foundation models (BFMs) can enable general-purpose EEG (electroencephalogram) analysis without task-specific supervision, successfully transferring a single pretrained backbone across diverse tasks like seizure detection and sleep staging. Driven by large open corpora [Obeid and Picone, 2016] and the adoption of vision and language architectures [He et al., 2022, Liu et al., 2023, Radford et al., 2019], the field has seen rapid progress. However, to leverage these architectures, existing BFMs [Jiang et al., 2024, Kostas et al., 2021, El Ouahidi et al., 2025, Yang et al., 2023a] have universally adopted a rigid training paradigm imported directly from computer vision: continuous neural waveforms are forcibly discretized into patches or learned codebooks, and a Transformer [Vaswani et al., 2017] is trained via discrete masked reconstruction or contrastive prediction.

However, for scalp EEG, this discretization introduces a fundamental representational mismatch. Brain rhythms are continuous, dynamic processes, yet existing approaches fragment them into arbitrary tokens with artificial boundaries. Consequently, the self-supervised objective is forced to model transitions between discrete chunks rather than capturing the underlying continuous temporal evolution of the neural signal [Rubanova et al., 2019, Zeng et al., 2023]. Furthermore, random

[1]`https://jd730.github.io/projects/BFM2`

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

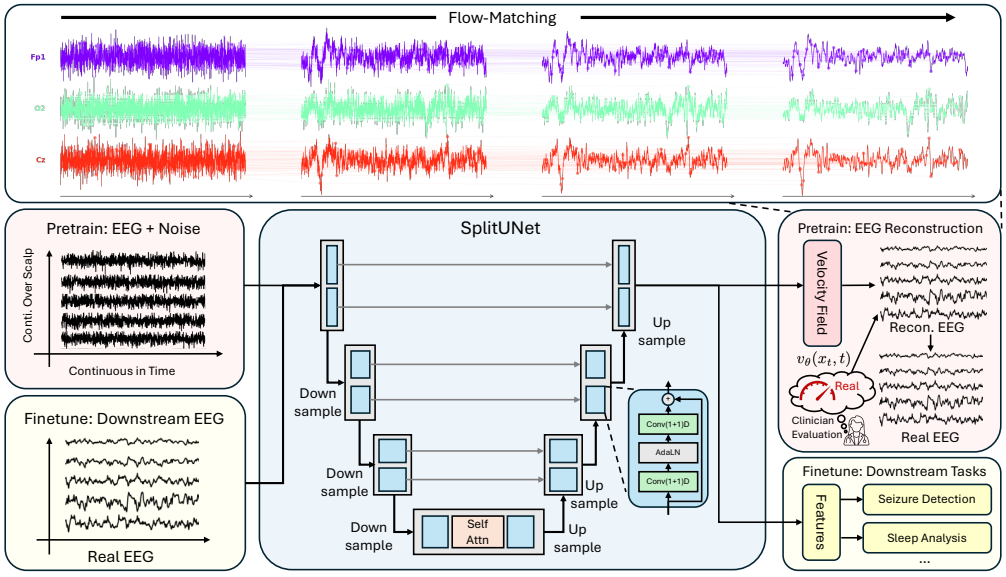


Figure 1: **Continuous-time generative pretraining for EEG. (Top)** The flow matching process maps Gaussian noise ($t = 0.0$) to continuous, multi-channel EEG signals ($t = 1.0$) along a continuous trajectory. **(Left/Center)** SplitUNet, the velocity network, reads the interpolant $\boldsymbol{x}_t$ together with t and predicts the velocity $v_\theta(\boldsymbol{x}_t, t)$ that points toward the data endpoint. Every spatiotemporal convolution inside it is factorized into a 1D convolution along time followed by a 1D convolution along electrodes, and down/upsampling acts on the time axis only, so the E electrodes survive at full resolution from input to output. **(Right)** Integrating the velocity field from noise generates new EEG, which two board-certified neurologists could not reliably separate from real recordings.

patch masking obscures the fine-grained temporal variations that are crucial for downstream clinical tasks [Dong et al., 2023].

To resolve this mismatch, we introduce B[FM]^2 (Brain Foundation Model via Flow Matching), which eliminates the discretization entirely. We directly pretrain on the continuous multi-channel waveform via flow matching [Lipman et al., 2023, Tong et al., 2024]. By corrupting the signal across all electrodes and timepoints simultaneously along a trajectory from Gaussian noise to data, the model learns a denoising vector field that inherently captures the continuous multi-scale dynamics of the brain. While continuous-time generative frameworks have yielded strong semantic representations in vision [Baranchuk et al., 2022, Tang et al., 2023, Xu et al., 2023], porting them to multi-channel EEG presents a distinct architectural challenge: dense temporal dynamics are observed only through the small, anatomically constrained set of scalp electrodes that the EEG montage permits.

To address this challenge, we propose *SplitUNet*, a velocity-network backbone designed for multi-channel EEG, where dense temporal dynamics are observed through a small, fixed set of electrodes. Each convolutional block factorizes spatiotemporal mixing into a 1D temporal convolution followed by a 1D electrode convolution; the encoder downsamples only along time, preserving the electrode dimension at every layer; and the network operates directly on the raw waveform—no patches, codebooks, or masking. With a continuous-time flow-matching objective, this backbone generates EEG by integrating a velocity field $v_\theta(\boldsymbol{x}_t, t)$ from Gaussian noise to the data manifold, producing a single velocity prediction over the full multi-channel signal at each step. Figure 1 depicts the pretraining-and-finetuning pipeline.

B[FM]^2 achieves state-of-the-art performance on 7 of 9 downstream tasks, outperforming prior models by up to 7.4 pp. Crucially, this is achieved with a pretraining set of only 36,895 segments ($\approx$ 307 h), one to two orders of magnitude smaller than corpora used by recent EEG foundation models. This establishes continuous-time flow matching as a markedly more sample-efficient pretraining framework than masked-token reconstruction. Furthermore, the resulting generative model produces highly realistic EEG that board-certified neurologists failed to distinguish from real TUEG data.

To summarize, the contributions of this paper are four-fold:

- We introduce a continuous-time flow matching paradigm [Lipman et al., 2023, Tong et al., 2024] for EEG foundation models, replacing the standard discrete masked-token framework. By operating directly on the raw multi-channel waveform, this generative pretext proves highly sample-efficient, requiring pretraining on only $36{,}895$ segments.
- We propose *SplitUNet*, an architecture tailored for continuous brain signals that factorizes temporal and electrode convolutions, preserving electrode topology end-to-end without relying on patches.
- We demonstrate that B[FM]^2 achieves state-of-the-art performance on 7 of 9 tasks across a diverse downstream suite.
- Simultaneously, B[FM]^2 can generate synthetic EEGs, which are demonstrated to be clinically indistinguishable from real TUEG data to board-certified neurologists.

2 B[FM]^2: Architecture and Pretraining

2.1 Preliminary: Flow Matching

We pretrain a velocity network v_θ on multi-channel EEG waveforms $\boldsymbol{x} \in \mathbb{R}^{E\times T}$, where E is the number of electrodes and T the number of timepoints. The training objective is the optimal-transport conditional flow-matching (OT-CFM) loss [Tong et al., 2024].

Conditional flow matching. Flow matching [Lipman et al., 2023] learns a velocity field $v_\theta(\boldsymbol{x}, t) : \mathbb{R}^{E\times T} \times [0,1] \to \mathbb{R}^{E\times T}$ whose induced flow transports an isotropic Gaussian prior q_0 into the data distribution q_1. Following the two-endpoint formulation of Tong et al. [2024], we specify a *coupling* π, defined as a joint distribution over $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ with noise marginal q_0 and data marginal q_1. We then define a conditional probability path $p_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1)$ interpolating between $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$, along with its conditional velocity field $u_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1)$. The conditional flow-matching loss,

$$\mathcal{L}_{\mathrm{CFM}}(\theta) = \mathbb{E}_{t\sim\mathcal{U}[0,1],\ (\boldsymbol{x}_0,\boldsymbol{x}_1)\sim\pi,\ \boldsymbol{x}\sim p_t(\cdot\mid\boldsymbol{x}_0,\boldsymbol{x}_1)} \left\| v_\theta(\boldsymbol{x}, t) - u_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1) \right\|^2 \tag{1}$$

shares the same gradient as the marginal flow-matching objective, meaning its minimizer is the true generative velocity. We take the straight interpolant $\boldsymbol{x}_t = (1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1$, for which $p_t(\cdot \mid \boldsymbol{x}_0, \boldsymbol{x}_1)$ is the Dirac delta at $\boldsymbol{x}_t$ and $u_t(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{x}_1) = \boldsymbol{x}_1 - \boldsymbol{x}_0$. Consequently, Eq. (1) reduces to regressing $v_\theta(\boldsymbol{x}_t, t)$ onto the displacement $\boldsymbol{x}_1 - \boldsymbol{x}_0$. The choice of coupling π is the only remaining free parameter. At inference, we sample by integrating $\dot{\boldsymbol{x}}_t = v_\theta(\boldsymbol{x}_t, t)$ from $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ at t=0 to t=1 using an off-the-shelf ordinary differential equation (ODE) solver.

Optimal transport coupling. Under the independent coupling $\pi = q_0 \otimes q_1$, straight-line interpolants between random noise-data pairs cross in $\mathbb{R}^{E\times T}$, and the marginal velocity at crossing points must average incompatible conditional velocities. This inflates gradient variance and curves the learned flow. OT-CFM [Tong et al., 2024] replaces π with the minibatch optimal-transport plan under squared-Euclidean cost, pairing each noise sample with its nearest data sample inside the batch. The resulting velocity field is straighter under the coupling, and few-step ODE sampling becomes accurate without distillation. We solve the assignment exactly with the Hungarian algorithm and use the resulting permuted pairs in Eq. (1).

2.2 SplitUNet: A Factorized Spatiotemporal Backbone for Multi-Channel Brain Signals

Generative pretraining has converged on two dominant backbones: Transformers [Ma et al., 2024, Peebles and Xie, 2023] and UNets [Ho et al., 2020, Karras et al., 2022]. However, Transformers [Vaswani et al., 2017] require tokenizing the signal, thereby reintroducing the very discretization our continuous-time objective is designed to avoid. A UNet-1D [Ronneberger et al., 2015] on the raw waveform removes that constraint, but, applied channel-wise, misses the inter-electrode correlations that carry most of the EEG's spatial information. A UNet-2D over the $(\text{electrode}, \text{time})$ tensor restores electrode coupling, and a rectangular kernel ($k_t \gg k_e$) can, in principle, match the per-axis scale—time is locally autocorrelated and benefits from a wide temporal receptive field, while the electrode axis is short (only 19 channels in the 10–20 montage).

We propose *SplitUNet*, a velocity-network backbone in which every $k_t \times k_e$ 2D convolution of a standard DDPM-style UNet [Ho et al., 2020] is replaced by the factorized (1+1)D operator

$$\mathrm{conv}_{(1+1)\mathrm{D}}(\boldsymbol{z}) \;=\; \mathrm{conv}_{\text{electrode}}\left(\sigma(\mathrm{conv}_{\text{time}}(\boldsymbol{z}))\right) \tag{2}$$

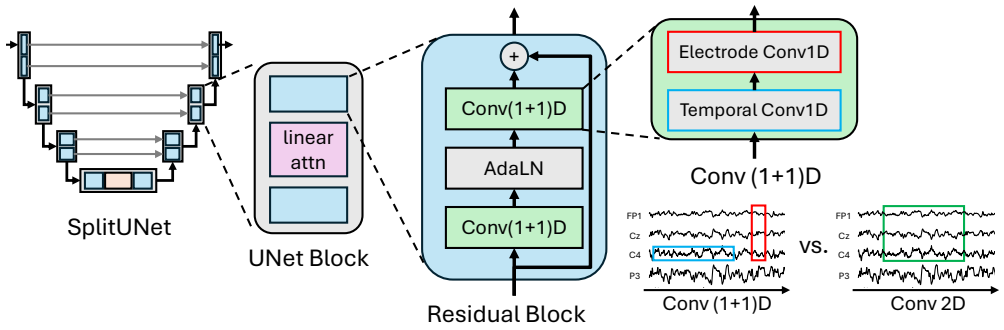


Figure 2: **Architecture of SplitUNet.** (left) The encoder halves only the time axis at each of its three downsampling steps (electrode dimension preserved); a self-attention bottleneck mixes globally; the decoder mirrors the encoder with skip connections and time-only upsampling. (center) Each UNet block contains two residual sub-blocks separated by a linear-attention layer [Shen et al., 2021]; each sub-block alternates Conv(1+1)D operators with AdaLN modulation conditioned on the flow time t. (right) Conv(1+1)D factorizes a 2D convolution into a temporal Conv1D (blue, kernel k_t) followed by an electrode Conv1D (red, kernel k_e, sliding over electrodes in the standard 10–20 order); on an EEG snippet, this contrasts with the joint $(\text{electrode}, \text{time})$ patch of a Conv2D (green, kernel $k_e \times k_t$).

where $\boldsymbol{z} \in \mathbb{R}^{C \times E \times T}$, the temporal convolution slides a kernel of size k_t along the time axis (weights shared across electrodes), the electrode convolution slides a kernel of size k_e along the electrode axis (weights shared across timepoints), and σ is a non-linear activation. This factorization, used in R(2+1)D for 3D video [Tran et al., 2018], has two advantages over the unfactorized 2D kernel: it interleaves a non-linearity between temporal and electrode mixing (doubling non-linear depth at no extra parameter cost), and it uses $k_t + k_e$ weights per filter rather than $k_t \cdot k_e$, redirecting the saved parameters into width or depth. Empirically, it outperforms UNet-2D on every task (see Table 5), consistent with EEG's temporal dynamics and inter-electrode coupling being largely separable; we do not claim a 2D kernel is universally inferior.

Beyond the factorization, SplitUNet reflects two properties of multi-channel EEG: time is densely sampled, and the electrode axis is short, with each channel anatomically meaningful—compressing it would discard distinct brain-region information. (i) *Time-only compression.* The encoder halves only the temporal resolution after each of its first three stages and preserves the electrode dimension at E throughout. (ii) *Attention placement.* Each resolution stage uses an efficient linear-attention block [Shen et al., 2021] for in-stage mixing, with a single full self-attention block at the bottleneck (where the time axis has been compressed by $8\times$) so that both axes are mixed once their dimensions are commensurate. In both block types the (E, T) feature map is flattened into a single sequence of length $E \cdot T$, so attention mixes over all electrode and time positions jointly and carries no axis-specific prior; the axis asymmetry is expressed by the convolutions and the downsampling schedule alone. The decoder mirrors the encoder with skip connections and time-only upsampling. Implementation details are in Appendix B, and the full architecture is illustrated in Figure 2.

2.3 Pretraining and Feature Extraction

We minimize the OT-CFM loss of Eq. (1) on the pretraining dataset (TUEG [Obeid and Picone, 2016]). Noise endpoints $\boldsymbol{x}_0$ are drawn from an isotropic Gaussian of matching shape, data endpoints $\boldsymbol{x}_1$ are sampled windows, and each minibatch is re-paired by an exact Hungarian algorithm before forming interpolants. Time t is sampled uniformly on $[0, 1]$.

For downstream transfer, we evaluate the velocity network on the input data $\boldsymbol{x}$ at $t = 0$. We extract features from the penultimate layer (immediately preceding the final output projection), yielding an activation tensor of shape $C \times E \times T$, where C is the feature channel width, E is the number of electrodes, and T is the number of timepoints. Applying global average pooling or flattening over both the electrode and time axes produces a fixed-size feature vector, which is then passed to a single linear classification head. The complete finetuning protocol is described in Section 3.1.

Table 1: Overview of downstream tasks and datasets.

Task	Dataset	# Channels	Duration	# Samples	Rate	# Classes
Mental Disorder	Mumtaz	19	5 s	7,143	256 Hz	2
Mental Stress	MAT	20	5 s	1,707	500 Hz	2
Seizure Detection	Siena	29	10 s	51,307	512 Hz	2
Sleep Staging	ISRUC	6	30 s	89,240	200 Hz	5
	HMC	4	30 s	137,243	256 Hz	5
Event Type	TUEV	16	5 s	112,491	256 Hz	6
Abnormal Detection	TUAB	16	10 s	409,455	256 Hz	2
Emotion Recognition	SEED-V	62	1 s	117,744	1000 Hz	5
Motor Imagery	BCIC-IV-2a	22	4 s	5,184	250 Hz	4

3 Experiment

B[FM]^2 replaces the masked-token recipe of existing EEG foundation models with continuous-time flow matching on the raw multi-channel waveform; we test this swap with two questions: **(Q1) Downstream task performance:** Does the same pretrained backbone, with no task-specific architectural changes, transfer competitively to standard EEG classification benchmarks under matched finetuning, despite using no patches, no codebooks, and no masking (see Section 3.2)? **(Q2) Clinical indistinguishability:** Does the same model also produce clinically realistic EEG, *i.e.*, samples indistinguishable from real held-out TUEG to two board-certified neurologists (see Section 3.3)? Implementation details and hyperparameters are in Appendix B.

3.1 Experimental Setup

Pretraining corpus and preprocessing. We pretrain on the Temple University Hospital EEG Corpus [Obeid and Picone, 2016] (TUEG): 69,652 recordings (27,062 h) from 14,987 subjects across 26,846 sessions. We follow established preprocessing pipelines [Jiang et al., 2024, El Ouahidi et al., 2025, Wang et al., 2025] (band-pass and notch filtering, segmentation, amplitude thresholding, normalization; full settings in Appendix B) on the standard 19-channel 10–20 montage [Klem et al., 1999] and its channel order, yielding 1,109,545 non-overlapping 30 s segments ($\approx$ 9,000 h).

From this pool we randomly sample a pretraining subset of 36,895 segments ($\approx$ 307 h)—roughly 3.3% of the preprocessed pool and 1% of the TUEG hours—which is 1–2 orders of magnitude smaller than the corpora used by recent baselines (LaBraM: 2,500 h [Jiang et al., 2024]; CBraMod: 9,000 h [Wang et al., 2025]; REVE: 60,000 h (25,000 subjects across multiple corpora) [El Ouahidi et al., 2025]).

Pretraining. The velocity network is trained by minimizing the OT-CFM loss (see Eq. (1)) with an MSE regression target. Each minibatch is re-paired by an exact optimal-transport assignment using the Hungarian algorithm before forming interpolants; t is sampled uniformly on $[0, 1]$. We train for 70 epochs with batch size 4 and maintain an exponential moving average of the weights for sampling and downstream feature extraction. Optimizer settings are in Appendix B.

Downstream tasks. We evaluate on nine classification tasks spanning anomaly, seizure, and sleep-stage detection, motor imagery, and emotion/stress recognition (see Table 1). We strictly follow published preprocessing pipelines, class-balance policies, and train/val/test splits: Zhou et al. [2025] for Siena, HMC, and SEED-V; prior work [Wang et al., 2025, Yang et al., 2023a, Jiang et al., 2024, El Ouahidi et al., 2025] for the remainder. Dataset specifics are in Appendix C.

Baselines. We compare B[FM]^2 against two groups: task-specific supervised architectures trained from scratch on each dataset, and pretrained EEG foundation models adapted by full-network finetuning. The supervised group comprises EEGNet [Lawhern et al., 2018], EEGConformer [Song et al., 2023], SPaRCNet [Jing et al., 2023], ContraWR [Yang et al., 2023b], a CNN–Transformer hybrid [Peh et al., 2022], FFCL [Li et al., 2022], and ST-Transformer [Song et al., 2021]. The foundation-model group comprises BIOT [Yang et al., 2023a], LaBraM-Base [Jiang et al., 2024], CBraMod [Wang et al., 2025], REVE [El Ouahidi et al., 2025], and CSBrain [Zhou et al., 2025]. All baseline numbers are taken directly from El Ouahidi et al. [2025] and Zhou et al. [2025]; both

Table 2: Balanced accuracy ($\pm$ std) across nine downstream classification tasks. B[FM]^2 reaches the highest suite-averaged balanced accuracy (0.754), sets a new state of the art on 7 of 9 tasks, is within 1.5 pp on TUAB, and achieves second-best performance on BCIC-IV-2a (motor imagery on healthy subjects). Bold and underline denote the best and the second best performance in each task.

Methods	Mumtaz	MAT	Siena	ISRUC	HMC
EEGNet	0.923 ± 0.010	0.677 ± 0.012	0.749 ± 0.052	0.715 ± 0.012	0.653 ± 0.012
EEGConformer	0.931 ± 0.012	0.680 ± 0.012	0.756 ± 0.021	0.740 ± 0.013	0.715 ± 0.009
SPaRCNet	0.932 ± 0.009	0.688 ± 0.011	0.657 ± 0.038	0.749 ± 0.007	0.476 ± 0.111
ContraWR	0.919 ± 0.011	0.663 ± 0.010	0.655 ± 0.031	0.740 ± 0.013	0.424 ± 0.054
CNN-Transformer	0.930 ± 0.007	0.678 ± 0.027	0.698 ± 0.056	0.736 ± 0.009	0.657 ± 0.014
FFCL	0.931 ± 0.004	0.680 ± 0.014	0.662 ± 0.039	0.728 ± 0.018	0.443 ± 0.070
ST-Transformer	0.913 ± 0.010	0.663 ± 0.017	0.753 ± 0.038	0.738 ± 0.021	0.256 ± 0.014
BIOT	0.936 ± 0.005	0.688 ± 0.019	0.735 ± 0.067	0.753 ± 0.012	0.686 ± 0.004
LaBraM-Base	0.941 ± 0.008	0.691 ± 0.013	0.708 ± 0.033	0.763 ± 0.010	0.728 ± 0.010
CBraMod	0.956 ± 0.006	0.726 ± 0.013	0.732 ± 0.065	0.786 ± 0.011	0.727 ± 0.004
REVE	$\underline{0.964 \pm 0.010}$	$\underline{0.766 \pm 0.035}$	0.740 ± 0.007	0.782 ± 0.008	$\underline{0.740 \pm 0.007}$
CSBrain	0.964 ± 0.015	0.756 ± 0.011	$\underline{0.766 \pm 0.047}$	$\underline{0.792 \pm 0.003}$	0.735 ± 0.005
B[FM]^2 (ours)	$\mathbf{1.000 \pm 0.000}$	$\mathbf{0.840 \pm 0.022}$	$\mathbf{0.776 \pm 0.010}$	$\mathbf{0.806 \pm 0.001}$	$\mathbf{0.764 \pm 0.002}$

	TUEV	TUAB	BCIC-IV-2a	SEED-V	Average
EEGNet	0.388 ± 0.014	0.764 ± 0.004	0.448 ± 0.009	0.296 ± 0.010	0.624 ± 0.020
EEGConformer	0.407 ± 0.016	0.776 ± 0.005	0.470 ± 0.011	0.354 ± 0.011	0.648 ± 0.013
SPaRCNet	0.416 ± 0.026	0.790 ± 0.002	0.464 ± 0.012	0.295 ± 0.008	0.607 ± 0.041
ContraWR	0.438 ± 0.035	0.775 ± 0.004	0.468 ± 0.013	0.355 ± 0.011	0.604 ± 0.025
CNN-Transformer	0.409 ± 0.016	0.778 ± 0.002	0.460 ± 0.011	0.368 ± 0.008	0.635 ± 0.023
FFCL	0.398 ± 0.010	0.785 ± 0.004	0.447 ± 0.014	0.364 ± 0.009	0.604 ± 0.029
ST-Transformer	0.398 ± 0.023	0.797 ± 0.002	0.458 ± 0.015	0.305 ± 0.007	0.587 ± 0.019
BIOT	0.528 ± 0.022	0.796 ± 0.006	0.475 ± 0.009	0.384 ± 0.019	0.664 ± 0.026
LaBraM-Base	0.641 ± 0.006	0.814 ± 0.002	0.487 ± 0.009	0.398 ± 0.014	0.686 ± 0.014
CBraMod	0.667 ± 0.011	$\underline{0.829 \pm 0.002}$	0.514 ± 0.007	0.409 ± 0.010	0.705 ± 0.023
REVE	0.676 ± 0.023	$\mathbf{0.832 \pm 0.001}$	$\mathbf{0.640 \pm 0.009}$	0.405 ± 0.002	$\underline{0.727 \pm 0.015}$
CSBrain	$\underline{0.690 \pm 0.006}$	0.817 ± 0.004	0.566 ± 0.007	$\underline{0.420 \pm 0.003}$	0.723 ± 0.017
B[FM]^2 (ours)	$\mathbf{0.715 \pm 0.013}$	0.819 ± 0.004	$\underline{0.570 \pm 0.008}$	$\mathbf{0.492 \pm 0.009}$	$\mathbf{0.754 \pm 0.010}$

papers score every baseline under finetuning on the same train/val/test splits we use, so the numbers in Table 2 are directly comparable to ours.

Finetuning protocol. We finetune the pretrained B[FM]^2 end-to-end on each downstream task with cross-entropy. The backbone is conditioned on a fixed timestep $t{=}0$; its output features are passed through a single linear head. The UNet-1D (joint) variant has an electrode-count-dependent first convolution, which we reinitialize per dataset before finetuning; the other backbones are electrode-count-invariant. All downstream results are averaged over the five highest-scoring seeds, with each run reported at its best epoch on the test split. Please refer to Appendix B for more details.

3.2 Downstream Classification

We address **Q1** (downstream transfer) by finetuning the pretrained B[FM]^2 backbone end-to-end on each of the nine tasks under the protocol of Section 3.1. Table 2 reports balanced accuracy per task; per-task breakdowns with task-appropriate companion metrics (Cohen's κ, Weighted F1, AUC-PR, AUROC) are provided in Appendix H. B[FM]^2 achieves a new state of the art on 7 of the 9 tasks (Mumtaz, MAT, Siena, ISRUC, HMC, TUEV, and SEED-V), and trails the leading model by less than 1.5 pp on TUAB. However, a performance gap remains on the motor imagery task, BCIC-IV-2a, where B[FM]^2 achieves the second-best performance but lags behind REVE (0.570 vs. 0.640). This gap is largely attributable to differences in the pretraining corpora. REVE explicitly leverages four motor imagery datasets [Cho et al., 2017, Ofner et al., 2017, Liu et al., 2024, Zhou et al., 2016] during pretraining, making its downstream evaluation on this task effectively in-domain. In contrast, B[FM]^2 is pretrained exclusively on TUEG—a clinical corpus of resting-state and pathological EEG. Thus,

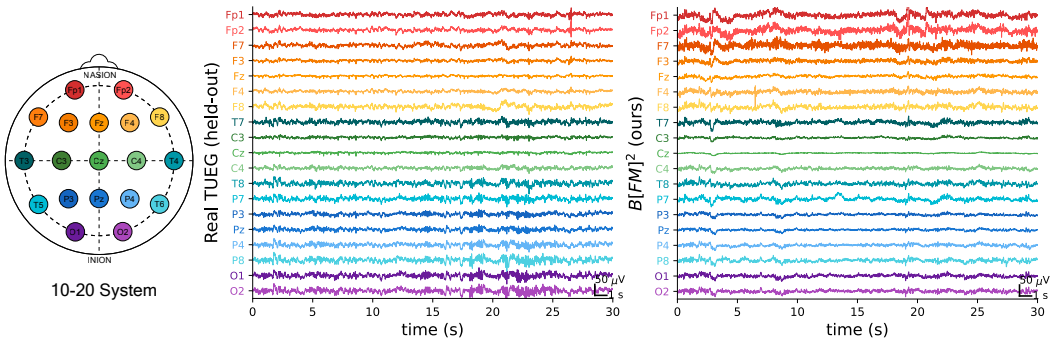


Figure 3: **Real vs. B[FM]^2 EEG segments.** A 30-second held-out TUEG segment (middle) and an unconditional B[FM]^2 sample (right), displayed in the 19-channel referential 10–20 montage (left; electrode layout from Ferrell et al. [2020]). Both panels exhibit both the spatial and temporal coherence characteristic of clinical EEG. For example, notice how adjacent electrodes in the same brain region (e.g., the frontal red traces or the occipital purple traces) share rhythmically aligned activity and correlated amplitude envelopes, rather than exhibiting independent, per-channel noise. The two sources are visually difficult to distinguish at a glance, consistent with the blinded neurologist evaluation in Table 3. Note that the right-hand traces are an *independent* unconditional sample drawn from Gaussian noise, not a reconstruction of a real segment. Calibration: 1 s, 50 μV.

Table 3: Blinded neurologist rating of B[FM]^2 samples. Two board-certified neurologists evaluated 50 randomly interleaved 30-second windows each (25 real held-out TUEG and 25 B[FM]^2 samples). Realness was scored on a 5-point Likert scale (1 = definitely real, 5 = definitely fake). Readers failed to distinguish synthetic samples from real segments, rating B[FM]^2 outputs as slightly more realistic on average.

Reader	Mean realness (↓ for real, ↑ for fake)		Mann–Whitney U		Agreement	
	Real Data	B[FM]^2	U	p	ρ	Cohen's κ
R1	2.96 ± 1.02	2.60 ± 1.04	251.5	0.204	0.052	-0.096
R2	3.08 ± 1.12	2.32 ± 0.80	188.5	0.011		
Pooled	3.02 ± 1.06	2.46 ± 0.93	876.0	0.007	-	-

while REVE enjoys target-domain familiarity, B[FM]^2 still achieves highly competitive cross-domain transfer, successfully adapting to motor imagery signals during finetuning despite never encountering them during pretraining.

3.3 Clinical Indistinguishability of Generated Samples

To address **Q2** (clinical indistinguishability), we generate unconditional samples using the same pretrained B[FM]^2 from Section 3.2. As shown in the example traces in Figures 3 and 5), the generated signals are visually compelling. In a blinded reading, two board-certified neurologists could not reliably distinguish them from real, held-out TUEG segments (see Table 3). Section 3.4 additionally showcases physiologically diverse patterns—sharp-wave transients, slow-wave activity, and posterior α rhythm—identified in generated samples (see Figure 4). The two neurologists each evaluated 50 randomly interleaved 30-second segments (25 real held-out TUEG and 25 unconditional B[FM]^2 samples). Segments were presented in the longitudinal bipolar 10–20 montage with the standard clinical display settings under which they habitually read EEG. The neurologists scored each segment on a 5-point Likert scale of perceived realness (1 = definitely real, 3 = unsure, 5 = definitely fake).

Pooled across readers, the mean realness score was 3.02 ± 1.06 for real segments versus 2.46 ± 0.93 for B[FM]^2 samples (Mann–Whitney $U = 876.0$, $p = 0.007$). This statistically significant gap points in an unexpected direction: B[FM]^2 samples were rated slightly *more* real-looking than the actual held-out EEG segments. Furthermore, at the segment level, the raters failed to agree on the origin of the samples (Spearman $\rho = 0.052$, Cohen's $\kappa = -0.096$). Neither reader could reliably classify

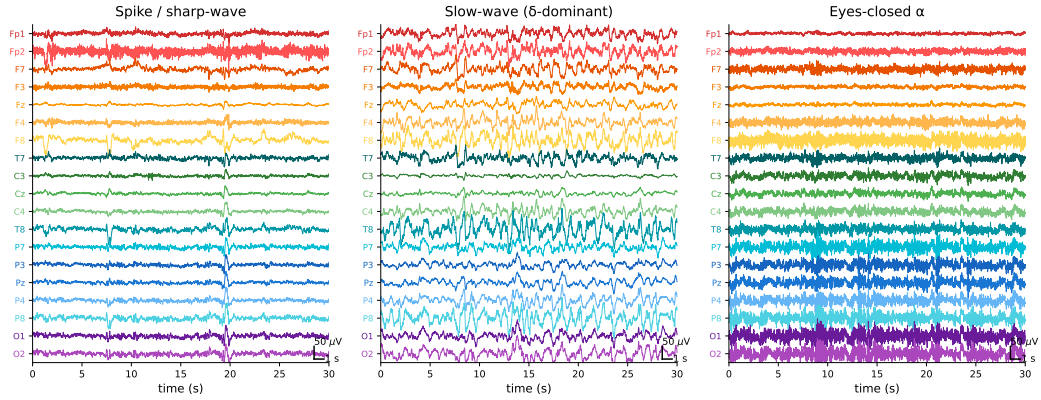


Figure 4: **B[FM]^2 generates physiologically diverse EEG patterns.** Three B[FM]^2 samples illustrating distinct brain-state patterns, curated from the unconditional generation pool by simple heuristics and displayed in the canonical 10–20 montage. (left) a sharp-wave / spike-like transient at $\sim$20 s. (center) δ-dominant slow-wave activity across all electrodes, characteristic of slow-wave sleep. (right) eyes-closed posterior α rhythm, with stronger 8–13 Hz power over O1/O2. Calibration bar: 1 s and 50 μV.

an individual segment as real or synthetic, indicating that the perceptual cues they relied upon were largely uncorrelated. The full reading protocol and clinical interface are detailed in Appendix D.

3.4 Recognizable Brain-State Patterns from Unconditional Generation

B[FM]^2 captures recognizable brain-state structure from a fully unconditional objective. Mining its generation pool with simple band-power heuristics (see Appendix E), we recover three canonical EEG patterns: a sharp-wave transient, δ-dominant slow-wave activity, and posterior α rhythm (see Figure 4). Each appears alongside spatially coherent activity across adjacent electrodes (see Figure 3), as expected of clinical recordings. This kind of qualitative inspection is itself uncommon in the EEG foundation-model literature: tokenized masked-token backbones [Jiang et al., 2024, Wang et al., 2025, El Ouahidi et al., 2025, Yang et al., 2023a] are not designed to generate novel waveforms in the input space and so cannot be probed for what they have learned about brain dynamics. That B[FM]^2 reveals interpretable structure without any class or pattern labels also points to a natural extension—class- and subject-conditional synthesis to augment scarce, long-tailed clinical EEG corpora (see Section 4).

3.5 Ablation Studies

We conduct ablation studies on the network architecture (see Tables 4 and 5), the pretraining objective (see Table 6), and pretraining itself (see Table 7) to assess the contribution of each component.

Network architecture. We replace SplitUNet with three backbones that differ in how they couple electrodes. **UNet-1D (indep.)** applies every temporal convolution to each electrode separately, so no information crosses electrodes anywhere in the network. **UNet-1D (joint)** treats the E electrodes as the input channels of the UNet. Its first convolution is reinitialized at finetuning to match each dataset's electrode count. **UNet-2D** convolves jointly over (electrode, time) at every layer. For generation, we draw $5{,}000$ unconditional samples from each backbone and measure the Wasserstein-1 (W_1) distance between their log-band-power distributions and those of held-out TUEG segments (see Table 4). Electrode coupling is necessary for generation, as UNet-1D (indep.) samples lie at W_1=1.903 against a real-vs-real floor of 0.156. SplitUNet attains the lowest distance at 0.195, compared with 0.248 for UNet-2D and 0.256 for UNet-1D (joint).

In downstream transfer (see Table 5), coupling electrodes without factorization raises the average balanced accuracy by at most 0.012 over UNet-1D (indep.), from 0.674 to 0.682 for UNet-1D (joint) and 0.686 for UNet-2D. UNet-1D (indep.) even exceeds both coupled backbones on TUEV and SEED-V. SplitUNet raises the average to 0.754 and leads all three backbones on every task except

Table 4: Sample quality of B[FM]^2 with four velocity-network backbones against held-out real TUEG segments (n=5,000, mean $\pm$ std). UNet-1D (indep.) processes each electrode as a separate time series, so no information passes between electrodes. UNet-1D (joint) takes all electrodes as the input channels of a temporal UNet, so electrodes are combined only in its first convolution. UNet-2D applies 2D convolutions over electrodes and time at every layer, and SplitUNet replaces each 2D convolution with a temporal convolution followed by an electrode convolution. Real-vs-Real compares two disjoint sets of real segments and gives the distance expected from sampling alone. Bold marks the best generator per column, excluding Real-vs-Real. PSNR and SSIM are computed without correspondence between generated and real segments and are reported for completeness.

Backbone	Band-power $W_1 \downarrow$	PSNR $\uparrow$	SSIM $\uparrow$
Real-vs-Real	0.156 ± 0.028	16.32 ± 0.26	0.021 ± 0.003
UNet-1D (indep.)	1.903 ± 0.098	6.74 ± 0.07	0.001 ± 0.000
UNet-1D (joint)	0.256 ± 0.002	16.88 ± 0.03	0.025 ± 0.001
UNet-2D	0.248 ± 0.001	17.14 ± 0.01	0.024 ± 0.001
SplitUNet (ours)	$\mathbf{0.195 \pm 0.002}$	$\mathbf{17.16 \pm 0.02}$	$\mathbf{0.026 \pm 0.001}$

Table 5: Balanced accuracy ($\pm$ std) of B[FM]^2 with four velocity-network backbones, each pretrained with OT-CFM on the same 307 h set and finetuned end-to-end on the nine downstream tasks. UNet-1D (indep.) processes each electrode as a separate time series, so no information passes between electrodes. UNet-1D (joint) takes all electrodes as the input channels of a temporal UNet. Bold denotes the best performance.

Dataset	UNet-1D (indep.)	UNet-1D (joint)	UNet-2D	SplitUNet (ours)
Mumtaz	0.896 ± 0.003	0.895 ± 0.006	0.909 ± 0.004	$\mathbf{1.000 \pm 0.000}$
MAT	0.528 ± 0.021	0.708 ± 0.036	0.715 ± 0.067	$\mathbf{0.840 \pm 0.022}$
Siena	0.710 ± 0.038	0.745 ± 0.016	0.723 ± 0.022	$\mathbf{0.776 \pm 0.010}$
ISRUC	0.789 ± 0.004	0.801 ± 0.001	0.800 ± 0.006	$\mathbf{0.806 \pm 0.001}$
HMC	0.727 ± 0.007	0.745 ± 0.005	0.749 ± 0.005	$\mathbf{0.764 \pm 0.002}$
TUEV	0.675 ± 0.023	0.559 ± 0.021	0.519 ± 0.016	$\mathbf{0.715 \pm 0.013}$
TUAB	0.800 ± 0.004	$\mathbf{0.826 \pm 0.004}$	0.817 ± 0.004	0.819 ± 0.004
BCIC-IV-2a	0.458 ± 0.029	0.420 ± 0.013	0.468 ± 0.007	$\mathbf{0.570 \pm 0.008}$
SEED-V	0.485 ± 0.009	0.438 ± 0.013	0.470 ± 0.011	$\mathbf{0.492 \pm 0.009}$
Average	0.674 ± 0.020	0.682 ± 0.016	0.686 ± 0.025	$\mathbf{0.754 \pm 0.010}$

TUAB, where UNet-1D (joint) is 0.007 higher. The largest margins over the best alternative appear on MAT ($+0.125$), BCIC-IV-2a ($+0.102$), and Mumtaz ($+0.091$).

Pretraining objective. We isolate the contribution of the flow-matching objective by replacing OT-CFM pretraining with masked reconstruction of either individual samples (*pointwise*) or contiguous 1 s blocks (*patchwise*), keeping SplitUNet and the pretraining data fixed (see Table 6). The pointwise variant masks 75% of the (E, T) samples uniformly at random and regresses the masked entries with an MSE loss on the waveform. This applies the MAE formulation and masking ratio [He et al., 2022] at the sample level rather than the patch level, so no tokenization is introduced. The patchwise variant instead masks 50% of each recording, following the masking ratio of CBraMod [Wang et al., 2025], in contiguous blocks of 200 samples (1 s at 200 Hz) spanning all electrodes. Masking entire blocks prevents the model from reconstructing a masked sample by interpolating from its immediate temporal neighbors. Each pretrained backbone is finetuned with the protocol of Section 3.1. OT-CFM raises the average balanced accuracy from 0.710 (pointwise) and 0.700 (patchwise) to 0.754 and leads on every task except TUAB, where patchwise masking is 0.004 higher.

Pretraining scale and initialization. Because the baselines are pretrained on 9,000–60,000 h, comparisons with them conflate the pretraining method with data volume. To separate the two, we pretrain CBraMod [Wang et al., 2025] on our 307 h pretraining set (36,895 segments) with its official codebase. At this scale, CBraMod's masked pretraining barely improves on training from scratch, raising the average balanced accuracy from 0.672 to 0.675 (see Table 7). Pretraining B[FM]^2 on the same data raises its average from 0.635 to 0.754, even though B[FM]^2 trails CBraMod when both

Table 6: Balanced accuracy (± std) of SplitUNet pretrained with pointwise masked reconstruction, patchwise (1 s) masked reconstruction, or OT-CFM on the same pretraining data. Bold denotes the best performance.

Dataset	Masked (pointwise)	Masked (patchwise)	OT-CFM (ours)
Mumtaz	0.878 ± 0.009	0.797 ± 0.033	$\mathbf{1.000 \pm 0.000}$
MAT	0.704 ± 0.096	0.732 ± 0.085	$\mathbf{0.840 \pm 0.022}$
Siena	0.764 ± 0.021	0.750 ± 0.023	$\mathbf{0.776 \pm 0.010}$
ISRUC	0.795 ± 0.004	0.794 ± 0.006	$\mathbf{0.806 \pm 0.001}$
HMC	0.755 ± 0.006	0.755 ± 0.007	$\mathbf{0.764 \pm 0.002}$
TUEV	0.669 ± 0.025	0.637 ± 0.030	$\mathbf{0.715 \pm 0.013}$
TUAB	0.822 ± 0.006	$\mathbf{0.823 \pm 0.005}$	0.819 ± 0.004
BCIC-IV-2a	0.522 ± 0.030	0.557 ± 0.016	$\mathbf{0.570 \pm 0.008}$
SEED-V	0.484 ± 0.014	0.457 ± 0.034	$\mathbf{0.492 \pm 0.009}$
Average	0.710 ± 0.036	0.700 ± 0.035	$\mathbf{0.754 \pm 0.010}$

Table 7: Balanced accuracy (± std) of CBraMod and B[FM]^2 trained from scratch on each downstream dataset or pretrained on our 307 h set. The CBraMod 9,000 h column is from Wang et al. [2025]. Bold denotes the best performance.

	CBraMod			B[FM]^2	
Dataset	Scratch	307 h	9,000 h	Scratch	307 h (ours)
Mumtaz	0.895 ± 0.009	0.909 ± 0.013	0.956 ± 0.006	0.863 ± 0.042	$\mathbf{1.000 \pm 0.000}$
MAT	0.623 ± 0.045	0.623 ± 0.046	0.726 ± 0.013	0.669 ± 0.059	$\mathbf{0.840 \pm 0.022}$
Siena	0.814 ± 0.014	$\mathbf{0.816 \pm 0.021}$	0.732 ± 0.065	0.595 ± 0.140	0.776 ± 0.010
ISRUC	0.776 ± 0.006	0.775 ± 0.006	0.786 ± 0.011	0.762 ± 0.063	$\mathbf{0.806 \pm 0.001}$
HMC	0.720 ± 0.009	0.720 ± 0.009	0.727 ± 0.004	0.736 ± 0.012	$\mathbf{0.764 \pm 0.002}$
TUEV	0.678 ± 0.017	0.648 ± 0.011	0.667 ± 0.011	0.593 ± 0.030	$\mathbf{0.715 \pm 0.013}$
TUAB	0.801 ± 0.011	0.790 ± 0.011	$\mathbf{0.829 \pm 0.002}$	0.824 ± 0.005	0.819 ± 0.004
BCIC-IV-2a	0.403 ± 0.020	0.386 ± 0.029	0.514 ± 0.007	0.253 ± 0.003	$\mathbf{0.570 \pm 0.008}$
SEED-V	0.334 ± 0.013	0.404 ± 0.005	0.409 ± 0.010	0.419 ± 0.023	$\mathbf{0.492 \pm 0.009}$
Average	0.672 ± 0.019	0.675 ± 0.021	0.705 ± 0.023	0.635 ± 0.058	$\mathbf{0.754 \pm 0.010}$

are trained from scratch. B[FM]^2 thus exceeds the matched-scale CBraMod by 0.079 on average and outperforms it on every task except Siena. It also surpasses the published CBraMod pretrained on $9{,}000$ h (0.705) with about one-thirtieth of the pretraining data.

4 Conclusion

We propose B[FM]^2, a brain foundation model that pretrains directly on the continuous multi-channel EEG waveform via flow matching. However, pretraining on EEG data presents a challenge of integrating two distinctive dimensions: a high-resolution temporal dimension and a fixed, low-resolution electrode dimension. We introduced SplitUNet, a velocity-network backbone that factorizes each 2D spatiotemporal convolution into a 1D temporal and a 1D electrode convolution and preserves the electrode dimension at every layer. B[FM]^2 establishes a new state of the art on 7 of 9 standard EEG downstream tasks, outperforming the standard practice of discretizing EEG into patches [El Ouahidi et al., 2025, Zhou et al., 2025] or tokens [Jiang et al., 2024]. Crucially, we achieve this using only $36{,}895$ segments—roughly $3.3\,\%$ of the pretraining pool—while unconditionally generating samples that two board-certified neurologists cannot reliably distinguish from real clinical data. Our paper has a limitation that B[FM]^2 is pretrained only on 307 h from a single clinical corpus with a fixed 19-channel montage, so whether its advantage persists at the $9{,}000$–$60{,}000$ h scale of existing models remains untested. Beyond EEG, SplitUNet may extend naturally to other multi-channel time series with structured sensor arrays such as MEG and ECoG. Class- and subject-conditional extensions also open the potential to synthesize rare clinical cases, accelerating clinical neuroscience research.

Acknowledgment

We thank the MIT Office of Research Computing and Data (ORCD) for support through ORCD Seed Fund Grants, which provided access to GPUs and additional funding support. We also thank NVIDIA and AMD for supporting GPUs. This work was supported by the Flemish Government under AI Research Program, METHUSALEM grant METH/26/003 (Methusalem-BioMedAI-Explainable and generative AI for accelerating biomedical discoveries: from genome to function) and VLAIO project HBC.2025.0582 (MediSynth). This work was also supported by the MIT-Belgium KU Leuven Seed Fund, part of the MIT Global Seed Funds[2], through the MIT-Belgium program[3].

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.

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. Chronos: Learning the language of time series. *TMLR*, 2024. ISSN 2835-8856.

Dmitry Baranchuk, Andrey Voynov, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. In *ICLR*, 2022.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

Hohyun Cho, Minkyu Ahn, Sangtae Ahn, Moonyoung Kwon, and Sung Chan Jun. EEG datasets for motor imagery brain–computer interface. *GigaScience*, 6(7):gix034, 2017.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *ICML*, 2024.

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.

Jiaxiang Dong, Haixu Wu, Haoran Zhang, Li Zhang, Jianmin Wang, and Mingsheng Long. SimMTM: A simple pre-training framework for masked time-series modeling. In *NeurIPS*, 2023.

Yassine El Ouahidi, Jonathan Lys, Philipp Thölke, Nicolas Farrugia, Bastien Pasdeloup, Vincent Gripon, Karim Jerbi, and Giulia Lioi. REVE: A foundation model for EEG–adapting to any setup with large-scale pretraining on 25,000 subjects. In *NeurIPS*, 2025.

Sean Ferrell, Vineetha Mathew, Matthew Refford, Vincent Tchiong, Tameem Ahsan, Iyad Obeid, and Joseph Picone. The Temple University Hospital EEG corpus: Electrode location and channel labels. *Institute for Signal and Information Processing Report*, 1(1), 2020.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *ICLR*, 2022.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Weibang Jiang, Liming Zhao, and Bao-Liang Lu. Large brain model for learning generic representations with tremendous EEG data in BCI. In *ICLR*, 2024.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-LLM: Time series forecasting by reprogramming large language models. In *ICLR*, 2024.

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.

[2] `https://cis.mit.edu/opportunities/seed-funding/available-funds/gsf-belgium`

[3] `https://cis.mit.edu/regions/europe/mit-belgium`

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *NeurIPS*, 2022.

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.

George H Klem, Hans O Lüders, Herbert H Jasper, and Christian Elger. The ten-twenty electrode system of the International Federation. The International Federation of Clinical Neurophysiology. *Electroencephalogr. Clin. Neurophysiol. Suppl.*, 52:3–6, 1999.

Marcel Kollovieh, Abdul Fatir Ansari, Michael Bohlke-Schneider, Jasper Zschiegner, Hao Wang, and Yuyang Wang. Predict, refine, synthesize: Self-guiding diffusion models for probabilistic time series forecasting. In *NeurIPS*, 2023.

Marcel Kollovieh, Marten Lienen, David Lüdke, Leo Schwinn, and Stephan Günnemann. Flow matching with Gaussian process priors for probabilistic time series forecasting. In *ICLR*, 2025.

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. DiffWave: A versatile diffusion model for audio synthesis. In *ICLR*, 2021.

Demetres Kostas, Stéphane 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.

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.

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.

Tianhong Li, Dina Katabi, and Kaiming He. Return of unconditional generation: A self-supervised representation generation method. In *NeurIPS*, 2024.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *ICLR*, 2023.

Haijie Liu, Penghu Wei, Haochong Wang, Xiaodong Lv, Wei Duan, Meijie Li, Yan Zhao, Qingmei Wang, Xinyuan Chen, Gaige Shi, Bo Han, and Junwei Hao. An EEG motor imagery dataset for brain computer interface in acute stroke patients. *Scientific Data*, 11(1):131, 2024.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.

Wei Liu, Jie-Lin Qiu, Wei-Long Zheng, and Bao-Liang Lu. Comparing recognition performance and robustness of multimodal deep learning models for multimodal emotion recognition. *IEEE Transactions on Cognitive and Developmental Systems*, 14(2):715–729, 2021.

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 model. In *ICLR*, 2026.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. SiT: Exploring flow and diffusion-based generative models with scalable interpolant Transformers. In *ECCV*, 2024.

Wajid Mumtaz, Likun Xia, Syed Saad Azhar Ali, Mohd Azhar Mohd Yasin, Muhammad Hussain, and Aamir Saeed Malik. Electroencephalogram (EEG)-based computer-aided technique to diagnose major depressive disorder (MDD). *Biomedical Signal Processing and Control*, 31:108–115, 2017.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021.

Iyad Obeid and Joseph Picone. The Temple University Hospital EEG data corpus. *Frontiers in Neuroscience*, 10:196, 2016.

Patrick Ofner, Andreas Schwarz, Joana Pereira, and Gernot R M"uller-Putz. Upper limb movements can be decoded from the time-domain of low-frequency EEG. *PLOS ONE*, 12(8):e0182578, 2017.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023.

Wei Yan Peh, Yuanyuan Yao, and Justin Dauwels. Transformer convolutional neural networks for automated artifact detection in scalp EEG. In *EMBC*, 2022.

Jia Hong Puah, Sim Kuan Goh, Ziwei Zhang, Zixuan Ye, Chow Khuen Chan, Kheng Seang Lim, Si Lei Fong, and Kok Sin Woon. EEGDM: EEG representation learning via generative diffusion model. *arXiv preprint arXiv:2508.14086*, 2025.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

Yulia Rubanova, Ricky TQ Chen, and David K Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In *NeurIPS*, 2019.

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

Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Efficient attention: Attention with linear complexities. In *WACV*, 2021.

Yonghao Song, Xueyu Jia, Lie Yang, and Longhan Xie. Transformer-based spatial-temporal feature learning for EEG decoding. *arXiv preprint arXiv:2106.11170*, 2021.

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.

Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *NeurIPS*, 2023.

Michael Tangermann, Klaus-Robert M"uller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J Miller, Gernot R M"uller-Putz, et al. Review of the BCI competition IV. *Frontiers in Neuroscience*, 6:55, 2012.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. CSDI: Conditional score-based diffusion models for probabilistic time series imputation. In *NeurIPS*, 2021.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *TMLR*, 2024. ISSN 2835-8856. Expert Certification.

Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

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 *NeurIPS*, 2024.

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 *ICLR*, 2025.

Peter Welch. The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, 15(2): 70–73, 1967.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *ICML*, 2024.

Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *CVPR*, 2023.

Chaoqi Yang, M. Brandon Westover, and Jimeng Sun. BIOT: Biosignal Transformer for cross-data learning in the wild. In *NeurIPS*, 2023a.

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(1): e46769, 2023b.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *AAAI*, 2023.

Daoze Zhang, Zhizhang Yuan, Yang Yang, Junru Chen, Jingjing Wang, and Yafeng Li. Brant: Foundation model for intracranial neural signal. In *NeurIPS*, 2023.

Bangyan Zhou, Xiaopei Wu, Zhao Lv, Lei Zhang, and Xiaojin Guo. A fully automated trial selection method for optimization of motor imagery based brain-computer interface. *PLOS ONE*, 11(9):e0162657, 2016.

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 *NeurIPS*, 2025.

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

A Related Work

A.1 Brain Foundation Model

EEG foundation models aim to learn a general representation from large-scale brain signals (*i.e.*, EEG) to have better performance on diverse downstream tasks, as opposed to task-specific methods [Lawhern et al., 2018, Song et al., 2023]. Training approaches for current models are generally classified into two categories: contrastive learning [Chen et al., 2020] and masked reconstruction [He et al., 2022]. Early contrastive approaches, such as BENDR [Kostas et al., 2021] and Brant [Zhang et al., 2023], demonstrated the feasibility of joint representation learning for physiological signals. Concurrently, BIOT [Yang et al., 2023a] uses a Transformer [Vaswani et al., 2017] with masked reconstruction [He et al., 2022], which became a standard technique for subsequent methods [Jiang et al., 2024, El Ouahidi et al., 2025, Wang et al., 2024, 2025, Zhou et al., 2025]. Recent works focus on incorporating spatiotemporal information via self-attention [Yang et al., 2023a] or bottleneck attention [Zhou et al., 2025]. On the other hand, El Ouahidi et al. [2025] proposes a unified framework for multiple pretraining datasets via 4D electrode positional encoding with block masking. CodeBrain [Ma et al., 2026] tokenizes both waveform and frequency for richer representation and leverages a state-space model [Gu et al., 2022].

Across these models, discrete tokenization and masked autoencoding remain central. However, discretizing continuous EEG signals into fixed temporal units creates a structural mismatch, forcing models to reconstruct proxy representations rather than the raw waveform. The representation operates at patch granularity, with cross-patch interactions summarized at the patch level—coarser than the time scale at which EEG dynamics actually unfold. In contrast, we eliminate discretization entirely, modeling EEG directly in continuous time via a generative objective.

A.2 Factorized Spatiotemporal Processing

Decoupling temporal and spatial axes is well-established in EEG models such as EEGNet [Lawhern et al., 2018] and ShallowNet [Schirrmeister et al., 2017], but they apply a disjoint front-end filter that immediately collapses the electrode dimension ($E \to 1$). In contrast, our Conv(1+1)D block, inspired by R(2+1)D video convolutions [Tran et al., 2018], alternates temporal and electrode convolutions inside every residual block. Together with time-only downsampling, SplitUNet preserves electrode topology and interleaves spatiotemporal mixing across the entire deep hierarchy.

A.3 Continuous-Time Generative Models

Continuous-time generative models learn a transport path from a simple prior to the data distribution by regressing onto a chosen interpolant. Diffusion-based variants [Ho et al., 2020, Karras et al.,

2022, Nichol and Dhariwal, 2021] simulate a stochastic process during training, whereas flow matching [Lipman et al., 2023] simplifies this by defining a regression target along a deterministic path, thereby removing the need for a noise schedule. Both frameworks have been applied to neural audio waveforms [Kong et al., 2021] as well as multivariate time-series imputation and forecasting [Tashiro et al., 2021, Kollovieh et al., 2023, 2025]. We adopt the optimal-transport conditional flow-matching (OT-CFM) [Tong et al., 2024], which pairs each noise sample with its nearest data sample within a minibatch; the resulting velocity field is provably straighter, enabling accurate, few-step sampling without the need for distillation.

Beyond synthesis, continuous-time generative pretraining has emerged as a robust approach for representation learning. Features from unconditional diffusion or flow models support label-efficient segmentation [Baranchuk et al., 2022, Xu et al., 2023], dense visual correspondence [Tang et al., 2023], and discriminative classification on par with contrastive features [Li et al., 2024]. B[FM]^2 extends this paradigm to multi-channel EEG waveforms, where the continuous-time generative objective serves as the foundation-model training signal. Concurrently, EEGDM [Puah et al., 2025] pretrains a structured state-space model (SSM) using DDPM diffusion [Ho et al., 2020] and adapts via a latent fusion Transformer. In contrast, B[FM]^2 pairs optimal-transport flow matching with our hierarchical SplitUNet backbone, extracts representations directly with a simple head, and benchmarks across nine diverse downstream tasks alongside clinical fidelity evaluations.

A.4 Time Series Foundation Model

Time-series foundation models pretrain a single backbone on large multi-domain time-series corpora to enable zero-shot forecasting and downstream task adaptation. Approaches in this space include autoregressive next-token Transformers [Ansari et al., 2024, Das et al., 2024], masked-encoder Transformers [Woo et al., 2024], and large-language-model adaptations [Jin et al., 2024]. While these differ in architecture, they all tokenize the input series—either into a vocabulary of quantized values or into a sequence of fixed-length patches—and the backbone operates on tokens rather than on raw continuous samples. Continuous-time generative pretraining on raw signals has not yet been adopted in this literature; diffusion- and flow-matching-based forecasting [Tashiro et al., 2021, Kollovieh et al., 2025] has remained task-specific rather than foundation-model-scale. B[FM]^2 differs in both respects: it operates on the raw multi-channel waveform without tokenization, and pretrains the backbone via continuous-time flow matching.

B Implementation Details

We provide additional details from Section 3.1.

Backbone. Our velocity network is a 6 M-parameter SplitUNet adapted from the UNet of Ho et al. [2020], via the widely used `denoising-diffusion-pytorch` reimplementation[4], with two EEG-driven modifications. **(a) (1+1)D Factorization.** Every 3×3 2D convolution is replaced by a pair of 1D convolutions: a temporal convolution along the time axis followed by an electrode-mixing convolution along the electrode axis with a $(3, 1)$ kernel and $(1, 0)$ padding. The initial input convolution uses a wider $(1, 7)$ kernel with $(0, 3)$ padding to widen the temporal receptive field. **(b) Time-only downsampling.** Down- and upsampling act only along time, using $(1, 4)$ kernels at stride $(1, 2)$ with $(0, 1)$ padding, preserving the number of electrodes and their topology rather than collapsing them at the bottleneck.

The network operates on input tensors of shape $(B, 1, E, T)$ where E and T denote the number of electrodes and time points, with base feature width 32, channel multipliers $(1, 2, 4, 8)$ across four encoder stages, and a mirrored decoder. Per-stage linear-attention blocks [Shen et al., 2021] use 4 heads of 32 dimensions; a single bottleneck full-self-attention block acts where the time axis has been compressed by $8\times$. The sinusoidal flow-time embedding ($\theta = 10{,}000$) feeds an MLP of width 128, injected into every residual block as a per-channel scale-and-shift on the post-normalization activations [Nichol and Dhariwal, 2021]. Self-conditioning is disabled and dropout is set to zero.

Sampling. Unconditional samples are drawn by integrating $\dot{\boldsymbol{x}}_t = v_\theta(\boldsymbol{x}_t, t)$ from $t{=}0$ to $t{=}1$ with a first-order (Euler) solver at 50 uniform steps. The straightness induced by the OT coupling makes

[4] `https://github.com/lucidrains/denoising-diffusion-pytorch`

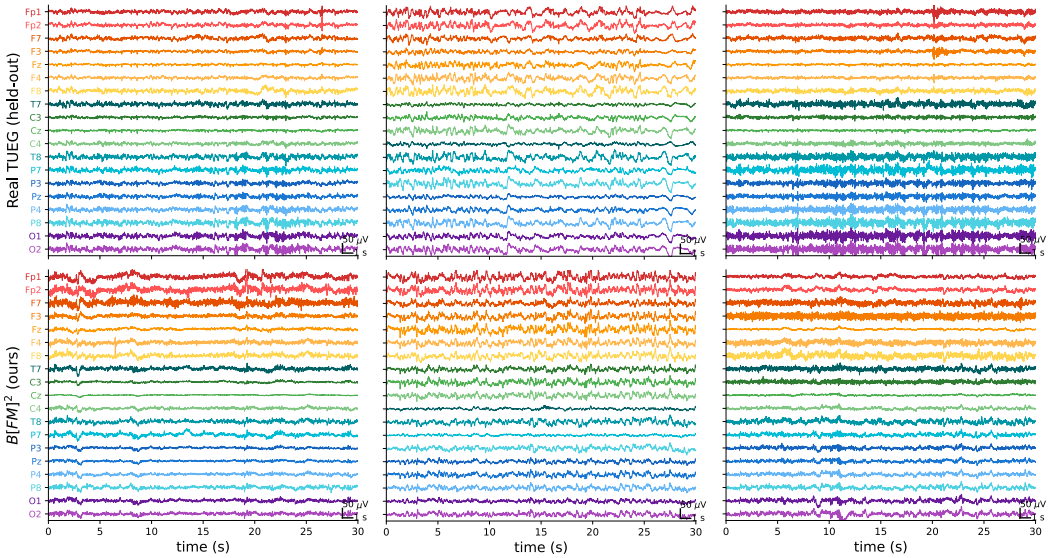


Figure 5: **Real vs. B[FM]^2 EEG segments.** Three 30 s held-out TUEG segments (top row) and three unconditional B[FM]^2 samples (bottom row), shown in the 19-channel referential 10–20 montage used for pretraining. Readers in the blinded study (see Table 3; full protocol in Appendix D) received no examples in advance and viewed segments at standard clinical display settings with randomly interleaved order. The traces shown here are illustrative; subtle textural differences (e.g., apparent smoothness in synthetic samples) are illustrated in Figure 4. Calibration bar: 1 s and 50 μV.

this budget sufficient. We did not observe further improvement in band-power W_1 from raising the step count.

Preprocessing. Recordings are resampled to 200 Hz, band-pass filtered to 0.3–75 Hz, notch-filtered at 60 Hz to remove signal from the US powerline, and partitioned into non-overlapping 30 s segments. Segments with any timepoint exceeding 100 μV in absolute amplitude are discarded; remaining segments are normalized to $[-1, +1]$ by dividing by 100 μV. We retain the 19 channels of the international 10–20 system [Klem et al., 1999], excluding the ear references A1 and A2: Fp1, Fp2, F7, F3, Fz, F4, F8, T3, C3, Cz, C4, T4, T5, P3, Pz, P4, T6, O1, O2.

Pretraining hyperparameters. We optimize with Adam (β_1=0.9, β_2=0.99, no weight decay) at a constant learning rate of 8×10^{-5}, with gradient clipping at $\|g\|_2 \leq 1.0$. Each step uses a batch of 4 segments on a single NVIDIA H100 80 GB GPU with fp16 mixed-precision training (Hugging Face Accelerate) and no gradient accumulation. The flow time $t \sim \mathcal{U}[0, 1]$ is rescaled by 1000 before entering the time MLP. We train for 70 epochs and maintain an EMA of the weights with decay 0.995 updated every 10 optimizer steps; the EMA copy is used for all sampling and downstream feature extraction.

Finetuning hyperparameters. We optimize cross-entropy with AdamW (β_1=0.9, β_2=0.999, weight decay 0.01) and gradient clipping at $\|g\|_2 \leq 1.0$. Training uses a batch size of 32 and runs for 40 epochs. For each task, we tune the learning rate ($5 \cdot 10^{-5}$, 10^{-4}, or $2 \cdot 10^{-4}$) and the feature-aggregation strategy (mean pooling across both electrode and time axes vs. flattening) based on validation performance. For each run we report the test balanced accuracy of the best-performing epoch on the test split, and each cell averages the five highest-scoring seeds. The same rule is applied to every model we train ourselves for the ablation experiments (see Tables 5, 6, and 7).

C Downstream Datasets

All recordings are resampled to 200 Hz before windowing. Sample counts below are rounded; precise values appear in Table 1.

TUAB. A binary normal-vs-abnormal benchmark drawn from the TUH EEG Corpus [Obeid and Picone, 2016], with labels assigned by clinical neurologists. We adopt the 16-channel bipolar montage of the CBraMod [Wang et al., 2025] pipelines and 10 s non-overlapping windows on the released subject-disjoint split: $\sim$ 409k windows from $\sim$ 2,993 patients.

TUEV. Six-class event classification from the same TUH EEG Corpus [Obeid and Picone, 2016]: SPSW (spike and sharp wave), GPED, PLED, EYEM (eye movement), ARTF (artifact), BCKG (background). Same montage as TUAB; 5 s windows; subject-disjoint split; $\sim$ 112k events. Class frequencies are strongly skewed toward BCKG.

BCIC-IV-2a. The four-class motor-imagery competition benchmark of Tangermann et al. [2012]: 9 subjects imagining left-hand, right-hand, both-feet, or tongue movements at 22 channels and 250 Hz. After resampling and 4 s windowing on the released competition partition: 5,184 trials.

ISRUC. Five-class AASM sleep staging from 100 subjects in subgroup 3 of Khalighi et al. [2016]. The released recordings include a chin EMG channel that CBraMod [Wang et al., 2025] pipelines treat as EEG; following El Ouahidi et al. [2025] we drop it and keep the six EEG derivations only. Subject-wise split, 30 s window from $\sim$ 89k samples.

Mumtaz. Binary major-depressive-disorder vs. healthy-control classification [Mumtaz et al., 2017] on 19 MDD patients and 15 controls, 19-channel 10–20 EEG at 256 Hz. The released subject-wise split with 5 s non-overlapping windows: $\sim$ 7,100 samples.

MAT (Mental Arithmetic). Binary stress-vs-rest contrast [Zyma et al., 2019]: 36 subjects performing timed serial subtraction against eyes-closed baseline, 20-channel EEG at 500 Hz, 5 s windowed. The smallest task in our suite at 1,707 samples and consequently the noisiest probe.

Siena. Binary seizure detection on 14 adult patients of Detti et al. [2020] (512 Hz, 10–20 montage). We retain the 29 channels available across all subjects and use 10 s windows (51,307 samples). Subjects PN16 and PN17 are held out as the test set; the remaining 12 are split 8 : 2 for training and validation.

HMC. Five-class AASM sleep staging from the 151-subject PSG corpus of Alvarez-Estevez and Rijsman [2021]. We use the four EEG channels (F4-M1, C4-M1, O2-M1, C3-M2) at 200 Hz with 30 s epochs on a subject-wise split: 137,243 samples.

SEED-V. Five-state emotion classification (happy, sad, neutral, disgust, fear) on the 16-subject corpus of Liu et al. [2021], three sessions per subject, 62 channels at 1000 Hz. We segment into 1 s windows on a subject-wise split: 117,744 samples.

D Neurologist Rating Protocol

We recruited two board-certified neurologists for the evaluation. Stimuli were presented through a single-page web application (see Figure 6) at standard clinical display settings (10 mm/s paper speed, 7 μV/mm sensitivity, 0.3–70 Hz bandpass, 60 Hz notch). Although the original data and generated samples are in a referential montage, we convert them to a longitudinal bipolar montage to match the neurologists' routine reading convention. Real and generated segments were matched in duration and post-preprocessing amplitude range and rendered identically; readers saw no metadata and were blind to provenance throughout. The full Likert anchors are 1= definitely real, 2= probably real, 3= unsure, 4= probably fake, 5= definitely fake; ratings were entered by mouse or via 1–5 keyboard shortcuts.

We report, per reader and pooled, the mean realness score separately for the real held-out and B[FM]^2-generated subsets, and the difference between the two subset means with its Mann–Whitney U test p value (an unpaired test, since real and synthetic segments are independent draws). We assess segment-level discriminability using two agreement statistics: Cohen's κ between each reader's binarized rating (real vs. fake) and the true real/synthetic label, and Spearman ρ between the two readers' per-segment ratings. Samples are deemed indistinguishable from real if neither reader can

reliably classify individual segments—that is, if Cohen's κ is near zero and inter-reader Spearman ρ is low—rather than from a failure to reject equality of means, which would not establish equivalence.

E Heuristics for Brain-State Pattern Mining for Qualitative Samples

Figure 4 shows the top-scoring generated sample for each of three brain-state patterns: a sharp-wave / spike-like transient, δ-dominant slow-wave activity, and an eyes-closed posterior α rhythm. We mine these samples from B[FM]^2's unconditional generation pool by ranking them with simple hand-crafted scores built from spectral and amplitude features. Each score is a product of (i) a *shape* term—a ratio that fires when the spatial or spectral profile of a target pattern is present—and (ii) a *magnitude* term $\log(1+\cdot)$ that suppresses low-amplitude samples which happen to satisfy the shape term by chance. The pattern labels are suggestive ("δ-rich"), not diagnostic.

Features. For each $E\times T$ generated sample (the standard 10–20 montage [Klem et al., 1999], 30 s at 200 Hz), we estimate the per-channel power spectral density $P(f)$ using Welch's method [Welch, 1967] with a 256-point Hann window and 50% overlap, and integrate it to obtain band powers in the three canonical EEG bands relevant to our targets:

$$b_\delta = \textstyle\int_{0.5}^{4} P(f)\,df, \qquad b_\alpha = \int_{8}^{13} P(f)\,df, \qquad b_\beta = \int_{13}^{30} P(f)\,df.$$

For the sharp-wave / spike-like transient pattern we use two amplitude features instead of band power: the median per-channel peak-to-peak amplitude A_{pp} and the count N_σ of large-amplitude excursions ($|z|>5$ in the per-channel z-scored signal).

Pattern scores. Let $\langle\cdot\rangle_S$ denote channel-averaging over an electrode subset S, where 'all' is the full 19-electrode set. The three pattern scores are:

$$s_{\text{spike}} = N_\sigma \cdot \log\bigl(1 + A_{\mathrm{pp}}\bigr) \qquad \text{(sharp-wave / spike-like transient)}$$

$$s_{\text{slow}} = \frac{\langle b_\delta\rangle_{\text{all}}}{\langle b_\delta\rangle_{\text{all}} + \langle b_\alpha\rangle_{\text{all}} + \langle b_\beta\rangle_{\text{all}}} \cdot \log\bigl(1 + \langle b_\delta\rangle_{\text{all}}\bigr) \qquad (\delta\text{-dominant slow-wave activity})$$

$$s_{\text{eye}} = \frac{\langle b_\alpha\rangle_{\{\mathrm{O1},\,\mathrm{O2}\}}}{\langle b_\alpha\rangle_{\text{all}}} \cdot \log\bigl(1 + \langle b_\alpha\rangle_{\text{all}}\bigr) \qquad \text{(posterior } \alpha \text{ rhythm).}$$

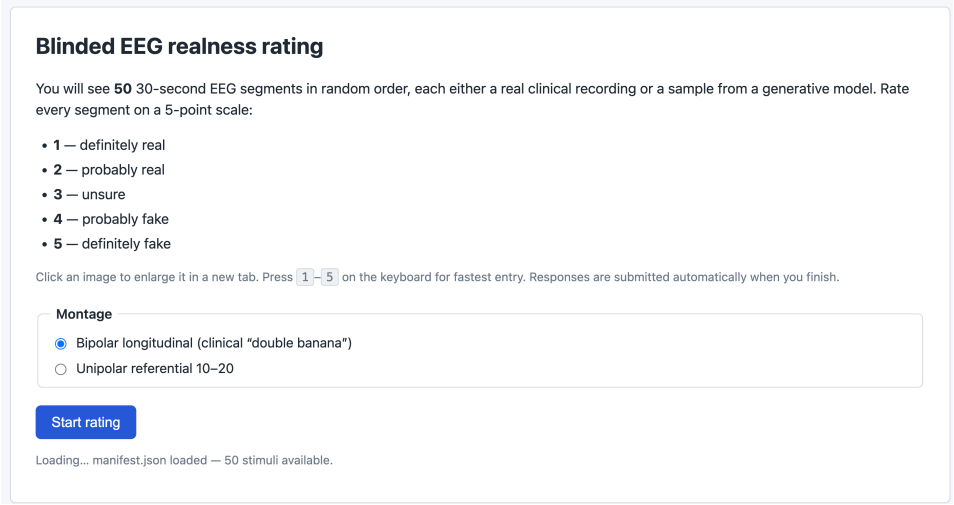
The first score fires when many large-amplitude excursions co-occur with high overall amplitude (as in interictal sharp-wave or spike-like transients); the second when low-frequency δ power dominates the spectrum (as in slow-wave sleep); and the third when α power is concentrated over the occipital electrodes (posterior dominant rhythm of eyes-closed wakefulness).

F Band-Power Distribution

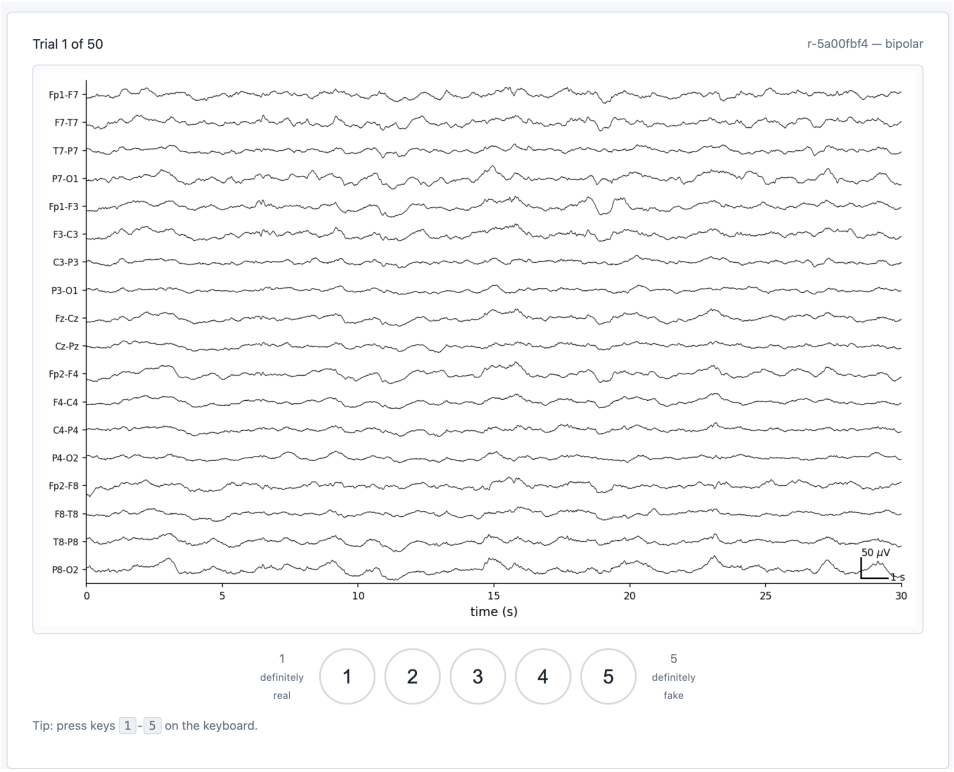
The five canonical bands are delta (1–4 Hz), theta (4–8 Hz), alpha (8–13 Hz), beta (13–30 Hz), and gamma (30–50 Hz). The per-channel log-power is computed in each band, and the Wasserstein-1 distance is averaged across channels and bands to give the single value reported in Table 4. Table 8 reports the per-band breakdown.

G Ablation Study on Feature-Extraction Timestep

For downstream transfer, B[FM]^2 conditions the velocity network on $t{=}0$, the noise endpoint of the probability path. Because a natural alternative is the clean-data endpoint ($t{=}1000$) or an intermediate flow state, we swept the conditioning timestep on TUAB across five settings spanning the interval with five seeds each (see Table 9). Downstream performance remains stable across all timesteps, with balanced accuracy spanning 0.802–0.819 and AUROC 0.882–0.892, a variation within typical seed-to-seed variance. Evaluating at $t{=}0$ is therefore a practical choice rather than a sensitive hyperparameter; it achieves the strongest overall metrics while evaluating directly without simulating an extra forward step or injecting noise.



(a) Welcome screen.



(b) Per-trial rating screen.

Figure 6: **Blinded neurologist rating interface.** (a) Welcome screen describing the task, the 5-point realness Likert, and selecting the longitudinal bipolar ("double banana") montage—the standard clinical reading layout, which removes reference-electrode artifacts and matches neurologists' habitual reading view. (b) Per-trial rating screen: a 30 s segment in the bipolar montage at clinical display settings, with a 50 μV scale bar and the Likert buttons below; trial index and an opaque stimulus identifier appear at the top so the reader cannot infer provenance. The interface randomly interleaves real held-out TUEG segments and unconditional B[FM]^2 samples and submits responses automatically; readers saw no metadata at any point.

Table 8: Sample quality of B[FM]^2 trained with four velocity-network backbones, measured as the Wasserstein-1 distance between log-band-power distributions of generated and held-out real TUEG segments (n=5,000 each; mean $\pm$ standard deviation across seeds). *Real-vs-Real* is a noise-floor baseline computed between two disjoint TUEG shards. Best generator in each column (excluding the baseline) in bold.

Backbone	δ	θ	α	β	γ	Average
Real-vs-Real	0.147 ± 0.064	0.147 ± 0.029	0.180 ± 0.039	0.165 ± 0.076	0.141 ± 0.060	0.156 ± 0.028
UNet-1D (joint)	0.266 ± 0.004	0.337 ± 0.004	0.231 ± 0.004	0.218 ± 0.003	$\mathbf{0.227 \pm 0.004}$	0.256 ± 0.002
UNet-1D (indep.)	1.379 ± 0.135	1.842 ± 0.200	1.944 ± 0.190	1.973 ± 0.019	2.375 ± 0.054	1.903 ± 0.098
UNet-2D	0.279 ± 0.002	$\mathbf{0.158 \pm 0.003}$	0.262 ± 0.010	0.230 ± 0.009	0.309 ± 0.010	0.248 ± 0.001
SplitUNet	$\mathbf{0.227 \pm 0.005}$	0.160 ± 0.004	$\mathbf{0.126 \pm 0.005}$	$\mathbf{0.197 \pm 0.008}$	0.265 ± 0.005	$\mathbf{0.195 \pm 0.002}$

Table 9: **Sensitivity to the feature-extraction timestep on TUAB.** The backbone is conditioned on a fixed timestep t during finetuning and inference; t is on the network's internal $[0, 1000]$ scale, so t=1000 is the clean-data endpoint. Bold denotes the best performance.

t	Balanced Accuracy	AUROC	AUC-PR
0 (ours)	$\mathbf{0.819 \pm 0.004}$	$\mathbf{0.892 \pm 0.008}$	$\mathbf{0.896 \pm 0.006}$
250	0.804 ± 0.012	0.882 ± 0.012	0.889 ± 0.012
500	0.807 ± 0.007	0.887 ± 0.008	0.892 ± 0.011
750	0.802 ± 0.005	0.882 ± 0.007	0.890 ± 0.010
1000	0.807 ± 0.007	0.889 ± 0.003	$\mathbf{0.896 \pm 0.004}$

H Per-task detailed results

For each of the nine downstream tasks, we report balanced accuracy and two task-appropriate companion metrics (Cohen's κ and Weighted F1 for multi-class; AUC-PR and AUROC for binary). External-baseline numbers are taken directly from El Ouahidi et al. [2025] and Zhou et al. [2025]. All B[FM]^2 entries are mean $\pm$ standard deviation over five seeds. Dataset descriptions are in Appendix C.

H.1 Mumtaz (Mental Disorder Diagnosis)

Table 10: Mumtaz (2-class, MDD vs. healthy control). External-baseline numbers from El Ouahidi et al. [2025].

Method	**Balanced Accuracy**	**AUC-PR**	**AUROC**
EEGNet	0.923 ± 0.010	0.963 ± 0.009	0.964 ± 0.009
EEGConformer	0.931 ± 0.012	0.968 ± 0.011	0.970 ± 0.010
SPaRCNet	0.932 ± 0.009	0.975 ± 0.006	0.978 ± 0.008
ContraWR	0.919 ± 0.011	0.959 ± 0.010	0.962 ± 0.009
CNN-Transformer	0.930 ± 0.007	0.976 ± 0.007	0.974 ± 0.006
FFCL	0.931 ± 0.004	0.972 ± 0.002	0.975 ± 0.003
ST-Transformer	0.913 ± 0.010	0.958 ± 0.009	0.959 ± 0.006
BIOT	0.936 ± 0.005	0.974 ± 0.003	0.976 ± 0.004
LaBraM-Base	0.941 ± 0.008	0.980 ± 0.009	0.978 ± 0.006
CBraMod	0.956 ± 0.006	0.992 ± 0.003	0.992 ± 0.003
REVE	$\underline{0.964 \pm 0.010}$	$\underline{0.996 \pm 0.001}$	$\underline{0.996 \pm 0.002}$
CSBrain	0.964 ± 0.015	0.994 ± 0.003	0.996 ± 0.002
B[FM]^2 (ours)	$\mathbf{1.000 \pm 0.000}$	$\mathbf{1.000 \pm 0.000}$	$\mathbf{1.000 \pm 0.000}$

H.2 MAT (Mental Stress Detection)

Table 11: MAT (2-class, mental arithmetic vs. rest). External-baseline numbers from El Ouahidi et al. [2025].

Method	Balanced Accuracy	AUC-PR	AUROC
EEGNet	0.677 ± 0.012	0.576 ± 0.010	0.732 ± 0.011
EEGConformer	0.680 ± 0.012	0.583 ± 0.013	0.742 ± 0.013
SPaRCNet	0.688 ± 0.011	0.583 ± 0.019	0.742 ± 0.013
ContraWR	0.663 ± 0.010	0.579 ± 0.016	0.733 ± 0.008
CNN-Transformer	0.678 ± 0.027	0.578 ± 0.029	0.726 ± 0.034
FFCL	0.680 ± 0.014	0.579 ± 0.027	0.733 ± 0.020
ST-Transformer	0.663 ± 0.017	0.567 ± 0.026	0.713 ± 0.017
BIOT	0.688 ± 0.019	0.600 ± 0.019	0.754 ± 0.014
LaBraM-Base	0.691 ± 0.013	0.600 ± 0.015	0.772 ± 0.009
CBraMod	0.726 ± 0.013	0.627 ± 0.010	0.790 ± 0.007
REVE	$\underline{0.766 \pm 0.035}$	$\underline{0.747 \pm 0.081}$	0.845 ± 0.051
CSBrain	0.756 ± 0.011	0.670 ± 0.022	$\underline{0.848 \pm 0.030}$
B[FM]^2 (ours)	$\mathbf{0.840 \pm 0.022}$	$\mathbf{0.806 \pm 0.038}$	$\mathbf{0.912 \pm 0.022}$

H.3 Siena (Seizure Detection)

Table 12: Siena (2-class, seizure vs. normal). External-baseline numbers from Zhou et al. [2025].

Method	Balanced Accuracy	AUC-PR	AUROC
EEGNet	0.749 ± 0.052	0.375 ± 0.087	0.869 ± 0.053
EEGConformer	0.756 ± 0.021	0.209 ± 0.079	0.816 ± 0.026
SPaRCNet	0.657 ± 0.038	0.316 ± 0.066	0.733 ± 0.086
ContraWR	0.655 ± 0.031	0.371 ± 0.041	0.782 ± 0.060
CNN-Transformer	0.698 ± 0.056	0.384 ± 0.071	0.872 ± 0.040
FFCL	0.662 ± 0.039	0.394 ± 0.090	0.815 ± 0.116
ST-Transformer	0.753 ± 0.038	0.364 ± 0.025	0.888 ± 0.009
BIOT	0.735 ± 0.067	0.381 ± 0.089	0.903 ± 0.030
LaBraM-Base	0.708 ± 0.033	0.312 ± 0.098	0.881 ± 0.033
CBraMod	0.732 ± 0.065	$\underline{0.411 \pm 0.072}$	$\underline{0.904 \pm 0.022}$
REVE	0.740 ± 0.007	0.381 ± 0.053	0.875 ± 0.030
CSBrain	$\underline{0.766 \pm 0.047}$	$\mathbf{0.487 \pm 0.034}$	$\mathbf{0.908 \pm 0.012}$
B[FM]^2 (ours)	$\mathbf{0.776 \pm 0.010}$	0.383 ± 0.036	0.883 ± 0.012

H.4 ISRUC (Sleep Staging)

Table 13: ISRUC (5-class, sleep staging). External-baseline numbers from El Ouahidi et al. [2025]. REVE numbers use the corrected six-channel montage described in Appendix C.

Method	Balanced Accuracy	Cohen's κ	Weighted F1
EEGNet	0.715 ± 0.012	0.704 ± 0.017	0.751 ± 0.012
EEGConformer	0.740 ± 0.013	0.714 ± 0.016	0.763 ± 0.015
SPaRCNet	0.749 ± 0.007	0.710 ± 0.013	0.762 ± 0.009
ContraWR	0.740 ± 0.013	0.718 ± 0.016	0.761 ± 0.014
CNN-Transformer	0.736 ± 0.009	0.713 ± 0.012	0.772 ± 0.011
FFCL	0.728 ± 0.018	0.702 ± 0.029	0.761 ± 0.020
ST-Transformer	0.738 ± 0.021	0.701 ± 0.035	0.768 ± 0.018
BIOT	0.753 ± 0.012	0.719 ± 0.023	0.779 ± 0.015
LaBraM-Base	0.763 ± 0.010	0.723 ± 0.018	0.781 ± 0.013
CBraMod	0.786 ± 0.011	0.744 ± 0.015	$\underline{0.801 \pm 0.010}$
REVE	0.782 ± 0.008	$\underline{0.750 \pm 0.016}$	0.800 ± 0.013
CSBrain	$\underline{0.792 \pm 0.003}$	0.741 ± 0.010	0.799 ± 0.009
B[FM]^2 (ours)	$\mathbf{0.806 \pm 0.001}$	$\mathbf{0.780 \pm 0.005}$	$\mathbf{0.829 \pm 0.004}$

H.5 HMC (Sleep Staging)

Table 14: HMC (5-class, sleep staging). External-baseline numbers from Zhou et al. [2025].

Method	Balanced Accuracy	Cohen's Kappa	Weighted F1
EEGNet	0.653 ± 0.012	0.589 ± 0.020	0.654 ± 0.017
EEGConformer	0.715 ± 0.009	0.643 ± 0.005	0.708 ± 0.004
SPaRCNet	0.476 ± 0.111	0.315 ± 0.132	0.411 ± 0.131
ContraWR	0.424 ± 0.054	0.234 ± 0.055	0.299 ± 0.029
CNN-Transformer	0.657 ± 0.014	0.596 ± 0.011	0.690 ± 0.006
FFCL	0.443 ± 0.070	0.254 ± 0.065	0.290 ± 0.049
ST-Transformer	0.256 ± 0.014	0.050 ± 0.018	0.143 ± 0.012
BIOT	0.686 ± 0.004	0.629 ± 0.011	0.709 ± 0.015
LaBraM-Base	0.728 ± 0.010	0.681 ± 0.005	0.745 ± 0.003
CBraMod	0.727 ± 0.004	0.668 ± 0.010	0.740 ± 0.009
REVE	$\underline{0.740 \pm 0.007}$	$\underline{0.698 \pm 0.008}$	$\underline{0.764 \pm 0.007}$
CSBrain	0.735 ± 0.005	0.682 ± 0.005	0.751 ± 0.004
B[FM]^2 (ours)	$\mathbf{0.764 \pm 0.002}$	$\mathbf{0.711 \pm 0.004}$	$\mathbf{0.775 \pm 0.004}$

H.6 TUEV (Event Type Classification)

Table 15: TUEV (6-class, epileptiform event). External-baseline numbers from El Ouahidi et al. [2025].

Method	Balanced Accuracy	Cohen's κ	Weighted F1
EEGNet	0.388 ± 0.014	0.358 ± 0.015	0.654 ± 0.012
EEGConformer	0.407 ± 0.016	0.397 ± 0.019	0.698 ± 0.015
SPaRCNet	0.416 ± 0.026	0.423 ± 0.018	0.702 ± 0.010
ContraWR	0.438 ± 0.035	0.391 ± 0.024	0.689 ± 0.014
CNN-Transformer	0.409 ± 0.016	0.382 ± 0.013	0.685 ± 0.029
FFCL	0.398 ± 0.010	0.373 ± 0.019	0.678 ± 0.012
ST-Transformer	0.398 ± 0.023	0.377 ± 0.031	0.682 ± 0.019
BIOT	0.528 ± 0.022	0.527 ± 0.025	0.749 ± 0.008
LaBraM-Base	0.641 ± 0.006	0.664 ± 0.009	0.831 ± 0.005
CBraMod	0.667 ± 0.011	0.677 ± 0.010	$\underline{0.834 \pm 0.006}$
REVE	0.676 ± 0.023	$\underline{0.678 \pm 0.020}$	$\mathbf{0.845 \pm 0.013}$
CSBrain	$\underline{0.690 \pm 0.006}$	$\mathbf{0.683 \pm 0.005}$	0.833 ± 0.006
B[FM]^2 (ours)	$\mathbf{0.715 \pm 0.013}$	0.625 ± 0.019	0.806 ± 0.010

H.7 TUAB (Abnormal Detection)

Table 16: TUAB (binary normal vs. abnormal). External-baseline numbers from El Ouahidi et al. [2025].

Method	Balanced Accuracy	AUC-PR	AUROC
EEGNet	0.764 ± 0.004	0.830 ± 0.004	0.841 ± 0.003
EEGConformer	0.776 ± 0.005	0.843 ± 0.005	0.845 ± 0.004
SPaRCNet	0.790 ± 0.002	0.841 ± 0.002	0.868 ± 0.001
ContraWR	0.775 ± 0.004	0.842 ± 0.010	0.846 ± 0.007
CNN-Transformer	0.778 ± 0.002	0.843 ± 0.004	0.846 ± 0.001
FFCL	0.785 ± 0.004	0.845 ± 0.006	0.857 ± 0.005
ST-Transformer	0.797 ± 0.002	0.852 ± 0.003	0.871 ± 0.002
BIOT	0.796 ± 0.006	0.879 ± 0.002	0.881 ± 0.004
LaBraM-Base	0.814 ± 0.002	0.896 ± 0.002	0.902 ± 0.001
CBraMod	$\underline{0.829 \pm 0.002}$	$\underline{0.926 \pm 0.001}$	$\underline{0.923 \pm 0.001}$
REVE	$\mathbf{0.832 \pm 0.001}$	$\mathbf{0.928 \pm 0.001}$	$\mathbf{0.924 \pm 0.001}$
CSBrain	0.817 ± 0.004	0.900 ± 0.007	0.896 ± 0.005
B[FM]^2 (ours)	0.819 ± 0.004	0.896 ± 0.006	0.892 ± 0.008

H.8 BCIC-IV-2a (Motor Imagery)

Table 17: BCIC-IV-2a (4-class, motor imagery) dataset. External-baseline numbers from El Ouahidi et al. [2025].

Method	Bal. Acc.	κ	W-F1
EEGNet	0.448 ± 0.009	0.269 ± 0.012	0.423 ± 0.011
EEGConformer	0.470 ± 0.011	0.292 ± 0.014	0.453 ± 0.013
SPaRCNet	0.464 ± 0.012	0.285 ± 0.015	0.443 ± 0.013
ContraWR	0.468 ± 0.013	0.290 ± 0.016	0.441 ± 0.014
CNN-Transformer	0.460 ± 0.011	0.280 ± 0.015	0.446 ± 0.011
FFCL	0.447 ± 0.014	0.263 ± 0.018	0.424 ± 0.014
ST-Transformer	0.458 ± 0.015	0.273 ± 0.020	0.447 ± 0.014
BIOT	0.475 ± 0.009	0.300 ± 0.014	0.461 ± 0.013
LaBraM-Base	0.487 ± 0.009	0.316 ± 0.015	0.476 ± 0.010
CBraMod	0.514 ± 0.007	0.352 ± 0.009	0.498 ± 0.009
REVE	$\mathbf{0.640 \pm 0.009}$	$\mathbf{0.519 \pm 0.013}$	$\mathbf{0.634 \pm 0.011}$
CSBrain	0.566 ± 0.007	0.421 ± 0.009	0.564 ± 0.009
B[FM]^2 (ours)	$\underline{0.570 \pm 0.008}$	$\underline{0.427 \pm 0.011}$	$\underline{0.567 \pm 0.008}$

H.9 SEED-V (Emotion Recognition)

Table 18: SEED-V (5-class, emotion). External-baseline numbers from Zhou et al. [2025].

Method	Balanced Accuracy	Cohen's κ	Weighted F1
EEGNet	0.296 ± 0.010	0.101 ± 0.014	0.275 ± 0.010
EEGConformer	0.354 ± 0.011	0.177 ± 0.017	0.349 ± 0.014
SPaRCNet	0.295 ± 0.008	0.112 ± 0.014	0.298 ± 0.008
ContraWR	0.355 ± 0.011	0.191 ± 0.019	0.354 ± 0.012
CNN-Transformer	0.368 ± 0.008	0.207 ± 0.018	0.364 ± 0.009
FFCL	0.364 ± 0.009	0.208 ± 0.020	0.364 ± 0.013
ST-Transformer	0.305 ± 0.007	0.108 ± 0.012	0.283 ± 0.011
BIOT	0.384 ± 0.019	0.226 ± 0.026	0.386 ± 0.020
LaBraM-Base	0.398 ± 0.014	0.239 ± 0.021	0.397 ± 0.011
CBraMod	0.409 ± 0.010	0.257 ± 0.014	0.410 ± 0.011
REVE	0.405 ± 0.002	0.262 ± 0.002	0.414 ± 0.002
CSBrain	$\underline{0.420 \pm 0.003}$	$\underline{0.279 \pm 0.003}$	$\underline{0.428 \pm 0.002}$
B[FM]^2 (ours)	$\mathbf{0.492 \pm 0.009}$	$\mathbf{0.355 \pm 0.016}$	$\mathbf{0.468 \pm 0.019}$

I Compute Resources

Pretraining, finetuning, and sampling were run on a single NVIDIA H100 GPU with PyTorch 2.6.0 (CUDA 12.4) and Hugging Face Accelerate; data were served from LMDB on a shared parallel filesystem.

J Use of Existing Assets

Datasets. The Temple University Hospital EEG Corpus (TUEG) [Obeid and Picone, 2016] is available for research after registration at the TUH EEG portal. The downstream datasets, summarized per-dataset in Appendix C, are released under the following terms: **Siena** [Detti et al., 2020] and **MAT** [Zyma et al., 2019] are available under the Open Data Commons Attribution License via PhysioNet; **HMC** [Alvarez-Estevez and Rijsman, 2021] is provided under the Creative Commons Attribution 4.0 International license; **TUAB** and **TUEV** [Obeid and Picone, 2016] follow the TUH EEG terms, matching those of TUEG; **ISRUC** [Khalighi et al., 2016] and **SEED-V** [Liu et al., 2021] require a research-use agreement and registration; **Mumtaz** [Mumtaz et al., 2017] is available for research use

under terms listed on its repository; and **BCIC-IV-2a** [Tangermann et al., 2012] is released under the Creative Commons Attribution-NoDerivatives 4.0 International license.

Code and libraries. Our implementation builds on PyTorch (BSD-3), NumPy (NumPy License), SciPy (BSD-3, used for the Hungarian assignment via `scipy.optimize.linear_sum_assignment`), scikit-learn (BSD), and Hugging Face Accelerate (Apache 2.0). The training loop and EMA scaffolding are adapted from the `denoising-diffusion-pytorch` implementation[5] (MIT). MNE-Python (BSD) is used for EEG rendering in the neurologist-rating web interface.

External-baseline numbers. External-anchor numbers in Table 2 and the per-task tables in Appendix H are taken directly from El Ouahidi et al. [2025] and Zhou et al. [2025]. Both source papers evaluate the publicly released checkpoints for BIOT [Yang et al., 2023a], LaBraM [Jiang et al., 2024], CBraMod [Wang et al., 2025], and REVE [El Ouahidi et al., 2025]; the CSBrain paper additionally evaluates its own checkpoint. All baselines are reported on the same train/val/test splits our model uses, with the single ISRUC chin-EMG correction described in Appendix C.

K Broader Impacts

Generating a clinically indistinguishable EEG introduces significant dual-use implications. While high-fidelity synthetic data enables privacy-preserving sharing and robust downstream augmentation, it simultaneously lowers the barrier for fabricating medical records.

L LLM Usage

During the preparation of this manuscript, we used the Large Language Model (LLM) to assist with grammar checking, typo correction, and paraphrasing.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please refer to Abstract and Section 1.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to Section 4.

   Guidelines:

[5] `https://github.com/lucidrains/denoising-diffusion-pytorch`

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: We do not have a theoretical result.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please refer to Section 3.1, Appendix B, and the project page (`https://jd730.github.io/projects/BFM2`).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.

- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: All datasets are publicly available (see Appendix J), and the code will be released through the project page (`https://jd730.github.io/projects/BFM2`).

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: Please refer to Section 3.1, Appendix B, and the project page (`https://jd730.github.io/projects/BFM2`).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Please refer to Tables 2-18.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Please refer to Appendix I.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have reviewed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please refer to Appendix K.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our brain foundation model does not have a high risk for misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Please refer to Appendix J.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: There are no new assets in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Clinicians were only involved for rating EEG signals, not conducting any experiments as subjects. For the evaluation protocol, please refer to Appendix D.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Clinicians were only involved for rating EEG signals, not conducting any experiments as subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Please refer to Appendix L.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.