Unsupervised Alignment using Latent Dynamics in Spiking Neural Networks for Stable and Energy-Efficient iBCI Decoding

1st Boshi Zhao#
Beijing University of Technology
Beijing 100124, China
Center for Excellence in Brain
Science and Intelligence Technology,
State Key Laboratory of Brain
Cognition and Brain-inspired
Intelligence Technology,
Institute of Neuroscience,
Chinese Academy of Sciences,
Shanghai 200031, China
1934479130@qq.com

1st Liyuan Han#
Center for Excellence in Brain
Science and Intelligence Technology,
State Key Laboratory of Brain
Cognition and Brain-inspired
Intelligence Technology,
Institute of Neuroscience,
Chinese Academy of Sciences,
Shanghai 200031, China
hanliyuan@ion.ac.cn

3rd Zhengtuo Zhao
Center for Excellence in Brain
Science and Intelligence Technology,
State Key Laboratory of Brain
Cognition and Brain-inspired
Intelligence Technology,
Institute of Neuroscience,
Chinese Academy of Sciences,
Shanghai 200031, China
University of Chinese Academy
of Sciences, Beijing, 100049, China
zhaozt@ion.ac.cn

4th Xue Li*
Center for Excellence in Brain
Science and Intelligence Technology,
State Key Laboratory of Brain
Cognition and Brain-inspired
Intelligence Technology,
Institute of Neuroscience,
Chinese Academy of Sciences,
Shanghai 200031, China
University of Chinese Academy
of Sciences, Beijing, 100049, China
xli@ion.ac.cn

5th Kebin Jia*
Beijing University of Technology
Beijing 100124, China
kebinj@bjut.edu.cn

6th Tielin Zhang*
Center for Excellence in Brain
Science and Intelligence Technology,
State Key Laboratory of Brain
Cognition and Brain-inspired
Intelligence Technology,
Institute of Neuroscience,
Chinese Academy of Sciences,
Shanghai 200031, China
School of Artificial Intelligence,
University of Chinese Academy
of Sciences, Beijing 100049, China
The Key Laboratory of Cognition
and Decision Intelligence for Complex
Systems, Institute of Automation,
Chinese Academy of Sciences,
Beijing 100190, China
zhangtielin@ion.ac.cn

***Abstract*—Invasive brain–computer interface (iBCI) enables direct interaction between the brain and external devices by decoding neural signals. However, cross-day variations in neural signal distributions degrade decoding performance, requiring frequent recalibration with labeled data and limiting long-term usability. Moreover, traditional artificial neural networks (ANNs) used for decoding are energy-intensive, posing challenges for future edge computing and deployment in iBCI systems. To address these issues, we propose a stable and energy-efficient iBCI decoding framework based on a sequential variational autoencoder (seqVAE). Our approach combines multi-day pre-training with a lightweight bias-based unsupervised alignment module, enabling reliable cross-day generalization using only 60 s unlabeled neural data. We further develop an SNN-compatible implementation by incorporating LIF spiking neurons into all core modules to enable low-power inference. Evaluations on two non-human primate datasets show that our framework achieves more stable cross-day behavioral decoding than existing unsupervised baselines, and the spiking variant reduces the estimated computational energy by over 50%, highlighting its potential for future neuromorphic iBCI applications.**

***Index Terms*—Invasive brain–computer interface (iBCI), cross-day decoding, unsupervised alignment, spiking neural networks (SNNs).**

I. Introduction

iBCI decodes neural signals from the cerebral cortex for neuroprosthetics and motor rehabilitation. Despite advances in decoding algorithms and acquisition technologies [1]–[3],

long-term stability remains a critical challenge due to inter-day shifts in neural signal distributions caused by electrode drift, tissue response, and neural plasticity [4]–[8].

Supervised adaptation methods [1], [9]–[11] often rely on large pre-trained models and labeled data from new sessions, limiting real-world applicability. Unsupervised alignment approaches [12], [13] reduce labeling dependence but typically use energy-intensive ANNs, hindering deployment in power-constrained iBCI systems.

SNNs offer event-driven, low-power computation compatible with neuromorphic hardware [14], yet their potential for stable cross-day decoding remains unexplored.

We present a decoding framework that integrates unsupervised alignment and energy-aware design. Using multi-day pre-training and a lightweight bias-based alignment module, we achieve cross-day generalization with only 60 s of unlabeled data. The core modules are converted to SNNs with LIF neurons, enabling low-power inference. Evaluations on two non-human primate datasets show stable decoding and over 50% estimated energy reduction.

Our main contributions are:

- A lightweight unsupervised alignment framework based on seqVAE, enabling cross-day generalization with minimal unlabeled data.
- Validation on two non-human primate iBCI datasets, showing improved cross-day decoding over existing baselines.
- An SNN-compatible implementation of core modules with LIF neurons that preserves latent manifold continuity while reducing energy cost.
- Energy estimation demonstrating over 50% reduction, highlighting suitability for power-constrained iBCI systems.

II. Related works

A. Unsupervised Cross-day Decoding

Degenhart et al. explored cross-day neural decoding through a linear subspace alignment approach, demonstrating strong robustness to severe electrode drift in multiple experiments of non-human primate [15]. While effective, such linear manifold methods are limited in modeling complex neural dynamics. To capture richer temporal structures, LFADS (Latent Factor Analysis via Dynamical Systems) used RNN-based model to extract latent dynamics and outperform traditional factor models on both synthetic and real neural datasets [16]. Building on this, NoMAD proposed a cross-day alignment framework that leverages the latent dynamical distribution learned by LFADS to calibrate unseen neural data, achieving stable behavioral decoding across multiple weeks in non-human primate motor tasks [17]. Nevertheless, this approach depends on source-day recordings and abundant target-day neural data to estimate latent distributions, which constrains its applicability in low-data scenarios. AlignVAE further introduced a source-free alignment strategy, where an alignment network projects new-day neural recordings onto the latent dynamics learned by a pre-trained seqVAE. This approach improves the performance of the cross-day decoding without labeled data [18].

Although these methods improve cross-day decoding stability, their reliance on complex neural networks and high computational cost hinders practical deployment in embedded iBCI systems.

B. Spiking Neural Networks in iBCI

SNNs offer a biologically inspired alternative to conventional architectures through sparse and event based computation. By operating on discrete spikes, SNNs naturally match the asynchronous and sparse characteristics of neural signals and have shown promising performance in supervised neural classification tasks [19], [20]. However, their use in continuous decoding tasks, particularly under unsupervised or cross-day conditions, remains limited. In particular, the combination of SNNs with domain adaptation or alignment strategies has rarely been explored in the context of long-term iBCI use.

SNNs are often considered compatible with neuromorphic processors such as Loihi, TrueNorth, and SpiNNaker, which are designed for event-based and low-power computation [21]–[24]. While this work does not target a specific hardware platform, we estimate the energy consumption of our spiking models using typical operation level costs reported in prior literature, such as the energy per multiply accumulate (MAC) or accumulate (AC) operation. This analysis provides a model independent estimate of potential energy savings and highlights the applicability of SNNs for power aware BCI inference, even without physical deployment.

III. Proposed method

The overall architecture of our decoding framework is illustrated in Fig. 1. It consists of three main components: 1) a seqVAE trained on multi-day neural recordings to learn stable latent dynamics; 2) a bias-based alignment module that adjusts new-day neural spike inputs without behavioral labels; 3) a spiking implementation of the core modules for low-power, event-driven inference.

A. seqVAE: Variational Modeling of Latent Dynamics

We first train a seqVAE to extract stable latent neural dynamics from multi-day recordings, as illustrated in Fig. 1. The framework combines a Transformer encoder to capture long-range spike dependencies, an MLP prior for smooth latent dynamics, a GRU decoder for behavior prediction, and a linear reconstruction decoder that preserves neural information to support robust latent representation learning.

Here, $\mathbf{x}_{1:T}$ denotes the observed neural activity over a sample of length T, and z_t represents the latent state at time t. The seqVAE consists of an encoder $q_\phi(z_t|\mathbf{x}_{1:T})$ that approximates the posterior distribution, together with decoder and prior networks $p_\theta(\mathbf{x}_t|z_t)$ and $p_\theta(z_t|z_{t-1})$ that model spike reconstruction and latent temporal transitions, respectively.

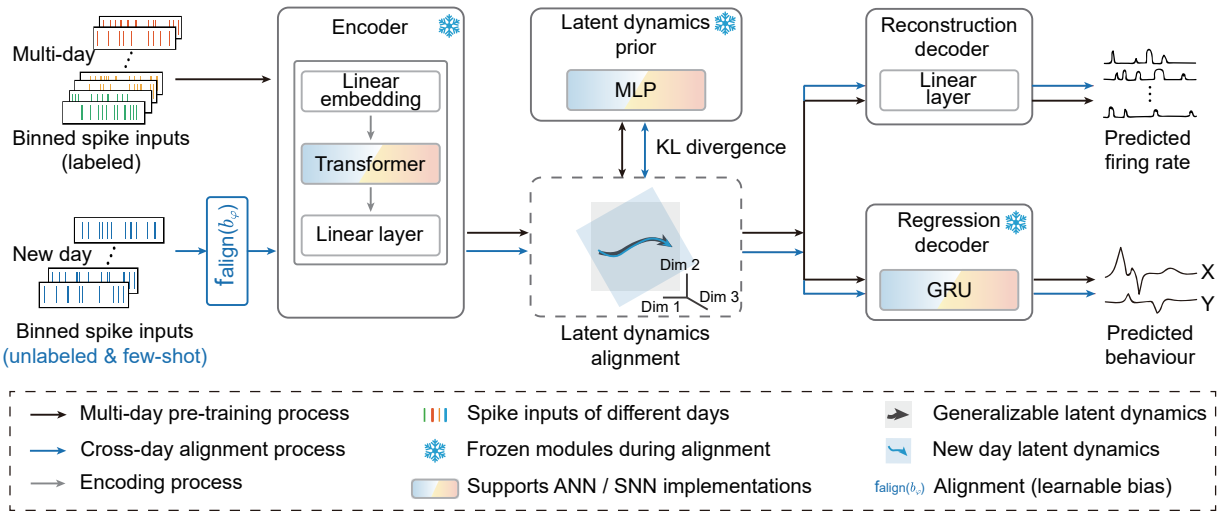

Fig. 1. Overview of the proposed iBCI decoding framework, which includes multi-day latent dynamics pre-training, unsupervised alignment with a learnable input bias, and spiking implementations of key modules within the seqVAE architecture. Black arrows indicate the multi-day pre-training flow, while blue arrows represent the alignment and inference path on new-day inputs.

The model is trained by maximizing the evidence lower bound (ELBO):

$$\begin{aligned}\mathcal{L}_{\text{ELBO}} = &\sum_{t=1}^{T} \mathbb{E}_{q_\phi(z_t|\mathbf{x}_{1:T})}\left[\log p_\theta(\mathbf{x}_t|z_t)\right] \\ &- \sum_{t=1}^{T} \text{KL}\left[q_\phi(z_t|\mathbf{x}_{1:T}) \,\|\, p_\theta(z_t|z_{t-1})\right]\end{aligned} \tag{1}$$

Both the latent prior $p_\theta(z_t|z_{t-1})$ and the posterior $q_\phi(z_t|\mathbf{x}_{1:T})$ are modeled as Gaussian distributions, with the initial latent state following a standard normal prior $p(z_1) = \mathcal{N}(0, I)$. To accurately characterize the statistics of spiking activity, the reconstruction likelihood $p_\theta(\mathbf{x}_t|z_t)$ is parameterized as a Poisson distribution.

To explicitly facilitate behavioral decoding, we introduce an additional supervised regression term, predicting behavior $\hat{y}_t$ from latent states z_t using mean-squared error, where y_t represents the observed behavior at time t:

$$\mathcal{L}_{\text{reg}} = \frac{1}{T}\sum_{t=1}^{T} \left|y_t - \hat{y}_t(z_t)\right|_2^2 \tag{2}$$

The final training loss combines these two terms with weight λ_1 and λ_2 to balance their contributions:

$$\mathcal{L}\text{train} = -\lambda_1 \mathcal{L}_{\text{ELBO}} + \lambda_2 \mathcal{L}_{\text{reg}} \tag{3}$$

Optimizing this objective yields a structured latent representation that captures cross-day neural dynamics while maintaining a stable behavioral decoder.

Notably, all key modules in our framework, including the Transformer encoder, MLP prior, GRU behavior decoder, can be converted to their SNN counterparts. This property enables subsequent spiking-domain fine-tuning and theoretical energy analysis.

B. Unsupervised Alignment using Latent Dynamics

To adapt the decoder to cross-day neural variations without labeled data, we propose a lightweight unsupervised alignment strategy, as illustrated in the bottom of Fig. 1. Specifically, we introduce a learnable bias term b_ψ that directly shifts new-day neural inputs:

$$f_{\text{align}}(\tilde{\mathbf{x}}_t) = \tilde{\mathbf{x}}_t + b_\psi \tag{4}$$

Aligned inputs are then passed through the frozen pre-trained encoder $q_\phi(\tilde{z}_t|f_{\text{align}}(\tilde{\mathbf{x}}_t))$.

To ensure temporal consistency, we align this latent distribution with the pre-trained latent dynamics prior $p_\theta(z_t|z_{t-1})$:

$$\mathcal{L}_{\text{KL}} = \sum_{t=1}^{T} \text{KL}\left[q_\phi(\tilde{z}_t|f_{\text{align}}(\tilde{\mathbf{x}}_t)) \,\|\, p_\theta(z_t|z_{t-1})\right] \tag{5}$$

We also impose a self-supervised reconstruction loss to preserve latent representation quality:

$$\mathcal{L}_{\text{rec}} = -\sum_{t=1}^{T} \log p_\theta(\tilde{\mathbf{x}}_t|\tilde{z}_t) \tag{6}$$

The alignment loss combines both terms with hyperparameter β_1 and β_2:

$$\mathcal{L}_{\text{align}} = -\beta_1 \mathcal{L}_{\text{KL}} + \beta_2 \mathcal{L}_{\text{rec}} \tag{7}$$

C. Spiking-Compatible Implementation and Energy Estimation

In the pre-training phase, we integrate Leaky Integrate-and-Fire (LIF) neurons into the key modules of the ANN model and perform the entire SNN conversion process using the SpikingJelly framework [25]. Specifically, we load the pre-trained ANN model weights into the corresponding SNN modules and fine-tune the neuron parameters using the original data. The fine-tuning process uses surrogate gradient methods to train

the neurons, adapting them to the SNN's characteristics [26]. Although the latent manifold remains continuous to preserve the variational inference structure, training in the spiking domain ensures that the decoder can be directly executed as an SNN during inference.

The membrane potential of an LIF neuron V_t is updated as:

$$V_t = \lambda V_{t-1} + I_t - S_{t-1} V_{\text{reset}} \tag{8}$$

where λ is the decay factor, I_t is the input current, and S_{t-1} indicates whether a spike was emitted at the previous time step. A spike is generated when V_t exceeds the threshold V_{th}:

$$S_t = \Theta(V_t - V_{\text{th}}) \tag{9}$$

where $\Theta(\cdot)$ is the Heaviside step function.

For energy estimation, we calculate the number of operations involved in the model inference, including multiply-accumulate operations (MACs) and accumulate operations (ACs). We estimate the energy consumption per operation, where the energy per MAC operation is 4.6 pJ and the energy per AC operation is 0.9 pJ [27]. With these theoretical values, we can estimate the total energy consumption during inference.

The total inference energy is divided into three parts: the alignment module, the Transformer encoder, and the GRU decoder:

$$E_{\text{total}} = E_{\text{align}} + E_{\text{encoder}} + E_{\text{decoder}} \tag{10}$$

1) ANN Energy Formulas: The energy consumption for ANN is calculated as:

$$E_{\text{ANN}} = N_{\text{MAC}} E_{\text{MAC}} + N_{\text{AC}} E_{\text{AC}} \tag{11}$$

where $E_{\text{MAC}} = 4.6\,\text{pJ}$ and $E_{\text{AC}} = 0.9\,\text{pJ}$ are the energy costs per multiply-accumulate and accumulate operation respectively, and N_{MAC}, N_{AC} represent the number of multiply-accumulate and accumulate operations respectively.

Alignment Module: The alignment module consists of a simple input bias component. The energy consumption of this module is given by:

$$E_{\text{align}} = N_{\text{align}} \cdot E_{\text{AC}} \tag{12}$$

where N_{align} represents the bias dimension of the alignment module.

Transformer Encoder: The encoder module consists of position encoding, linear projections, and stacked Transformer blocks. The total energy consumption of the encoder is composed of the following components:

$$\begin{cases} E^{\text{ac}}_{\text{proj}} = N^{\text{in}}_{\text{len}}\, N^{\text{trans}}_{\text{cha}}\, E_{\text{AC}} \\ E^{\text{mac}}_{\text{proj}} = N^{\text{in}}_{\text{len}}\, N^{\text{in}}_{\text{neu}}\, N^{\text{trans}}_{\text{cha}}\, E_{\text{MAC}} \\ E^{\text{ac}}_{\text{pos}} = N^{\text{in}}_{\text{len}}\, N^{\text{trans}}_{\text{cha}}\, E_{\text{AC}} \\ E^{\text{ac}}_{\text{trans}} = (7N^{\text{in}}_{\text{len}}\, N^{\text{trans}}_{\text{cha}} + N^{\text{in}}_{\text{len}}\, N^{\text{fd}}_{\text{cha}})\, E_{\text{AC}} \\ E^{\text{mac}}_{\text{trans}} = N_{\text{lay}}\big(4N^{\text{in}}_{\text{len}}(N^{\text{trans}}_{\text{cha}})^2 + 2(N^{\text{in}}_{\text{len}})^2 N^{\text{head}}_{\text{att}} \\ \qquad\qquad + 2N^{\text{in}}_{\text{len}} N^{\text{trans}}_{\text{cha}} N^{\text{fd}}_{\text{cha}}\big) E_{\text{MAC}} \\ E_{\text{encoder}} = E_{\text{proj}} + E_{\text{pos}} + E_{\text{trans}} \end{cases} \tag{13}$$

where $N^{\text{in}}_{\text{len}}$ denotes the length of the input sequence, $N^{\text{trans}}_{\text{cha}}$ represents the dimensionality of the model within each channel of the Transformer block, $N^{\text{in}}_{\text{neu}}$ indicates the dimension of the input neuron, $N^{\text{fd}}_{\text{cha}}$ refers to the channel dimension of the feed-forward layer, N_{lay} is the number of layers, and $N^{\text{head}}_{\text{att}}$ is the number of attention heads.

GRU Decoder: The decoder module consists of gated recurrent units (GRU) with gate mechanisms and an additional MLP layer for output projection. The energy consumption breakdown is as follows:

$$\begin{cases} E^{\text{ac}}_{\text{GRU}} = (5N^{\text{in}}_{\text{len}} N_{\text{lay}} N^{\text{hid}}_{\text{neu}} + N^{\text{out}}_{\text{neu}}) E_{\text{AC}} \\ E^{\text{mac}}_{\text{in}} = 3(N^{\text{in}}_{\text{neu}} + N^{\text{hid}}_{\text{neu}}) N^{\text{hid}}_{\text{neu}} E_{\text{MAC}} \\ E^{\text{mac}}_{\text{hid}} = 3(2N^{\text{hid}}_{\text{neu}}) N^{\text{hid}}_{\text{neu}} E_{\text{MAC}} \\ E_{\text{GRU}} = N^{\text{in}}_{\text{len}}[E^{\text{mac}}_{\text{in}} + (N_{\text{lay}} - 1) E^{\text{mac}}_{\text{hid}}] + E^{\text{ac}}_{\text{GRU}} \\ E_{\text{MLP}} = (N^{\text{in}}_{\text{neu}} N^{\text{hid}}_{\text{neu}} + N^{\text{hid}}_{\text{neu}} N^{\text{out}}_{\text{neu}}) E_{\text{MAC}} \\ \qquad\qquad + (N^{\text{hid}}_{\text{neu}} + N^{\text{out}}_{\text{neu}}) E_{\text{AC}} \\ E_{\text{decoder}} = E_{\text{GRU}} + E_{\text{MLP}} \end{cases} \tag{14}$$

where $N^{\text{hid}}_{\text{neu}}$ denotes the hidden neuron dimension, and $N^{\text{out}}_{\text{neu}}$ represents the output neuron dimension.

2) SNN Energy Formulas: In SNN, event-based computations are performed only during spike events, therefore compared to ANN, the cost in SNN is additionally multiplied by the value of duty cycle (DC). The energy consumption for SNN is calculated as: The energy consumption for SNN is calculated based on event-driven computation with sparsity consideration:

$$E_{\text{SNN}} = N_{\text{MAC}} \cdot E_{\text{MAC}} \cdot DC + N_{\text{AC}} \cdot E_{\text{AC}} \cdot DC \tag{15}$$

IV. Experiments

We evaluated the unsupervised cross-day generalization ability of our framework for both ANN and SNN models, as well as the theoretically estimated energy consumption, on two invasive BCI datasets. The results verify the effectiveness and potential application of our method.

A. Datasets

a) Center-out reaching task: Data were collected from Monkey D (Fig. 2a) while controlling a joystick to reach eight targets. Neural signals (64 channels, 30 kHz) and 2D joystick voltages (1 kHz) were binned into 20 ms windows. Trials were split 8:2 for training/testing, with 2 s samples (100 ms stride). Recordings spanned 04/01/2025–04/29/2025.

b) Isometric wrist task: Monkey J performed wrist force control (Fig. 3a). Neural activity (96-channel Utah array, 30 kHz) and cursor velocity (1 kHz) were aligned and binned into 20 ms windows. Trials were split 8:2, using 600 ms samples (120 ms stride). Recordings from 07/30/2015–09/06/2015 were used.

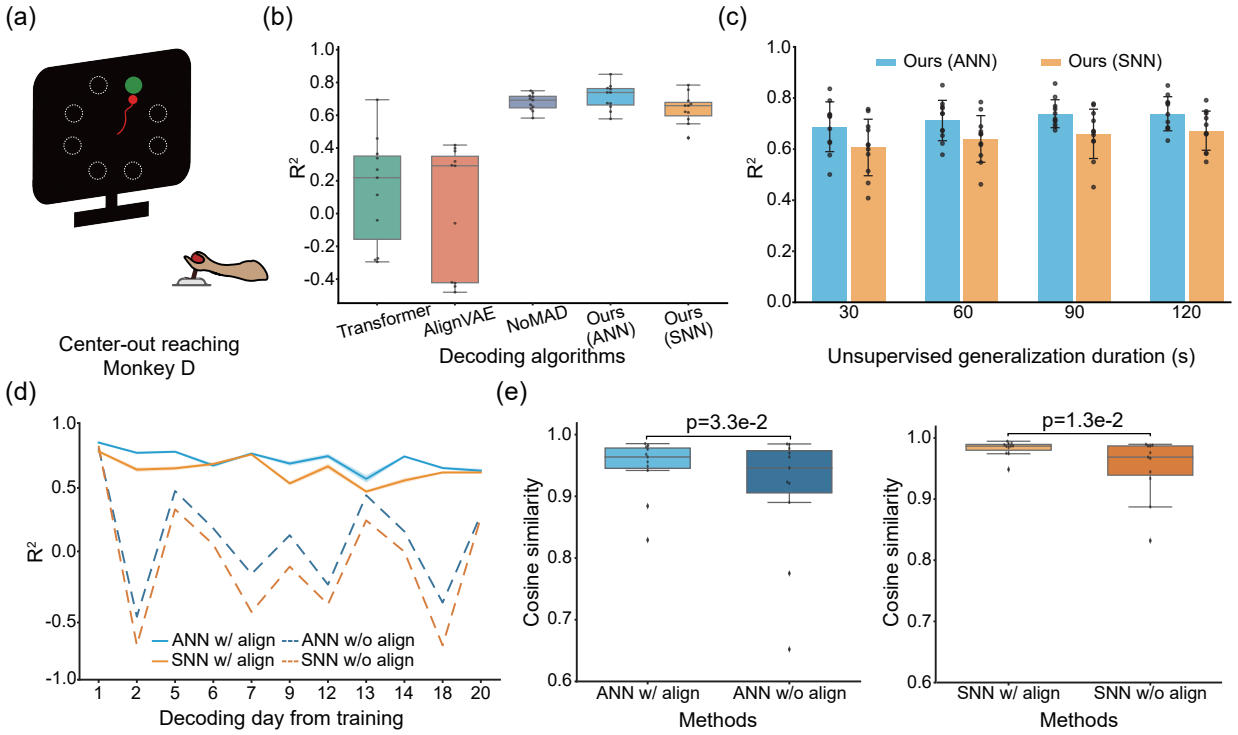

Fig. 2. Results on the center-out (CO) reaching task from Monkey D. (a) Experimental paradigm of the CO task performed by the Monkey. (b) Cross-day decoding accuracy across different methods, shown as box plots over target days. (c) Effect of unsupervised adaptation duration on decoding performance. (d) Decoding accuracy with (w/) and without (w/o) alignment for both ANN and SNN models. (e) Cosine similarity between the target-day latent representation and the pre-trained latent manifold, computed with and without alignment, for both ANN and SNN models.

B. Experimental Setup

We compared against three baselines: Vanilla Transformer [28] (no alignment), AlignVAE [18] (GRU-VAE with alignment), and NoMAD [17] (RNN latent dynamics with KL alignment). For fair comparison, all methods used approximately 120K parameters and the same 60 s of unlabeled target-day data.

Baselines were pre-trained on a single source day (04/09/2025 for Monkey D; 08/06/2015 for Monkey J). Our method used multi-day pre-training (5 days for Monkey D, 4 days for Monkey J) with AdamW (lr=5e-4, batch=128, 200 epochs). Loss weights: $\lambda_1 = 1\text{e}{-3}$ (ELBO), $\lambda_2 = 1$ (MSE). For alignment, we used lr=5e-3, 100 epochs, weight decay=5e-3, and $\beta_1 = 1\text{e}{-2}$, $\beta_2 = 1\text{e}{-3}$. SNN variants used the same settings with simulation windows of 10 (Monkey D) or 30 (Monkey J). Hyperparameters for baselines are in Table I.

TABLE I
HYPERPARAMETER CONFIGURATIONS FOR BASELINE MODELS USED IN COMPARATIVE ANALYSIS

Parameter	Transformer	AlignVAE	NoMAD
Learning rate	1×10^{-4}	5×10^{-4}	1×10^{-3}
Batch size	128	128	128
Training epochs	100	1000	100
Optimizer	Adam	AdamW	Adam
Loss	MSE	ELBO + NLL	NLL+MSE+KL

C. Cross-Day Generalization with Unsupervised Alignment

1) Baseline Comparison: As shown in Fig. 2 and Fig. 3 (b), we compared our proposed decoding framework with the aforementioned baselines across both datasets. Under the condition of using only 60 s of unlabeled neural data from each target day. Our method, which combining multi-day pre-training with a bias-based alignment strategy, achieves the highest cross-day decoding accuracy with ANN decoders, 0.71 ± 0.08 on Monkey D and 0.75 ± 0.04 on Monkey J.

NoMAD, performs competitively on some days but is generally less stable (0.68 ± 0.05 on Monkey D and 0.73 ± 0.02 on Monkey J). This is likely due to its reliance on a large amount of target day neural data to estimate latent distributions, which is insufficient under our limited data setting.

AlignVAE demonstrates sporadic generalization on Monkey D (0.03 ± 0.37) but fails to transfer on Monkey J (-0.04 ± 0.07). We attribute this to three key factors: (i) the limited representational capacity induced by single-day training; (ii) the weaker modeling power of the GRU encoder; and (iii) the difficulty of training a linear or MLP alignment head from only a few of unlabeled data, which may result in overfitting and misalignment of latent trajectories. In contrast, our multi-day pre-training strategy and lightweight bias-based alignment effectively address these limitations by enhancing manifold structure and reducing the complexity of alignment.

Although the overall accuracy is low (0.14 ± 0.32 on

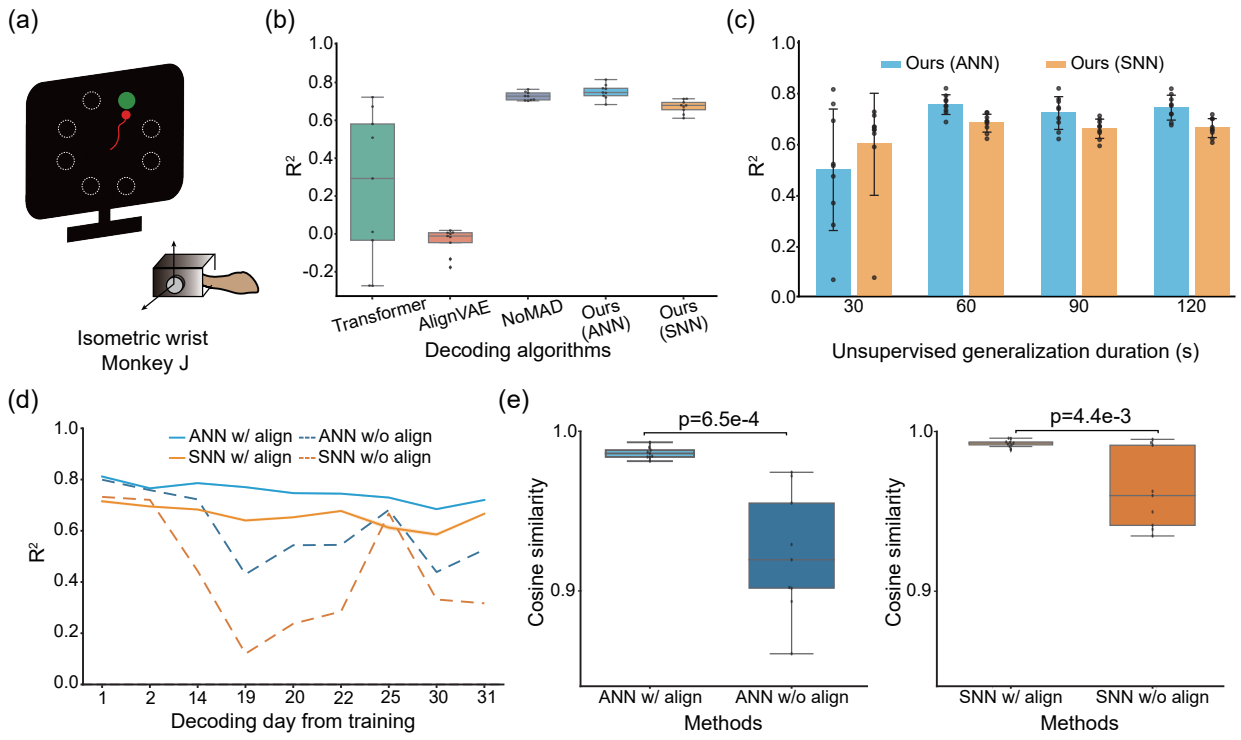

Fig. 3. Results on the isometric wrist task from Monkey J. (a) Experimental paradigm of the wrist force control task. (b) Cross-day decoding accuracy across different methods, shown as box plots over target sessions. (c) Effect of unsupervised adaptation duration on decoding performance. (d) Decoding accuracy before and after alignment for both ANNs and SNNs models. (e) Cosine similarity between the target-day latent representation and the pre-trained latent manifold, computed before and after alignment, for both ANNs and SNNs.

Monkey D; 0.25 ± 0.37 on Monkey J), the Vanilla Transformer baseline shows non-trivial generalization on certain days, suggesting inherent robustness and supporting its use as our backbone encoder.

In the SNN setting, our method continues to outperform both Transformer and AlignVAE. Although its decoding accuracy is moderately lower than ANN and NoMAD (0.64 ± 0.09 on Monkey D and 0.68 ± 0.03 on Monkey J), it achieves a trade-off between accuracy and energy efficiency, as shown in Fig. 4.

2) Effect of Alignment: We further investigated how the amount of unlabeled target day data affects the quality of alignment by varying adaptation durations from 30 to 120 s. As shown in Fig. 2 (c), on Monkey D, the accuracy of ANN improves from 0.69 ± 0.10 (30 s) to 0.74 ± 0.07 (120 s), while SNN improves from 0.61 ± 0.11 to 0.67 ± 0.08. On Monkey J (Fig. 3 (c)), ANN rises from 0.50 ± 0.24 (30 s) to 0.75 ± 0.04 (60 s), and SNN from 0.60 ± 0.20 to 0.68 ± 0.03. Although slight fluctuations occur beyond 60 s, performance remains consistently higher than the 30 s baseline, indicating that less than 60 seconds of data may be insufficient for stable alignment, while gains tend to saturate beyond that point.

Beyond average performance, we also examined decoding consistency across all target days. As shown in Fig. 2 (d) and Fig. 3 (d), our method maintains stable cross-day decoding (both ANN and SNN) across a 20-day span for Monkey D and a 30-day span for Monkey J. In contrast, removing the alignment module results in substantial variability and performance degradation, further underscoring the importance of alignment in maintaining temporal generalization.

To further understand the mechanism behind this improvement, we analyzed the cosine similarity between the target-day latent representations and the pre-trained manifold before and after alignment. On Monkey D (Fig. 2 (e)), the similarity increases from 0.91 ± 0.10 to 0.95 ± 0.05 for ANN, and from 0.95 ± 0.05 to 0.98 ± 0.01 for SNN. On Monkey J (Fig. 3 (e)), ANN improves from 0.92 ± 0.04 to 0.99 ± 0.003, and SNN from 0.96 ± 0.02 to 0.99 ± 0.002. Pair t-tests with one tail on all target days confirm that improvements are statistically significant ($p < 0.05$), indicating that our alignment module consistently restores the underlying latent structure, even with limited adaptation data.

3) Ablation Study: To further quantify the contributions of each component in our framework, we conducted an ablation study as summarized in Table II. The full model consistently achieves the best performance across both datasets and architectures, confirming the effectiveness of combining multi-day pret-raining with the bias-based alignment module. When the alignment module is removed, the model shows a clear drop in performance. For example, on Monkey D, the ANN accuracy decreases from 0.71 to 0.07, and the SNN accuracy drops from 0.64 to –0.07, indicating that pre-training alone is insufficient

TABLE II
ABLATION STUDY ON CENTER-OUT REACHING TASK AND ISOMETRIC WRIST TASK DATASETS FOR BOTH ANN AND SNN MODELS. WE REPORT AVERAGE R^2 DECODING ACCURACY ACROSS TARGET DAYS.

Dataset	Settings	ANN R^2 (↑)	SNN R^2 (↑)
Center-out reaching task Monkey D	**Full model (ours)**	**0.71 ± 0.08**	**0.64 ± 0.09**
	w/o align	0.07 ± 0.39	-0.07 ± 0.48
	w/o multi-day pre-training	0.66 ± 0.13	0.63 ± 0.10
	w/o both	0.00 ± 0.39	-0.19 ± 0.52
Isometric wrist task Monkey J	**Full model (ours)**	**0.75 ± 0.04**	**0.68 ± 0.03**
	w/o align	0.61 ± 0.13	0.43 ± 0.22
	w/o multi-day pre-training	0.54 ± 0.13	0.42 ± 0.12
	w/o both	0.08 ± 0.42	-0.14 ± 0.44

to address the larger distribution shifts present in this dataset. In contrast, on Monkey J, the model still achieves moderate generalization without alignment (e.g., 0.54 with ANN), suggesting that multi-day pre-training alone can provide a strong prior when the target-day distribution is relatively stable. Nevertheless, adding alignment consistently improves performance across both datasets and architectures. When both components are removed, the model fails completely. For instance, the SNN on Monkey J drops to –0.14. These results demonstrate that multi-day pre-training and alignment play complementary roles. Pre-training helps establish a stable and generalizable latent dynamics across days, while alignment enables efficient unsupervised adaptation to target day distribution shifts using only a few of unlabeled neural data.

D. Energy Efficiency of SNN

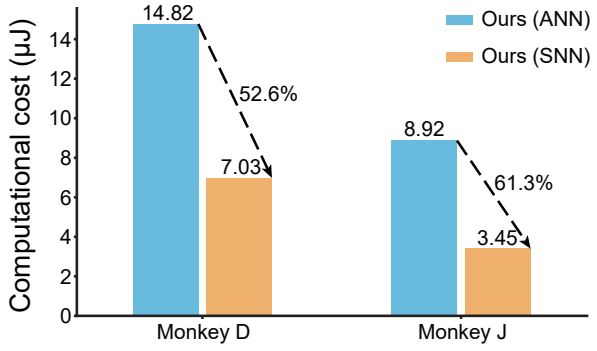

Fig. 4. Energy comparison between ours ANN and SNN.

In addition to the accuracy of decoding, we also analyzed the energy efficiency of our framework. Fig. 4 shows the theoretical energy consumption per inference sample for ANN and SNN implementations. On Monkey D, the SNN version reduces the computational cost from 14.82 μJ (ANN) to 7.03 μJ, resulting in a 52.6% reduction. On Monkey J, the energy consumption decreases from 8.92 μJ to 3.45 μJ, corresponding to a reduction of 61.3%. While the SNN decoder exhibits slightly lower accuracy than its ANN counterpart, it provides substantial energy savings, making it a compelling choice for real-time deployment in power-constrained edge iBCI systems.

V. CONCLUSION

We propose a fully unsupervised framework for cross-day decoding in invasive brain–computer interfaces. It integrates multi-day pretraining of latent neural dynamics with a lightweight bias-based alignment module to enable stable generalization without behavioral labels. Experiments on two non-human primate datasets show that our method consistently outperforms existing ANN baselines and maintains competitive accuracy under spiking neural network architectures. Ablation studies confirm the complementary roles of pretraining and alignment, while cosine similarity analysis verifies that alignment effectively restores latent structure. Theoretical energy analysis further indicates that the SNN variant reduces computational cost by over 50% compared to the ANN version, highlighting its potential for deployment on low-power neuromorphic hardware.

Despite promising results under realistic cross-day conditions, our framework has limitations. The current additive alignment module may be suboptimal under large or non-linear distribution shifts. Our evaluation focuses on short-to-medium timescales (within 30 days); validation over longer periods and with more diverse decoding targets remains for future work. Moreover, energy estimates are theoretical, and real-time deployment on neuromorphic hardware requires further verification. It should be noted that current neuromorphic platforms such as Loihi and SpiNNaker do not natively support several key operators used in our model (e.g., Transformer attention, GRU gating, and variational sampling), making direct deployment infeasible without structural adaptation.

To address these operator-level constraints, we have designed a custom manycore neuromorphic processor, currently under fabrication. We plan to evaluate its on-chip implementation and real-time decoding performance once the hardware is available [29].

ACKNOWLEDMENT

This work was supported by the Brain Science and Brain-like Intelligence Technology-National Science and Technology Major Project (2025ZD0217200), Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDB1010302), CAS Project for Young Scientists in Basic Research (YSBR-116), Youth Innovation

Promotion Association CAS, Shanghai Leading Talent Program of Eastern Talent Plan, the Lingang Laboratory Fund (Grant No. LG-GG-202402-06-07, LGL-1987-09), the Shanghai Municipal Science and Technology Project (Grant No. 25ZR1401370, 25LN3200400), Special Support Project of Guangdong Province (Grant No. 0720240209). The numerical calculations in this study were carried out on the ORISE Supercomputer. Author contributions: T.Z. designed the study. T.Z. performed the discussion and wrote the paper. Competing interests: The authors declare that they have no competing interests.

REFERENCES

[1] J. Ye, F. Rizzoglio, A. Smoulder, H. Mao, X. Ma, P. Marino, R. Chowdhury, D. Moore, G. Blumenthal, W. Hockeimer *et al.*, "A generalist intracortical motor decoder," *bioRxiv*, 2025.

[2] Z. Zhao, H. Zhu, X. Li, L. Sun, F. He, J. E. Chung, D. F. Liu, L. Frank, L. Luan, and C. Xie, "Ultraflexible electrode arrays for months-long high-density electrophysiological mapping of thousands of neurons in rodents," *Nature biomedical engineering*, vol. 7, no. 4, pp. 520–532, 2023.

[3] Y. Fang, W. Jinfen, G. Shouliang, M. Du, and L. Zou, "Implantable flexible neural microelectrode comb, and preparation method and implantation method therefor," Jun. 28 2022, uS Patent 11,369,302.

[4] J. E. Chung, K. K. Sellers, M. K. Leonard, L. Gwilliams, D. Xu, M. E. Dougherty, V. Kharazia, S. L. Metzger, M. Welkenhuysen, B. Dutta *et al.*, "High-density single-unit human cortical recordings using the neuropixels probe," *Neuron*, vol. 110, no. 15, pp. 2409–2421, 2022.

[5] K. Woeppel, C. Hughes, A. J. Herrera, J. R. Eles, E. C. Tyler-Kabara, R. A. Gaunt, J. L. Collinger, and X. T. Cui, "Explant analysis of utah electrode arrays implanted in human cortex for brain-computer-interfaces," *Frontiers in Bioengineering and Biotechnology*, vol. 9, p. 759711, 2021.

[6] J. W. Salatino, K. A. Ludwig, T. D. Kozai, and E. K. Purcell, "Glial responses to implanted electrodes in the brain," *Nature biomedical engineering*, vol. 1, no. 11, pp. 862–877, 2017.

[7] M. E. Franklin, C. Bennett, M. Arboite, A. Alvarez-Ciara, N. Corrales, J. Verdelus, W. D. Dietrich, R. W. Keane, J. P. de Rivero Vaccari, and A. Prasad, "Activation of inflammasomes and their effects on neuroinflammation at the microelectrode-tissue interface in intracortical implants," *Biomaterials*, vol. 297, p. 122102, 2023.

[8] M. Grosse-Wentrup, D. Mattia, and K. Oweiss, "Using brain–computer interfaces to induce neural plasticity and restore function," *Journal of neural engineering*, vol. 8, no. 2, p. 025004, 2011.

[9] M. Azabou, V. Arora, V. Ganesh, X. Mao, S. Nachimuthu, M. Mendelson, B. Richards, M. Perich, G. Lajoie, and E. Dyer, "A unified, scalable framework for neural population decoding," *Advances in Neural Information Processing Systems*, vol. 36, pp. 44 937–44 956, 2023.

[10] J. Ye, J. Collinger, L. Wehbe, and R. Gaunt, "Neural data transformer 2: multi-context pretraining for neural spiking activity," *Advances in Neural Information Processing Systems*, vol. 36, pp. 80 352–80 374, 2023.

[11] A. Antoniades, Y. Yu, J. Canzano, W. Wang, and S. L. Smith, "Neuroformer: Multimodal and multitask generative pretraining for brain data," *arXiv preprint arXiv:2311.00136*, 2023.

[12] A. Farshchian, J. A. Gallego, J. P. Cohen, Y. Bengio, L. E. Miller, and S. A. Solla, "Adversarial domain adaptation for stable brain-machine interfaces," *arXiv preprint arXiv:1810.00045*, 2018.

[13] X. Ma, F. Rizzoglio, K. L. Bodkin, E. Perreault, L. E. Miller, and A. Kennedy, "Using adversarial networks to extend brain computer interface decoding accuracy over time," *elife*, vol. 12, p. e84296, 2023.

[14] Q. T. Pham, T. Q. Nguyen, P. C. Hoang, Q. H. Dang, D. M. Nguyen, and H. H. Nguyen, "A review of snn implementation on fpga," in *2021 international conference on multimedia analysis and pattern recognition (MAPR)*. IEEE, 2021, pp. 1–6.

[15] A. D. Degenhart, W. E. Bishop, E. R. Oby, E. C. Tyler-Kabara, S. M. Chase, A. P. Batista, and B. M. Yu, "Stabilization of a brain–computer interface via the alignment of low-dimensional spaces of neural activity," *Nature biomedical engineering*, vol. 4, no. 7, pp. 672–685, 2020.

[16] D. Sussillo, R. Jozefowicz, L. Abbott, and C. Pandarinath, "Lfads-latent factor analysis via dynamical systems," *arXiv preprint arXiv:1608.06315*, 2016.

[17] B. M. Karpowicz, Y. H. Ali, L. N. Wimalasena, A. R. Sedler, M. R. Keshtkaran, K. Bodkin, X. Ma, D. B. Rubin, Z. M. Williams, S. S. Cash *et al.*, "Stabilizing brain-computer interfaces through alignment of latent dynamics," *Nature Communications*, vol. 16, no. 1, pp. 1–17, 2025.

[18] A. Vermani, I. M. Park, and J. Nassar, "Leveraging generative models for unsupervised alignment of neural time series data," in *The Twelfth International Conference on Learning Representations*, 2023.

[19] Y. Song, L. Han, T. Zhang, and B. Xu, "Multiscale fusion enhanced spiking neural network for invasive bci neural signal decoding," *Frontiers in Neuroscience*, vol. 19, p. 1551656, 2025.

[20] Q. Wei, L. Han, and T. Zhang, "Learning and controlling multiscale dynamics in spiking neural networks using recursive least square modifications," *IEEE Transactions on Cybernetics*, vol. 54, no. 8, pp. 4603–4616, 2024.

[21] G. Indiveri, B. Linares-Barranco, T. J. Hamilton, A. v. Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Häfliger, S. Renaud *et al.*, "Neuromorphic silicon neuron circuits," *Frontiers in neuroscience*, vol. 5, p. 73, 2011.

[22] M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain *et al.*, "Loihi: A neuromorphic manycore processor with on-chip learning," *Ieee Micro*, vol. 38, no. 1, pp. 82–99, 2018.

[23] F. Akopyan, J. Sawada, A. S. Cassidy, R. Alvarez-Icaza, J. V. Arthur, P. A. Merolla, N. Imam, Y. Nakamura, P. Datta, G.-J. Nam *et al.*, "A 20x22 core configuration of a 1 million neuron programmable neurosynaptic chip using the truenorth architecture," *IEEE Journal of Solid-State Circuits*, vol. 50, no. 1, pp. 123–135, 2015.

[24] S. B. Furber, F. Galluppi, S. Temple, and L. A. Plana, "The spinnaker project," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 652–665, 2014.

[25] W. Fang, Y. Chen, J. Ding, Z. Yu, T. Masquelier, D. Chen, L. Huang, H. Zhou, G. Li, and Y. Tian, "Spikingjelly: An open-source machine learning infrastructure platform for spike-based intelligence," *Science Advances*, vol. 9, no. 40, p. eadi1480, 2023.

[26] E. O. Neftci, H. Mostafa, and F. Zenke, "Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 51–63, 2019.

[27] M. Horowitz, "1.1 computing's energy problem (and what we can do about it)," in *2014 IEEE international solid-state circuits conference digest of technical papers (ISSCC)*. IEEE, 2014, pp. 10–14.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[29] Q. Li, Y. Song, X. Liu, W. Song, B. Zhao, Z. Wang, A. Chen, T. Zhang, and L. Chen, "Taibai: A fully programmable brain-inspired processor with topology-aware efficiency," *arXiv preprint arXiv:2508.18961*, 2025.