DETECT-THEN-ADAPT: CONTINUOUS-STREAM NEURAL DECODING FOR ONLINE ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Intracortical brain-computer interfaces (iBCIs) hold great promise for restoring motor and communication abilities in individuals with paralysis. However, the inherent nonstationarity of neural systems poses a major challenge to the long-term stability of iBCI systems. To avoid frequent recalibration, existing approaches typically apply predefined domain adaptation techniques to adjust the neural decoder prior to each use, yet they fail to consider when and how the adaptation should be performed. As a result, decoder performance can vary substantially under different neural signal conditions. In this work, we propose a *Detect-Then-Adapt* framework for long-term iBCI operation. The framework first *detects* whether decoder re-adaptation is needed based on continuous neural input, and then automatically constructs a suitable adapter tailored to the current state of neural signals when adaptation is required. Experiments on multi-session motor recordings from two non-human primates demonstrate that continuous neural activity provides informative cues for both detecting and readapting to neural signal drifts. Our method consistently outperforms domain-adaptation and neural-alignment baselines, highlighting the potential of leveraging continuous neural streams as a practical foundation for reliable, long-term iBCI adaptation.

1 INTRODUCTION

Intracortical brain-computer interfaces (iBCIs) translate neural activity into movement and offer a promising way to restore motor and speech function in people with paralysis (Willett et al., 2021; Metzger et al., 2023; Willett et al., 2023; Wairagkar et al., 2025; Littlejohn et al., 2025; Card et al., 2024). However, changes in both electrode recordings and neural activity continuously alter the mapping between the neural signals and behavior, causing a decoder trained on source sessions to degrade in later sessions (Sussillo et al., 2016; Degenhart et al., 2020). Given the significant burden of repeatedly performing recalibration for the target sessions, achieving reliable cross-session adaptation without target labels has become a key issue in long-term iBCIs (Wilson et al., 2025).

Existing methods enhance cross-session neural decoding mainly by improving adaptation to newly recorded neural activity. Sussillo *et al.* introduce historical neural data during the training stage to enable the decoder to adapt in advance to possible recording changes (Sussillo et al., 2016); While more methods choose to perform unsupervised alignment of neural representations in new sessions, including matching low-dimensional neural manifolds (Dyer et al., 2017; Degenhart et al., 2020), learning session-invariant representations (Farshchian et al., 2019; Ma et al., 2023; Jude et al., 2022), and using a relatively stable potential neural dynamics constraint objective representation (Karpowicz et al., 2025); Furthermore, recent methods recalibrate the decoder based on inferred latent motion intentions (Wilson et al., 2025). However, these methods primarily react to neural activity observed during target task execution and generally lack mechanisms for determining whether an observed change has actually impaired the source decoder. In clinical applications, drift may occur before or between tasks, which may lead to a sudden drop in decoding performance at the task beginning.

A central difficulty is that neural nonstationarity is not simply a discrete difference between source and target sessions. In reality, neural states may continuously evolve before, during, and between tasks. Current research indicates that neural activity and its relationship with behavior may accumulate and change over time, leading to a gradual mismatch in the decoder before recalibration is performed (Wilson et al., 2025; Pun et al., 2024). The resulting continuous neural stream may

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

therefore provide intermediate evidence about how the current neural state departs from the source condition. Importantly, even minor changes in neural activity can alter the observed neural distribution, yet still maintain sufficient compatibility between the behaviorally relevant structures and the decoder (Jude et al., 2022; Karpowicz et al., 2025). Conversely, transient decoded anomalies may reflect neural noise rather than persistent model drift. Therefore, determining *when* to recalibrate requires complementary evidence that neural variation is present and that it propagates through the source decoder into incompatible output dynamics. Moreover, unconstrained adaptation may remove useful source structure or overreact to unlabeled observations. Determining *how* to recalibrate thus requires balancing target plasticity against preservation of the reliable source solution.

Based on this principle, we propose a *Detect-Then-Adapt* framework for neural decoder adaptation. In the detection stage, neural-space deviations measure changes in recorded activity and latent structure, while decoding-space deviations determine whether these changes disrupt source-compatible output dynamics. Their complementary evidence can identify functionally relevant drift and determine when adaptation is needed. In the adaptation stage, the method constructs a conservative stable path and a source-constrained plastic path. A reliability-aware gate controls the contribution of the plastic path based on its source compatibility and adaptation consistency, thereby determining the degree to which the decoder should adapt. Experiments on multi-session motor recordings from two nonhuman primates (O'Doherty et al., 2020) show that observed unlabeled neural streams provide informative evidence for detecting and recalibrating functional drift. The proposed method outperforms the representative domain adaptation and dynamic alignment methods in cross-session decoding. These results suggest that reliable long-term iBCIs require not only adapting to a target session, but also detecting whether and how strongly recalibration is needed from the neural activity.

2 Related Work

Cross-Session Neural Decoding. Cross-session neural decoding methods aim to preserve decoding performance under changes in neural recordings. One direction improves robustness before deployment by training on long-term recordings or synthetic neural perturbations (Sussillo et al., 2016). Another direction learns behaviorally relevant representations that are less sensitive to session identity. For example, adversarial approaches suppress session-specific information while retaining movement-related features (Farshchian et al., 2019; Ma et al., 2023), and SABLE combines sequential latent-variable modeling with domain adaptation to learn representations shared across recording sessions (Jude et al., 2022). Unsupervised domain adaptation methods further align target-session activity with source distributions or low-dimensional neural manifolds (Dyer et al., 2017; Degenhart et al., 2020). More recently, NoMAD aligns new recordings with latent neural dynamics learned from historical sessions, allowing a fixed behavioral decoder to operate on the adapted representation (Karpowicz et al., 2025). These methods can construct stable and reliable source domain representations, or adapt to the task-related target domain neural signals. However, they usually treat each target session as an independent domain and conduct task-related adaptation. In contrast, our method views the non-stationarity of neural signals as a continuous change process. It uses unlabeled neural data streams to recalibrate the decoder before and during task execution.

3 Methodology

3.1 Problem Setting and Method Overview

We consider cross-session neural decoding from one labeled source session to a later, unlabeled target session. The source data are:

$$\mathcal{S}_s = (\mathbf{X}_s, \mathbf{Y}_s), \qquad \mathbf{X}_s \in \mathbb{R}^{T_s \times C}, \quad \mathbf{Y}_s \in \mathbb{R}^{T_s \times d_y}, \tag{1}$$

where C is the number of neural channels and $\mathbf{Y}_s$ contains the corresponding kinematics. At recalibration time, we observe only a continuous target spike stream $\mathbf{X}_t^{\mathrm{cal}}$. This stream may contain pre-task, inter-task, and task-period activity, but we do not use target kinematics or task/rest markers.

Without target labels, a flexible adapter may match source statistics yet degrade decoding accuracy. We therefore retain two target predictions. A *stable* path applies only channel-wise moment correction, whereas a *plastic* path adds a learned but bounded temporal residual. A session-level gate then

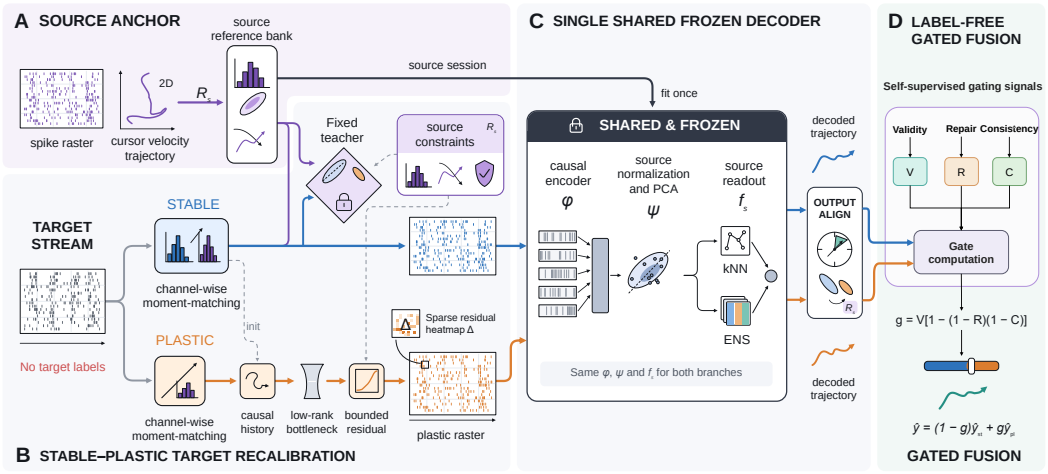


Figure 1: Overview of our decoding framework. **(A)** Source data establish a reference bank for anchoring target adaptation. **(B)** Stable moment matching and source-constrained plastic residual correction recalibrate target neural activity without target labels. **(C)** Both branches share a frozen source-trained decoder, followed by output alignment. **(D)** Label-free validity, repair, and consistency signals gate the fusion of their predictions.

determines how much of the plastic proposal is admitted:

$$\widehat{\mathbf{Y}}_t = (1-g)\widetilde{\mathbf{Y}}_{\mathrm{st}} + g\widetilde{\mathbf{Y}}_{\mathrm{pl}}, \qquad 0 \le g \le 1. \tag{2}$$

The complete procedure follows the visual order of the main figure: (i) fit one source backbone and cache one source reference bank, (ii) construct the Stable and Plastic target candidates, (iii) optimize only the Plastic candidate under the fixed source references, (iv) pass both candidates through the same frozen backbone and output alignment, and (v) compute g from label-free source-compatibility tests. Target kinematics are accessed only after these steps for offline evaluation.

3.2 Frozen Source Model and Reference Statistics

Source Representation and Readout. A causal feature extractor ϕ concatenates averages over several windows of the current and preceding neural activity. Source standardization followed by PCA maps these features to a source latent space:

$$\mathbf{z} = \psi(\phi(\mathbf{x})) = \left[\frac{\phi(\mathbf{x}) - \boldsymbol{\mu}_{\phi,s}}{\boldsymbol{\sigma}_{\phi,s} + \epsilon} - \mathbf{m}_s\right]\mathbf{P}_s^{\top}. \tag{3}$$

All quantities in ψ are fitted on the source session and are subsequently frozen. The final source readout combines a local RBF-weighted nearest-neighbor regressor with a kernel ensemble,

$$f_s(\mathbf{z}) = (1-\omega)f_{\mathrm{knn}}(\mathbf{z}) + \omega f_{\mathrm{ens}}(\mathbf{z}). \tag{4}$$

Because this readout is not fully differentiable, we additionally fit a ridge proxy $q_s(\mathbf{z}) = \mathbf{W}_q\mathbf{z} + \mathbf{b}_q$. The proxy supplies gradients during adaptation; all reported predictions are produced by the frozen readout f_s. We use $E_s = \psi \circ \phi$ to denote the frozen causal encoder and latent projection. The pair (E_s, f_s) is the single-source backbone shown in the main figure: it is fitted once on the source session and reused by both target paths.

Source-Compatible References. The source session also supplies the references used to train and assess the target adapter. From the continuous source stream, we record channel moments $(\boldsymbol{\mu}_{x,s}, \boldsymbol{\sigma}_{x,s})$, source latent moments and scale $(\boldsymbol{\mu}_{z,s}, \boldsymbol{\Sigma}_{z,s}, \boldsymbol{\sigma}_{z,s})$, and the source-stream predictions $\widehat{\mathbf{U}}_s$. From $\widehat{\mathbf{U}}_s$ we retain its moments, the moments of its first differences, normalized zero- and one-lag covariances $(\mathbf{C}_{0,s}, \mathbf{C}_{1,s})$, and state-conditioned output moments used for alignment and gate diagnostics. We denote this single frozen collection by $\mathcal{R}_s$ (written as R_s in the main figure). It describes plausible source-scale activity and output dynamics; it does not prescribe a particular target trajectory or act as a decoder.

For compactness, define a standardized moment discrepancy for any $\mathbf{U} \in \mathbb{R}^{N\times d}$:

$$\mathcal{M}(\mathbf{U};\boldsymbol{\mu},\boldsymbol{\sigma}) = \frac{1}{d}\left\|\frac{\mathrm{mean}(\mathbf{U})-\boldsymbol{\mu}}{\boldsymbol{\sigma}+\epsilon}\right\|_2^2 + \frac{1}{2d}\left\|\frac{\mathrm{std}(\mathbf{U})-\boldsymbol{\sigma}}{\boldsymbol{\sigma}+\epsilon}\right\|_2^2. \tag{5}$$

This single discrepancy will be reused in neural, decoded-output, and dynamic spaces.

3.3 Stable and Plastic Target Paths

Stable Path. Let $(\boldsymbol{\mu}_{x,t}, \boldsymbol{\sigma}_{x,t})$ be channel-wise moments estimated from the unlabeled target stream, and let Π denote element-wise clipping to the valid spike-count range. The stable adapter is:

$$A_{\mathrm{st}}(\mathbf{x}) = \Pi\left[\boldsymbol{\mu}_{x,s} + \frac{\boldsymbol{\sigma}_{x,s}+\epsilon}{\boldsymbol{\sigma}_{x,t}+\epsilon}\odot(\mathbf{x}-\boldsymbol{\mu}_{x,t})\right]. \tag{6}$$

Before clipping, this transform exactly matches the first two channel-wise moments. It neither mixes channels nor introduces new temporal interactions, which makes it a conservative reference path.

Plastic Path. Moment correction alone cannot represent state-dependent or cross-channel drift. The plastic adapter therefore augments a trainable diagonal transform with a low-rank causal residual. Let $\mathbf{h}_t$ concatenate recent windowed neural activity and let $\overline{\mathbf{h}}_t$ be its target-normalized, clipped version. We write:

$$\begin{aligned} A_{\mathrm{diag}}(\mathbf{x}_t) &= \Pi\left[\exp(\boldsymbol{\ell})\odot\mathbf{x}_t + \boldsymbol{\beta}\right], \\ \mathbf{r}_\theta(t) &= \rho\,\mathbf{s}_s \odot \tanh(\overline{\mathbf{h}}_t\mathbf{D}\mathbf{U}), \\ A_\theta(\mathbf{x}_t) &= \Pi\left[A_{\mathrm{diag}}(\mathbf{x}_t) + \mathbf{r}_\theta(t)\right]. \end{aligned} \tag{7}$$

Here $\mathbf{D} \in \mathbb{R}^{d_h\times r}$ and $\mathbf{U} \in \mathbb{R}^{r\times C}$ restrict the residual to rank r; $\mathbf{s}_s$ is a clipped source channel scale; and ρ bounds the residual amplitude. The log-gain is itself bounded in implementation. The plastic path is initialized from the stable path,

$$\boldsymbol{\ell} \leftarrow \log\left([\mathbf{a}]_{a_{\min}}^{a_{\max}}\right), \qquad \boldsymbol{\beta} \leftarrow \mathbf{b}, \qquad \mathbf{U} \leftarrow \mathbf{0}, \tag{8}$$

where $\mathbf{a}$ and $\mathbf{b}$ are the scale and bias implicit in equation 6, and the brackets enforce the plastic gain range. The other low-rank factor $\mathbf{D}$ is randomly initialized. Consequently, the residual is initially zero, the diagonal component starts from the closest admissible stable transform, and gradients can immediately update $\mathbf{U}$. Both paths remain aligned to the same target time axis: the Stable path changes channel-wise moments, whereas the Plastic path retains that correction and can add only a bounded, context-dependent residual.

3.4 Plastic Training with a Fixed Teacher

This subsection corresponds to the training-only teacher and constraint capsule in the main figure. During target calibration, E_s, f_s, and $\mathcal{R}_s$ remain frozen; only the Plastic parameters θ are updated. The Stable path is never optimized. The plastic adapter is fitted on target stream blocks $\{\mathbf{B}_j\}_{j=1}^{J}$. Its objective has three roles: matching source-compatible statistics, preserving the target structure already captured by the stable path, and limiting the size of the learned correction.

Fixed Latent Teacher. We first pass stable-adapted target blocks through the frozen encoder E_s, obtaining $\mathbf{Z}_{\mathrm{st}}$. A shrinkage-CORAL transform moves their aggregate covariance toward that of the continuous source stream. With target and source latent moments $(\boldsymbol{\mu}_{z,t}, \widetilde{\boldsymbol{\Sigma}}_{z,t})$ and $(\boldsymbol{\mu}_{z,s}, \widetilde{\boldsymbol{\Sigma}}_{z,s})$, respectively, define:

$$\mathbf{T}_z = \widetilde{\boldsymbol{\Sigma}}_{z,t}^{-1/2}\widetilde{\boldsymbol{\Sigma}}_{z,s}^{1/2}, \qquad \mathbf{Z}^{\mathrm{tea}} = \mathbf{Z}_{\mathrm{st}} + \eta_z\left[(\mathbf{Z}_{\mathrm{st}}-\boldsymbol{\mu}_{z,t})\mathbf{T}_z + \boldsymbol{\mu}_{z,s} - \mathbf{Z}_{\mathrm{st}}\right]. \tag{9}$$

The corresponding rows are stored for every target block and detached before optimization. The teacher therefore supplies a fixed, source-oriented structure rather than a moving self-training target.

For the current plastic adapter, let:

$$\mathbf{Z}_{\theta,j} = E_s(A_\theta(\mathbf{B}_j)), \qquad \widehat{\mathbf{U}}_\theta = q_s(\mathbf{Z}_\theta), \tag{10}$$

where $\mathbf{Z}_\theta$ concatenates the block-wise latent features. In the figure, the resulting constraints are summarized by moments, teacher consistency, decoded dynamics, and a trust-region shield; the following three groups give their mathematical form.

First, distribution matching constrains raw neural activity, decoded output, and decoded first differences:

$$\begin{aligned}\mathcal{L}_{\text{match}} = \lambda_{\text{str}}\mathcal{M}\big(A_\theta(\mathbf{X}_t^{\text{cal}}); \boldsymbol{\mu}_{x,s}, \boldsymbol{\sigma}_{x,s}\big) \\ + \lambda_{\text{pri}}\mathcal{M}\Big(\widehat{\mathbf{U}}_\theta; \boldsymbol{\mu}_{\hat{y},s}, \boldsymbol{\sigma}_{\hat{y},s}\Big) \\ + \lambda_{\text{dyn}}\mathcal{M}\Big(\Delta\widehat{\mathbf{U}}_\theta; \boldsymbol{\mu}_{\Delta\hat{y},s}, \boldsymbol{\sigma}_{\Delta\hat{y},s}\Big)\,.\end{aligned} \tag{11}$$

Second, structure preservation combines the fixed teacher with decoded trajectory geometry:

$$\begin{aligned}\mathcal{L}_{\text{tea}} &= \frac{1}{J}\sum_{j=1}^{J}\text{MSE}\left(\frac{\mathbf{Z}_{\theta,j} - \mathbf{Z}_j^{\text{tea}}}{\boldsymbol{\sigma}_{z,s} + \epsilon}\right), \\ \mathcal{L}_{\text{geo}} &= \frac{1}{2}\sum_{\ell\in\{0,1\}}\text{MSE}\Big(\mathbf{C}_\ell(\widehat{\mathbf{U}}_\theta) - \mathbf{C}_{\ell,s}\Big), \\ \mathcal{L}_{\text{pres}} &= \lambda_{\text{tea}}\mathcal{L}_{\text{tea}} + \lambda_{\text{geo}}\mathcal{L}_{\text{geo}}.\end{aligned} \tag{12}$$

Here $\mathbf{C}_0$ and $\mathbf{C}_1$ are normalized zero-lag and one-lag covariances. Thus, the teacher preserves sample-level latent structure, while the geometry term preserves coarse decoded dynamics.

Finally, a trust-region penalty discourages the residual from replacing the source decoder with an unconstrained target-specific mapping:

$$\begin{aligned}\mathcal{L}_{\text{tru}} &= \mathbb{E}_{\mathbf{x}\sim\mathbf{X}_t^{\text{sample}}}\left[\frac{1}{C}\left\|\frac{\mathbf{r}_\theta(\mathbf{x})}{\mathbf{s}_{\text{str}}}\right\|_2^2\right] + \frac{1}{J}\sum_{j=1}^{J}\mathbb{E}_{\mathbf{x}\sim\mathbf{B}_j}\left[\frac{1}{C}\left\|\frac{\mathbf{r}_\theta(\mathbf{x})}{\mathbf{s}_{\text{str}}}\right\|_2^2\right], \\ \mathcal{L}_{\text{par}} &= \text{mean}(\mathbf{D}^{\odot 2}) + \text{mean}(\mathbf{U}^{\odot 2}), \qquad \mathcal{L}_{\text{reg}} = \lambda_{\text{tru}}\mathcal{L}_{\text{tru}} + \lambda_{\text{par}}\mathcal{L}_{\text{par}}.\end{aligned} \tag{13}$$

Thus $\mathcal{L}_{\text{tru}}$ checks both sampled stream rows and the selected contiguous target blocks. The complete target objective is simply:

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{match}} + \mathcal{L}_{\text{pres}} + \mathcal{L}_{\text{reg}}. \tag{14}$$

3.5 Shared Frozen Backbone and Output Alignment

After adaptation, both paths enter the same physical source backbone used in Panel A. There are not separate Stable and Plastic decoders:

$$\widehat{\mathbf{Y}}_p^0 = f_s(E_s(A_p(\mathbf{X}_t)))\,, \qquad p \in \{\text{st}, \text{pl}\}. \tag{15}$$

The shared backbone contains the causal feature extractor, source normalization/PCA, and the fixed kNN–ensemble readout. Thus the Stable and Plastic paths differ only in their neural input transform; the decoder itself is identical and frozen.

Residual scale and covariance mismatch can remain in decoded space. We correct it using one output-alignment rule applied to both colored paths, with path-specific stream statistics. The rule is a state-conditioned output CORAL transform estimated from the continuous decoded target stream. States are inferred from the predictions themselves: the lowest-speed bin forms one state, while the remaining speed bins are subdivided by movement direction. For path p and state k, let $(\boldsymbol{\mu}_{p,k}, \widetilde{\boldsymbol{\Sigma}}_{p,k})$ and $(\boldsymbol{\mu}_{s,k}, \widetilde{\boldsymbol{\Sigma}}_{s,k})$ be the target-path and source-stream output statistics. Then:

$$\mathbf{T}_{p,k} = \widetilde{\boldsymbol{\Sigma}}_{p,k}^{-1/2}\widetilde{\boldsymbol{\Sigma}}_{s,k}^{1/2}, \qquad O_p(\mathbf{y}) = \mathbf{y} + \eta_o\left[(\mathbf{y} - \boldsymbol{\mu}_{p,k})\mathbf{T}_{p,k} + \boldsymbol{\mu}_{s,k} - \mathbf{y}\right]. \tag{16}$$

Shrinkage covariances are used throughout, and a global transform replaces a state-specific transform when either stream has too few samples in that state. The aligned predictions are $\widetilde{\mathbf{Y}}_p = O_p(\widehat{\mathbf{Y}}_p^0)$. The same O_p is applied to the full decoded target stream used by the gate. No target kinematics are needed to infer the states or estimate these transforms.

3.6 Label-Free Stable–Plastic Gate

Low adaptation loss does not by itself imply better unknown target accuracy. The gate therefore asks three separate questions: Is the plastic output source-compatible? Does it repair a mismatch in the stable path? Did the plastic adapter remain close to its fixed teacher and residual budget?

Let $D_{\mathrm{pri}}(p)$ be the moment discrepancy in equation 5 between the aligned stream output of path p and the source-stream output. Let $D_{\mathrm{state}}(p)$ be the analogous sample-weighted discrepancy computed within source-defined speed bins. Smaller values indicate greater source compatibility. We use the overflow and relative repair functions:

$$\Omega(d;\tau) = \left[\frac{d}{\tau} - 1\right]_+, \qquad \Gamma(d_0, d_1) = \left[\frac{d_0 - d_1}{d_0 + \epsilon}\right]_0^1, \tag{17}$$

where $[\cdot]_+$ is the positive part and $[\cdot]_0^1$ clips to $[0, 1]$.

Validity. The plastic path is valid when its decoded prior remains inside the source-compatible corridor:

$$V = \exp\left\{-\Omega(D_{\mathrm{pri}}(\mathrm{pl});\tau_{\mathrm{pri}})^2\right\}. \tag{18}$$

Thus $V = 1$ inside the corridor and decays smoothly beyond it.

Repair. Plasticity is more useful when the stable path lies outside the corridor and the plastic path reduces that mismatch. We combine global and speed-conditioned repair in one score:

$$R = \max\left\{\Gamma(D_{\mathrm{state}}(\mathrm{st}), D_{\mathrm{state}}(\mathrm{pl})),\ \sqrt{[\Omega(D_{\mathrm{pri}}(\mathrm{st});\tau_{\mathrm{pri}})]_0^1\,\Gamma(D_{\mathrm{pri}}(\mathrm{st}), D_{\mathrm{pri}}(\mathrm{pl}))}\right\}. \tag{19}$$

The global term is activated only when the stable prior exceeds its permitted range; state-conditioned improvement can independently provide evidence of repair.

Consistency. Teacher and residual losses directly measure whether optimization respected its two main safeguards. Their normalized penalties and consistency score are:

$$P_k = \left[\frac{\mathcal{L}_k - \tau_k}{w_k}\right]_0^1, \quad k \in \{\mathrm{tea}, \mathrm{tru}\}, \qquad C = 1 - \max(P_{\mathrm{tea}}, P_{\mathrm{tru}}). \tag{20}$$

The final gate is the soft logical rule:

$$g = \left[V\big(1 - (1-R)(1-C)\big)\right]_0^1. \tag{21}$$

Validity acts as a prerequisite, whereas $1-(1-R)(1-C)$ is a soft OR between demonstrated repair and consistency with the bounded adaptation process. The resulting scalar g is computed once per target session and is used in equation 2 for all predictions from that session.

For interpretability, we additionally report a broader microdrift audit based on decoded-prior and decoded-dynamics mismatch, neural variance expansion, population covariance, eigenspectrum deviation, and lost-channel count. This audit explains which aspects of a target session leave the source-compatible corridor. In the default triad gate, however, it is diagnostic only and does not introduce an additional gating factor.

3.7 Default Configuration and Evaluation Protocol

The causal feature extractor uses five windows spanning the current bin through the preceding 60 bins. The source PCA has 32 components, the nearest-neighbor readout uses 20 neighbors, and the ensemble contribution in equation 4 is $\omega = 0.45$. The plastic residual has rank 64 and is optimized for 40 Adam steps with learning rate 0.01. The loss weights $(\lambda_{\mathrm{str}}, \lambda_{\mathrm{tea}}, \lambda_{\mathrm{pri}}, \lambda_{\mathrm{dyn}}, \lambda_{\mathrm{geo}}, \lambda_{\mathrm{tru}}, \lambda_{\mathrm{par}})$ are $(0.35, 0.30, 0.15, 0.08, 0.02, 0.12, 10^{-4})$. Output alignment uses four speed bins, four direction bins for moving states, and interpolation strength $\eta_o = 0.25$; latent teacher construction uses $\eta_z = 1.25$. The default gate uses $\tau_{\mathrm{pri}} = 0.01$, $(\tau_{\mathrm{tea}}, w_{\mathrm{tea}}) = (0.05, 0.03)$, and $(\tau_{\mathrm{tru}}, w_{\mathrm{tru}}) = (0.015, 0.03)$. These thresholds are fixed before target evaluation and are not selected from target kinematics. For each animal-array chain, the earliest session is the labeled source, and all later sessions are targets. Source reach annotations and kinematics train the source representation and readout. Every target-side quantity-moment initialization, teacher construction, adapter optimization, output alignment, and gate computation uses only the continuous target spike stream. Target reach kinematics are loaded exclusively to form the offline test set and report decoding accuracy.

More details of the method are provided in Appendix A.

Table 1: *One2All* cross-session decoding ($100R^2$, higher is better). **Bold** and underline denote best and second-best means. '†' indicates no significant difference from Ours.

Method	Indy-96	Indy-192	Loco-192	Avg.
KF	-70.8 ± 0.0	-70.0 ± 0.0	-275.9 ± 0.0	-112.6 ± 0.0
LSTM	7.1 ± 0.6	19.1 ± 3.7	-2.5 ± 2.2	6.8 ± 0.9
Transformer	13.1 ± 1.8	17.2 ± 5.7	-21.9 ± 10.6	6.5 ± 2.9
CORAL	38.5 ± 0.2	54.0 ± 0.3	9.4 ± 1.0	34.6 ± 0.3
MMD	39.2 ± 0.4	54.0 ± 0.5	9.8 ± 0.7	35.2 ± 0.3
DANN	39.8 ± 1.5	54.2 ± 0.7†	5.4 ± 0.9	34.7 ± 1.1
NoMAD	38.9 ± 0.8	**56.4 ± 0.8**†	5.9 ± 3.9	34.5 ± 0.9
CEBRA	39.5 ± 0.3	51.5 ± 0.4	12.4 ± 0.7	35.6 ± 0.3
Ours	**41.3 ± 0.1**	55.8 ± 0.2	**18.2 ± 0.1**	**38.6 ± 0.1**

4 Experiments and Results

4.1 Experimental Setup

Datasets. We use the public reaching dataset from two primates, Indy and Loco (O'Doherty et al., 2020), comprising continuous neural recordings and movement kinematics from 96-channel M1 or 192-channel M1/S1 arrays. The recordings form three channel-consistent sequences: Indy-96, Indy-192, and Loco-192, with 29, 6, and 9 target sessions, respectively.

Baselines. We compare against source-only decoders KF, LSTM (Hochreiter, 1997), and Transformer (Vaswani et al., 2017); domain-adaptation methods CORAL (Sun et al., 2016), MMD (Gretton et al., 2012), and DANN (Ganin et al., 2016); and neural representation and dynamics methods CEBRA (Schneider et al., 2023) and NoMAD (Karpowicz et al., 2025). All methods share source-target splits and protocol-specific target-data access.

Training and Evaluation Details. We use a 32-dimensional source representation and a rank-64 plastic adapter, optimized for 40 Adam steps at a learning rate of 0.01. Remaining loss weights, alignment strengths, and gate thresholds are selected using source-derived pseudo-target streams and fixed before target evaluation. We report $100R^2$ (higher is better) as mean ± sample standard deviation over five matched independent runs, with two-sided paired t-tests. Full preprocessing, baseline, optimization, and statistical details appear in Appendix B.

4.2 Cross-Session Decoding

Main Comparison. To assess transfer without target labels, we evaluate *One2All* decoding from each sequence's earliest labeled session to all subsequent target sessions. For each column in Table 1, we first average the relevant session scores within each run, then report the mean and sample standard deviation across five runs; Avg. weights all 44 target sessions equally. Our method achieves the highest overall score of **38.6±0.1**, exceeding CEBRA by 3.0 points on the $100R^2$ scale, with a particularly strong 5.8-point gain on Loco-192. Although NoMAD obtains the highest mean on the smaller Indy-192 sequence, our method ranks second and remains competitive in this setting. Taken together, these results show that the proposed method provides the strongest aggregate generalization across subjects, channel configurations, and heterogeneous target sessions.

Days-Apart. Neural drift is a continuously evolving process, and the degree of mismatch between a fixed source decoder and a target session may change substantially with the temporal relationship between their recordings. To examine transfer to progressively later recordings, we keep the earliest session as the sole training source and evaluate sessions beyond chronological cutoffs of 20%, 40%, 60%, and 80%. Consequently, increasing r evaluates the same fixed source decoder on more distant target sessions, which is usually more challenging. Notably, the relative ranking of the competing methods changes: CEBRA performs better than NoMAD at $r = 0.2$ and $r = 0.4$, whereas NoMAD is better when $r > 0.4$. This crossover suggests that these methods have different strengths under different degrees or forms of long-term neural drift, and neither provides uniformly reliable adaptation across the entire temporal range. In contrast, our method leads at every cutoff in Figure 2(a), achieving an average score of **35.5±0.1**. Its sustained advantage across temporal splits highlights the ability of constrained plastic adaptation to accommodate evolving cross-session mismatch.

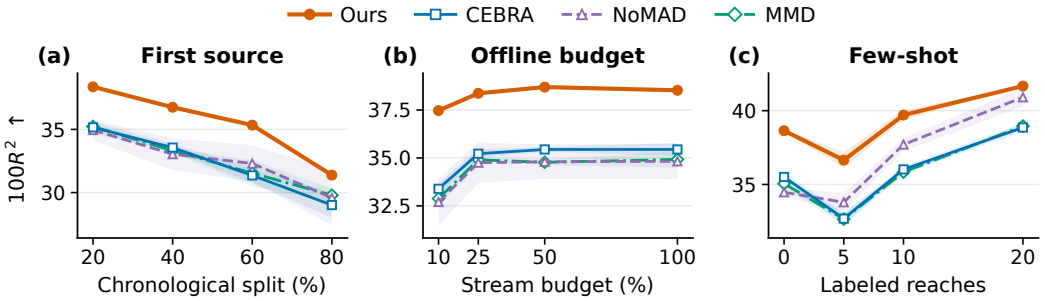


Figure 2: Cross-session transfer and calibration budgets. **(a)** First-source transfer across chronological splits. **(b)** Offline adaptation with varying unlabeled stream budgets. **(c)** Few-shot output calibration. Shading indicates one sample standard deviation across five runs.

Offline Stream-Budget. In practice, recalibration should be completed before the actual target session, and its utility depends on whether informative drift evidence can be extracted from a short, unlabeled neural stream. To measure the amount of unlabeled activity required for recalibration, we adapt using the first 10%, 25%, 50%, or 100% of each target stream and evaluate all target reaches. As shown in Figure 2(b), our method leads across budgets, reaching 37.5±0.1 with only 10% of the stream and 38.7±0.1 with 50%. Performance remains near this level with the full stream, indicating that limited unlabeled observations already provide useful source-compatible correction.

Few-Shot Calibration. To examine how limited supervision can further enhance adaptation, we apply the same affine output calibration to all methods using $K \in \{0, 5, 10, 20\}$ labeled target reaches and evaluate on the remaining reaches. Figure 2(c) shows that our method reaches 39.71±0.34 and 41.67±0.02 with 10 and 20 labeled reaches, respectively, with mean scores 1.1 and 3.0 points above its unsupervised setting. These results suggest that a small calibration set can complement unsupervised adaptation once sufficient labeled data are available.

Complete results, protocol details, and statistical comparisons are provided in Appendix C.

4.3 Ablation Study

To examine how target plasticity, source guidance, and adaptive prediction contribute to recalibration, we conduct ablations on the 29 target sessions of Indy-96. Figure 3 compares the full model with three variants: removing the plastic path, removing the fixed teacher, and replacing adaptive fusion with a fixed equal weight. Each point is one session's performance averaged over five independent runs.

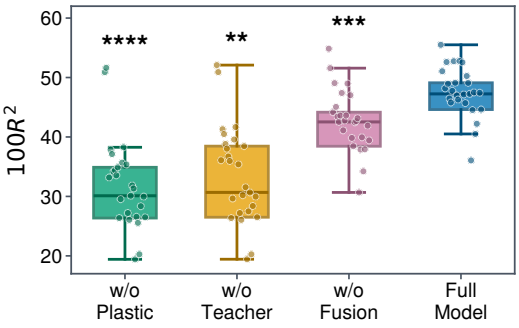


Figure 3: Ablation study. Significance is measured against the full model using two-sided paired t-tests: $^{**}p < 0.01$, $^{***}p < 0.001$, and $^{****}p < 0.0001$ (n=29).

The full model improves over these variants by **12.9**, **11.4**, and **3.7** points on the $100R^2$ scale, respectively. The largest gain comes from introducing plastic adaptation beyond the stable reference, highlighting the importance of learning target-specific corrections to accommodate cross-session drift. The substantial contribution of the fixed teacher further shows the value of guiding these corrections toward source-compatible latent structure. Adaptive fusion also improves over equal-weight averaging, supporting session-dependent control of the plastic prediction's contribution. Together, these results highlight the complementary roles of plasticity, structural guidance, and reliability-aware weighting in effective neural recalibration. Complete ablations and statistical details are provided in Appendix C.7.

4.4 Adaptation and Representation Analyses

Stabilized Adaptation. To examine adaptation performance across recording days, we evaluate transfer from a fixed source session to nine Loco-192 target sessions recorded 3-20 days later, recal-

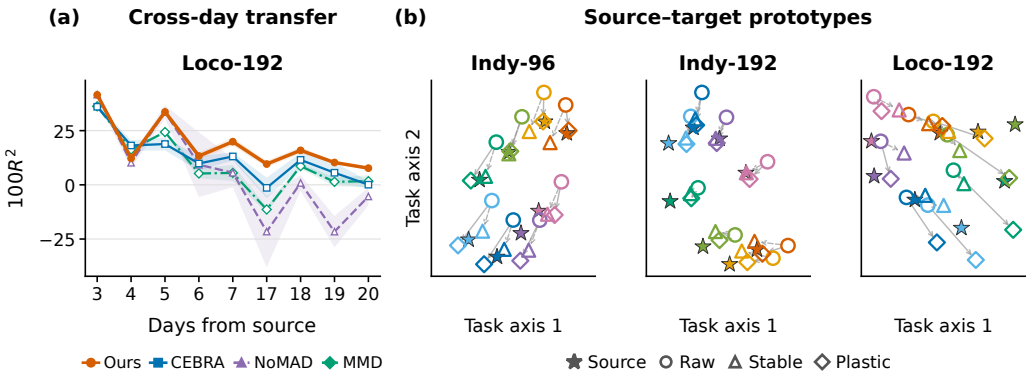


Figure 4: Stable cross-day performance and representation analysis. **(a)** Loco decoding across nine target recording days, 3-20 days after the source recording. **(b)** Matched movement-direction prototypes from fixed examples in three recording chains. Colors identify matched direction groups.

ibrating each session using its unlabeled neural stream. Figure 4(a) reports daily means and standard deviations across five independent runs. Our method maintains positive R^2 throughout the evaluated period, with a particularly clear advantage on days 17-20, where it exceeds the strongest baseline at each recording day by **4.4-11.0** points on the $100R^2$ scale. These results highlight the ability of source-constrained adaptation to preserve useful decoding performance as recording conditions evolve, including later sessions with substantial cross-day non-stationarity.

Representation Alignment. To understand the complementary roles of the two adaptation paths, we visualize eight movement-direction prototypes from fixed examples in three recording chains. Within each chain, all representations share a frozen source-defined task projection, allowing direct comparison of matched prototypes in Figure 4(b). Both Stable and Plastic paths bring target prototypes closer to their source counterparts on average, while exhibiting different correction behaviors. Stable provides a relatively restrained channel-wise adjustment, whereas Plastic introduces context-dependent corrections that achieve stronger overall alignment. This distinction is particularly visible in Loco-192, where Plastic reduces the mean source-prototype distance from 0.850 for Stable to 0.558, drawing several displaced direction groups closer to their source locations. Individual prototypes further reveal the value of retaining both paths. For the blue directions in Indy-96, the Stable triangle approaches the source star, while the Plastic moves beyond it, illustrating how a stronger correction can overshoot an already nearby reference. These examples motivate reliability-aware output fusion: Stable supplies a restrained reference, Plastic provides greater adaptation capacity, and the gate regulates their contributions to the final prediction. The resulting design combines the flexibility of target-specific correction with the stability of a source-anchored alternative.

5 Conclusion and Limitations

Conclusion. We presented a *Detect-Then-Adapt* framework for adaptation from unlabeled neural streams. Rather than treating all observed neural changes as drift evidence, the proposed method evaluates variations in both neural and decoded spaces, constructs complementary stable and plastic predictions through a shared frozen decoding stack, and adaptively combines them according to source-session reliability. Experiments across multiple cross-session settings show that the method consistently improves recalibration across recording configurations, temporal separations, and limited stream budgets. Ablations highlight the contributions of the plastic path, fixed teacher, and adaptive fusion. Adaptation and representation analyses further illustrate sustained transfer performance and complementary correction behaviors, supporting the value of balancing flexible adaptation with source-anchored stability. Together, these findings highlight the value of source-constrained plasticity and reliability-guided fusion for balancing target-specific adaptation with stable neural decoding.

Limitations. Our evaluation focuses on primate reaching recordings; extending the framework to human intracortical BCIs and other behavioral tasks is an important next step. The current gate assigns one fusion weight per session, motivating finer-grained reliability estimation for rapid within-session changes. Source-derived references and adaptation constraints also offer opportunities for adaptive reference updating and objective simplification. Future work will explore these directions alongside broader clinical evaluation and end-to-end real-time deployment.

AI USE STATEMENT

In this work, we used generative AI tools to manuscript polishing, including improving grammar, clarity, and presentation. We have not used generative AI tools to generate results, conduct experiments, analyze data, or produce scientific claims. All AI-assisted text was reviewed and revised by the authors, who verified its correctness and consistency with the underlying work. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work develops methods for reliable neural decoder recalibration, with the long-term goal of supporting assistive motor BCIs. Our experiments use publicly available nonhuman-primate reaching recordings (O'Doherty et al., 2020) and consist of secondary computational analyses. The original dataset documentation describes the recording procedures and data-collection context. Future translation to clinical BCIs should prioritize neural-data privacy, informed consent, user agency, and systematic safety evaluation, particularly when adaptive updates affect decoder behavior.

REPRODUCIBILITY STATEMENT

Section 3 specifies the stable and plastic paths, fixed-teacher adaptation objectives, shared frozen decoder, and reliability-guided fusion. Section 4 describes the datasets, comparison methods, evaluation protocols, and performance metrics. Appendix A details causal preprocessing, source references, target sampling, optimization, and the recalibration algorithm. Appendix C provides extended comparative results, ablations, robustness evaluations, and statistical procedures, including the hyperparameter analyses in Appendix C.10.

REFERENCES

Nicholas S. Card, Maitreyee Wairagkar, Carrina Iacobacci, Xianda Hou, Tyler Singer-Clark, Francis R. Willett, Erin M. Kunz, Chaofei Fan, Maryam Vahdati Nia, Darrel R. Deo, Aparna Srinivasan, Eun Young Choi, Matthew F. Glasser, Leigh R. Hochberg, Jaimie M. Henderson, Kiarash Shahlaie, Sergey D. Stavisky, and David M. Brandman. An accurate and rapidly calibrating speech neuroprosthesis. *New England Journal of Medicine*, 391(7):609–618, 2024. doi: 10.1056/NEJMoa2314132. URL https://doi.org/10.1056/NEJMoa2314132.

Alan D. Degenhart, William E. Bishop, Emily R. Oby, Elizabeth C. Tyler-Kabara, Steven M. Chase, Aaron P. Batista, and Byron M. Yu. Stabilization of a brain–computer interface via the alignment of low-dimensional spaces of neural activity. *Nature Biomedical Engineering*, 4(7):672–685, 2020. doi: 10.1038/s41551-020-0542-9.

Eva L. Dyer, Mohammad Gheshlaghi Azar, Matthew G. Perich, Hugo L. Fernandes, Stephanie Naufel, Lee E. Miller, and Konrad P. Körding. A cryptography-based approach for movement decoding. *Nature Biomedical Engineering*, 1(12):967–976, 2017. doi: 10.1038/s41551-017-0169-7.

Ali Farshchian, Juan Álvaro Gallego, Joseph Paul Cohen, Yoshua Bengio, Lee E. Miller, and Sara A. Solla. Adversarial domain adaptation for stable brain–machine interfaces. In *7th International Conference on Learning Representations, ICLR 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=Hyx6Bi0qYm.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016. URL http://jmlr.org/papers/v17/15-239.html.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. URL https://www.jmlr.org/papers/v13/gretton12a.html.

S Hochreiter. Long short-term memory. *Neural Computation MIT-Press*, 1997.

Justin Jude, Matthew Perich, Lee E. Miller, and Matthias H. Hennig. Robust alignment of cross-session recordings of neural population activity by behaviour via unsupervised domain adaptation. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 10462–10475. PMLR, 2022.

Brianna M. Karpowicz, Yahia H. Ali, Lahiru N. Wimalasena, Andrew R. Sedler, Mohammad Reza Keshtkaran, Kevin Bodkin, Xuan Ma, Daniel B. Rubin, Ziv M. Williams, Sydney S. Cash, Leigh R. Hochberg, Lee E. Miller, and Chethan Pandarinath. Stabilizing brain–computer interfaces through alignment of latent dynamics. *Nature Communications*, 16(1):4662, 2025. doi: 10.1038/s41467-025-59652-y.

Kaylo T. Littlejohn, Cheol Jun Cho, Jessie R. Liu, Alexander B. Silva, Bohan Yu, Vanessa R. Anderson, Cady M. Kurtz-Miott, Samantha Brosler, Anshul P. Kashyap, Irina P. Hallinan, Adit Shah, Adelyn Tu-Chan, Karunesh Ganguly, David A. Moses, Edward F. Chang, and Gopala K. Anumanchipalli. A streaming brain-to-voice neuroprosthesis to restore naturalistic communication. *Nature Neuroscience*, 28(4):902–912, 2025. doi: 10.1038/s41593-025-01905-6. URL https://doi.org/10.1038/s41593-025-01905-6.

Xuan Ma, Fabio Rizzoglio, Kevin L. Bodkin, Eric Perreault, Lee E. Miller, and Ann Kennedy. Using adversarial networks to extend brain computer interface decoding accuracy over time. *eLife*, 12: e84296, 2023. doi: 10.7554/eLife.84296.

Sean L. Metzger, Kaylo T. Littlejohn, Alexander B. Silva, David A. Moses, Margaret P. Seaton, Ran Wang, Maximilian E. Dougherty, Jessie R. Liu, Peter Wu, Michael A. Berger, Inga Zhuravleva, Adelyn Tu-Chan, Karunesh Ganguly, Gopala K. Anumanchipalli, and Edward F. Chang. A high-performance neuroprosthesis for speech decoding and avatar control. *Nature*, 620(7976): 1037–1046, 2023. doi: 10.1038/s41586-023-06443-4. URL https://doi.org/10.1038/s41586-023-06443-4.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16888–16905. PMLR, 2022. URL https://proceedings.mlr.press/v162/niu22a.html.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=g2YraF75Tj.

Joseph E. O'Doherty, Mariana M. B. Cardoso, Joseph G. Makin, and Philip N. Sabes. Nonhuman primate reaching with multichannel sensorimotor cortex electrophysiology, May 2020. URL https://doi.org/10.5281/zenodo.3854034.

Tsam Kiu Pun, Mona Khoshnevis, Tommy Hosman, Guy H. Wilson, Anastasia Kapitonava, Foram Kamdar, Jaimie M. Henderson, John D. Simeral, Carlos E. Vargas-Irwin, Matthew T. Harrison, and Leigh R. Hochberg. Measuring instability in chronic human intracortical neural recordings towards stable, long-term brain-computer interfaces. *Communications Biology*, 7:1363, 2024. doi: 10.1038/s42003-024-06784-4.

Steffen Schneider, Jin Hwa Lee, and Mackenzie Weygandt Mathis. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617(7960):360–368, 2023. doi: 10.1038/s41586-023-06031-6.

Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, pp. 2058–2065, 2016. doi: 10.1609/aaai.v30i1.10306.

David Sussillo, Sergey D. Stavisky, Jonathan C. Kao, Stephen I. Ryu, and Krishna V. Shenoy. Making brain–machine interfaces robust to future neural variability. *Nature Communications*, 7(1): 13749, 2016. doi: 10.1038/ncomms13749.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Maitreyee Wairagkar, Nicholas S. Card, Tyler Singer-Clark, Xianda Hou, Carrina Iacobacci, Lee M. Miller, Leigh R. Hochberg, David M. Brandman, and Sergey D. Stavisky. An instantaneous voice-synthesis neuroprosthesis. *Nature*, 644(8075):145–152, 2025. doi: 10.1038/s41586-025-09127-3. URL https://doi.org/10.1038/s41586-025-09127-3.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno A. Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uXl3bZLkr3c.

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7191–7201, 2022. doi: 10.1109/CVPR52688.2022.00706.

Francis R. Willett, Donald T. Avansino, Leigh R. Hochberg, Jaimie M. Henderson, and Krishna V. Shenoy. High-performance brain-to-text communication via handwriting. *Nature*, 593(7858): 249–254, 2021. doi: 10.1038/s41586-021-03506-2. URL https://doi.org/10.1038/s41586-021-03506-2.

Francis R. Willett, Erin M. Kunz, Chaofei Fan, Donald T. Avansino, Guy H. Wilson, Eun Young Choi, Foram Kamdar, Matthew F. Glasser, Leigh R. Hochberg, Shaul Druckmann, Krishna V. Shenoy, and Jaimie M. Henderson. A high-performance speech neuroprosthesis. *Nature*, 620 (7976):1031–1036, 2023. doi: 10.1038/s41586-023-06377-x. URL https://doi.org/10.1038/s41586-023-06377-x.

Guy H. Wilson, Elias A. Stein, Foram Kamdar, Donald T. Avansino, Tsam Kiu Pun, Ronnie Gross, Tommy Hosman, Tyler Singer-Clark, Anastasia Kapitonava, Leigh R. Hochberg, John D. Simeral, Krishna V. Shenoy, Shaul Druckmann, Jaimie M. Henderson, and Francis R. Willett. Long-term unsupervised recalibration of cursor-based intracortical brain–computer interfaces using a hidden markov model. *Nature Biomedical Engineering*, 2025. doi: 10.1038/s41551-025-01536-z.

A METHOD DETAILS

Table 2: Notation used in the proposed cross-session recalibration framework.

Notation	Description	Notation	Description
$\mathcal{S} = \{X_s, Y_s\}$	Labeled source neural and kinematic sequences.	X_t^{cal}	Unlabeled target stream used for recalibration.
T_s, C, d_y	Source length, neural-channel count, and output dimension.	$\phi(\cdot), \psi(\cdot)$	Causal feature extractor and frozen latent mapping.
P_s, m_P	Source projection matrix and centering vector.	$\mu_{\phi,s}, \sigma_{\phi,s}$	Source feature mean and standard deviation.
$q(\cdot)$	Differentiable readout used for target optimization.	$f_s(\cdot)$	Frozen nonlinear decoder used for final prediction.
$f_{\mathrm{knn}}, f_{\mathrm{nys}}$	Local RBF and Nyström-RBF source decoders.	ω	Mixture weight of the two source decoders.
$\mathcal{R}_s^{\mathrm{N}}$	Source neural and latent references.	$\mathcal{R}_s^{\mathrm{D}}$	Source decoded-output references.
$\mu_{x,s}, \sigma_{x,s}$	Source channel-wise neural statistics.	$\mu_{z,s}, \sigma_{z,s}, \Sigma_{z,s}$	Source latent moments and covariance.
$\widehat{Y}_s^{\mathrm{oof}}$	Out-of-fold source predictions.	$C_{0,s}, C_{1,s}$	Zero- and one-lag output covariances.
A_{st}	Stable channel-wise affine adapter.	a, b	Stable-path scale and offset.
A_θ	Trainable plastic neural adapter.	$\theta = \{\ell, \beta, D, U\}$	Plastic diagonal and low-rank parameters.
$\bar{h}_t$	Normalized causal-window statistics.	r, ρ, s_s	Residual rank, amplitude bound, and source scale.
Z_{st}	Stable-path latent sequence.	Z_{tea}	Fixed latent teacher constructed using source mean and covariance references.
Z_θ	Plastic-path latent sequence.	$U_\theta = q(Z_\theta)$	Auxiliary decoded trajectory for adaptation.
$\widehat{Y}_p^{\,0}$	Unaligned prediction from path p.	$p \in \{\mathrm{st}, \mathrm{pl}\}$	Stable or plastic prediction path.
$\mathcal{L}_{\mathrm{N}}$	Neural-space compatibility objective.	$\mathcal{L}_{\mathrm{D}}$	Decoded-space compatibility objective.
$\mathcal{L}_{\mathrm{B}}$	Bounded-plasticity regularization.	$\mathcal{L}(\theta)$	Complete target adaptation objective.
$d_{\mathrm{mom}}, d_{\mathrm{lat}}$	Moment and latent-teacher discrepancies.	$d_{\mathrm{geo}}, d_{\mathrm{cov}}$	Geometry and covariance discrepancies.
$D_{\mathrm{pri}}(p)$	Global moment discrepancy of the aligned output from path p.	$D_{\mathrm{state}}(p)$	Sample-weighted moment discrepancy within source-defined speed bins.
$\Omega(d; \tau)$	Overflow beyond a source-compatible threshold.	$\Gamma(d_0, d_1)$	Clipped relative reduction in discrepancy.
V, R, C	Plastic validity, repair, and consistency.	g	Session-level plastic fusion weight.
$\mathcal{O}_p$	Path-specific output-alignment transform.	$\widehat{Y}$	Final stable–plastic prediction.
$\odot, \oslash, \Delta$	Element-wise product, division, and temporal difference.	ϵ	Small numerical-stability constant.

A.1 ONLINE RECALIBRATION PROTOCOL

For each target session, let:

$$X_t = \left(X_t^{\mathrm{cal}}, X_t^{\mathrm{pred}}\right) \tag{22}$$

denote an observed calibration stream followed by a non-overlapping prediction stream. All target-dependent quantities, including neural moments, the fixed teacher, plastic parameters, output-alignment transforms, and the fusion weight, are estimated exclusively from X_t^{cal}. They are frozen before predicting X_t^{pred}.

In the pre-task protocol, X_t^{cal} contains the neural activity recorded before the first scored movement interval. When acquisition metadata identify an explicit resting-state recording, that interval is included without using its neural content to infer a task/rest label. In the continual prefix protocol, the same procedure may be repeated at predetermined checkpoints:

$$0 < \tau_1 < \cdots < \tau_M < T_t, \tag{23}$$

where the model recalibrated from $X_{t,1:\tau_m}$ predicts only the subsequent interval $(\tau_m, \tau_{m+1}]$. No neural sample is evaluated after it has already contributed to its own recalibration.

For completeness, a full-stream transductive protocol may use all unlabeled target neural samples to construct one session-level model. Because that protocol uses neural samples from the entire session, it is reported separately and is not interpreted as causal prefix-to-future evaluation.

Target reach boundaries may be used to define scored intervals and reset temporal context only when they are available from the experimental controller at inference time. They are never derived from target kinematics and do not enter adapter optimization or gate construction.

A.2 Causal Features and Neural Preprocessing

For W non-overlapping causal windows, the feature at time t is:

$$\phi_t(X) = \mathrm{concat}_{w=1}^{W} \left[\frac{1}{|\mathcal{I}_w|} \sum_{\delta \in \mathcal{I}_w} x_{t-\delta} \right]. \tag{24}$$

We use five windows covering the current bin and the preceding 1–5, 6–15, 16–30, and 31–60 bins. Bins without the full required context are excluded. When online reach-onset metadata are available, history is reset at the onset; otherwise, the feature extractor remains continuous across the calibration and prediction streams.

All clipping operations use channel-wise source-derived ranges:

$$\mathcal{X}_s = \prod_{c=1}^{C} [l_{s,c}, u_{s,c}], \tag{25}$$

which are estimated once from source training data and stored with the source model. For a vector v,

$$\mathrm{clip}_{\mathcal{X}_s}(v)_c = \min\left\{u_{s,c}, \max\left\{l_{s,c}, v_c\right\}\right\}. \tag{26}$$

This definition replaces an unexplained hard-coded $[0, 1]$ range and ensures that all target clipping bounds are source-only quantities.

The temporal residual input is standardized using calibration-prefix statistics:

$$\bar{h}_t = \mathrm{clip}_{[-c_h, c_h]}\left[(h_t - \mu_{h,t}) \oslash (\sigma_{h,t} + \epsilon)\right]. \tag{27}$$

The source residual scale is:

$$s_s = \mathrm{clip}_{[s_{\min}, s_{\max}]}\left(\sigma_{x,s}\right). \tag{28}$$

The constants c_h, $s_{\min}$, and $s_{\max}$ are selected using source pseudo-target streams.

A.3 Source Decoder and Out-of-Fold References

The source nonlinear decoder combines an RBF-weighted nearest-neighbor regressor and a Nyström kernel ensemble:

$$f_s(z) = (1-\omega)\frac{\sum_{i \in \mathcal{N}_K(z)} \exp\left(-\|z - z_i\|_2^2 / \tau_{\mathrm{knn}}\right) y_i}{\sum_{i \in \mathcal{N}_K(z)} \exp\left(-\|z - z_i\|_2^2 / \tau_{\mathrm{knn}}\right)} + \omega f_{\mathrm{nys}}(z). \tag{29}$$

The Nyström component averages fixed RBF feature maps followed by Ridge heads.

All decoding-space source references are constructed from K_{cv}-fold out-of-fold predictions. For fold k, the representation and readouts are fitted on $\mathcal{S} \setminus \mathcal{S}^{(k)}$ and applied to $\mathcal{S}^{(k)}$. Concatenating the predictions gives:

$$\widehat{Y}_s^{\mathrm{oof}} = \mathrm{concat}_{k=1}^{K_{\mathrm{cv}}} f_s^{(-k)}\left(Z_s^{(k)}\right). \tag{30}$$

The final deployment decoder is subsequently refitted on the complete source session. Out-of-fold construction prevents a source sample from serving as its own nearest neighbor when defining decoded-output references and gate thresholds.

Speed and direction states are defined using thresholds estimated from $\widehat{Y}_s^{\mathrm{oof}}$. The same fixed source thresholds are applied to target decoded streams; target-specific quantile boundaries are not estimated.

A.4 Discrepancy Measures

For a d-dimensional sequence V and source moments (μ, σ), define:

$$d_{\mathrm{mom}}(V; \mu, \sigma) = \frac{1}{d} \left\| (\mu(V) - \mu) \oslash (\sigma + \epsilon) \right\|_2^2 + \frac{1}{2d} \left\| (\sigma(V) - \sigma) \oslash (\sigma + \epsilon) \right\|_2^2 . \tag{31}$$

For a latent sequence Z, let:

$$\bar{\Sigma}(Z) = (1 - \alpha_{\mathrm{sh}})\Sigma(Z) + \alpha_{\mathrm{sh}} \operatorname{diag}(\Sigma(Z)) + \epsilon_{\mathrm{cov}} I. \tag{32}$$

The normalized covariance discrepancy is:

$$d_{\mathrm{cov}}(Z; \Sigma_{z,s}) = \frac{1}{d_z^2} \left\| \bar{\Sigma}(Z) - \bar{\Sigma}_{z,s} \right\|_F^2 . \tag{33}$$

For a decoded sequence Y, define:

$$\bar{Y} = (Y - \mu_{\hat{y},s}) \oslash (\sigma_{\hat{y},s} + \epsilon) . \tag{34}$$

Its complete decoding-space discrepancy is:

$$\begin{aligned} D_{\mathrm{D}}(Y) = {} & d_{\mathrm{mom}}(Y; \mu_{\hat{y},s}, \sigma_{\hat{y},s}) \\ & + \lambda_{\Delta D} d_{\mathrm{mom}}(\Delta Y; \mu_{\Delta\hat{y},s}, \sigma_{\Delta\hat{y},s}) \\ & + \frac{\lambda_{CD}}{2} \sum_{k=0}^{1} \mathrm{MSE}\left(C_k(\bar{Y}), C_{k,s}\right) . \end{aligned} \tag{35}$$

Differences and lagged covariances are computed separately inside each contiguous block and are then averaged, preventing artificial transitions between unrelated blocks.

The normalized latent teacher discrepancy is:

$$d_{\mathrm{lat}}(Z_\theta, Z_{\mathrm{tea}}) = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{|B_k| d_z} \sum_{t \in B_k} \left\| (z_{\theta,t} - z_{\mathrm{tea},t}) \oslash (\sigma_{z,s} + \epsilon) \right\|_2^2 . \tag{36}$$

The decoded geometry loss is:

$$d_{\mathrm{geo}}(U_\theta; C_{0,s}, C_{1,s}) = \frac{1}{2} \sum_{k=0}^{1} \mathrm{MSE}\left(C_k(\bar{U}_\theta), C_{k,s}\right) . \tag{37}$$

A.5 Fixed Latent Teacher

The fixed teacher provides a source-referenced latent target for plastic adaptation. We first encode stable-adapted target calibration blocks using the frozen source encoder to obtain Z_{st}. Let $(\mu_{z,t}, \widetilde{\Sigma}_{z,t})$ denote the mean and shrinkage covariance of the stable-adapted target latents, and let $(\mu_{z,s}, \widetilde{\Sigma}_{z,s})$ denote the corresponding frozen source-stream statistics. The source-referenced CORAL transform is:

$$T_z = \widetilde{\Sigma}_{z,t}^{-1/2} \widetilde{\Sigma}_{z,s}^{1/2}, \qquad Z_{\mathrm{cor}} = (Z_{\mathrm{st}} - \mu_{z,t}) T_z + \mu_{z,s}. \tag{38}$$

The teacher follows the same definition as Eq. equation 9.

$$Z_{\mathrm{tea}} = Z_{\mathrm{st}} + \eta_z (Z_{\mathrm{cor}} - Z_{\mathrm{st}}). \tag{39}$$

Here, Z_{tea} denotes the teacher $\mathbf{Z}^{\mathrm{tea}}$ in the main text. The transform uses source mean and covariance references to define the latent correction. Matrix square roots are computed through symmetric eigendecomposition with eigenvalues lower-bounded by ϵ_{eig}.

The coefficient η_z controls the correction strength, with $\eta_z = 0$ recovering the stable latents and $\eta_z = 1$ recovering the CORAL-transformed latents. The default $\eta_z = 1.25$ applies an extrapolated correction along the same transformation direction.

The teacher is constructed once for each target session from its unlabeled calibration stream. Teacher rows corresponding to each calibration block are cached and detached before plastic optimization. The teacher, source references, and source-model parameters remain fixed throughout the optimization. Thus, "fixed" refers to the teacher's invariance during adaptation within a session, while each target session has its own calibration-derived teacher.

A.6 Target Sampling and Optimization

Global channel moments are estimated using at most 12,000 uniformly spaced rows from X_t^{cal}. Temporal constraints use at most eight contiguous blocks, each containing at most 2,000 bins. When the prefix is shorter, all available rows are used, and block lengths are reduced without overlap.

The residual loss is:

$$\mathcal{L}_{\mathrm{res}} = \mathrm{MSE}\left(r_{\mathrm{stat}} \oslash (s_s + \epsilon), 0\right) + \frac{1}{K}\sum_{k=1}^{K} \mathrm{MSE}\left(r_{B_k} \oslash (s_s + \epsilon), 0\right). \tag{40}$$

The diagonal proximity penalty is:

$$\mathcal{L}_{\mathrm{diag}} = \frac{1}{C}\left\|\ell - \log(a + \epsilon)\right\|_2^2 + \frac{1}{C}\left\|(\beta - b) \oslash (s_s + \epsilon)\right\|_2^2. \tag{41}$$

The low-rank penalty is:

$$\mathcal{L}_{\mathrm{lr}} = \frac{1}{WCr}\|D\|_F^2 + \frac{1}{rC}\|U\|_F^2. \tag{42}$$

In addition to these penalties, the implementation clamps:

$$\ell - \log(a + \epsilon) \in [-c_\ell, c_\ell], \qquad (\beta - b) \oslash (s_s + \epsilon) \in [-c_\beta, c_\beta] \tag{43}$$

after each optimizer update.

Plastic adaptation uses Adam for 40 updates by default. Only $\theta = \{\ell, \beta, D, U\}$ receives gradients. The representation, auxiliary readout, nonlinear decoder, source samples, source references, and stable teacher remain frozen.

A.7 Label-Free Stable-Plastic Gate

The session-level gate evaluates the source compatibility and reliability of the plastic candidate after adaptation and output alignment. For each path $p \in \{\mathrm{st}, \mathrm{pl}\}$, let $\widehat{Y}_p^{\,0}$ denote its decoded calibration output and let $\widetilde{Y}_p = \mathcal{O}_p(\widehat{Y}_p^{\,0})$ denote its aligned output. The path-specific transform $\mathcal{O}_p$ is the output-alignment map O_p in equation 16, estimated from the unlabeled calibration stream. The global discrepancy $D_{\mathrm{pri}}(p)$ is the moment discrepancy in equation 5 between $\widetilde{Y}_p$ and the source-stream output. The state-conditioned discrepancy $D_{\mathrm{state}}(p)$ is the analogous sample-weighted moment discrepancy computed within source-defined speed bins. Both discrepancies use aligned calibration outputs and frozen source references.

Overflow and Relative Repair. The overflow function measures deviation beyond a source-compatible threshold, and the relative-repair function measures the fractional reduction in discrepancy.

$$\Omega(d; \tau) = \left[\frac{d}{\tau} - 1\right]_+, \qquad \Gamma(d_0, d_1) = \left[\frac{d_0 - d_1}{d_0 + \epsilon}\right]_0^1, \tag{44}$$

here, $[\cdot]_+$ denotes the positive part and $[\cdot]_0^1$ clips to $[0, 1]$.

Validity. Validity measures whether the plastic output remains inside the source-compatible decoded-prior corridor.

$$V = \exp\left\{-\Omega(D_{\mathrm{pri}}(\mathrm{pl}); \tau_{\mathrm{pri}})^2\right\}. \tag{45}$$

The score equals one inside the corridor and decays smoothly as the plastic discrepancy exceeds the threshold.

Repair. Repair combines the relative improvement in the sample-weighted state-conditioned discrepancy with global improvement when the stable prior exceeds the corridor.

$$R = \max\left\{\Gamma(D_{\mathrm{state}}(\mathrm{st}), D_{\mathrm{state}}(\mathrm{pl}))\,, \sqrt{[\Omega(D_{\mathrm{pri}}(\mathrm{st}); \tau_{\mathrm{pri}})]_0^1\, \Gamma(D_{\mathrm{pri}}(\mathrm{st}), D_{\mathrm{pri}}(\mathrm{pl}))}\right\}. \tag{46}$$

State-conditioned improvement can provide evidence of repair even when the stable global prior remains inside the corridor.

Consistency. Consistency uses the final teacher loss $\mathcal{L}_{\mathrm{tea}}$ and residual trust loss $\mathcal{L}_{\mathrm{tru}}$ defined in the main text.

$$P_k = \left[\frac{\mathcal{L}_k - \tau_k}{w_k}\right]_0^1, \quad k \in \{\mathrm{tea}, \mathrm{tru}\}, \qquad C = 1 - \max(P_{\mathrm{tea}}, P_{\mathrm{tru}}). \tag{47}$$

The thresholds τ_k and widths w_k specify the tolerated loss levels and the ranges over which consistency decreases.

Session-Level Fusion. The final gate follows the same validity-weighted soft logical rule as equation 21.

$$g = \left[V\big(1 - (1-R)(1-C)\big)\right]_0^1. \tag{48}$$

Validity acts as a prerequisite, while the soft OR combines evidence of repair with consistency under the bounded adaptation process. The gate is computed once from the calibration stream and is held fixed when fusing the aligned stable and plastic predictions on the subsequent prediction stream. All gate thresholds are fixed before target evaluation using the source-only selection procedure in Appendix A.9.

Microdrift Diagnostics. The broader microdrift audit describes decoded-prior and decoded-dynamics mismatch, neural variance expansion, population covariance, eigenspectrum deviation, and lost-channel count. These quantities support interpretation of source-compatibility deviations, while the default fusion weight is fully specified by V, R, and C above.

A.8 Path-Specific Output Alignment

For each path $p \in \{\mathrm{st}, \mathrm{pl}\}$, let $(\mu_{\hat{y},t,p}, \bar{\Sigma}_{\hat{y},t,p})$ be the moments of its unaligned calibration output. The whitening-coloring transform is:

$$\mathcal{W}_p(y) = (y - \mu_{\hat{y},t,p})\, \bar{\Sigma}_{\hat{y},t,p}^{-1/2} \bar{\Sigma}_{\hat{y},s}^{1/2} + \mu_{\hat{y},s}, \tag{49}$$

and its softened form is:

$$\mathcal{O}_p(y) = (1 - \gamma_o) y + \gamma_o \mathcal{W}_p(y). \tag{50}$$

When supported, separate transforms are estimated within source-defined speed and direction states. Both source and target outputs are assigned using the same thresholds obtained from out-of-fold source predictions. A state-specific transform is used only when its source and target sample counts exceed $n_{\min}$; otherwise, the global transform is applied.

Output alignment is estimated using X_t^{cal} and applied to both the decoded calibration outputs and the future predictions from X_t^{pred}. The aligned calibration outputs determine D_{pri} and D_{state}, which are used to compute validity V and repair R. Consistency C is computed from the final teacher and residual trust losses. The output-alignment transforms and the resulting gate are frozen before predicting X_t^{pred}.

A.9 Source-Only Configuration Selection

Architecture choices, ranks, loss weights, optimizer settings, clipping limits, covariance shrinkage, output-alignment strength, and all gate thresholds are selected without evaluating any real target-session kinematics. Pseudo-target streams are constructed exclusively from source activity using:

- identity transformations;
- channel-wise gain and bias changes;
- partial channel attenuation;
- activity-dependent low-rank channel mixing;
- block-wise changes in gain or bias; and
- combinations of persistent and state-dependent perturbations.

Each pseudo-target is divided into a calibration and prediction portion using the same protocol as real target sessions. Because its source kinematics remain available, candidate configurations can be ranked without exposing real target labels. The selected configuration is frozen before all real-session experiments. Real target labels are excluded from adapter gradients, early stopping, gate thresholds, seed selection, and session-specific control decisions.

A.10 Recalibration Algorithm

Algorithm 1 Detect-Then-Adapt Adaptation from an Unlabeled Neural Stream

Require: Labeled source session $\mathcal{S}$; target calibration prefix X_t^{cal}; future stream X_t^{pred}
Ensure: Future movement prediction $\widehat{Y}_t^{\mathrm{pred}}$

Train source representation $\psi \circ \phi$, auxiliary readout q, and final decoder f_s
Construct neural references $\mathcal{R}_s^{\mathrm{N}}$
Construct out-of-fold predictions $\widehat{Y}_s^{\mathrm{oof}}$ and decoding references $\mathcal{R}_s^{\mathrm{D}}$
Freeze all source components and references
Estimate $(\mu_{x,t}, \sigma_{x,t})$ from X_t^{cal}
Construct stable transform $A_{\mathrm{st}}(x) = \mathrm{clip}_{\mathcal{X}_s}\,(a \odot x + b)$
Compute stable latents Z_{st} from the unlabeled calibration stream.
Construct Z_{tea} using equation 39 with $\eta_z = 1.25$, and cache the detached teacher rows for each calibration block.
Initialize $\ell \leftarrow \log(a + \epsilon)$, $\beta \leftarrow b$, and $U \leftarrow 0$
for $j = 1, \ldots, J$ **do**
    Sample global rows and contiguous blocks from X_t^{cal}
    Compute Z_θ and auxiliary output U_θ
    Evaluate $\mathcal{L}_{\mathrm{N}}$, $\mathcal{L}_{\mathrm{D}}$, and $\mathcal{L}_{\mathrm{B}}$
    Update only $\theta = \{\ell, \beta, D, U\}$ with Adam
    Clamp diagonal deviations to their source-selected ranges
end for
Decode unaligned calibration candidates $\widehat{Y}_{\mathrm{st}}^{\,0}$ and $\widehat{Y}_{\mathrm{pl}}^{\,0}$ using the shared frozen decoder.
Estimate $\mathcal{O}_{\mathrm{st}}$ and $\mathcal{O}_{\mathrm{pl}}$ from the calibration candidates and frozen source references.
Align the calibration candidates as $\widetilde{Y}_p = \mathcal{O}_p(\widehat{Y}_p^{\,0})$ for $p \in \{\mathrm{st}, \mathrm{pl}\}$.
Compute $D_{\mathrm{pri}}(p)$ and $D_{\mathrm{state}}(p)$ from the aligned calibration candidates.
Compute validity V, repair R, and consistency C according to Appendix A.7.
Set $g \leftarrow \big[V\big(1 - (1 - R)(1 - C)\big)\big]_0^1$.
Freeze the plastic parameters, output-alignment transforms, and fusion weight for future prediction.
Decode future candidates $\widehat{Y}_{\mathrm{st}}^{\,0}$ and $\widehat{Y}_{\mathrm{pl}}^{\,0}$ from X_t^{pred} using the shared frozen decoder.
return $(1 - g)\mathcal{O}_{\mathrm{st}}(\widehat{Y}_{\mathrm{st}}^{\,0}) + g\mathcal{O}_{\mathrm{pl}}(\widehat{Y}_{\mathrm{pl}}^{\,0})$.

A.11 Parameters and Computational Cost

The plastic adapter contains:

$$P_{\text{adapter}} = 2C + WCr + rC \tag{51}$$

trainable parameters. With $W = 5$ and $r = 64$,

$$P_{\text{adapter}} = 386C, \tag{52}$$

corresponding to 37,056 parameters for 96 channels and 74,112 parameters for 192 channels.

With maintained running window sums, the low-rank temporal residual requires:

$$\mathcal{O}\left(Cr(W+1)\right) \tag{53}$$

operations per predicted bin. The nearest-neighbor decoder additionally requires distances to stored source latent samples, while the Nyström component applies fixed kernel maps and Ridge heads. Runtime and memory are measured using the stated calibration-stream budget and hardware configuration.

B Experimental Details

B.1 Dataset Details

Dataset and Decoding Task. We use the publicly available Nonhuman Primate Reaching with Multichannel Sensorimotor Cortex Electrophysiology dataset (O'Doherty et al., 2020). The recordings were collected from two nonhuman primates, Indy and Loco, performing a two-dimensional reaching task. Neural activity was recorded using 96-channel M1 or 192-channel M1/S1 configurations, together with synchronized movement kinematics. The decoding task predicts two-dimensional cursor velocity from the current and preceding neural activity. Each session provides a continuous neural stream, synchronized velocity measurements, and reach boundaries used to construct source supervision and define evaluation intervals.

Recording Sequences. We organize the 47 recording sessions into three chronological sequences with consistent animal identity and channel configuration: Indy-96, Indy-192, and Loco-192. Table 3 summarizes their composition and recording periods. The recordings contain approximately 13.74 hours of neural activity and 25,487 reaches in total. A separate source model is fitted for each sequence, preserving its channel correspondence throughout cross-session evaluation.

Table 3: Composition of the three recording sequences used for cross-session evaluation.

Sequence	Channels	Sessions	Sources	Targets	Recording period
Indy-96	96	30	1	29	2016-06-22 to 2017-01-31
Indy-192	192	7	1	6	2016-04-07 to 2016-04-26
Loco-192	192	10	1	9	2017-02-10 to 2017-03-02
Total	–	47	3	44	–

Neural Representation and Temporal Resolution. Neural activity is represented in 4-ms bins, corresponding to a sampling rate of 250 Hz. The standardized channel representation records whether at least one spike event occurs within each bin. We use the synchronized two-dimensional velocity supplied with the dataset as the regression target. For each channel, the causal feature extractor averages neural activity over five lag windows: $[0, 0]$, $[1, 5]$, $[6, 15]$, $[16, 30]$, and $[31, 60]$ bins. Concatenating these averages produces a $5C$-dimensional feature vector for a recording with C channels. The maximum history is 240 ms, and each feature depends exclusively on the current and preceding neural observations. Reach-based source training and evaluation retain samples beginning 60 bins after reach onset, ensuring that every scored sample has the required within-reach history.

Source-Target Partition. In the One2All setting, the earliest session in each recording sequence serves as the labeled source, and all subsequent sessions serve as targets. This produces 29, 6, and 9 target sessions for Indy-96, Indy-192, and Loco-192, respectively. The source representation and decoder are fitted once per sequence and reused across its target sessions. Each target adapter is initialized independently from that session's unlabeled neural statistics. Target-specific adapter parameters are reset between sessions, so the One2All comparison measures adaptation from a common source anchor. The Overall score assigns equal weight to each of the 44 target sessions.

Target-Stream Access. The unlabeled adaptation stream can contain pre-task, inter-task, and task-period activity. Target kinematics and task/rest labels are excluded from the construction and optimization of the proposed unsupervised adapter. The offline stream-budget setting uses the first 10%, 25%, 50%, or 100% of the target neural stream for recalibration and evaluates predictions on the common offline evaluation set. This setting measures the effect of adaptation-data availability under a transductive protocol. The online stream-budget setting uses the first 10%, 25%, or 50% of the stream for recalibration and evaluates performance beginning after the calibration cutoff. For online evaluation, all target-dependent parameters, output-alignment transforms, and gate values are frozen before the subsequent prediction interval. The few-shot experiment separately introduces a specified number of labeled calibration reaches and evaluates the remaining reaches, as detailed in the corresponding experimental subsection.

B.2 Training Details

Source Training and Sampling. Source training pairs causal neural features with synchronized two-dimensional velocity from the labeled source session. We use at most 20,000 valid source samples, selected at uniformly spaced positions along the ordered training support when more samples are available. Target evaluation uses at most 4,000 valid samples per session, with the same temporal coverage strategy. Sampling is restricted to the intervals permitted by each evaluation protocol. For moment estimation, we use at most 12,000 uniformly spaced observations from the available neural stream. Temporal adaptation losses additionally use up to eight representative contiguous blocks, each containing at most 2,000 bins, corresponding to 8 s of activity. The number and length of these blocks are reduced when the available stream is shorter.

Frozen Source Representation and Decoder. We standardize the causal features using source-training means and standard deviations and fit a 32-dimensional PCA representation. The source readout combines an RBF-weighted nearest-neighbor regressor with an ensemble of Nyström-RBF ridge regressors. The nearest-neighbor component uses 20 neighbors, with its local bandwidth determined from the neighborhood distances. The ensemble contains three regressors, each using at most 2,048 kernel components and a ridge coefficient of 1.0. Their predictions are combined as $f_s = 0.55 f_{\mathrm{knn}} + 0.45 f_{\mathrm{ens}}$, matching Eq. equation 4. A separate ridge proxy maps source latent features to velocity and supplies differentiable output constraints during target adaptation. All reported stable and plastic predictions use the shared frozen nonlinear readout f_s. Source normalization, PCA, the nonlinear readout, the ridge proxy, and the source reference statistics remain fixed throughout target adaptation.

Adapter Initialization and Optimization. For each target session, the stable path is constructed by matching channel-wise target moments to the source references. The plastic path starts from the same affine transform and introduces a rank-64 temporal residual. Its residual output factor is initialized to zero, so the initial plastic transform coincides with the stable transform. The fixed latent teacher is constructed once from stable-adapted target blocks using Eq. equation 9 and is detached before optimization. We optimize only the plastic parameters $\theta = \{\ell, \beta, D, U\}$ using Adam for 40 updates with a learning rate of 0.01. Each update evaluates the adaptation objective on the sampled neural observations and contiguous blocks. The final adapter is retained after the prescribed update budget, with target kinematics excluded from optimization and stopping decisions. The bounded residual parameterization and diagonal-parameter constraints are maintained throughout optimization.

Adaptation Objective. We use the matching, preservation, and regularization terms defined in equation 11, equation 12, and equation 13. The default coefficients are

$$(\lambda_{\mathrm{str}}, \lambda_{\mathrm{tea}}, \lambda_{\mathrm{pri}}, \lambda_{\mathrm{dyn}}, \lambda_{\mathrm{geo}}, \lambda_{\mathrm{tru}}, \lambda_{\mathrm{par}}) = (0.35, 0.30, 0.15, 0.08, 0.02, 0.12, 10^{-4}). \tag{54}$$

These terms constrain neural moments, fixed-teacher agreement, decoded-output moments and dynamics, decoded geometry, residual energy, and residual-factor magnitude. Temporal differences and lagged covariances are evaluated within contiguous blocks to preserve their temporal interpretation. The latent teacher uses the alignment strength $\eta_z = 1.25$ specified in the main text.

Output Alignment and Fusion. After plastic optimization, both paths are decoded using the same frozen source model. Path-specific output alignment uses four speed bins and four direction bins for moving states, with alignment strength $\eta_o = 0.25$. State assignments are derived from decoded predictions, and a global transform is used when a state has insufficient source or target support. The aligned calibration outputs determine the validity and repair scores, while the final teacher and residual trust losses determine consistency. The default gate parameters are $\tau_{\mathrm{pri}} = 0.01$, $(\tau_{\mathrm{tea}}, w_{\mathrm{tea}}) = (0.05, 0.03)$, and $(\tau_{\mathrm{tru}}, w_{\mathrm{tru}}) = (0.015, 0.03)$. The session-level fusion weight is $g = [V(1 - (1 - R)(1 - C))]_0^1$, following Appendix A.7. For online evaluation, the fitted output transforms and fusion weight are applied unchanged to the subsequent prediction stream.

Configuration Selection and Comparison Settings. The proposed method selects its adapter structure, loss coefficients, alignment strengths, and gate parameters using source-derived pseudo-target streams. The selected configuration is fixed before real target-session evaluation. All methods use the same source–target partitions, causal neural observations, and protocol-specific evaluation support. Source-only decoders are trained using labeled source data, while the adaptation baselines additionally receive the unlabeled target stream permitted by each protocol. For each cross-session baseline and recording sequence, configuration selection uses an equal budget of 128 candidates. Candidates are ranked on a fixed target-calibration subset by mean session-level R^2 minus half its session-level standard deviation. This baseline configuration-selection stage uses target-calibration scores, whereas the subsequent within-session adaptation procedure uses unlabeled neural observations. Method-specific architectures, objectives, and search spaces are described in Appendix D.

Evaluation Metric and Aggregation. For each target session, we compute the coefficient of determination jointly over both velocity dimensions:

$$R^2 = 1 - \frac{\sum_{t \in \mathcal{I}} \|\mathbf{y}_t - \widehat{\mathbf{y}}_t\|_2^2}{\sum_{t \in \mathcal{I}} \|\mathbf{y}_t - \overline{\mathbf{y}}\|_2^2}, \qquad \overline{\mathbf{y}} = \frac{1}{|\mathcal{I}|} \sum_{t \in \mathcal{I}} \mathbf{y}_t. \tag{55}$$

Here, $\mathcal{I}$ denotes the valid evaluation samples for the session and protocol. All reported scores use the $100R^2$ scale, with higher values indicating better decoding. We conduct five matched independent runs and first average target-session scores within each run and evaluation condition. The reported mean and sample standard deviation are then calculated across these five run-level averages. For One2All, the Overall score averages all 44 target sessions, equivalently weighting the three sequence-level means by their target counts of 29, 6, and 9. The main method comparisons use two-sided paired t-tests on matched run-level aggregates, as detailed in Appendix C.2.

C Additional Experimental Results

Unless otherwise stated, all results are measured by $100R^2$, where higher values indicate better movement decoding. Results are reported as the mean and sample standard deviation across five independent runs. Within each run, performance is first aggregated over all target sessions included in the corresponding evaluation condition. Bold and underlined values denote the best and second-best means in each column, respectively.

C.1 Statistical Tests for the One2All Comparison

The main paper reports the One2All performance across Indy-96, Indy-192, and Loco-192, covering 44 target sessions in total. To determine whether the observed improvements are consistent

across independent runs, we additionally compare each baseline against the proposed method using two-sided paired t-tests. For each run, session-level performance is first aggregated within the corresponding recording sequence, after which the matched run-level results are used for statistical testing.

Table 4: Two-sided paired t-test results for the One2All comparison. Each entry reports the p-value of the corresponding baseline against Ours. † denotes a difference that is not statistically significant at $p < 0.05$.

Method vs. Ours	Indy-96	Indy-192	Loco-192	Overall
KF	9.02×10^{-8}	6.02×10^{-7}	6.12×10^{-7}	1.16×10^{-7}
LSTM	1.32×10^{-4}	2.82×10^{-3}	1.96×10^{-3}	8.81×10^{-5}
Transformer	3.75×10^{-4}	9.63×10^{-3}	2.42×10^{-2}	1.80×10^{-3}
CORAL	1.33×10^{-4}	3.26×10^{-3}	4.94×10^{-3}	6.39×10^{-4}
MMD	1.06×10^{-2}	1.51×10^{-3}	1.56×10^{-3}	4.00×10^{-3}
DANN	4.88×10^{-2}	$5.59 \times 10^{-2\dagger}$	4.65×10^{-4}	1.04×10^{-2}
NoMAD	4.74×10^{-2}	$8.37 \times 10^{-2\dagger}$	8.22×10^{-3}	3.29×10^{-3}
CEBRA	1.28×10^{-2}	1.60×10^{-3}	6.91×10^{-3}	7.85×10^{-3}

Results and Analysis. As shown in Table 4, Ours differs significantly from every baseline in the 44-session Overall comparison. The same conclusion holds for all comparisons on Indy-96 and Loco-192. On the smaller Indy-192 sequence, the differences from DANN ($p = 5.59 \times 10^{-2}$) and NoMAD ($p = 8.37 \times 10^{-2}$) do not reach the significance threshold. This finding is consistent with the mean results: NoMAD obtains the highest Indy-192 mean, whereas Ours remains competitive and ranks second. Overall, the tests support a consistent aggregate advantage across the complete benchmark without claiming significant superiority over DANN or NoMAD on the six-session Indy-192 subset.

C.2 Paired Bootstrap Confidence Intervals for One2All

We quantify uncertainty in the performance differences between our method and the five cross-session adaptation baselines using 44 target sessions and five matched independent runs. For baseline m, target session i, and run s, the paired difference is:

$$d_{i,s}^{(m)} = 100 \left(R^2_{\text{ours},i,s} - R^2_{m,i,s} \right). \tag{56}$$

Pairing matches the source session, target session, and run. The reported effect is the average paired difference across runs and target sessions. For the Overall comparison, every target session receives equal weight, preserving the contributions of 29 Indy-96, 6 Indy-192, and 9 Loco-192 sessions.

Bootstrap Procedure. We construct 95% confidence intervals using 5,000 paired bootstrap replicates. Within each replicate, target sessions are sampled with replacement separately within each recording sequence, preserving its original session count. Five run indices are also sampled with replacement, and the same sampled run indices are applied globally across all sessions. This crossed resampling preserves the shared run structure while accounting for variation across target sessions and model repetitions. The same resampling indices are used for both methods in each comparison and across the compared baselines. The confidence limits are the 2.5th and 97.5th percentiles of the bootstrap effect distribution.

Overall confidence intervals lie above zero, supporting an aggregate decoding advantage across the evaluated sessions. Against CEBRA, the strongest overall baseline, the mean improvement is 3.00 points with a 95% confidence interval of $[1.686, 4.327]$. The CEBRA comparison also yields positive confidence intervals within each recording sequence. All five comparisons on Loco-192 have positive confidence intervals. On Indy-96, the intervals for DANN and NoMAD include zero, while the Indy-192 intervals for CORAL, DANN, and NoMAD include zero. The sequence-level intervals clarify the heterogeneity of the gains alongside the consistent aggregate advantage.

Table 5: Paired performance improvements of Ours over the cross-session baselines in One2All. Each entry reports the mean difference and its 95% paired bootstrap confidence interval in $100R^2$ points. Positive differences favor Ours.

Baseline	Indy-96	Indy-192	Loco-192	Overall
CORAL	2.85 [0.60, 5.31]	1.80 [−0.26, 4.37]	8.88 [3.81, 14.23]	3.94 [2.08, 5.92]
MMD	2.17 [0.15, 4.75]	1.71 [0.20, 3.71]	8.46 [4.19, 12.76]	3.40 [1.77, 5.27]
DANN	1.54 [−0.79, 4.79]	1.55 [−0.38, 3.90]	12.85 [7.35, 18.36]	3.85 [1.85, 6.17]
NoMAD	2.41 [−1.25, 7.89]	−0.66 [−2.20, 1.05]	12.38 [4.57, 21.13]	4.03 [0.93, 7.99]
CEBRA	1.85 [0.24, 3.33]	4.30 [2.09, 6.62]	5.82 [1.89, 9.46]	3.00 [1.69, 4.33]

C.3 Complete Days-Apart Evaluation

The Days-Apart evaluation examines whether a decoder trained from the earliest labeled source session remains effective on progressively later recordings. For each recording sequence, sessions are ordered chronologically, and a cutoff ratio $r \in \{0.2, 0.4, 0.6, 0.8\}$ defines a chronological prefix. The source model is always trained only on the earliest session, while all sessions after the cutoff form the target set. The four settings contain 38, 29, 19, and 10 target sessions, respectively. Sessions inside the prefix other than the first source session are not added to source training. The Average column is the macro-average over the four cutoff settings.

Table 6: Complete Days-Apart results under the first-source protocol. Performance is measured by $100R^2$. † denotes no significant difference from Ours according to the finalized statistical record.

Method	$r = 0.2$	$r = 0.4$	$r = 0.6$	$r = 0.8$	Average
KF	-121.96 ± 0.00	-130.83 ± 0.00	-113.64 ± 0.00	-134.07 ± 0.00	-125.13 ± 0.00
LSTM	5.65 ± 0.32	4.73 ± 0.36	2.75 ± 0.61	2.72 ± 0.76	3.96 ± 0.46
Transformer	5.17 ± 2.33	4.77 ± 2.48	2.36 ± 2.81	0.30 ± 4.16	3.15 ± 2.75
CORAL	34.89 ± 0.20	32.32 ± 0.31	30.52 ± 0.40	29.13 ± 0.84†	31.71 ± 0.21
MMD	35.23 ± 0.26	33.31 ± 0.19	31.54 ± 1.03	29.79 ± 0.57	32.47 ± 0.29
DANN	33.88 ± 0.38	31.26 ± 1.69	30.01 ± 1.13	29.03 ± 1.18	31.05 ± 0.64
NoMAD	34.95 ± 0.81	33.04 ± 1.26	32.30 ± 1.45	29.59 ± 2.19	32.47 ± 0.76
CEBRA	35.16 ± 0.48	33.56 ± 0.31	31.36 ± 0.58	29.01 ± 1.01	32.27 ± 0.32
Ours	**38.37 ± 0.08**	**36.75 ± 0.12**	**35.34 ± 0.12**	**31.38 ± 0.07**	**35.46 ± 0.07**

Results and Analysis. Ours achieves the highest mean at every cutoff and obtains an average performance of 35.46 ± 0.07. Its improvements over the strongest competing mean are 3.14, 3.19, 3.04, and 1.59 points at $r = 0.2$, 0.4, 0.6, and 0.8, respectively. The average improvement over the strongest baseline average is 2.99 points. Performance decreases for all methods as the cutoff advances and the evaluation shifts toward later target sessions. The strongest baseline also varies across settings: MMD ranks second at $r = 0.2$ and $r = 0.8$, CEBRA ranks second at $r = 0.4$, and NoMAD ranks second at $r = 0.6$. In contrast, Ours remains the top-performing approach throughout the complete chronological range. These results indicate that the stable–plastic framework is less dependent on a particular source–target separation regime than methods based on a single alignment or representation-learning mechanism.

C.4 Complete Target-Stream Budget Evaluation

Motivation and Protocol. A practical recalibration method should remain useful when only a limited amount of unlabeled target activity is available. We therefore evaluate two complementary

stream-budget protocols. In the offline protocol, only an $r \in \{0.1, 0.25, 0.5, 1.0\}$ fraction of the target stream is available for constructing the target-session model, while all methods are evaluated on the common offline evaluation set. This protocol isolates sensitivity to the amount of adaptation data but is transductive and is not interpreted as causal deployment. In the online protocol, the first $r \in \{0.1, 0.25, 0.5\}$ fraction of each target session is used for recalibration, after which all target-dependent quantities are frozen and evaluated on the subsequent non-overlapping stream.

Table 7: Complete offline target-stream budget results. The Average column is the macro-average over the four target-stream fractions.

Method	$r = 0.1$	$r = 0.25$	$r = 0.5$	$r = 1.0$	Average
KF	-112.63 ± 0.00	-112.63 ± 0.00	-112.63 ± 0.00	-112.63 ± 0.00	-112.63 ± 0.00
LSTM	6.57 ± 0.46	6.57 ± 0.46	6.57 ± 0.46	6.57 ± 0.46	6.57 ± 0.46
Transformer	6.65 ± 2.27	6.65 ± 2.27	6.65 ± 2.27	6.65 ± 2.27	6.65 ± 2.27
CORAL	32.87 ± 0.38	34.76 ± 0.06	34.77 ± 0.43	34.57 ± 0.20	34.24 ± 0.23
MMD	32.88 ± 0.42	34.90 ± 0.10	34.77 ± 0.15	34.92 ± 0.42	34.37 ± 0.12
DANN	31.53 ± 2.53	33.70 ± 1.68	33.99 ± 0.70	33.50 ± 0.69	33.18 ± 0.57
NoMAD	32.70 ± 1.22	34.75 ± 1.05	34.80 ± 0.89	34.82 ± 0.93	34.27 ± 0.43
CEBRA	33.39 ± 0.32	35.22 ± 0.24	35.44 ± 0.12	35.44 ± 0.34	34.87 ± 0.09
Ours	**37.47 ± 0.09**	**38.37 ± 0.08**	**38.69 ± 0.07**	**38.53 ± 0.05**	**38.26 ± 0.02**

Table 8: Complete online prefix-to-future target-stream budget results. Recalibration uses only the observed prefix, and evaluation uses the subsequent non-overlapping stream.

Method	$r = 0.1$	$r = 0.25$	$r = 0.5$	Average
KF	-115.71 ± 0.00	-124.82 ± 0.00	-133.36 ± 0.00	-124.63 ± 0.00
LSTM	6.62 ± 0.42	6.44 ± 0.41	6.27 ± 0.33	6.44 ± 0.38
Transformer	6.47 ± 2.31	6.01 ± 2.41	5.53 ± 2.48	6.01 ± 2.39
CORAL	33.15 ± 0.45	35.23 ± 0.13	35.26 ± 0.41	34.55 ± 0.26
MMD	33.16 ± 0.40	35.26 ± 0.18	35.15 ± 0.26	34.52 ± 0.08
DANN	31.66 ± 2.52	34.01 ± 1.86	34.24 ± 0.60	33.30 ± 0.83
NoMAD	33.49 ± 1.26	35.37 ± 0.98	35.58 ± 0.96	34.81 ± 0.60
CEBRA	33.73 ± 0.33	35.69 ± 0.13	35.85 ± 0.18	35.09 ± 0.08
Ours	**37.86 ± 0.11**	**39.06 ± 0.05**	**39.35 ± 0.09**	**38.75 ± 0.05**

Results and Analysis. In the offline protocol, Ours ranks first at every budget and improves over the strongest competing method by 4.08, 3.15, 3.25, and 3.09 points at $r = 0.1$, 0.25, 0.5, and 1.0, respectively. Its performance increases from 37.47 ± 0.09 with 10% of the stream to 38.69 ± 0.07 with 50%, after which it remains similar at 38.53 ± 0.05 with the complete stream. The small difference between the 50% and 100% settings suggests that the target statistics and source-constrained adaptation objective stabilize before the complete recording is observed.

The same ranking is observed under the causal online protocol. Ours outperforms the strongest baseline by 4.13, 3.37, and 3.50 points at the three prefix budgets. Compared with the offline result at the same nominal budget, the online result is higher by 0.39–0.69 points. More importantly, the online experiment confirms that the method retains its advantage when all target-dependent quantities are estimated from an observed prefix and frozen before future prediction. Together, these results show that the method does not require the complete target stream to achieve stable cross-session improvements.

C.5 Few-Shot Evaluation

Motivation and Protocol. Although the primary setting assumes no target labels, a practical system may occasionally obtain a small number of labeled target reaches. We therefore evaluate

whether the proposed framework can benefit from limited supervised calibration. For each target session, the fixed split provides $K \in \{0, 5, 10, 20\}$ labeled reaches for target calibration. Only these declared calibration labels are used, while the remaining reaches are reserved for evaluation. The $K = 0$ setting corresponds to the fully unsupervised One2All protocol.

Table 9: Few-shot cross-session decoding results. † denotes no significant difference from Ours according to a two-sided paired t-test.

Method	$K = 0$	$K = 5$	$K = 10$	$K = 20$	Average
KF	-112.63 ± 0.00	-113.34 ± 0.77	-112.88 ± 1.42	-110.70 ± 1.62	-112.39 ± 1.16
LSTM	6.57 ± 0.46	6.71 ± 0.28	6.72 ± 0.05	6.78 ± 0.10	6.70 ± 0.09
Transformer	6.65 ± 2.27	6.59 ± 0.21	6.64 ± 0.17	6.74 ± 0.14	6.65 ± 0.06
CORAL	34.72 ± 0.19	34.93 ± 0.10	34.97 ± 0.44	35.27 ± 0.23	34.97 ± 0.20
MMD	35.06 ± 0.24	35.44 ± 0.40	35.09 ± 0.11	35.39 ± 0.54	35.25 ± 0.14
DANN	34.24 ± 1.72	34.73 ± 0.81	33.31 ± 0.80	34.16 ± 0.62	34.11 ± 0.59
NoMAD	34.47 ± 0.32	35.45 ± 0.75†	35.45 ± 0.98	35.49 ± 0.93	35.21 ± 0.22
CEBRA	35.51 ± 0.30	36.01 ± 0.23†	36.06 ± 0.20	36.11 ± 0.39	35.92 ± 0.15
Ours	**38.64 ± 0.09**	**36.65 ± 0.55**	**39.71 ± 0.34**	**41.67 ± 0.02**	**39.17 ± 0.09**

Results and Analysis. Ours obtains the highest mean under all four label budgets. The $K = 5$ result remains higher than all baselines but does not improve over the unsupervised $K = 0$ configuration; its differences from NoMAD and CEBRA are also not significant. This suggests that an extremely small labeled subset may not yet provide sufficient coverage of target-session variability and may remain sensitive to the selected calibration reaches.

The benefit becomes clear when more labeled reaches are available. Relative to $K = 0$, Ours improves by 1.06 points at $K = 10$ and by 3.03 points at $K = 20$, reaching 41.67 ± 0.02 in the latter setting. The small run-to-run variation at $K = 20$ further suggests that adequate target supervision can complement the source-constrained recalibration objective. These results show that the framework remains effective without target labels while also benefiting from limited supervision once the calibration set provides sufficient behavioral coverage.

C.6 Robustness Evaluation

To evaluate whether the proposed recalibration remains reliable under additional recording degradation, we corrupt only the target neural signals while keeping source training unchanged. The same corruption is applied to both the target adaptation stream and the target samples used for evaluation.

Channel Dropout. Channel dropout simulates the loss or attenuation of a fraction of recorded channels. We evaluate dropout rates $r \in \{0.05, 0.10, 0.20\}$ in addition to the clean setting.

Relative to the clean result, Ours decreases by 0.63 and 1.54 points at dropout rates of 0.05 and 0.10, respectively. At the strongest rate of 0.20, performance decreases by 4.72 points but remains the highest mean, exceeding CEBRA by 3.32 points. The larger standard deviation at $r = 0.20$ indicates that severe channel loss affects different runs or sessions less uniformly. Moreover, the differences from CEBRA and MMD at the strongest corruption level do not reach statistical significance. Overall, the gradual degradation rather than abrupt failure indicates that the bounded plastic path and stable-path fusion retain useful decoding structure under progressive channel loss.

Gaussian Noise. Gaussian count noise evaluates sensitivity to increasingly noisy target neural observations. We apply noise levels $r \in \{0.05, 0.10, 0.20\}$ to both the target adaptation and evaluation streams.

Ours decreases by 0.67, 2.01, and 6.63 points relative to the clean condition as the noise level increases. Despite the larger degradation under the strongest perturbation, it retains the highest mean at every noise level and exceeds the strongest competing result by 2.40 points at $r = 0.20$. Gaussian noise causes a larger reduction than channel dropout at the highest tested level, suggesting that

Table 10: Robustness to target-channel dropout. † denotes no significant difference from Ours according to a two-sided paired t-test.

Method	Clean	$r = 0.05$	$r = 0.10$	$r = 0.20$
KF	-112.63 ± 0.00	-117.37 ± 0.00	-130.41 ± 16.52	-130.72 ± 51.93
LSTM	6.57 ± 0.46	5.22 ± 0.56	4.76 ± 0.43	3.53 ± 0.73
Transformer	6.65 ± 2.27	5.12 ± 2.55	4.51 ± 2.78	4.14 ± 1.21
CORAL	34.72 ± 0.19	33.09 ± 0.21	30.50 ± 0.40	28.90 ± 0.28
MMD	35.06 ± 0.24	33.44 ± 0.45	31.78 ± 0.16	30.41 ± 0.33†
DANN	34.24 ± 1.72	32.36 ± 0.46	30.62 ± 1.14	28.27 ± 1.41
NoMAD	34.47 ± 0.32	33.78 ± 0.70	30.20 ± 0.62	28.01 ± 0.68
CEBRA	35.51 ± 0.30	34.23 ± 0.30	32.32 ± 0.20	30.60 ± 0.27†
Ours	**38.64 ± 0.09**	**38.01 ± 0.23**	**37.11 ± 0.37**	**33.92 ± 1.50**

Table 11: Robustness to Gaussian noise added to the target neural signal.

Method	Clean	$r = 0.05$	$r = 0.10$	$r = 0.20$
KF	-112.63 ± 0.00	-127.02 ± 0.00	-259.47 ± 0.18	-878.39 ± 0.84
LSTM	6.57 ± 0.46	6.37 ± 0.80	-0.08 ± 0.04	-15.08 ± 0.03
Transformer	6.65 ± 2.27	3.18 ± 3.21	1.93 ± 0.01	-2.28 ± 0.05
CORAL	34.72 ± 0.19	31.99 ± 0.30	30.21 ± 0.11	24.61 ± 0.21
MMD	35.06 ± 0.24	33.34 ± 0.13	32.43 ± 0.20	28.18 ± 0.22
DANN	34.24 ± 1.72	30.24 ± 0.36	28.68 ± 0.17	23.50 ± 0.03
NoMAD	34.47 ± 0.32	32.47 ± 0.49	31.80 ± 0.77	29.61 ± 0.71
CEBRA	35.51 ± 0.30	34.09 ± 0.12	32.89 ± 0.18	28.81 ± 0.24
Ours	**38.64 ± 0.09**	**37.97 ± 0.15**	**36.63 ± 0.18**	**32.01 ± 0.35**

widespread perturbation of the observed counts is more difficult than losing a limited subset of channels. Nevertheless, the method preserves its performance advantage throughout the complete corruption range. Together, the two robustness evaluations show that the proposed source-constrained adaptation degrades continuously rather than catastrophically under increasingly corrupted target observations.

C.7 Complete Ablation Study

Motivation and Protocol. The main paper visualizes representative ablations of the stable-plastic framework. We provide the complete numerical results for five principal variants on the 29 target sessions of Indy-96. For each session, performance is first averaged across five independent runs. The resulting 29 matched session-level averages are then compared against the full model using a two-sided paired t-test. The variant performance is recovered from the full-model mean, and the corresponding performance difference is reported in the finalized experimental record.

Results and Analysis. Removing the plastic path causes the largest degradation, reducing performance from 41.46 to 28.59. This 12.87-point decrease confirms that the conservative stable correction alone cannot recover the target-specific changes encountered across Indy-96. The plastic path is therefore the primary source of cross-session adaptation. Nevertheless, removing the stable path also reduces performance from 41.46 to 41.10 ($p = 1.42 \times 10^{-2}$), showing that the stable prediction remains a useful complement when plastic correction is unnecessary or less reliable. Removing the fixed teacher decreases performance to 30.08, corresponding to an 11.38-point degradation. This result demonstrates that the plastic path requires a fixed target-specific anchor rather than unconstrained self-adjustment. Replacing adaptive fusion with a fixed equal-weight rule reduces performance to 37.76, supporting session-dependent control over the stable and plastic contributions. Removing the trust constraint causes a smaller but significant decrease to 40.84. Together,

Table 12: Complete ablation results on the Indy-96 target sessions. Performance is measured by $100R^2$. The p-values compare each variant against the full model using two-sided paired t-tests over the matched session-level averages ().

Method	$100R^2$	p-value
w/o Plastic path	28.59	3.99×10^{-5}
w/o Stable path	41.10	1.42×10^{-2}
w/o Adaptive fusion	37.76	2.80×10^{-4}
w/o Fixed teacher	30.08	1.29×10^{-3}
w/o Trust constraint	40.84	5.33×10^{-4}
Full model	**41.46**	–

these results show that reliable cross-session recalibration requires not only sufficient plasticity, but also a stable complementary path, a fixed teacher that constrains the direction of adaptation, and a reliability-aware fusion rule.

C.8 Representation Alignment and Branch Complementarity

We examine how Stable and Plastic corrections affect distribution alignment and task information across natural target sessions. The analysis covers 29 Indy-96, 6 Indy-192, and 9 Loco-192 target sessions, each evaluated with five matched model runs. Within each recording chain and run, Raw, Stable, and Plastic representations share the same frozen source-fitted coordinates. We measure the normalized covariance gap to the source, target velocity R^2 from a frozen source-trained ridge probe, and domain separability defined as $2|\mathrm{AUROC} - 0.5|$. Lower covariance gap and domain separability indicate stronger statistical alignment, while higher probe R^2 indicates better transfer of the source-trained linear readout. This offline analysis uses overlapping target support for unlabeled adaptation and representation evaluation, with target kinematics used exclusively for post-hoc scoring. We compute 95% pointwise percentile bootstrap intervals from 5,000 draws, resampling sessions within each recording chain and sharing resampled model-run indices across sessions, states, and metrics.

Alignment and Task Information. Figure 5 shows that Plastic achieves a lower covariance gap than Stable on all three datasets. The probe comparison reveals dataset-dependent advantages: Plastic improves over Stable by 0.065 on Indy-96 (95% CI $[0.031, 0.091]$) and 0.036 on Indy-192 ($[0.017, 0.057]$), while Stable performs better on Loco-192, where the Plastic-Stable difference is -0.065 ($[-0.114, -0.016]$). On Loco-192, this probe advantage for Stable coexists with substantially lower domain separability for Plastic, illustrating a tradeoff between statistical alignment and preservation of source-readable task information. Indy-96 exhibits a complementary pattern: Stable achieves lower domain separability than Plastic, whereas Plastic provides better probe transfer.

These results show that distribution alignment and task information preservation capture distinct aspects of adaptation. The relative advantages of Stable and Plastic depend on both the dataset and the diagnostic, supporting the retention of both paths and motivating reliability-aware control of their contributions. This analysis characterizes branch representations; session-level gate evaluation separately assesses whether the gate selects useful plastic corrections.

C.9 Session-Level Gate Reliability

We examine whether the label-free gate identifies useful plastic corrections under a causal calibration-to-prediction protocol. For each of the 44 target sessions, the first 25% of the neural stream is used for adaptation, output alignment, and gate estimation. All target-dependent quantities are then frozen, and decoding is evaluated on subsequent complete reaches. Stable, Plastic, and Fusion share identical evaluation samples, and target kinematics are accessed only after predictions are fixed. Branch scores and gate values are averaged across five independent runs within each session. We define the session-level plastic advantage as:

$$\Delta_i = R^2_{\mathrm{Plastic},i} - R^2_{\mathrm{Stable},i}. \tag{57}$$

This yields 39 sessions with $\Delta_i > 0$ and 5 with $\Delta_i < 0$.

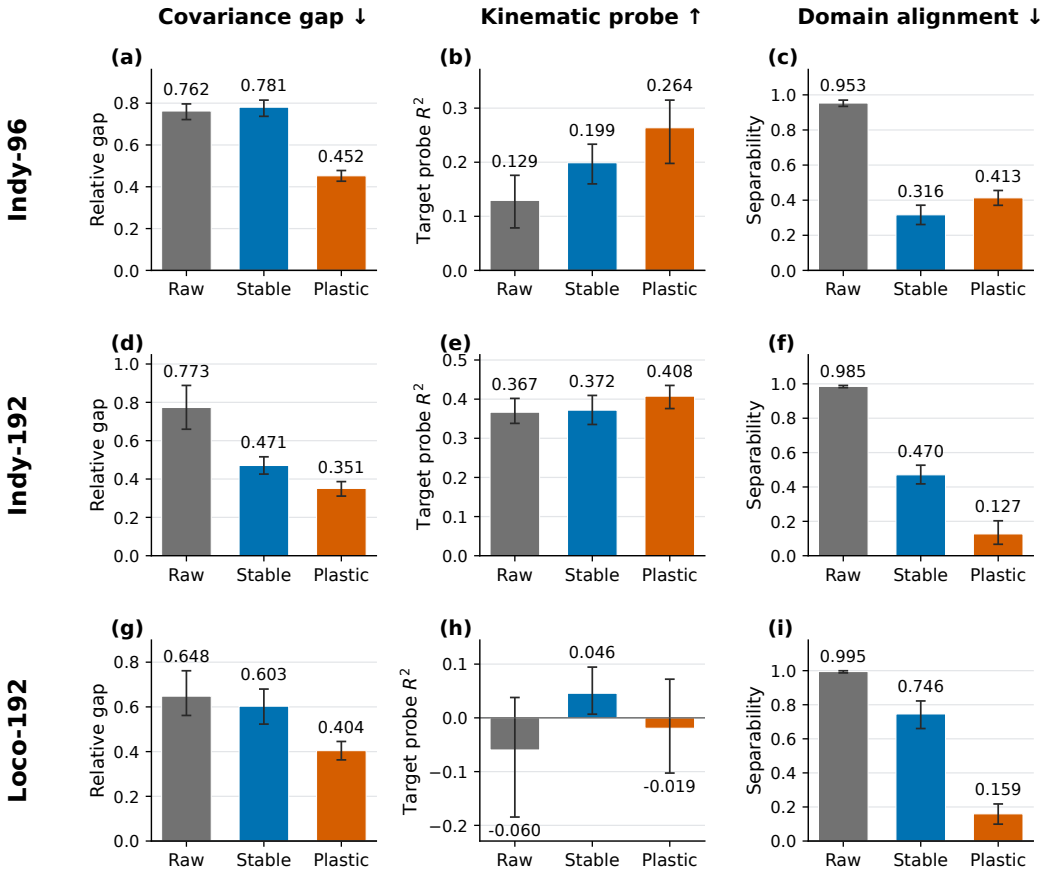


Figure 5: Representation alignment across natural target sessions. Rows correspond to Indy-96, Indy-192, and Loco-192; columns show covariance gap, frozen-source probe R^2, and domain separability. Bars report means over target sessions and model runs, with 95% bootstrap intervals. Arrows indicate the preferred direction.

Gate Discrimination and Admission. The final gate achieves an AUROC of 0.692 for identifying sessions with positive plastic advantage. Its average precision is 0.945, compared with a positive-session proportion of 0.886. Using the existing diagnostic threshold $g_i > 0.2$, the gate admits 38 of the 39 beneficial sessions and rejects two of the five harmful sessions. The Stable-only need score, which summarizes neural and decoded-output discrepancies, achieves an AUROC of 0.482. These observations support assessing the reliability of the optimized plastic candidate when determining its contribution to prediction (Figure 6). The need score serves as a diagnostic comparator and contributes no additional multiplicative factor to the default gate.

Decoding Gains and Downside Control. The full gate achieves an average improvement of 11.70 points over Stable on the $100R^2$ scale. Relative to using the complete Plastic branch, fusion reduces the number of sessions performing below Stable from five to two. The mean gain among the worst 10% of sessions improves from -4.47 to -3.19 points. These results demonstrate useful adaptation gains together with reduced observed downside across target sessions.

Branch advantage and fusion utility characterize complementary aspects of the decision. A negative Plastic-Stable difference identifies a harmful complete replacement, while partial mixing can still improve the stable prediction. Accordingly, the analysis evaluates both branch admission and the decoding performance of the actual fused output. The observed reductions in harmful sessions and tail loss support reliability-guided fusion, with the harmful-session rejection estimate based on five sessions.

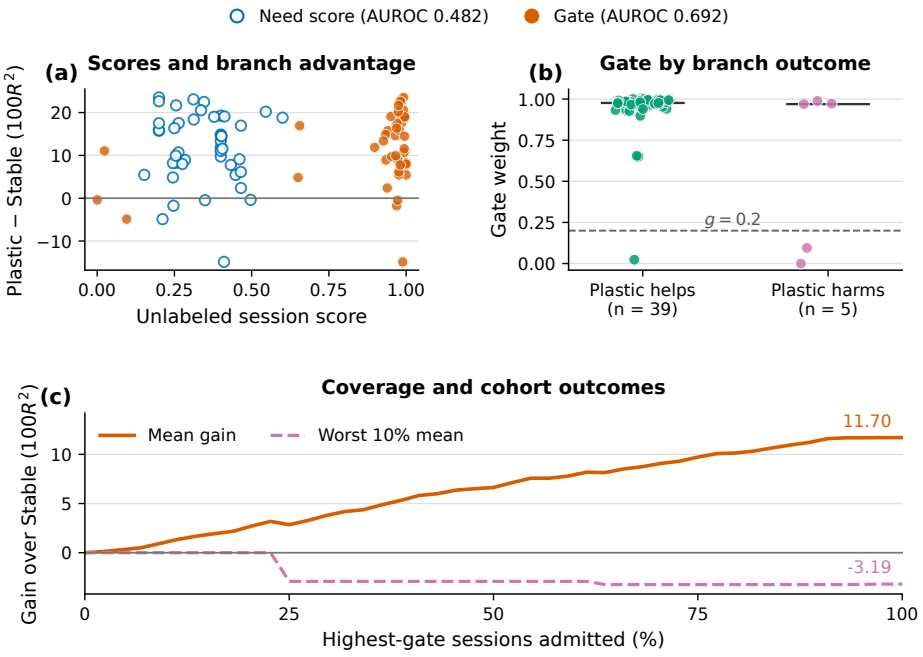


Figure 6: Session-level gate analysis under causal evaluation. **(a)** Unlabeled scores versus Plastic–Stable advantage on future evaluation samples. **(b)** Gate weights grouped by the sign of the plastic advantage; the dashed line marks the diagnostic threshold $g = 0.2$. **(c)** Coverage analysis admits sessions in descending gate order, applies soft fusion to admitted sessions, and retains Stable predictions elsewhere. Both curves use all 44 sessions, and the worst-10% statistic averages the five lowest session gains.

C.10 Additional Hyperparameter Sensitivity

The main paper examines sensitivity to adaptation steps, learning rate, output-alignment strength, and teacher weight. We further evaluate four objective weights that regulate decoded-output compatibility and fusion reliability: the readout, prior, dynamics, and trust weights. Each parameter is varied independently over seven values on the 29 Indy-96 target sessions, while all other settings remain fixed. The default values are 0.45, 0.15, 0.08, and 0.12, respectively. Each point reports the mean and sample standard deviation over five independent runs. These target-session sweeps are used only as sensitivity diagnostics; the main configuration remains the one selected from source-derived pseudo-target streams.

Figure 7(a) shows that increasing the readout weight from 0.15 to 0.60 improves performance from 40.43 ± 0.09 to 41.65 ± 0.06. Performance remains high at 0.75, where it reaches 41.56 ± 0.04. Thus, the default value of 0.45 lies inside a broad high-performance region rather than at a narrow isolated optimum. This pattern indicates that sufficiently weighting the differentiable decoded-output constraint is beneficial, whereas further increases yield only minor changes.

The prior weight in Figure 7(b) is comparatively insensitive. Across the complete range from 0.0375 to 0.45, mean performance remains between 41.36 and 41.66. The higher-weight settings produce larger run-to-run variation, suggesting that stronger prior matching does not consistently improve the solution even though its mean remains competitive. The default value of 0.15 therefore provides a stable balance without relying on the highest tested weight.

The dynamics weight is the most stable parameter. As shown in Figure 7(c), all seven settings remain between 41.41 and 41.47, corresponding to a total range of only 0.06 points. This result indicates that the temporal-dynamics term acts primarily as a regularizing constraint and that performance is not sensitive to its precise coefficient.

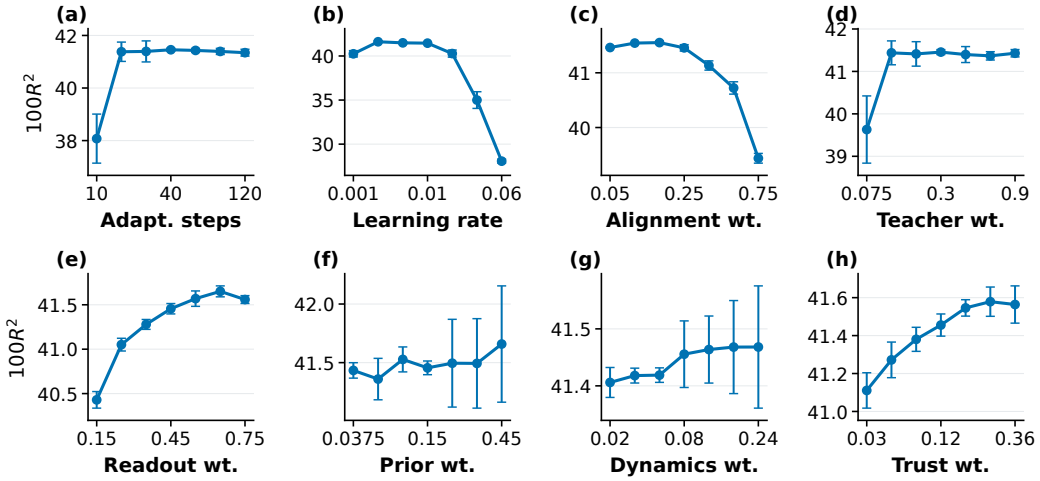


Figure 7: Hyperparameter sensitivity on Indy-96. Each panel varies one hyperparameter while keeping the others at their default values. Points and error bars show the mean and one sample standard deviation across five runs.

Finally, the trust weight produces a gradual improvement from 41.11 ± 0.09 at 0.03 to 41.58 ± 0.08 at 0.24, followed by a similar result of 41.56 ± 0.10 at 0.36. The broad plateau between 0.18 and 0.36 supports the role of trust regularization while showing that the method does not depend on a single exact value. Overall, the four additional sweeps place the default configuration within stable high-performing regions. Together with the sensitivity results in the main paper, they indicate that the reported gains do not result from narrowly tuned target-session hyperparameters.

D Baselines and Settings

We compare our method with KF, LSTM, and Transformer for source-trained decoding, CORAL, MMD, and DANN for distribution alignment, and NoMAD and CEBRA for dynamics-guided and contrastive representation learning. The five cross-session baselines undergo extensive configuration searches on the One2All calibration subset, with 128 candidates per method and recording sequence. Configurations are ranked by mean calibration-session R^2 minus half its session-level standard deviation. The calibration subset is contained in the reported One2All cohort, while within-session adaptation uses unlabeled neural activity. The tables summarize up to five principal search dimensions per method. Tuples list the final values for Indy-96, Indy-192, and Loco-192, respectively, and "Fixed" identifies settings without a reported search range.

D.1 Our Method

Our framework combines stable channel-wise correction with a bounded plastic adapter and reliability-guided output fusion. The source representation and decoder remain frozen, while the plastic adapter learns from unlabeled target activity under fixed-teacher and source-compatibility constraints. The configuration follows the source-only selection protocol described in Section 3.7.

D.2 Kalman Filter

KF represents velocity as a linear Gaussian dynamical state and neural features as observations. The transition, observation, and noise parameters are estimated from labeled source data with ridge stabilization. Target velocity is decoded through recursive filtering with frozen model parameters.

Table 13: Configuration of our method.

Hyperparameter	Search space	Final value
PCA dimension	Fixed	32
Adapter rank	Fixed	64
Adaptation updates	Fixed	40
Learning rate	$\{3, 5, 10, 50\} \times 10^{-3}$	$(10, 5, 10) \times 10^{-3}$
Teacher weight λ_{tea}	$\{0.1, 0.2, 0.3, 0.4, 0.5\}$	$(0.3, 0.3, 0.4)$
Trust weight λ_{tru}	$\{0.10, 0.12, 0.15, 0.20\}$	$(0.12, 0.15, 0.12)$
Teacher strength η_z	$\{1.00, 1.25, 1.50, 2.00\}$	$(1.25, 1.25, 1.50)$
Output alignment η_o	$\{0.20, 0.22, 0.25, 0.30\}$	$(0.25, 0.25, 0.25)$

Table 14: Search ranges and recorded final settings for KF.

Hyperparameter	Search space	Final value
Ridge coefficient λ	$\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$	10^{-3}
Process covariance regularizer ϵ_Q	$\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}\}$	10^{-5}
Observation variance floor ϵ_R	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-4}
Initial covariance regularizer ϵ_{P_0}	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$	10^{-4}

D.3 LSTM

LSTM uses gated recurrent states to model temporal dependencies in neural activity (Hochreiter, 1997). Our source-trained implementation processes causal neural-feature sequences and predicts velocity through a linear readout. Training uses Adam and chronological source validation, and the selected checkpoint remains frozen during target decoding.

Table 15: Search ranges and recorded final settings for LSTM.

Hyperparameter	Search space	Final value
Hidden size	$\{32, 64, 128, 256\}$	64
Number of layers	$\{1, 2, 3\}$	1
Sequence length	$\{16, 32, 64, 128\}$	32
Learning rate	$\{10^{-4}, 3 \times 10^{-4}, 10^{-3}, 3 \times 10^{-3}\}$	10^{-3}
Batch size	$\{64, 128, 256\}$	128

D.4 TRANSFORMER

Transformer models temporal interactions through self-attention (Vaswani et al., 2017). Our implementation uses causal attention, learned positional embeddings, and a linear velocity head. It follows the same source supervision, sequence length, and checkpoint-selection procedure as the LSTM.

D.5 CORAL

CORAL reduces distribution differences through correlation alignment (Sun et al., 2016). Our neural adaptation variant jointly matches source and target moments at the hidden and latent layers while preserving source velocity supervision. The learned representation is decoded through the source-trained nonlinear readout.

D.6 MMD

MMD measures distribution differences using kernel mean embeddings (Gretton et al., 2012). Our implementation combines source velocity regression with an RBF-kernel MMD loss on source and

Table 16: Search ranges and recorded final settings for Transformer.

Hyperparameter	Search space	Final value
Model dimension d_{model}	$\{32, 64, 128, 256\}$	64
Number of layers	$\{1, 2, 4\}$	2
Attention heads	$\{2, 4, 8\}$	4
Dropout	$\{0, 0.05, 0.1, 0.2\}$	0.05
Learning rate	$\{10^{-4}, 3 \times 10^{-4}, 10^{-3}, 3 \times 10^{-3}\}$	10^{-3}

Table 17: CORAL search dimensions and final values.

Hyperparameter	Search space	Final value
Latent dimension	$\{8, 16, 32, 48, 64\}$	$(16, 16, 48)$
Hidden dimension	$\{64, 128, 256, 512\}$	$(256, 256, 256)$
Training updates	$\{250, 400, 600\}$	$(250, 400, 600)$
Learning rate	$\{3, 5, 10, 50\} \times 10^{-4}$	$(10, 3, 10) \times 10^{-4}$
Alignment weight	$\{0.1, 0.15, 0.3, 0.6, 1, 2\}$	$(1, 0.3, 1)$

target latent representations. The kernel bandwidth is determined from pairwise feature distances, and target observations contribute only to distribution alignment.

Table 18: MMD search dimensions and final values.

Hyperparameter	Search space	Final value
Latent dimension	$\{8, 16, 32, 64, 128\}$	$(16, 16, 64)$
Hidden dimension	$\{64, 128, 256, 512\}$	$(128, 256, 256)$
Training updates	$\{250, 400, 600\}$	$(400, 400, 400)$
Learning rate	$\{3, 5, 10, 50\} \times 10^{-4}$	$(3, 10, 5) \times 10^{-4}$
MMD weight	$\{0.1, 0.15, 0.3, 0.6, 1, 2\}$	$(1, 0.15, 0.6)$

D.7 DANN

DANN learns task-relevant, domain-invariant representations through a domain discriminator and gradient reversal (Ganin et al., 2016). Our implementation replaces the classification objective with source velocity regression and trains the discriminator using source and target domain identities. The reversal strength increases linearly during optimization.

D.8 NoMAD

NoMAD aligns new neural recordings to a reference latent dynamical model (Karpowicz et al., 2025). Our adapted implementation retains the source-anchored read-in adaptation principle using a source-trained encoder with reconstruction and temporal regularization. Target optimization updates the read-in transform using reconstruction, latent-moment, temporal-increment, and identity constraints while keeping the source encoder frozen. This implementation uses a feedforward latent model in place of the original LFADS generator. The source model receives 900 optimization updates, followed by 450 target read-in updates.

D.9 CEBRA

CEBRA learns neural embeddings through contrastive objectives conditioned on temporal or behavioral information (Schneider et al., 2023). Our adapted implementation uses temporal positive pairs and an InfoNCE objective, together with source velocity supervision and an optional latent MMD constraint. Target activity contributes temporal and distributional information, while behavioral su-

Table 19: DANN search dimensions and final values.

Hyperparameter	Search space	Final value
Latent dimension	$\{8, 16, 32, 48, 64\}$	$(48, 32, 32)$
Hidden dimension	$\{64, 128, 256, 512\}$	$(256, 256, 128)$
Training updates	$\{250, 400, 600\}$	$(400, 400, 250)$
Maximum GRL strength	$\{0.05, 0.15, 0.3, 0.6, 1\}$	$(0.6, 0.3, 0.3)$
Domain-loss weight	$\{0.1, 0.2, 0.4, 0.8, 1.0\}$	$(0.4, 0.2, 0.2)$

Table 20: Search dimensions for the NoMAD implementation.

Hyperparameter	Search space	Final value
Latent dimension	$\{8, 16, 32, 48, 64\}$	$(64, 64, 48)$
Learning rate	$\{3, 5, 10, 50\} \times 10^{-4}$	$(10, 10, 3) \times 10^{-4}$
Latent-alignment weight	$\{0.1, 0.2, 0.4, 0.8, 1.0\}$	$(0.2, 0.8, 0.2)$
Dynamics-alignment weight	$\{0.01, 0.02, 0.1, 0.4, 1.0\}$	$(0.4, 0.02, 0.4)$
Identity weight	$\{0.005, 0.01, 0.02, 0.05, 0.1\}$	$(0.05, 0.01, 0.05)$

pervision remains restricted to the source. The contrastive temperature is 0.2, and each anchor uses 16 negative samples.

Table 21: Search dimensions for the CEBRA implementation.

Hyperparameter	Search space	Final value
Output dimension	$\{8, 16, 32, 48, 64\}$	$(16, 16, 32)$
Learning rate	$\{3, 5, 10, 50\} \times 10^{-4}$	$(5, 5, 10) \times 10^{-4}$
Time offset	$\{1, 3, 5, 10\}$	$(5, 3, 5)$
Temporal contrastive weight	$\{0.01, 0.05, 0.1, 0.4, 0.8, 1.0\}$	$(0.05, 0.05, 0.8)$
Domain-alignment weight	$\{0.05, 0.1, 0.2, 1.0\}$	$(0.1, 0.2, 0.2)$

E More Related Work

Online Recalibration and Test-Time Adaptation. Online recalibration updates a deployed decoder from newly observed data without requiring explicit target labels. In neural decoding, retrospective inference estimates latent movement intentions from task trajectories and uses them to continually retrain the decoder (Wilson et al., 2025). More broadly, test-time adaptation updates pretrained models using unlabeled deployment data. Tent minimizes prediction entropy by adapting a restricted set of parameters (Wang et al., 2021); EATA filters unreliable or redundant samples and constrains parameter changes (Niu et al., 2022); CoTTA combines teacher-based consistency, augmentation, and partial source-model restoration (Wang et al., 2022); and SAR improves adaptation stability through reliable-sample selection and model recovery (Niu et al., 2023). These methods are mainly used to determine which target samples or parameters should be updated. The model after adaptive adjustment usually replaces the source model and is used for subsequent predictions. However, changes in neural distribution do not necessarily indicate the presence of behavior-related drift. Unconstrained updates of noisy neural data streams may reduce decoding performance. Therefore, we simultaneously monitor deviations in the neural and decoding spaces. Only when the evidence in both spaces is consistent do we introduce bounded plasticity correction and maintain a conservative and stable operation path through a prediction gating mechanism with reliability perception capabilities.