LET THE QUERY LEAD: FACTORIZED REFERENCE CONDITIONING FOR WEARABLE RECOGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Wearable recognition models must generalize to unseen wearers whose sensor patterns can differ from those observed during training. Unlabeled recordings from a new wearer can provide useful context, but their statistics mix persistent wearer- or acquisition-dependent variation with the activities or states that happen to be recorded. Such context can be informative without specifying how an individual prediction should change. We introduce Query–Subject Factorization (QSF), a query-led interface for reference-conditioned wearable recognition. Starting from a frozen population recognizer, QSF learns a query-only correction that activates shared correction modes. References supply gains that act only through these query-dependent activations. Activity-relative residual aggregation with bounded per-class evidence yields a compact reference state that is reusable across queries and requires no target labels or gradient updates at deployment. The interface exactly recovers the reference-free predictor when references are absent and provides an explicit bound tying reference-induced changes to query activation strength. Across eight activity, sleep, and affect recognition datasets, the complete system ranks first in accuracy on all six activity-recognition tasks and among the top three on every task. With a common frozen upstream predictor, QSF achieves higher mean weighted F1 than a compact joint-conditioning baseline on all eight tasks without a larger conditioning head. Targeted decision-level analyses link this advantage to fewer harmful changes to correct query-only predictions, while controlled activation interventions show that the query-activated mode mixture affects recognition. These findings support treating reference context as a modulator of query-led corrections rather than an independent source of prediction changes.

1 INTRODUCTION

Wearable recognition models infer activities, sleep stages, and affective states from sensor measurements. A model trained on one population must often recognize the same classes in previously unseen wearers, despite differences in movement style, physiology, sensor placement, and acquisition conditions (Qian et al., 2021; Wang et al., 2023). For a sensor window awaiting classification (the query), other unlabeled recordings from the same wearer can serve as references. These references offer personalization context; the challenge is to determine how they should influence the query's prediction.

Reference observations mix persistent wearer- or acquisition-dependent variation with the activities or states that happen to be recorded. Consider two reference sets collected from the same wearer under the same sensing configuration, one dominated by walking and the other by sitting. Their pooled feature statistics can differ substantially even though the wearer and measurement setup have not changed. An adjustment driven by this difference need not be appropriate for the query and can disturb an already-correct prediction. This exposes a distinction between having informative context and having evidence for a particular correction. In short, *context is not correction*.

Existing cross-user recognition methods address wearer variation through representations that generalize across people (Qian et al., 2021), while test-time adaptation exploits unlabeled target observations through self-supervision, entropy minimization, normalization updates, or classifier adjustment (Sun et al., 2020; Wang et al., 2021; Iwasawa & Matsuo, 2021; Niu et al., 2022; Wang et al., 2023). These approaches provide mechanisms for incorporating target observations into prediction; we study a

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.

complementary question: *how should the current query constrain the influence of reference context?* Our premise is that the query should determine which correction modes are active, while references modulate their effect on the prediction.

We introduce Query–Subject Factorization (QSF), a query-led reference-conditioning interface (Figure 1). Starting from a frozen source-trained population recognizer, we learn a query-only correction to obtain the reference-free DIRECT predictor. This correction activates a small set of shared modes. A compact, class-relative summary of the unlabeled references supplies gains that modulate these activations, producing a bounded increment to the DIRECT logits. Consequently, a reference gain cannot affect the prediction through a mode that the query leaves inactive. This restriction is built into the prediction interface rather than learned solely from data.

We evaluate QSF across eight activity, sleep, and affect recognition datasets. In our complete-system benchmark, QSF achieves the highest mean accuracy on all six activity-recognition tasks and ranks among the top three on every task. Our contributions are threefold:

- We distinguish informative reference context from query-specific correction and assign them complementary roles: the query activates correction modes, while references condition their effect.
- QSF provides a bound tying reference-induced logit changes to query activation strength and exactly recovers DIRECT when references are absent. All learned parameters remain fixed at deployment; reference states require no target labels or gradient updates and can be reused across queries sharing the same eligible reference set.
- With a common frozen upstream predictor, QSF achieves higher mean weighted F1 than compact joint conditioning on all eight tasks without a larger conditioning head. Targeted analyses link these gains to better preservation of correct DIRECT predictions and demonstrate the predictive role of the query-activated mode mixture.

2 RELATED WORK

Cross-user recognition and target-time adaptation. GILE learns representations for cross-person activity recognition without target-domain data (Qian et al., 2021), whereas OFTTA uses unlabeled target observations to adjust normalization and prototypes without gradient optimization (Wang et al., 2023). Recent work addresses the sensitivity of adaptation to target-data composition and unnecessary updates. TRIBE combines balanced normalization with anchored self-training under class imbalance and continual shifts (Su et al., 2024). POEM detects shifts in predictive entropy and adapts the classifier by online entropy matching (Bar et al., 2024). NEO instead re-centers latent features without optimization (Murphy et al., 2026). QSF studies a complementary constraint on the prediction interface: reference influence is restricted to correction modes activated by the current query.

Residual adaptation and conditional modulation. FiLM conditions feature-wise affine transformations on auxiliary information (Perez et al., 2018). More recently, COSA applies gated residual corrections to a frozen time-series forecaster, using context from recently observed ground truth and updating its adapter at test time (Im & Kwon, 2026). QSF uses unlabeled references while all learned parameters remain fixed at deployment. Its reference gains act only through query-dependent correction activations, so an inactive mode cannot transmit reference influence.

Low-rank factorization and reference aggregation. Low-rank bilinear pooling factorizes multiplicative interactions (Kim et al., 2017), while LoRA factorizes parameter updates (Hu et al., 2022). HydraLoRA shares one low-rank factor across expert-specific factors and routes their contributions according to the input (Tian et al., 2024). Deep Sets provides a general foundation for permutation-invariant set aggregation (Zaheer et al., 2017). Our factorization acts on class-score corrections, with modulation gains supplied by an activity-relative, permutation-invariant reference descriptor.

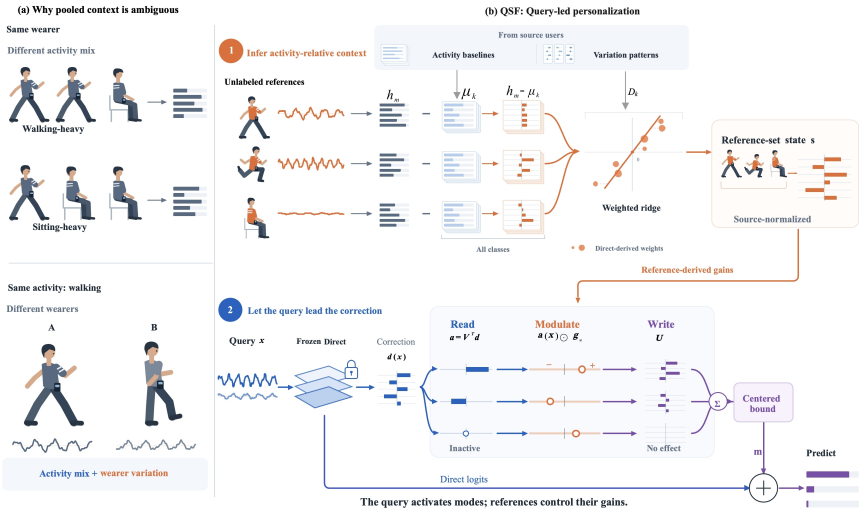


Figure 1: Query-led reference conditioning. (a) Pooled context mixes activity composition and wearer variation, leaving the appropriate query-specific correction ambiguous. (b) The query's learned correction activates shared modes, while an activity-relative reference state supplies their gains. The resulting bounded increment is added to DIRECT logits, confining reference influence to query-activated modes.

3 METHOD

3.1 QUERY CORRECTIONS AS THE PERSONALIZATION INTERFACE

Let $\mathcal{D}_s = \{(\boldsymbol{x}_i, y_i, u_i)\}_{i=1}^{N}$ contain labeled source-user windows from $K \geq 2$ classes. A source-trained population recognizer supplies $(\boldsymbol{h}, \boldsymbol{z}_0) = f_0(\boldsymbol{x}) \in \mathbb{R}^D \times \mathbb{R}^K$ and is then frozen. At deployment, the predictor receives a query $\boldsymbol{x}$ and an unordered, unlabeled reference set $\mathcal{R}_u$ from the same wearer under a fixed sensing configuration. Reference eligibility excludes the query and follows the declared evaluation protocol. The population model, correction heads, and global statistics are fitted on source users; the training and evaluation settings are described in Section 4.1. The computation has two roles: the query proposes a correction, and the reference set controls its subject-dependent modulation.

Softmax ignores a common logit shift. We therefore use the centered space $\mathcal{C} = \{\boldsymbol{a} \in \mathbb{R}^K : \mathbf{1}^\top \boldsymbol{a} = 0\}$ and its projector $\boldsymbol{P} = \boldsymbol{I}_K - K^{-1}\mathbf{1}\mathbf{1}^\top$. For $\tau > 0$, define

$$\mathcal{B}_\tau(\boldsymbol{a}) = \boldsymbol{P}[\tau \tanh(\boldsymbol{P}\boldsymbol{a}/\tau)], \tag{1}$$

with coordinatewise $\tanh$. Let $\boldsymbol{q}(\boldsymbol{x})$ be a descriptor of the frozen feature and prediction statistics (Appendix B.2). A small head forms

$$\boldsymbol{d}(\boldsymbol{x}) = \mathcal{B}_{\tau_d}(r_\theta(\boldsymbol{q}(\boldsymbol{x}))), \qquad \boldsymbol{z}_D(\boldsymbol{x}) = \boldsymbol{z}_0(\boldsymbol{x}) + \boldsymbol{d}(\boldsymbol{x}). \tag{2}$$

Its output layer starts at zero. Source classification with a correction-norm penalty learns the reference-free predictor DIRECT, which is then frozen. QSF uses $\boldsymbol{d}(\boldsymbol{x})$ as the query-side input to reference conditioning. The Stage-I objective is specified in Appendix B.2.

3.2 SUBJECT MODULATION OF QUERY-ACTIVATED MODES

The reference branch supplies a standardized state $\boldsymbol{s}_u \in \mathbb{R}^R$ and an attenuation coefficient $\rho_u \in [0, 1]$, constructed in Section 3.3. We use q shared read/write modes $\boldsymbol{V}, \boldsymbol{U} \in \mathbb{R}^{K \times q}$ with column-normalized factors, and a gain map $\boldsymbol{W} \in \mathbb{R}^{q \times R}$:

$$\begin{aligned} \boldsymbol{a}(\boldsymbol{x}) &= \boldsymbol{V}^\top \boldsymbol{d}(\boldsymbol{x}), & \boldsymbol{g}_u &= \alpha \rho_u \tanh(\boldsymbol{W}\boldsymbol{s}_u), \\ \boldsymbol{m}_u(\boldsymbol{x}) &= \mathcal{B}_{\tau_m}(\boldsymbol{U}[\boldsymbol{a}(\boldsymbol{x}) \odot \boldsymbol{g}_u]), & \boldsymbol{z}_S(\boldsymbol{x}, \mathcal{R}_u) &= \boldsymbol{z}_D(\boldsymbol{x}) + \boldsymbol{m}_u(\boldsymbol{x}), \end{aligned} \tag{3}$$

where $\alpha \geq 0$ and $\tau_m = \tau_d \max\{\alpha, 10^{-6}\}$. The query activates $a_j(\boldsymbol{x}) = \boldsymbol{V}_{:j}^\top \boldsymbol{d}(\boldsymbol{x})$; the gain $g_{u,j}$ scales or reverses this activation, and the write direction $\boldsymbol{U}_{:j}$ maps the interaction into class scores. Read/write directions are shared across wearers, while the gains depend on their reference states. The factors store $q(2K + R)$ trainable coefficients.

Reference influence follows query activation. For fixed references, the centered pre-saturation operator is $\boldsymbol{L}_u = \boldsymbol{P}\boldsymbol{U}\,\mathrm{diag}(\boldsymbol{g}_u)\boldsymbol{V}^\top\boldsymbol{P}$, with rank at most $\min\{q, K-1\}$. Here $\|\cdot\|_2$ denotes the Euclidean norm for vectors and the spectral norm for matrices. Since $\mathcal{B}_{\tau_m}$ is nonexpansive,

$$\|\boldsymbol{m}_u(\boldsymbol{x})\|_2 \leq \min\left\{\tau_m\sqrt{K}, \alpha\rho_u \|\boldsymbol{P}\boldsymbol{U}\|_2 \|\boldsymbol{a}(\boldsymbol{x})\|_2\right\}. \tag{4}$$

The bound ties reference influence to query activation strength, rather than assigning it a query-independent magnitude. If $\boldsymbol{a}(\boldsymbol{x}) = \boldsymbol{0}$, the prediction is exactly DIRECT for every reference state; $\rho_u = 0$ also gives exact recovery. These properties hold at finite correction amplitudes. Appendix B.6 extends the analysis to changes between reference sets and their effect on pairwise class margins.

3.3 ACTIVITY-RELATIVE REFERENCE INFERENCE

Here, class denotes an activity, sleep stage, or affective state. Each source class has a prototype $\boldsymbol{\mu}_k$ and a fixed residual dictionary $\boldsymbol{D}_k \in \mathbb{R}^{D\times R}$. The dictionaries use classwise SVD and seeded completion, as detailed in Appendix B.3. Together they define an activity-relative reference descriptor used consistently during source learning and deployment. We use $T > 0$, $\tau_c > 0$, and $\gamma > 0$ (with $\gamma = 2$ by default). For window m, let $\boldsymbol{p}_m = \mathrm{softmax}(\boldsymbol{z}_D(\boldsymbol{x}_m)/T)$ and $\boldsymbol{r}_{m,k} = \boldsymbol{h}_m - \boldsymbol{\mu}_k$. Set $\widetilde{p}_{m,k} = \max\{p_{m,k}, \epsilon\}$, with $\epsilon = 10^{-8}$, and use

$$a_k = \sum_m p_{m,k}, \qquad w_{m,k} = (1 - e^{-a_k/\tau_c})\frac{\widetilde{p}_{m,k}^{\gamma}}{\max\{\sum_n \widetilde{p}_{n,k}^{\gamma}, \epsilon\}}. \tag{5}$$

With the default $\gamma = 2$, the power emphasizes higher-confidence assignments within each predicted class. The total evidence weight assigned to each predicted class satisfies $\sum_m w_{m,k} \leq 1 - e^{-a_k/\tau_c} \leq 1$, with a cap that saturates as class mass increases. The resulting state solves one weighted ridge problem:

$$\begin{aligned}
\widehat{\boldsymbol{s}}_u &= \arg\min_{\boldsymbol{s}} \left\{ \frac{1}{2}\sum_{m,k} w_{m,k} \|\boldsymbol{r}_{m,k} - \boldsymbol{D}_k\boldsymbol{s}\|_2^2 + \frac{\lambda_s}{2}\|\boldsymbol{s}\|_2^2 \right\}, \\
\boldsymbol{\Lambda}_u &= \lambda_s \boldsymbol{I}_R + \sum_k \Big(\sum_m w_{m,k}\Big)\boldsymbol{D}_k^\top\boldsymbol{D}_k, \qquad \boldsymbol{b}_u = \sum_{m,k} w_{m,k}\boldsymbol{D}_k^\top\boldsymbol{r}_{m,k}, \\
\widehat{\boldsymbol{s}}_u &= \mathrm{solve}(\boldsymbol{\Lambda}_u, \boldsymbol{b}_u), \qquad \nu_u = R^{-1}\,\mathrm{tr}(\boldsymbol{\Lambda}_u^{-1}).
\end{aligned} \tag{6}$$

Positive λ_s makes the solution unique. Source episodes provide a subject-balanced center $\boldsymbol{m}_s$, positive coordinate scales $\boldsymbol{a}_s$, and a global mean-square scale v_s. For nonempty references,

$$\boldsymbol{s}_u = (\widehat{\boldsymbol{s}}_u - \boldsymbol{m}_s) \oslash \boldsymbol{a}_s, \qquad \rho_u = \frac{R^{-1}\|\boldsymbol{s}_u\|_2^2}{R^{-1}\|\boldsymbol{s}_u\|_2^2 + \nu_u/v_s + \epsilon}. \tag{7}$$

The state conditions the gains through $\boldsymbol{W}$, while the source-centered energy–curvature coefficient ρ_u attenuates their magnitude. For an empty set, $\boldsymbol{s}_u = \boldsymbol{0}$ and $\rho_u = 0$. All reference aggregations are symmetric, yielding permutation-invariant states for a fixed selected set. Source normalization, reference selection, and numerical conventions are given in Appendices B.4–B.5.

3.4 SOURCE LEARNING AND DEPLOYMENT

With DIRECT, the source dictionaries, and normalization statistics fixed, we learn the raw read/write factors and $\boldsymbol{W}$ from source episodes. Each episode pairs a labeled query with query-excluding same-user references, whose labels are not supplied to state inference. Learning uses DIRECT as a common anchor for both classification and regularization. Write $\ell_i^S = \mathrm{CE}(\boldsymbol{z}_{S,i}, y_i)$ and $\ell_i^D = \mathrm{CE}(\boldsymbol{z}_{D,i}, y_i)$

for a batch of B queries:

$$\mathcal{L}_S = \frac{1}{B}\sum_i \left[\ell_i^S + \lambda_a[\ell_i^S - \ell_i^D]_+ + \frac{\lambda_m}{K}\|\boldsymbol{m}_i\|_2^2\right] + \frac{\lambda_p}{|\mathcal{H}|}\sum_{i\in\mathcal{H}} \mathrm{KL}(\mathrm{softmax}(\boldsymbol{z}_{D,i}) \,\|\, \mathrm{softmax}(\boldsymbol{z}_{S,i}))\,. \tag{8}$$

Here $[t]_+ = \max\{t, 0\}$ and the loss weights are nonnegative. The set $\mathcal{H}$ contains source queries that DIRECT classifies correctly with a prescribed probability margin; its term is zero when $\mathcal{H}$ is empty. The anchor-relative hinge upweights the classification gradient when the personalized loss exceeds the query-only loss. The norm and distribution terms regularize the magnitude of the change and its effect on confident correct anchor predictions. Appendix B.7 specifies the coefficients, exact reductions, and small-random initialization.

Deployment keeps all learned quantities fixed and uses an $R \times R$ state solve and gain map without target labels or gradient updates (Appendix B.8). Reuse and measured costs are examined in Section 4.7.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets and metrics. We evaluate six activity-recognition datasets: DSADS (Altun et al., 2010), HAPT (Reyes-Ortiz et al., 2016), MotionSense (Malekzadeh et al., 2019), Shoaib-2014 (Shoaib et al., 2014), UCI-HAR (Anguita et al., 2013), and UniMiB SHAR (Micucci et al., 2017). We also evaluate SleepEDF-20 (Kemp et al., 2000; Kemp, 2013) for sleep staging and WESAD (Schmidt et al., 2018) for affect recognition. All datasets use disjoint source and target users; detailed configurations are provided in Appendix A. Benchmark accuracy (ACC) and weighted F1 (W-F1) pool predictions across held-out target users within each dataset; cross-dataset summaries weight datasets equally. We report means and sample standard deviations over five model seeds on fixed subject partitions. Default references for QSF are unlabeled, query-excluded windows from the same wearer's full session (temporal alternatives: Appendix C.9).

Baselines. We compare with CNN (Huang et al., 2023), Residual (Zhang et al., 2025), DeepConvLSTM (Ordóñez & Roggen, 2016), CALANet (Park et al., 2024), CEHAR (Huang et al., 2023), BIOT (Yang et al., 2023), DDLearn (Qin et al., 2023), DIVERSIFY (Lu et al., 2023), Decompose-WHAR (Xie et al., 2025), InvConvNet (Germain et al., 2025), and OFTTA (Wang et al., 2023). These eleven baselines cover supervised recognition, representation learning, and test-time adaptation. Appendix A.3 specifies their implementations and target-data access.

Model selection and target access. All architecture, hyperparameter, and checkpoint choices use source-side data only. Benchmark checkpoints maximize pooled ACC on the fixed held-out source users in Table 6, retaining the earliest epoch in a tie. The eight-task interface comparisons share a common source-selected Population/DIRECT pair and evaluate conditioning heads at the prespecified epoch-120 endpoint; matched training and evaluation settings are given in Appendix C.1. Interventions keep the trained model fixed unless explicitly identified as source-retrained controls. Target labels are used only for evaluation, except in marked diagnostics where they define reference composition; they never enter parameter fitting, checkpoint selection, or reference-state inference.

4.2 CROSS-TASK RECOGNITION BENCHMARK

Table 1 compares the complete QSF system with eleven recognition and adaptation baselines across eight activity, sleep, and affect recognition datasets. Among the evaluated systems, QSF achieves the highest equal-dataset mean accuracy of 93.45%, exceeding the strongest competing average of 90.53% for InvConvNet (Germain et al., 2025) by 2.91 percentage points. It achieves the highest mean accuracy on all six activity-recognition datasets and ranks among the top three on every task. Weighted F1 yields the same six activity-task wins and top-three rankings on all eight datasets. We next examine the contribution of the conditioning interface with the upstream predictor held fixed. Appendix D provides complementary training-recipe and capacity controls.

Table 1: Benchmark results (%). Each model reports ACC on the upper row and W-F1 on the lower row, using the same checkpoint. Entries are five-seed means with sample SD in subscripts; **bold** and underlined means mark the best and second-best results per dataset and metric.

Model	DSADS	HAPT	MotionSense	Shoaib-2014	UCI-HAR	UniMiB	SleepEDF-20	WESAD
CNN†	86.90$_{0.41}$	94.53$_{0.10}$	93.68$_{0.26}$	96.03$_{1.76}$	97.18$_{0.23}$	75.94$_{0.22}$	88.20$_{0.19}$	82.47$_{0.35}$
	87.33$_{0.48}$	94.50$_{0.10}$	93.70$_{0.26}$	95.85$_{1.91}$	97.17$_{0.24}$	75.33$_{0.26}$	88.00$_{0.26}$	80.16$_{0.89}$
Residual	85.94$_{1.69}$	94.88$_{0.17}$	94.82$_{0.27}$	95.53$_{2.43}$	96.89$_{0.10}$	75.60$_{0.55}$	89.09$_{0.41}$	85.35$_{0.63}$
	86.40$_{1.58}$	94.84$_{0.17}$	94.81$_{0.28}$	95.32$_{2.65}$	96.88$_{0.10}$	75.19$_{0.59}$	88.90$_{0.48}$	83.06$_{0.83}$
DeepConvLSTM	87.53$_{0.61}$	92.66$_{1.17}$	94.09$_{0.35}$	99.03$_{0.41}$	93.30$_{0.59}$	69.26$_{1.78}$	90.38$_{0.38}$	**87.69**$_{2.03}$
	87.53$_{0.50}$	92.68$_{1.16}$	94.08$_{0.35}$	99.03$_{0.41}$	93.30$_{0.59}$	69.45$_{1.52}$	90.34$_{0.44}$	**85.87**$_{2.91}$
CALANet	84.78$_{1.42}$	94.02$_{0.44}$	95.88$_{1.17}$	96.30$_{1.92}$	96.17$_{0.77}$	74.71$_{1.04}$	90.88$_{0.16}$	80.54$_{2.31}$
	84.81$_{1.39}$	94.11$_{0.35}$	95.89$_{1.15}$	96.27$_{1.93}$	96.15$_{0.77}$	74.84$_{0.89}$	90.71$_{0.14}$	74.98$_{3.46}$
BIOT	85.49$_{2.35}$	91.73$_{0.24}$	94.88$_{0.36}$	98.49$_{0.33}$	95.32$_{0.45}$	72.05$_{0.47}$	**91.97**$_{0.20}$	74.69$_{0.99}$
	85.09$_{2.42}$	91.77$_{0.22}$	94.86$_{0.37}$	98.50$_{0.32}$	95.30$_{0.45}$	71.76$_{0.50}$	**91.50**$_{0.35}$	70.27$_{0.62}$
CEHAR	90.11$_{0.18}$	95.23$_{0.16}$	93.70$_{0.41}$	99.59$_{0.05}$	97.04$_{0.08}$	73.53$_{1.00}$	90.54$_{0.24}$	84.18$_{0.45}$
	89.90$_{0.16}$	95.18$_{0.16}$	93.68$_{0.41}$	99.59$_{0.05}$	97.03$_{0.08}$	73.11$_{1.06}$	90.42$_{0.25}$	78.49$_{1.08}$
DDLearn	90.07$_{0.64}$	93.81$_{0.54}$	94.90$_{0.69}$	99.62$_{0.11}$	95.36$_{0.88}$	71.30$_{0.85}$	68.88$_{11.03}$	55.74$_{6.64}$
	89.99$_{0.72}$	93.69$_{0.53}$	94.90$_{0.68}$	99.62$_{0.11}$	95.35$_{0.88}$	70.75$_{0.93}$	57.52$_{17.07}$	41.27$_{10.76}$
OFTTA*	87.01$_{1.43}$	91.09$_{1.35}$	76.35$_{3.03}$	95.36$_{1.10}$	92.56$_{0.84}$	73.52$_{0.22}$	82.32$_{0.73}$	82.11$_{2.99}$
	86.67$_{1.90}$	91.22$_{1.00}$	75.98$_{3.15}$	95.37$_{1.10}$	92.49$_{0.83}$	74.15$_{0.17}$	83.29$_{0.97}$	79.05$_{4.91}$
DecomposeWHAR	88.39$_{0.66}$	93.23$_{0.30}$	95.40$_{0.39}$	99.37$_{0.39}$	95.52$_{0.44}$	71.52$_{0.53}$	82.93$_{0.20}$	83.76$_{0.48}$
	88.42$_{0.65}$	93.14$_{0.33}$	95.40$_{0.38}$	99.37$_{0.39}$	95.51$_{0.45}$	71.30$_{0.55}$	82.11$_{0.58}$	80.73$_{1.13}$
DIVERSIFY	85.35$_{1.51}$	91.65$_{0.33}$	91.87$_{1.31}$	97.85$_{1.32}$	93.93$_{0.48}$	66.98$_{1.14}$	88.71$_{0.66}$	83.06$_{1.20}$
	85.09$_{1.71}$	91.60$_{0.25}$	91.90$_{1.29}$	97.83$_{1.35}$	93.93$_{0.48}$	66.66$_{1.10}$	88.37$_{0.54}$	78.59$_{1.02}$
InvConvNet	89.00$_{0.51}$	96.55$_{0.18}$	96.43$_{0.35}$	98.36$_{1.06}$	97.48$_{0.25}$	72.82$_{0.55}$	91.74$_{0.11}$	81.89$_{0.39}$
	88.62$_{0.59}$	96.53$_{0.17}$	96.45$_{0.34}$	98.34$_{1.08}$	97.47$_{0.24}$	72.53$_{0.57}$	91.46$_{0.27}$	75.16$_{0.27}$
QSF (ours)	**93.52**$_{0.44}$	**97.18**$_{0.27}$	**97.03**$_{0.61}$	**99.78**$_{0.08}$	**98.86**$_{0.22}$	**84.58**$_{0.60}$	91.19$_{0.23}$	85.43$_{1.26}$
	93.43$_{0.45}$	**97.14**$_{0.25}$	**97.03**$_{0.61}$	**99.78**$_{0.08}$	**98.86**$_{0.22}$	**84.04**$_{0.69}$	91.17$_{0.20}$	81.99$_{2.54}$

†CNN uses its original UniMiB input layout; see Appendix A.3. *OFTTA uses target adaptation.

4.3 Why Factorize Reference Conditioning?

With the upstream predictor held fixed, QSF achieves higher mean W-F1 than compact Joint on all eight tasks (Figure 2; Appendix C.2). Joint maps the query correction and reference state through an MLP, sharing QSF's reference inference, attenuation, bounded output map, and source-training objective (Appendix C.1).

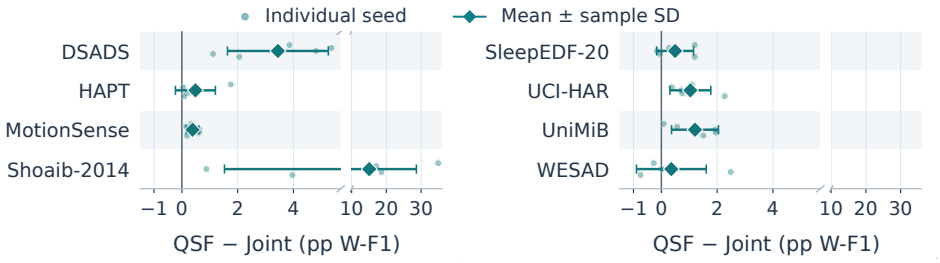


Figure 2: **QSF versus Joint across eight tasks.** Paired pooled-query W-F1 differences with a common frozen upstream predictor. Circles show individual seeds; diamonds and error bars show the five-seed mean ± sample SD. Both columns use the same broken-axis scale.

On HAPT and MotionSense, better preservation of correct DIRECT predictions is the main source of the accuracy advantage over compact Joint. QSF reduces mean harm rates by approximately 79% and 53%, respectively; Joint's additional repairs do not offset its greater harm (Figure 3a). Relative to DIRECT, QSF reduces the mean net accuracy decrease from 0.63 to 0.14 pp on HAPT and from 0.51 to 0.12 pp on MotionSense.

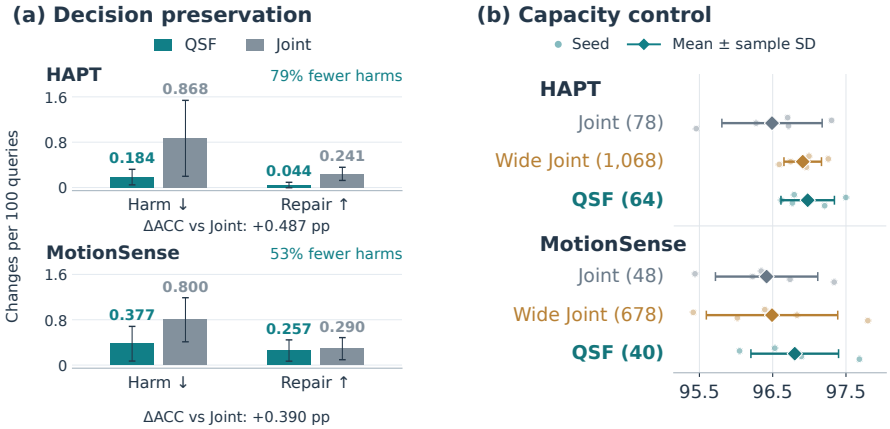


Figure 3: **Decision preservation and capacity controls on HAPT and MotionSense.** (a) Repairs and harms per 100 queries relative to DIRECT; ΔACC annotations give QSF–Joint differences in percentage points. (b) Pooled-query W-F1 (%) for compact and widened Joint heads and QSF; parentheses report added Stage-II parameters. Bars in (a) and diamonds in (b) show five-seed means, with sample-SD error bars; circles in (b) show individual seeds.

Table 2: **Conditioning across backbones.** Paired ACC differences in percentage points (mean ± sample SD over five seeds). Positive values favor QSF.

Dataset	Backbone	QSF–DIRECT	QSF–Joint	QSF–Logit
DSADS	InvConvNet	0.0000 ± 0.0000	-0.1404 ± 0.2370	$+0.0175 \pm 0.0392$
	DeepConvLSTM	$+0.0175 \pm 0.0500$	$+0.1930 \pm 0.1743$	$+0.0175 \pm 0.0500$
MotionSense	InvConvNet	$+0.0506 \pm 0.0498$	$+0.0689 \pm 0.1065$	$+0.0368 \pm 0.0448$
	DeepConvLSTM	$+0.0138 \pm 0.0206$	-0.0184 ± 0.0638	$+0.0092 \pm 0.0262$
Shoaib-2014	InvConvNet	$+0.1437 \pm 0.3021$	$+0.4070 \pm 0.4708$	$+0.1516 \pm 0.3618$
	DeepConvLSTM	$+0.0080 \pm 0.0178$	$+0.0080 \pm 0.2317$	0.0000 ± 0.0000

The advantage also holds against Wide Joint: QSF achieves higher mean W-F1 by 0.07 and 0.31 points on HAPT and MotionSense, respectively, using 64/40 versus 1,068/678 conditioning parameters (Figure 3b). With both Stage-II heads trained using cross-entropy alone, the mean QSF–Joint advantage is +1.49 W-F1 points on HAPT and +0.59 on MotionSense (Appendix C.3). Together, these capacity and objective controls support the factorized interface as a compact, effective way to incorporate reference context.

4.4 CONDITIONING ACROSS BACKBONES

To test whether query-led conditioning also benefits different recognition backbones, we evaluate InvConvNet and DeepConvLSTM on DSADS, MotionSense, and Shoaib-2014. Within each backbone, dataset, and seed, QSF, Joint, and Logit share the same frozen Population/DIRECT pair (Appendix C.13). QSF improves mean ACC over DIRECT in five of six combinations and matches it in the sixth (Table 2). Its mean ACC also matches or exceeds Logit in all six combinations. On MotionSense with InvConvNet, QSF improves over DIRECT, Joint, and Logit in four of five seeds; on DSADS with DeepConvLSTM, it improves over Joint in all five seeds. These results support query-led reference conditioning as a way to augment different frozen recognizers, extending its benefit beyond the population encoder used in the main benchmark.

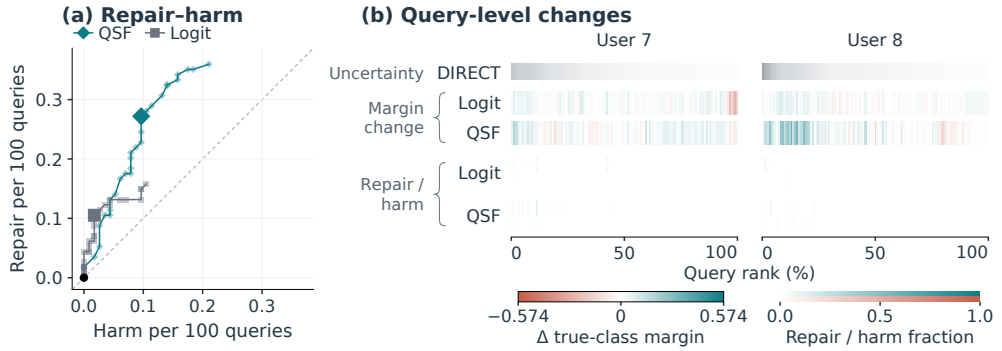


Figure 4: **Decision-level effects of reference conditioning on DSADS.** (a) Five-seed mean repair and harm rates, averaging users equally, as each frozen head's increment is scaled by $\beta = 0, 0.05, \ldots, 1.5$. Large filled markers denote each head's trained scale $\beta = 1$; the origin is DIRECT and the diagonal denotes zero net accuracy change. (b) Seed-averaged true-class margin changes and repair/harm fractions in 128 bins per user (2,280 queries). Queries are ranked by decreasing mean DIRECT entropy (top row, normalized). Heads share ordering, bins, and color scales.

4.5 Decision-Level Effects of Reference Conditioning

We next examine how the query-side read input affects decision changes within the factorized interface. The Logit control retains the same multiplicative form but reads source-energy-scaled centered DIRECT logits instead of the learned query correction, with DIRECT serving as the common decision anchor (Appendix C.1).

On DSADS, reading the learned correction yields a more favorable repair–harm balance: QSF achieves a larger mean net accuracy gain over DIRECT than Logit (0.175 versus 0.088 pp), as its additional repairs outweigh its additional harmful changes on average. Its paired pooled-query W-F1 gains are $+0.154 \pm 0.186$ pp over DIRECT and $+0.062 \pm 0.264$ pp over Logit. A plausible explanation is that the learned correction encodes how the population prediction is adjusted, providing a more direct signal for reference modulation than the current DIRECT class scores. Appendix C.4 reports the full repair–harm decomposition and seed variability.

Scaling each frozen head's own increment traces the repair–harm trade-off as conditioning strength varies (Figure 4a); $\beta = 1$ denotes each head's trained scale. The seed-averaged, binned view in Figure 4b shows margin shifts across many query bins but sparse repairs and harms, distinguishing changes to class scores from changes in prediction correctness.

4.6 Query-Gated Influence and Activation Geometry

Activation strength constrains reference influence. For a fixed query, Equation (30) bounds logit changes under reference substitution in terms of query activation strength, reference-gain differences, and the learned write operator. On HAPT and MotionSense, we compare same-wearer references with cross-wearer sets matched in ground-truth class counts. Higher source-defined activation bins exhibit larger mean logit changes (Figure 5a), and all evaluated changes satisfy their individual bounds. Appendix C.10 details the diagnostic construction and observed-to-bound ratios.

Activation direction affects recognition. To test the predictive role of the mode mixture at a fixed pre-write interaction norm, we rotate activations in the frozen two-mode interface, keeping queries, references, gains, and read/write factors fixed. After applying the gains, each rotated interaction is rescaled to match its native pre-write norm. A radial control instead scales the native interaction by $\cos\theta$. Rotation yields a larger mean normalized area under the accuracy-loss curve (AUL) than radial attenuation on seven of eight tasks. Figure 5b shows DSADS, Shoaib-2014, and UCI-HAR; Appendix C.11 reports the full paired results. On six tasks, mean rotation AUL is positive and mean radial AUL negative, suggesting that the selected mixture carries useful correction structure even when weaker reference influence is beneficial.

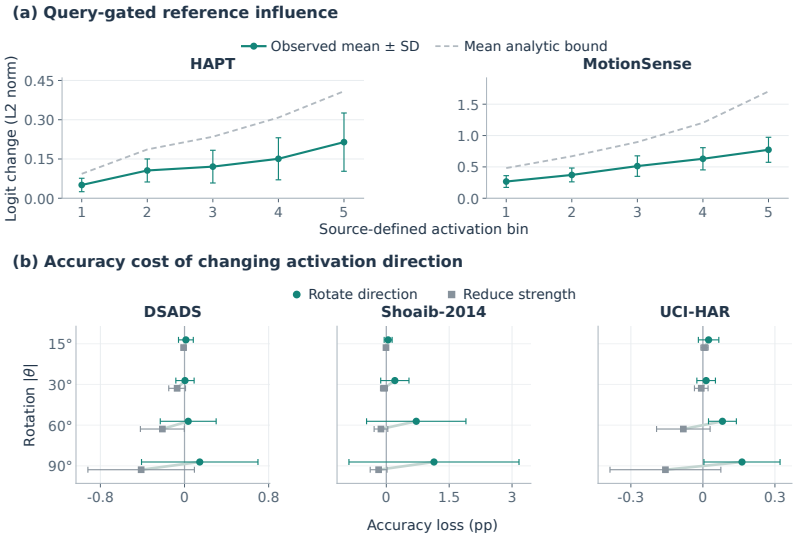


Figure 5: **Query-gated reference influence and activation geometry.** (a) Logit changes under reference substitution with matched ground-truth class counts, grouped by source-quantile bins of $\|\boldsymbol{a}(\boldsymbol{x})\|_2$; dashed curves show mean analytic bounds. (b) Accuracy loss under activation rotation with matched native pre-write interaction norm (circles) or radial attenuation (squares). Loss is native minus intervened ACC; positive values favor the native model. Points and bars show five-seed means and sample SDs. Dataset axes differ. Appendices C.10 and C.11 give the corresponding protocols.

Table 3: **GPU deployment latency (ms).** Warm QSF queries reuse cached context; preparation includes candidate encoding, selection, and state inference. Overheads use unrounded medians.

Dataset	DIRECT warm	QSF warm	Overhead	Reference prep.
HAPT	1.785	1.918	+7.4%	17.69
MotionSense	1.969	2.100	+6.7%	39.58
WESAD	2.003	2.149	+7.3%	76.70

4.7 AMORTIZED DEPLOYMENT COST

For queries sharing an eligible reference set, one preparation step supplies the state and gains reused by subsequent QSF queries. Across the three profiled tasks, cached QSF adds 6.7–7.4% to DIRECT's warm single-query latency on an RTX 4090 (Table 3). Appendix C.12 reports the full timing and amortization results.

5 CONCLUSION

We introduced QSF to make reference conditioning query-led: queries activate shared correction modes, and unlabeled references modulate their gains. This division of roles yields bounded, query-dependent reference influence. Across eight wearable-recognition tasks, the complete system achieves the highest mean accuracy on all six activity-recognition benchmarks, while fixed-upstream comparisons show higher mean weighted F1 than compact Joint on all eight tasks. Targeted analyses connect the interface advantage to decision preservation, and activation interventions show that the mode mixture matters for recognition. Together, these findings support treating informative context not as an independent instruction to change a prediction, but as evidence whose effect is structured by the query.

AI USE STATEMENT

AI tools assisted with coding for experiments, manuscript polishing, visual refinement of experimental figures, and grammar checking.

REFERENCES

Kerem Altun, Billur Barshan, and Orkun Tunçel. Comparative study on classifying human activities with miniature inertial and magnetic sensors. *Pattern Recognition*, 43(10):3605–3620, 2010.

Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz. A public domain dataset for human activity recognition using smartphones. In *Proceedings of the 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 437–442, 2013.

Yarin Bar, Shalev Shaer, and Yaniv Romano. Protected test-time adaptation via online entropy matching: A betting approach. In *Advances in Neural Information Processing Systems*, volume 37, pp. 85467–85499, 2024.

Thibaut Germain, Chrysoula Kosma, and Laurent Oudre. Time series representations with hard-coded invariances. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 19172–19195. PMLR, 2025.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Wenbo Huang, Lei Zhang, Hao Wu, Fuhong Min, and Aiguo Song. Channel-Equalization-HAR: A light-weight convolutional neural network for wearable sensor based human activity recognition. *IEEE Transactions on Mobile Computing*, 22(9):5064–5077, 2023.

Jeonghwan Im and Hyuk-Yoon Kwon. COSA: Context-aware output-space adapter for test-time adaptation in time series forecasting. In *International Conference on Learning Representations*, volume 2026, pp. 24845–24874, 2026.

Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. In *Advances in Neural Information Processing Systems*, volume 34, pp. 2427–2440, 2021.

Bob Kemp. Sleep-EDF Database Expanded. PhysioNet, 2013. Version 1.0.0.

Bob Kemp, Aeilko H. Zwinderman, Bert Tuk, Hilbert A. C. Kamphuisen, and Josefien J. L. Oberyé. Analysis of a sleep-dependent neuronal feedback loop: The slow-wave microcontinuity of the EEG. *IEEE Transactions on Biomedical Engineering*, 47(9):1185–1194, 2000.

Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Hadamard product for low-rank bilinear pooling. In *International Conference on Learning Representations*, 2017.

Wang Lu, Jindong Wang, Xinwei Sun, Yiqiang Chen, and Xing Xie. Out-of-distribution representation learning for time series classification. In *International Conference on Learning Representations*, 2023.

Mohammad Malekzadeh, Richard G. Clegg, Andrea Cavallaro, and Hamed Haddadi. Mobile sensor data anonymization. In *Proceedings of the International Conference on Internet of Things Design and Implementation*, pp. 49–58. ACM, 2019.

Daniela Micucci, Marco Mobilio, and Paolo Napoletano. UniMiB SHAR: A dataset for human activity recognition using acceleration data from smartphones. *Applied Sciences*, 7(10):1101, 2017.

Alexander Murphy, Michal Danilowski, Soumyajit Chatterjee, and Abhirup Ghosh. NEO—no-optimization test-time adaptation through latent re-centering. In *International Conference on Learning Representations*, volume 2026, pp. 128762–128789, 2026.

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.

Francisco Javier Ordóñez and Daniel Roggen. Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16(1):115, 2016.

Jaegyun Park, Dae-Won Kim, and Jaesung Lee. CALANet: Cheap all-layer aggregation for human activity recognition. In *Advances in Neural Information Processing Systems*, volume 37, pp. 69419–69444, 2024.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), 2018.

Hangwei Qian, Sinno Jialin Pan, and Chunyan Miao. Latent independent excitation for generalizable sensor-based cross-person activity recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(13):11921–11929, 2021.

Xin Qin, Jindong Wang, Shuo Ma, Wang Lu, Yongchun Zhu, Xing Xie, and Yiqiang Chen. Generalizable low-resource activity recognition with diverse and discriminative representation learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1943–1953, 2023.

Jorge-L. Reyes-Ortiz, Luca Oneto, Albert Samà, Xavier Parra, and Davide Anguita. Transition-aware human activity recognition using smartphones. *Neurocomputing*, 171:754–767, 2016.

Philip Schmidt, Attila Reiss, Robert Dürichen, Claus Marberger, and Kristof Van Laerhoven. Introducing WESAD, a multimodal dataset for wearable stress and affect detection. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pp. 400–408. ACM, 2018.

Muhammad Shoaib, Stephan Bosch, Özlem Durmaz Incel, Hans Scholten, and Paul J. M. Havinga. Fusion of smartphone motion sensors for physical activity recognition. *Sensors*, 14(6):10146–10176, 2014.

Yongyi Su, Xun Xu, and Kui Jia. Towards real-world test-time adaptation: Tri-net self-training with balanced normalization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(13): 15126–15135, 2024.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9229–9248. PMLR, 2020.

Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Chengzhong Xu. HydraLoRA: An asymmetric LoRA architecture for efficient fine-tuning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 9565–9584, 2024.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.

Shuoyuan Wang, Jindong Wang, Huajun Xi, Bob Zhang, Lei Zhang, and Hongxin Wei. Optimization-free test-time adaptation for cross-person activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 7(4):183:1–183:27, 2023.

Haoyu Xie, Haoxuan Li, Chunyuan Zheng, Haonan Yuan, Guorui Liao, Jun Liao, and Li Liu. Decomposing and fusing intra- and inter-sensor spatio-temporal signal for multi-sensor wearable human activity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 14441–14449, 2025.

Chaoqi Yang, M. Brandon Westover, and Jimeng Sun. BIOT: Biosignal transformer for cross-data learning in the wild. In *Advances in Neural Information Processing Systems*, volume 36, pp. 78240–78260, 2023.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Junru Zhang, Cheng Peng, Zhidan Liu, Lang Feng, Yuhan Wu, Yabo Dong, and Duanqing Xu. DI2SDiff++: Activity style decomposition and diffusion-based fusion for cross-person generalization in activity recognition. *IEEE Transactions on Mobile Computing*, 24(10):10760–10777, 2025.

A Dataset and Evaluation Details

Table 4 summarizes the datasets, sensor inputs, and subject-disjoint partitions used in our evaluation (Altun et al., 2010; Reyes-Ortiz et al., 2016; Malekzadeh et al., 2019; Shoaib et al., 2014; Anguita et al., 2013; Micucci et al., 2017; Kemp et al., 2000; Kemp, 2013; Schmidt et al., 2018). Counts refer to preprocessed windows. Window and stride are in samples unless seconds are shown explicitly.

Table 4: Dataset configurations. Subject counts are total (source/target), and the last column gives source/target windows. Source counts precede the internal source-validation split in Table 6. WESAD is resampled from 700 to 100 Hz.

Dataset	Subjects	Classes	Input used in this work	Rate	Window/stride	Windows
DSADS	8 (6/2)	19	acc. + gyro. + magnetometer at five body locations (45 ch.)	25 Hz	5 s / 5 s	6,840/2,280
HAPT	30 (21/9)	12	accelerometer + gyroscope (6 ch.)	50 Hz	128/64	7,752/3,156
MotionSense	24 (19/5)	6	user acceleration + rotation rate (6 ch.)	50 Hz	128/64	17,187/4,352
Shoaib-2014	10 (8/2)	7	acc., linear acc., gyro., magnetometer at five locations (60 ch.)	50 Hz	100/50	10,024/2,506
SleepEDF-20	20 (16/4)	5	Fpz–Cz and Pz–Oz EEG (2 ch.)	100 Hz	30 s / 30 s	43,695/10,892
UCI-HAR	30 (21/9)	6	body/total acceleration + gyroscope (9 ch.)	50 Hz	128/64	7,352/2,947
UniMiB SHAR	30 (20/10)	17	smartphone triaxial acceleration	50 Hz	3 s event / –	7,734/4,037
WESAD	15 (12/3)	3	chest ECG, EDA, EMG, respiration, temperature (5 ch.)	100 Hz	10 s / 5 s	5,239/1,334

HAPT and UCI-HAR follow their published subject-disjoint partitions. For Shoaib-2014 and MotionSense, which do not define the partitions used here, we fix subjects 1–8/9–10 and 1–19/20–24, respectively. SleepEDF-20 selects the first recording night for Sleep-EDF Expanded subjects 00–19, merges stages 3 and 4 into N3, removes movement/unscored epochs, and does not trim mapped wake epochs. WESAD uses subjects S2–S11, S13, and S14 for training and S15–S17 for evaluation, with five RespiBAN chest channels and a fixed subject-disjoint partition. In every case, a subject contributes windows to exactly one split.

A.1 Evaluation Settings and Data Access

Table 5 instantiates the model-selection and target-access policy defined in Section 4.1 for the evaluation settings. The eight-task interface study reuses the same frozen Population/DIRECT pair and prespecified final-epoch conditioning heads.

Table 5: Evaluation settings under the protocol in Section 4.1.

Setting	Fitting and model selection	Reference access	Metrics
Benchmark	Source fitting; held-out source users select checkpoints and hyperparameters	QSF: same-wearer, query-excluded full-session references; baselines: Appendix A.3	Pooled ACC and W-F1
Eight-task interface comparisons	Common Population/DIRECT pair; identical source episodes and 120-epoch head training	Shared reference sets across heads	Pooled W-F1; paired seed differences
Cross-backbone comparisons	Shared upstream pair within each backbone, dataset, and seed; source-validation ACC selects checkpoints	Shared reference sets; budget 64	ACC; paired seed differences
Reference / activation interventions	Trained models fixed; source-retrained controls marked separately	Protocol-defined pools; GT matching where stated	Equal-user reference W-F1; query-averaged logit changes; pooled angular ACC losses

Source validation for the benchmark. We hold out complete source users according to Table 6: all their windows form the validation set, and the remaining source users are used for fitting. The split is fixed across model seeds. Candidate user groups target approximately 20% of source users and are ranked using source information only: first by fewer missing classes, then by smaller label-distribution discrepancy, and finally by a validation-window fraction closer to the target; remaining ties are resolved lexicographically by user IDs. Benchmark checkpoint selection uses pooled validation ACC, updating only on a strict increase and retaining the earliest epoch in a tie, without secondary ranking by F1 or loss. ACC and W-F1 are reported from the same selected checkpoint.

Table 6: Fixed whole-user source-validation splits for the benchmark. The user-count denominator is the original source partition in Table 4; user IDs retain the protocol's numbering. These splits are unchanged across model seeds.

Dataset	Validation user IDs	Validation / source users	Validation windows
DSADS	1	1/6	1,140
HAPT	6, 25, 27, 29	4/21	1,534
MotionSense	6, 8, 9, 15	4/19	3,576
Shoaib-2014	1, 2	2/8	2,506
SleepEDF-20	2, 13, 15	3/16	8,234
UCI-HAR	6, 11, 17, 28	4/21	1,391
UniMiB	2, 10, 15, 16	4/20	1,759
WESAD	4, 14	2/12	871

Aggregation and decision changes. Pooled metrics combine the predictions of all evaluation users before scoring; equal-user metrics score each user and then average. Dataset-level summaries weight datasets equally. Paired differences are formed within seed and summarized as means ± sample SD over five seeds, except for the three-seed upstream controls in Appendix D. Repair and harm count, respectively, wrong-to-correct and correct-to-wrong changes relative to DIRECT per 100 queries; their difference is the accuracy change in percentage points.

A.2 POPULATION ARCHITECTURE AND TRAINING

The Population recognizer combines temporal residual convolutions, token projection, and attention/max pooling (Table 7). It is distinct from the external Residual baseline. A semantic $T \times C$ window is passed as $1 \times T \times C$: convolutions act along time while retaining sensor positions, and token projection mixes the sensor channels.

Table 7: Default Population architecture. $L = \lceil T/8 \rceil$. Conv denotes bias-free Conv2d; GroupNorm uses eight groups. Each residual block has a 3×1 Conv–GN–SiLU–Dropout2d followed by a 3×1 Conv–GN, identity addition, and SiLU. Temporal padding preserves the stated sizes.

Stage	Operations	Output per window
Stem + block	7×1 Conv, stride $(2,1)$, padding $(3,0)$; GN, SiLU, dropout 0.05; residual block	$128 \times \lceil T/2 \rceil \times C$
Downsample + block	5×1 Conv, stride $(2,1)$, padding $(2,0)$; GN, SiLU, dropout 0.05; residual block	$256 \times \lceil T/4 \rceil \times C$
Downsample + block	Same operations, 384 output channels	$384 \times L \times C$
Temporal tokens	Reshape; Linear $(384C) \to 256$, LayerNorm, SiLU	$L \times 256$
Pooling	Linear $256 \to 1$, softmax over time; concatenate attention-weighted sum and temporal max	512
Feature projection	Linear $512 \to 256$, LayerNorm, SiLU	256
Classifier	LayerNorm, dropout 0.20, Linear $256 \to K$	K logits

The default encoder uses widths $(128, 256, 384)$ and a 256-dimensional feature. For the DSADS benchmark, the same block structure uses widths $(64, 128, 192)$ and a 128-dimensional feature, with convolutional and classifier dropout 0.05 and 0.20. The eight-task interface comparisons use

the default encoder. The Population training recipe uses AdamW (learning rate 0.001, weight decay 0.0001), cosine decay over 120 epochs, batch size 64 with the final partial batch retained, norm-5 gradient clipping, label smoothing 0.1, and mixup parameter 0.2. Source-fitted channel standardization uses a 10^{-6} SD floor. Augmentations apply noise SD 0.015, scale 0.9–1.1, temporal shift, and a contiguous mask with probability 0.25. Dataset-dependent operations are listed below. The default rotation augmentation applies a common matrix to the specified channel triads in source-standardized coordinates.

Table 8: QSF input dimensions, mode counts, and augmentations. All tasks use $R = 8$, $M = 64$, ridge 0.1, and $\tau_d = 2$. The two q columns distinguish the benchmark (B) and fixed-upstream comparison (F); α is the modulation scale.

Task	C	T	K	α	q_B	q_F	Rotation groups	Shift / mask
UCI-HAR	9	128	6	2	2	2	Three triads	±5 / 4–12
Shoaib-2014	60	100	7	3	2	2	First two triads only	±5 / 4–12
HAPT	6	128	12	1	2	2	Two triads	±5 / 4–12
MotionSense	6	128	6	2.5	5	2	Two triads	±5 / 4–12
SleepEDF-20	2	3000	5	1	2	2	Disabled	±5 / 4–12
DSADS	45	125	19	1	2	2	Fifteen triads	±3 / 2–6
WESAD	5	1000	3	2	2	2	Disabled	±5 / 4–12
UniMiB	3	151	17	1	2	2	One triad	±5 / 4–12

The same rotation matrix, bounded by 15 degrees, acts on the declared triads within each window. In the DSADS benchmark, rotations of up to 30° are sampled independently for the five body locations. Each location shares one matrix across its accelerometer, gyroscope, and magnetometer triads; rotations act in raw sensor units before source-fitted standardization. DIRECT adds 120 epochs of source fitting on the frozen cache (AdamW, learning rate 0.0005, weight decay 0.0001, batch 512, cosine schedule, clipping at 5). The conditioning stage freezes DIRECT, uses learning rate 0.005 and zero weight decay, and retains the complete objective in Equation (8). Model-selection endpoints follow Section 4.1.

A.3 BENCHMARK METHODS AND EVALUATION PROTOCOL

Comparison methods. We compare QSF with eleven methods spanning supervised recognition, representation learning, and adaptation. CNN and Residual represent convolutional and residual-network backbone families evaluated in prior wearable HAR studies (Huang et al., 2023; Zhang et al., 2025). CEHAR denotes Channel-Equalization-HAR (Huang et al., 2023) in this paper. DeepConvLSTM (Ordóñez & Roggen, 2016) models within-window temporal structure; CALANet (Park et al., 2024) and CEHAR improve feature aggregation and channel representations. InvConvNet (Germain et al., 2025) combines ordinary and hard-coded invariant convolutions for time-series classification. BIOT (Yang et al., 2023) uses a biosignal transformer, while DDLearn (Qin et al., 2023) and DIVERSIFY (Lu et al., 2023) learn representations for distribution shift. OFTTA (Wang et al., 2023) updates target normalization and prototypes without gradient optimization. BIOT is trained from random initialization using its supervised HAR configuration. OFTTA additionally uses unlabeled target windows; its target-data access differs from that of source-only predictors and from QSF's query-excluded reference inference.

Inputs and preprocessing. The comparison preserves the train/test users, class labels, and input windows of each dataset. The additional methods use the QSF input layout and source-fitted channel standardization, $(x-\mu)/\max(\sigma, 10^{-6})$. CNN retains its original standardization, $(x-\mu)/(\sigma+10^{-8})$, and its UniMiB input reshapes the stored 453 values into a 151×3 array in memory order; the other methods reconstruct the three axis blocks. Method-specific architectural dimensions accommodate the original channel counts and window lengths.

Architectures and transferred configurations. DeepConvLSTM uses the authors' later PyTorch implementation and initialization. CALANet retains all-layer aggregation and max-pooling gradients. CEHAR uses the authors' UCI-HAR and UniMiB networks on those datasets and the UCI-HAR architecture with adjusted input dimensions elsewhere, retaining both channel-equalization branches.

BIOT uses four transformer layers, eight heads, FFT length 100 and hop length 5 samples, with random initialization and source-fitted channel standardization. InvConvNet retains the complete author model, with ordinary, offset-invariant, and trend-invariant branches at kernel sizes 51, 75, 101, and 125, width 32, and dropout 0.1. Its input layout and dataset-specific input/output dimensions match the sensor windows used here. DDLearn jointly optimizes its classification, augmentation, diversity, and supervised-contrastive objectives with weights $1, 1, 10, 1$; spatial augmentations are restricted to sensor-axis triplets, while physiological channels use nonspatial transformations. DecomposeWHAR (Xie et al., 2025) retains its Mamba and cross-sensor attention modules, with sensor groups defined by physical devices and embedding stride $s = \max(1, \lfloor L/6 \rfloor)$ and kernel $2s$ for window length L. Its cross-sensor module receives one token on single-device datasets. DIVERSIFY retains all three training stages and latent-domain reassignment.

Training and target access. The additional supervised and representation-learning baselines are trained for 120 epochs, except that DIVERSIFY uses 120 outer rounds with three source-training traversals per round. InvConvNet retains the author's RAdam, gradient clipping, and warmup/cosine recipe with batch size 16, using the 120-epoch training budget of this benchmark. OFTTA trains its source classifier for 120 epochs and processes each target user's windows in their stored order, using batches of 64 and an entropy-filtered support limit of 25 per class. Its target state is reset between users and source checkpoints, and no target labels or target gradient steps are used in adaptation.

Benchmark evaluation. Benchmark model development follows the source-only selection protocol in Section 4.1. For each method and seed, ACC and W-F1 are computed from the same selected checkpoint on the complete held-out test partition. Method-specific unlabeled target access follows the inference procedures described above. Table rankings use unrounded means. The benchmark compares complete systems; the paired controls in the main text assess conditioning interfaces with a common frozen upstream predictor.

Mode count. MotionSense uses five modes for the benchmark and two for the fixed-upstream comparison, corresponding to 100 and 40 conditioning parameters. The benchmark mode count is held fixed across the five seeds; the data partitions, upstream models, learning rates, and reference settings are shared.

B Method Derivations and Algorithmic Details

We give the derivations and implementation details for the three parts of QSF: the source-trained query correction, activity-relative reference inference, and subject-dependent correction modes. We then specify source learning, inference, and computational costs. The reference-set protocol is reported separately in Appendix A.

B.1 Centered logits and coordinatewise saturation

Softmax obeys $\mathrm{softmax}(\boldsymbol{z} + c\mathbf{1}) = \mathrm{softmax}(\boldsymbol{z})$. The orthogonal projector $\boldsymbol{P} = \boldsymbol{I}_K - \mathbf{1}\mathbf{1}^\top/K$ therefore removes a prediction-invariant direction. Corrections are stored in K logit coordinates, with their common shift removed by centering. The map $\mathcal{B}_\tau$ in Equation (1) first centers, then saturates each coordinate, and centers again to keep its output in $\mathcal{C}$.

Proposition 1 (Centered bounded correction)**.** *For $\tau > 0$, $\mathcal{B}_\tau$ maps $\mathbb{R}^K$ into $\mathcal{C}$, preserves zero, and is one-Lipschitz in the Euclidean norm. It satisfies*

$$\begin{aligned} \|\mathcal{B}_\tau(\boldsymbol{a})\|_2 &\le \min\{\|\boldsymbol{P}\boldsymbol{a}\|_2, \tau\sqrt{K}\}, \\ \|\mathcal{B}_\tau(\boldsymbol{a})\|_\infty &\le 2\tau(1 - 1/K), \qquad D\mathcal{B}_\tau(\mathbf{0}) = \boldsymbol{P}. \end{aligned} \tag{9}$$

For any $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^K$, $\|\mathcal{B}_\tau(\boldsymbol{a}) - \mathcal{B}_\tau(\boldsymbol{b})\|_2 \le \|\boldsymbol{P}(\boldsymbol{a} - \boldsymbol{b})\|_2$. Near zero, $\mathcal{B}_\tau(\boldsymbol{a}) = \boldsymbol{P}\boldsymbol{a} + O(\|\boldsymbol{P}\boldsymbol{a}\|_2^3/\tau^2)$.

Proof. Centering follows from $\mathbf{1}^\top\boldsymbol{P} = 0$, and zero preservation follows from $\tanh(0) = 0$. The Jacobian is

$$D\mathcal{B}_\tau(\boldsymbol{a}) = \boldsymbol{P}\,\mathrm{diag}\big(\mathrm{sech}^2(\boldsymbol{P}\boldsymbol{a}/\tau)\big)\,\boldsymbol{P}. \tag{10}$$

Both projectors have operator norm one and the diagonal entries lie in $(0, 1]$. Integrating the Jacobian on a line segment proves nonexpansiveness. Since $\mathcal{B}_\tau(\boldsymbol{a}) = \mathcal{B}_\tau(\boldsymbol{P}\boldsymbol{a})$, the two-point bound also holds with the centered difference $\boldsymbol{P}(\boldsymbol{a}-\boldsymbol{b})$. Since $|\tanh t| \le |t|$ and $|\tau \tanh t| \le \tau$, the Euclidean bounds follow by contraction of $\boldsymbol{P}$. For any vector $\boldsymbol{b}$ with $|b_k| \le \tau$, $(\boldsymbol{P}\boldsymbol{b})_k = (1-1/K)b_k - K^{-1}\sum_{\ell\neq k} b_\ell$, which gives the infinity bound. The derivative and expansion follow from the scalar Taylor expansion of $\tanh$. □

The saturation scale τ acts before the final centering; the resulting Euclidean and coordinate bounds are given in Equation (9). The model uses positive budgets.

B.2 Query descriptor and Stage-I objective

Let $\boldsymbol{p}_0 = \mathrm{softmax}(\boldsymbol{z}_0)$, let $p_{0,(1)} \ge p_{0,(2)}$ be its two largest entries, and define

$$\begin{aligned} c_0 &= p_{0,(1)}, \qquad \Delta_0 = p_{0,(1)} - p_{0,(2)}, \\ H_0 &= -\frac{1}{\log K}\sum_k p_{0,k}\log\max\{p_{0,k}, \epsilon\}, \\ \boldsymbol{q}(\boldsymbol{x}) &= [\boldsymbol{h}; \boldsymbol{z}_0; \boldsymbol{p}_0; c_0; \Delta_0; H_0] \in \mathbb{R}^{D+2K+3}. \end{aligned} \tag{11}$$

The head consists of LayerNorm, a linear layer, GELU, dropout, and a K-dimensional output layer. Default hidden width and dropout are 256 and 0.1. Both the output weight and bias start at zero, so DIRECT initially matches the population predictor. With the latter frozen, its default batch objective is

$$\mathcal{L}_D = \frac{1}{B}\sum_i \mathrm{CE}(\boldsymbol{z}_{0,i} + \boldsymbol{d}_i, y_i) + \frac{\lambda_d}{BK}\sum_i \|\boldsymbol{d}_i\|_2^2, \qquad \lambda_d = 10^{-3}. \tag{12}$$

The norm uses the final centered correction. Once trained, DIRECT is kept in evaluation mode, including its dropout layer, during state inference and Stage II. All descriptor quantities come from frozen population outputs.

B.3 Source activity prototypes and residual dictionaries

For source class k, let $\mathcal{I}_k = \{i : y_i = k\}$ and $\mathcal{I}_{v,k} = \{i : u_i = v, y_i = k\}$. For a nonempty class, the prototype is the window-weighted mean

$$\boldsymbol{\mu}_k = \frac{1}{|\mathcal{I}_k|}\sum_{i\in\mathcal{I}_k} \boldsymbol{h}_i. \tag{13}$$

If a class has no source examples, the routine uses the global source feature mean. A user–activity cell is eligible for dictionary estimation when it has at least two examples. Let n_k be the number of eligible users for class k. Each contributes one row $(\bar{\boldsymbol{h}}_{v,k} - \boldsymbol{\mu}_k)^\top$ to $\boldsymbol{A}_k \in \mathbb{R}^{n_k\times D}$, where $\bar{\boldsymbol{h}}_{v,k} = |\mathcal{I}_{v,k}|^{-1}\sum_{i\in\mathcal{I}_{v,k}} \boldsymbol{h}_i$. For $n_k \ge 2$, subtract the row mean $\overline{\boldsymbol{a}}_k$ and compute a thin SVD:

$$\boldsymbol{A}_k^c = \boldsymbol{A}_k - \mathbf{1}\overline{\boldsymbol{a}}_k^\top = \boldsymbol{F}_k\,\mathrm{diag}(\boldsymbol{\sigma}_k)\boldsymbol{H}_k^\top, \qquad r_k = \min\{R, n_k, D\}. \tag{14}$$

For $n_k < 2$, set $r_k = 0$ and retain the seeded initialization below. One row per eligible user gives equal user weighting in the SVD; centering removes their unweighted row mean. The prototype in Equation (13) remains window weighted. For $n_k \ge 2$, row centering implies $\mathrm{rank}(\boldsymbol{A}_k^c) \le \min\{n_k - 1, D\}$. Thus r_k counts selected thin-SVD columns, not nonzero singular directions; selected columns may include null-space completion directions. No numerical rank threshold is applied.

Initially every entry of $\boldsymbol{D}_k^{\mathrm{raw}} \in \mathbb{R}^{D\times R}$ is sampled from $\mathcal{N}(0, 0.01^2)$ using a source-statistics seed. For $j \le r_k$, replace its jth column by

$$(\boldsymbol{D}_k^{\mathrm{raw}})_{:j} = (\boldsymbol{H}_k)_{:j}\sqrt{\max\{\sigma_{k,j}, 10^{-6}\}}. \tag{15}$$

Unreplaced columns retain their random initialization. The frozen dictionary is then

$$(\boldsymbol{D}_k)_{:j} = 0.1\,\frac{(\boldsymbol{D}_k^{\mathrm{raw}})_{:j}}{\max\{\|(\boldsymbol{D}_k^{\mathrm{raw}})_{:j}\|_2, 10^{-3}\}}. \tag{16}$$

For every selected SVD column, $\|(\boldsymbol{H}_k)_{:j}\|_2 = 1$ and the raw norm is $\sqrt{\max\{\sigma_{k,j}, 10^{-6}\}} \geq 10^{-3}$. Hence, in exact arithmetic,

$$(\boldsymbol{D}_k)_{:j} = 0.1(\boldsymbol{H}_k)_{:j}, \qquad j \leq r_k.$$

The construction therefore retains the selected directions but not their singular-value amplitudes, including when $\sigma_{k,j} = 0$. Columns beyond r_k use the normalized seeded initialization, giving every dictionary width R.

Reference-state coordinates. The classwise SVD bases are fixed after construction and need not align across classes. They define coordinates of an activity-relative reference descriptor; no identifiable class-independent wearer latent factor is assumed. The gain map is learned with these same dictionaries held fixed.

B.4 Reference selection, weighting, and the ridge system

Reference eligibility and selection. The selector operates on the protocol-defined eligible pool $\mathcal{E}(\boldsymbol{x})$ (Appendix A.1) and chooses at most M references using frozen prediction scores and feature diversity. For a pool of size N_e, the selector returns all indices if $N_e \leq M$ (default $M = 64$). Otherwise it normalizes the frozen features to $\bar{\boldsymbol{h}}_j$ and forms probabilities $\boldsymbol{p}_j^{\mathrm{sel}}$ from the supplied frozen selection logits. Define

$$c_j = \max\left\{10^{-3}, \max_k p_{j,k}^{\mathrm{sel}}\left(1 + \frac{\sum_k p_{j,k}^{\mathrm{sel}} \log \max\{p_{j,k}^{\mathrm{sel}}, \epsilon\}}{\log K}\right)\right\}. \tag{17}$$

A fixed, seed-dependent perturbation satisfying $|t_j| < 10^{-9}$ is used for tie breaking. The first selected index maximizes $c_j + t_j$; subsequent indices maximize

$$\sqrt{c_j}\min_{\ell \in \mathcal{S}}(1 - \bar{\boldsymbol{h}}_j^\top \bar{\boldsymbol{h}}_\ell) + t_j, \qquad j \notin \mathcal{S}, \tag{18}$$

where $\mathcal{S}$ is the selected set. The experiment configuration specifies the frozen logits used for selection. Reference-state inference uses DIRECT logits, as defined below. In the default feature-based protocol, selection and state estimation do not use reference labels.

Frozen reference probabilities. For each reference, compute

$$\boldsymbol{p}_m = \mathrm{softmax}\big([\boldsymbol{z}_{0,m} + \mathcal{B}_{\tau_d}(r_\theta(\boldsymbol{q}(\boldsymbol{x}_m)))]/\max\{T, 10^{-4}\}\big). \tag{19}$$

These probabilities are fixed during Stage II. With no padding, use Equation (5) with $\tau_c^{\mathrm{num}} = \max\{\tau_c, 10^{-4}\}$ in place of τ_c. All K class assignments participate in this weighting rule. The soft class masses are prediction-derived and require no target labels.

Padding convention. Let $e_m \in \{0, 1\}$ indicate a valid reference row; an unpadded set has $e_m = 1$ for every row. The numerical rule masks probabilities before applying the ϵ floor:

$$\begin{aligned} &p_{m,k}^e = e_m p_{m,k}, && \widetilde{p}_{m,k}^e = \max\{p_{m,k}^e, \epsilon\}, \quad a_k = \sum_m p_{m,k}^e, \\ &w_{m,k} = (1 - e^{-a_k/\tau_c^{\mathrm{num}}})\frac{(\widetilde{p}_{m,k}^e)^\gamma}{\max\{\sum_n (\widetilde{p}_{n,k}^e)^\gamma, \epsilon\}}. \end{aligned} \tag{20}$$

For a padded row, $\widetilde{p}_{m,k}^e = \epsilon$, so

$$0 \leq w_{m,k} \leq (1 - e^{-a_k/\tau_c^{\mathrm{num}}})\epsilon^{\gamma - 1}.$$

This weight is at most 10^{-8} for the default $\gamma = 2$, but need not be zero in a partially padded set. Its contribution to $\boldsymbol{b}_u$ satisfies $\left\|w_{m,k}\boldsymbol{D}_k^\top \boldsymbol{r}_{m,k}\right\|_2 \leq w_{m,k}\left\|\boldsymbol{D}_k\right\|_2 \left\|\boldsymbol{r}_{m,k}\right\|_2$; the weight bound alone does not make the state effect negligible independently of residual magnitudes and the number of padded rows. Padding uses finite feature values. In an entirely masked set, $a_k = 0$ makes all weights exactly zero. Permutation invariance applies to simultaneous permutations of the selected rows and their masks.

Closed-form inference. With fixed weights, the reference objective is

$$E_u(\boldsymbol{s}) = \frac{1}{2}\sum_{m,k} w_{m,k}\,\|\boldsymbol{h}_m - \boldsymbol{\mu}_k - \boldsymbol{D}_k\boldsymbol{s}\|_2^2 + \frac{\lambda_s}{2}\,\|\boldsymbol{s}\|_2^2\,. \tag{21}$$

Its gradient is $\nabla E_u(\boldsymbol{s}) = \boldsymbol{\Lambda}_u\boldsymbol{s} - \boldsymbol{b}_u$. For every nonzero $\boldsymbol{a}$,

$$\boldsymbol{a}^\top\boldsymbol{\Lambda}_u\boldsymbol{a} = \lambda_s\,\|\boldsymbol{a}\|_2^2 + \sum_{m,k} w_{m,k}\,\|\boldsymbol{D}_k\boldsymbol{a}\|_2^2 > 0. \tag{22}$$

Thus $\widehat{\boldsymbol{s}}_u$ is the unique global minimizer. A linear solve gives the state, and the average trace of the inverse Hessian gives ν_u. The fixed-weight quadratic energy also defines a Gaussian working density proportional to $\exp[-E_u(\boldsymbol{s})]$, with mean $\widehat{\boldsymbol{s}}_u$ and covariance $\boldsymbol{\Lambda}_u^{-1}$. We use this inverse curvature to scale state attenuation in Equation (7).

Since $\sum_m w_{m,k} \le 1$, the precision is bounded in the positive-semidefinite order:

$$\lambda_s\boldsymbol{I}_R \preceq \boldsymbol{\Lambda}_u \preceq \lambda_s\boldsymbol{I}_R + \sum_k \boldsymbol{D}_k^\top\boldsymbol{D}_k. \tag{23}$$

Repeating reference windows therefore does not yield unbounded precision. This is the algebraic consequence of class-mass saturation.

B.5 Source-balanced normalization and attenuation

Compute raw ridge states on source reference episodes. Let $v = 1, \dots, J$ index the source users with nonempty episode collections $\mathcal{E}_v$; at least one such user is required. First average within each user:

$$\bar{\boldsymbol{s}}_v = \frac{1}{|\mathcal{E}_v|}\sum_{e\in\mathcal{E}_v}\widehat{\boldsymbol{s}}_e, \qquad \boldsymbol{m}_s = \frac{1}{J}\sum_{v=1}^{J}\bar{\boldsymbol{s}}_v. \tag{24}$$

For $J > 1$, set

$$a_{s,r} = \max\left\{10^{-3}, \sqrt{\frac{1}{J-1}\sum_{v=1}^{J}(\bar{s}_{v,r} - m_{s,r})^2}\right\}. \tag{25}$$

For $J = 1$, the empirical standard deviation uses denominator J and the same floor. In either case, set

$$v_s = \max\{R^{-1}\sum_r a_{s,r}^2, 10^{-6}\}.$$

These buffers are fixed before subject-conditioned learning.

For a padded episode, let $b_u = \mathbb{I}[\sum_m e_m > 0.5]$. The exact normalization and attenuation are

$$\begin{aligned} \boldsymbol{s}_u &= b_u(\widehat{\boldsymbol{s}}_u - \boldsymbol{m}_s) \oslash \boldsymbol{a}_s, \\ A_u &= R^{-1}\,\|\boldsymbol{s}_u\|_2^2\,, \qquad N_u = \nu_u/v_s, \\ \rho_u &= b_u\,\mathrm{clip}_{[0,1]}\left(\frac{A_u}{A_u + N_u + \epsilon}\right). \end{aligned} \tag{26}$$

For a nonempty set $N_u > 0$, and the clip only guards numerical effects. With no support, the explicit mask gives $\boldsymbol{s}_u = \boldsymbol{0}$ and $\rho_u = 0$ even when the source center is nonzero. For present references, the neutral point is the source center $\widehat{\boldsymbol{s}}_u = \boldsymbol{m}_s$, not necessarily raw state zero.

The coordinate scales $\boldsymbol{a}_s$ standardize the state components, while the global mean-square scale v_s normalizes the scalar inverse-curvature summary. These two operations give Equation (26) its energy-to-curvature form. At fixed N_u, the score increases with standardized state energy; at fixed energy, it decreases with inverse curvature. The gain map uses this score as a continuous shrinkage factor around the source center. For present references, with $\boldsymbol{A}_s = \mathrm{diag}(\boldsymbol{a}_s)$, the mean coordinate variance of the standardized working density is $R^{-1}\,\mathrm{tr}(\boldsymbol{A}_s^{-1}\boldsymbol{\Lambda}_u^{-1}\boldsymbol{A}_s^{-1})$. In general this differs from N_u, which uses the single global scale v_s. Accordingly, ρ_u is a deterministic energy–curvature attenuation coefficient, not a calibrated confidence probability.

B.6 Correction operators and their structural properties

The learned raw factors are $\boldsymbol{U}^{\mathrm{raw}}, \boldsymbol{V}^{\mathrm{raw}} \in \mathbb{R}^{K\times q}$. At each forward pass, the implementation uses

$$\boldsymbol{U}_{:j} = \frac{\boldsymbol{U}^{\mathrm{raw}}_{:j}}{\max\{\left\|\boldsymbol{U}^{\mathrm{raw}}_{:j}\right\|_2, 10^{-6}\}}, \qquad \boldsymbol{V}_{:j} = \frac{\boldsymbol{V}^{\mathrm{raw}}_{:j}}{\max\{\left\|\boldsymbol{V}^{\mathrm{raw}}_{:j}\right\|_2, 10^{-6}\}}. \tag{27}$$

Columns have norm at most one, with their relative angles learned freely. The linear-operator rank is measured after centering, using $\boldsymbol{L}_u = \boldsymbol{P}\boldsymbol{U}\,\mathrm{diag}(\boldsymbol{g}_u)\boldsymbol{V}^\top\boldsymbol{P}$.

Proposition 2 (Factorized correction interface)**.** *For positive correction budgets, $\lambda_s > 0$, and fixed finite inputs: (i) $\mathrm{rank}(\boldsymbol{L}_u) \le \min\{q, K-1\}$; (ii) $\boldsymbol{V}^\top\boldsymbol{d} = \boldsymbol{0}$ implies $\boldsymbol{m}_u = \boldsymbol{0}$, including $\boldsymbol{d} = \boldsymbol{0}$; (iii) an empty or fully masked reference set gives $\boldsymbol{z}_S = \boldsymbol{z}_D$; (iv) Equation* (4) *holds; and (v) the state and prediction are invariant to a simultaneous permutation of the selected reference features, logits, and masks.*

Proof. The rank bound follows by matrix-product rank and $\mathrm{rank}(\boldsymbol{P}) = K-1$. If $\boldsymbol{a} = \boldsymbol{V}^\top\boldsymbol{d} = \boldsymbol{0}$, the mode interaction and its saturated output are zero. For absent references, Equation (26) gives $\rho_u = 0$, so the gain and increment vanish. Proposition 1 and $\|\boldsymbol{g}_u\|_\infty \le \alpha\rho_u$ give

$$\|\boldsymbol{m}_u\|_2 \le \|\boldsymbol{P}\boldsymbol{U}\,\mathrm{diag}(\boldsymbol{g}_u)\boldsymbol{a}\|_2 \le \alpha\rho_u\,\|\boldsymbol{P}\boldsymbol{U}\|_2\,\|\boldsymbol{a}\|_2 \tag{28}$$

$$\le \alpha\rho_u\,\|\boldsymbol{P}\boldsymbol{U}\|_2\,\|\boldsymbol{P}\boldsymbol{V}\|_2\,\|\boldsymbol{d}\|_2 \le \alpha\rho_u q\,\|\boldsymbol{d}\|_2\,. \tag{29}$$

Here $\boldsymbol{d} \in \mathcal{C}$, and column normalization gives $\|\boldsymbol{U}\|_2 \le \|\boldsymbol{U}\|_F \le \sqrt{q}$, with the same bound for $\boldsymbol{V}$. The absolute bound $\tau_m\sqrt{K}$ proves Equation (4). Finally, reference logits and residuals are per-row functions, while weights, normal equations, and curvature use symmetric sums. The fixed normalization and query map preserve this invariance. □

Proposition 3 (Reference sensitivity and decision margins)**.** *For a fixed query and a fixed trained model, let two admissible reference sets produce gains $\boldsymbol{g}$ and $\boldsymbol{g}'$. Their personalized logits satisfy*

$$\|\boldsymbol{z}_S(\boldsymbol{x},\mathcal{R}) - \boldsymbol{z}_S(\boldsymbol{x},\mathcal{R}')\|_2 \le \|\boldsymbol{P}\boldsymbol{U}\|_2\,\|\boldsymbol{a}(\boldsymbol{x})\|_2\,\|\boldsymbol{g}-\boldsymbol{g}'\|_\infty\,. \tag{30}$$

For distinct classes k, ℓ, the absolute logit-margin change is at most $\sqrt{2}\,\|\boldsymbol{m}_u(\boldsymbol{x})\|_2$. If DIRECT *has a unique winning class with a top-two logit gap larger than this bound, personalization preserves that winner.*

Proof. For the same query, $\boldsymbol{z}_D$ and $\boldsymbol{a}$ are fixed. The nonexpansiveness of $\mathcal{B}_{\tau_m}$ on centered inputs gives

$$\|\boldsymbol{m}(\boldsymbol{g}) - \boldsymbol{m}(\boldsymbol{g}')\|_2 \le \|\boldsymbol{P}\boldsymbol{U}\,\mathrm{diag}(\boldsymbol{g}-\boldsymbol{g}')\boldsymbol{a}\|_2 \tag{31}$$

$$\le \|\boldsymbol{P}\boldsymbol{U}\|_2\,\|\boldsymbol{g}-\boldsymbol{g}'\|_\infty\,\|\boldsymbol{a}\|_2\,. \tag{32}$$

For a class pair, the margin change is $(\boldsymbol{e}_k - \boldsymbol{e}_\ell)^\top\boldsymbol{m}$. Cauchy–Schwarz and $\|\boldsymbol{e}_k - \boldsymbol{e}_\ell\|_2 = \sqrt{2}$ give the stated bound. If the original winning margin exceeds it for every competitor, all winning margins remain positive. □

Equation (30) states the division of labor in quantitative form: reference variation acts through gain differences, while the current query sets their influence through $\|\boldsymbol{a}(\boldsymbol{x})\|_2$. Substituting the upper bound in Equation (4) gives a query-dependent sufficient condition for preserving the query-only winner.

The low-dimensional query interface. For fixed references, the reference-induced increment depends on the query correction through $\boldsymbol{a} = \boldsymbol{V}^\top\boldsymbol{d} \in \mathbb{R}^q$. Write $\widetilde{\boldsymbol{m}}_u = \boldsymbol{U}[\boldsymbol{a}\odot\boldsymbol{g}_u]$. On the centered correction space, its Jacobian is

$$D_{\boldsymbol{d}}\boldsymbol{m}_u = \boldsymbol{P}\,\mathrm{diag}\big(\mathrm{sech}^2(\boldsymbol{P}\widetilde{\boldsymbol{m}}_u/\tau_m)\big)\,\boldsymbol{P}\boldsymbol{U}\,\mathrm{diag}(\boldsymbol{g}_u)\boldsymbol{V}^\top\boldsymbol{P}, \tag{33}$$

whose rank is at most $\min\{q, K-1\}$. Thus the personalization branch retains a q-dimensional activation interface through the coordinatewise nonlinearity. Its centered pre-saturation image lies in $\mathrm{span}(\boldsymbol{P}\boldsymbol{U})$; after saturation it is parameterized by the same q activations, but need not lie in that linear span. The complete correction $\boldsymbol{d} + \boldsymbol{m}_u$ has Jacobian $\boldsymbol{I}_{\mathcal{C}} + D_{\boldsymbol{d}}\boldsymbol{m}_u$ with respect to $\boldsymbol{d}$, retaining the query-only identity path.

Source-centered shrinkage. For $\boldsymbol{d} \in \mathcal{C}$ and the linear subject gain $\boldsymbol{W}\boldsymbol{s}$, the read–gain–write expression is the bilinear tensor contraction

$$\boldsymbol{P}\boldsymbol{U}[(\boldsymbol{V}^\top \boldsymbol{d}) \odot (\boldsymbol{W}\boldsymbol{s})] = \mathcal{T} \times_2 \boldsymbol{d} \times_3 \boldsymbol{s}, \qquad \mathcal{T}_{abr} = \sum_{j=1}^{q} (\boldsymbol{P}\boldsymbol{U})_{aj} (\boldsymbol{P}\boldsymbol{V})_{bj} W_{jr}. \tag{34}$$

This is a rank-at-most-q CP parameterization of the bilinear core. The energy attenuation determines the local order of the complete map. For a nonempty set, hold the curvature summary fixed and write $n = \nu_u / v_s + \epsilon > 0$. As $\boldsymbol{s} \to \boldsymbol{0}$,

$$\begin{aligned} \rho(\boldsymbol{s}) &= \frac{\|\boldsymbol{s}\|_2^2}{Rn} + O(\|\boldsymbol{s}\|_2^4), \\ \boldsymbol{g}(\boldsymbol{s}) &= \frac{\alpha}{Rn} \|\boldsymbol{s}\|_2^2 \boldsymbol{W}\boldsymbol{s} + O(\|\boldsymbol{s}\|_2^5). \end{aligned} \tag{35}$$

For a fixed bounded query correction and fixed positive τ_m, Proposition 1 consequently gives

$$\boldsymbol{m}(\boldsymbol{d}, \boldsymbol{s}) = \frac{\alpha}{Rn} \|\boldsymbol{s}\|_2^2 \boldsymbol{P}\boldsymbol{U}[(\boldsymbol{V}^\top \boldsymbol{d}) \odot (\boldsymbol{W}\boldsymbol{s})] + O(\|\boldsymbol{s}\|_2^5). \tag{36}$$

The saturation remainder is of order $\|\boldsymbol{s}\|_2^9$ here. The personalized increment is therefore $O(\|\boldsymbol{s}\|_2^3)$ near the source center, with the displayed cubic coefficient determining its leading effect when nonzero. This smooth shrinkage damps small state deviations more strongly than the unattenuated bilinear interaction.

B.7 Source episodic learning and implementation settings

Stage II retains the trained DIRECT head in evaluation mode and freezes its parameters, the activity dictionaries, and the state-normalization statistics. It learns the read/write factors $\boldsymbol{U}, \boldsymbol{V}$ and the gain map $\boldsymbol{W}$. Reference states therefore keep the same inference map throughout episodic learning.

The query-only anchor. Let $\pi_i^D = \mathrm{softmax}(\boldsymbol{z}_{D,i})$, $\pi_i^S = \mathrm{softmax}(\boldsymbol{z}_{S,i})$, and define the frozen anchor set

$$\mathcal{H} = \{i : \arg\max_k z_{D,i,k} = y_i,\ \pi_{i,(1)}^D - \pi_{i,(2)}^D \geq \delta_p\}, \qquad \delta_p = 0.6. \tag{37}$$

The modulation penalty is $\mathcal{L}_m = (BK)^{-1} \sum_i \|\boldsymbol{m}_i\|_2^2$. The anchor-relative term is

$$\mathcal{L}_a = \frac{1}{B} \sum_i [\ell_i^S - \ell_i^D + \delta_a]_+, \qquad \delta_a = 0 \text{ by default.} \tag{38}$$

The preservation loss is $\mathcal{L}_{\mathrm{keep}} = |\mathcal{H}|^{-1} \sum_{i \in \mathcal{H}} \mathrm{KL}(\pi_i^D \| \pi_i^S)$ for $|\mathcal{H}| > 0$, and zero otherwise. The preservation term is averaged over $\mathcal{H}$ and uses the frozen DIRECT distribution as the teacher target.

The Stage-II objective is

$$\mathcal{L}_S = \mathrm{CE}_{\mathrm{batch}}(\boldsymbol{z}_S, y) + \lambda_m \mathcal{L}_m + \lambda_a \mathcal{L}_a + \lambda_p \mathcal{L}_{\mathrm{keep}}. \tag{39}$$

Equation (8) uses $\delta_a = 0$, with unweighted leading cross-entropy. The read/write columns are normalized as in Equation (27), and their relative directions are learned through this objective.

Interpretation of the anchor-relative term. For fixed ℓ_i^D, away from the hinge boundary,

$$\nabla_{\boldsymbol{z}_{S,i}} \left(\ell_i^S + \lambda_a [\ell_i^S - \ell_i^D + \delta_a]_+\right) = \left(1 + \lambda_a \mathbb{I}[\ell_i^S - \ell_i^D + \delta_a > 0]\right) (\pi_i^S - \boldsymbol{e}_{y_i}). \tag{40}$$

Thus the anchor-relative term emphasizes queries whose personalized loss exceeds the frozen anchor's threshold. It changes the weight of the ordinary classification gradient while retaining its direction. At the boundary, use the hinge subgradient. The magnitude and distribution terms regularize the extent of personalization relative to the same query-only anchor.

Initialization and defaults. The query head's output layer starts at zero. Stage II initializes the raw read/write matrices and gain map with independent $\mathcal{N}(0, 0.05^2)$ entries. Table 9 gives the default settings; Table 8 specifies task-dependent mode counts and modulation scales.

Table 9: Core configuration defaults. The modulation scale α is specified in Table 8.

Quantity	Symbol	Default
State width / operator width	R / q	8 / 2
State ridge	λ_s	0.1
Pseudo-label temperature / power	T / γ	1 / 2
Class-mass scale	τ_c	1
DIRECT saturation scale	τ_d	2
Modulation scale	α	1 (profile-specific)
DIRECT norm penalty	λ_d	10^{-3}
Modulation norm / anchor-relative penalty	λ_m / λ_a	10^{-3} / 0.5
Preservation penalty	λ_p	0.1
Relative-loss / preservation margin	δ_a / δ_p	0 / 0.6
Gain initialization standard deviation	—	0.05
Reference budget	M	64

Algorithm 1 QSF inference with an admissible same-user reference set

Require: Query $\boldsymbol{x}$; selected reference rows $\mathcal{R}_u$ with support mask e_m (all ones if unpadded); frozen f_0, r_θ, $\{\boldsymbol{\mu}_k, \boldsymbol{D}_k\}$, normalizer $(\boldsymbol{m}_s, \boldsymbol{a}_s, v_s)$; learned $\boldsymbol{U}_{\text{raw}}, \boldsymbol{V}_{\text{raw}}, \boldsymbol{W}$; α, bounds τ_d, τ_m, and inference settings (Appendix B.7)

$(\boldsymbol{h}, \boldsymbol{z}_0) \leftarrow f_0(\boldsymbol{x})$
$\boldsymbol{d} \leftarrow \mathcal{B}_{\tau_d}(r_\theta(\boldsymbol{q}(\boldsymbol{x}))); \boldsymbol{z}_D \leftarrow \boldsymbol{z}_0 + \boldsymbol{d}$
if $\mathcal{R}_u = \varnothing$ or $\sum_m e_m = 0$ **then**
    return $\mathrm{softmax}(\boldsymbol{z}_D)$
for each reference row m **do**
    Obtain frozen $\boldsymbol{h}_m, \boldsymbol{z}_D(\boldsymbol{x}_m)$; use finite values for padded rows
    Compute $\boldsymbol{p}_m$ by Equation (19)
Mask $p^e_{m,k} \leftarrow e_m p_{m,k}$, then floor $\widetilde{p}^e_{m,k} \leftarrow \max\{p^e_{m,k}, \epsilon\}$
Form $a_k, w_{m,k}$ by Equation (20)
Form $\boldsymbol{\Lambda}_u, \boldsymbol{b}_u$; solve $\widehat{\boldsymbol{s}}_u \leftarrow \mathrm{solve}(\boldsymbol{\Lambda}_u, \boldsymbol{b}_u)$
$\nu_u \leftarrow R^{-1} \operatorname{tr}(\boldsymbol{\Lambda}_u^{-1})$
Form $\boldsymbol{s}_u, \rho_u$ using Equation (26)
Normalize the read/write columns by Equation (27)
$\boldsymbol{g}_u \leftarrow \alpha \rho_u \tanh(\boldsymbol{W} \boldsymbol{s}_u)$
$\boldsymbol{m}_u \leftarrow \mathcal{B}_{\tau_m}(\boldsymbol{U}[(\boldsymbol{V}^\top \boldsymbol{d}) \odot \boldsymbol{g}_u])$
return $\mathrm{softmax}(\boldsymbol{z}_D + \boldsymbol{m}_u)$

B.8 INFERENCE ALGORITHM, COMPLEXITY, AND CACHING

The early-return and vectorized fully masked paths recover the same DIRECT logits before softmax. The numerical implementation uses the floors specified above, with floating-point comparisons evaluated to numerical tolerance.

Reference-state algebra. All K classes are used for every reference, so residual projection costs $O(MKDR)$. Forming class Gram matrices costs $O(KDR^2)$, precision accumulation costs $O(KR^2)$, and both the solve and inverse cost $O(R^3)$. The reference implementation forms the Gram matrices on each call. Caching these source-fixed matrices removes the $O(KDR^2)$ term while preserving predictions. Explicitly materializing reference residuals and projections requires $O(MKD+MKR+KR^2+R^2)$ working memory per reference episode, besides frozen features and the $O(KDR)$ dictionaries.

Query-time modulation. Computing subject gains costs $O(qR)$. Once gains are cached, the incremental read–gain–write computation and saturation cost $O(Kq+K)$ per query. The parameter tensors store $q(2K+R)$ scalars. Shared references permit caching of the state, curvature summary, attenuation, and gains. Query-specific eligible sets use their corresponding inferred states.

End-to-end cost. End-to-end computation comprises feature extraction, reference selection, state inference, and query modulation. If references are preselected, encoding requires M frozen population and DIRECT passes. If feature-based selection first examines the eligible pool, uncached preparation instead requires N_e such passes, followed by state inference on the selected M windows. With running minimum distances, greedy selection of M references from N_e eligible feature vectors costs $O(MN_eD)$. Separating these preparation costs from the cached per-query computation makes the cost of each deployment protocol explicit.

C MATCHED HEADS AND REFERENCE INTERVENTIONS

C.1 CONDITIONING CONTROLS AND SHARED EVALUATION SETTINGS

The controls test which information should drive a prediction change and how query and reference signals should interact.

Conditioning interfaces. Each control adds a bounded increment to the common DIRECT logits $\boldsymbol{z}_D$. For a two-layer GELU network F, the query-only head uses $\mathcal{B}_{\tau_m}(\alpha F(\boldsymbol{d}))$, the additive-state head uses $\mathcal{B}_{\tau_m}(\alpha\rho_u F(\boldsymbol{s}_u))$, and Joint uses $\mathcal{B}_{\tau_m}(\alpha\rho_u F([\boldsymbol{d};\boldsymbol{s}_u]))$. For the compact MLP controls, we choose the smallest integer hidden width for which the added parameter count is at least that of QSF. The query-only head remains active without references; the additive-state and Joint heads recover DIRECT when references are absent. Joint and QSF share reference-state inference, attenuation ρ_u, the bounded output map $\mathcal{B}_{\tau_m}$, and the source-training objective.

QSF and Logit share the multiplicative conditioning form and differ in the read input: the learned query correction $\boldsymbol{d}$ versus energy-scaled centered logits $c_z\boldsymbol{P}\boldsymbol{z}_D$. The scale is fitted on source data as $c_z^2 = (\mathbb{E}\|\boldsymbol{d}\|_2^2 + \epsilon)/(\mathbb{E}\|\boldsymbol{P}\boldsymbol{z}_D\|_2^2 + \epsilon)$. Logit's query activation is $\boldsymbol{a}_Z(\boldsymbol{x}) = \boldsymbol{V}_Z^\top c_z\boldsymbol{P}\boldsymbol{z}_D(\boldsymbol{x})$; it obeys the same forms of the norm and reference-substitution bounds in Equations (4) and (30). QSF and Logit each add 64/40/28 conditioning parameters on HAPT/MotionSense/WESAD. Wide Joint has 32 hidden units and adds 1,068/678 parameters on HAPT/MotionSense.

Shared training settings. The eight-task interface study and its associated controls use the common frozen Population/DIRECT pair defined in Section 4.1. Within each seed, all conditioning heads share source supervision, source query–reference episodes, and target queries. Heads are trained for 120 epochs and evaluated at the prespecified final checkpoint. Auxiliary control-head learning rates are selected using source-only development and then fixed. QSF and Logit share initial modes and gain maps. The eight-task interface comparison and its HAPT/MotionSense/WESAD controls use AdamW at learning rate 0.005, zero weight decay, cosine decay, batch size 512, and gradient clipping at norm 5.

Unless varied, $q = 2$, $R = 8$, $\lambda_s = 0.1$, $T = 1$, $\gamma = 2$, $\tau_c = 1$, $\tau_d = 2$, and the reference budget is 64. The modulation scales are $\alpha = 1$, 2.5, and 2 for HAPT, MotionSense, and WESAD. In Equation (8), $(\lambda_a, \lambda_m, \lambda_p) = (0.5, 0.001, 0.1)$ and $\delta_p = 0.6$; loss weights and reductions follow Appendix B.7. A changed state encoder receives a refitted source-user-balanced normalizer and a newly trained head.

Decision-change metrics. Aggregation follows Appendix A.1. In temporal comparisons, F1 is computed within each user, policy, gap, and draw before averaging.

For correctness indicators $C_{D,i}$ and $C_{M,i}$ of DIRECT and a conditioned head M, repair and harm per 100 queries are

$$R_{100}(M) = \frac{100}{N}\sum_{i=1}^{N}\mathbf{1}\{\neg C_{D,i} \wedge C_{M,i}\}, \tag{41}$$

$$H_{100}(M) = \frac{100}{N}\sum_{i=1}^{N}\mathbf{1}\{C_{D,i} \wedge \neg C_{M,i}\}. \tag{42}$$

Their difference is the accuracy change from DIRECT in percentage points; W-F1 is computed separately from the confusion matrix.

C.2 QSF VERSUS JOINT ACROSS EIGHT TASKS

QSF achieves higher mean W-F1 than the compact Joint head on all eight tasks (Table 10). Table 11 gives absolute scores for the conditioning controls on HAPT, MotionSense, and WESAD. These fixed-endpoint comparisons share frozen Population/DIRECT pairs and use the matched configurations, distinct from the complete-system benchmark settings.

Table 10: QSF versus Joint across eight tasks. Paired pooled-query W-F1 differences (pp), mean ± sample SD over five seeds under the matched protocol in Section 4.1. Both heads share the upstream predictor and target queries within seed. These contrasts appear in Figure 2.

Dataset	QSF − Joint (pp W-F1)
DSADS	$+3.443 \pm 1.808$
HAPT	$+0.487 \pm 0.716$
MotionSense	$+0.384 \pm 0.225$
Shoaib-2014	$+15.088 \pm 13.560$
SleepEDF-20	$+0.493 \pm 0.661$
UCI-HAR	$+1.040 \pm 0.733$
UniMiB	$+1.206 \pm 0.841$
WESAD	$+0.358 \pm 1.251$

Table 11: Conditioning controls with a common frozen upstream predictor. Pooled-query W-F1 (%), mean ± sample SD over five seeds, under the protocol in Section 4.1. Parameter counts cover only the added Stage-II head and are ordered HAPT/MotionSense/WESAD.

Head	Added Stage-II parameters	HAPT	MotionSense	WESAD
Population recognizer	0/0/0	97.22 ± 0.17	96.74 ± 0.50	80.93 ± 2.14
DIRECT	0/0/0	97.12 ± 0.22	96.92 ± 0.50	81.20 ± 2.79
Query-only head (Q)	87/45/31	96.79 ± 0.43	96.60 ± 0.61	81.01 ± 3.22
Additive state (A)	75/51/39	96.93 ± 0.34	96.82 ± 0.54	81.12 ± 2.86
Joint	78/48/33	96.49 ± 0.68	96.42 ± 0.70	81.34 ± 3.43
Logit	64/40/28	96.99 ± 0.29	96.87 ± 0.58	81.78 ± 2.63
QSF	64/40/28	96.98 ± 0.36	96.80 ± 0.60	81.69 ± 2.63

Widening Joint to 32 hidden units improves its mean W-F1 to 96.910 ± 0.256 on HAPT and 96.491 ± 0.897 on MotionSense, narrowing the mean gaps to QSF to 0.068 and 0.310 points. Wide Joint uses 1,068/678 conditioning parameters, compared with QSF's 64/40. Thus, the factorized interface retains its mean W-F1 advantage on these two tasks with substantially fewer conditioning parameters than the widened joint interface.

C.3 DECISION CHANGES AND TRAINING OBJECTIVES

On HAPT and MotionSense, preserving correct DIRECT predictions explains most of the QSF–Joint accuracy advantage. Compact Joint's additional repairs do not compensate for its greater disruption rate (Table 12). The mean net accuracy changes relative to DIRECT are $-0.140/-0.120$ pp for QSF and $-0.627/-0.510$ pp for Joint on HAPT/MotionSense, respectively. Figure 3a shows the mean repair and harm rates with sample-SD error bars; its annotations report the resulting QSF–Joint accuracy differences.

Table 12: Decision changes in Figure 3a. Entries are mean ± sample SD over five seeds, in events per 100 queries, pooled within dataset and seed. Heads share DIRECT predictions and query identities.

Dataset	Head	Repair	Harm
HAPT	QSF	0.044 ± 0.048	0.184 ± 0.137
	Joint	0.241 ± 0.116	0.868 ± 0.670
MotionSense	QSF	0.257 ± 0.188	0.377 ± 0.306
	Joint	0.290 ± 0.195	0.800 ± 0.389

Objective ablation. The cross-entropy-only control sets $\lambda_a = \lambda_m = \lambda_p = 0$ during Stage-II head training, keeping the upstream predictor frozen and the remaining training protocol fixed. QSF retains a higher mean W-F1 than Joint when both conditioning heads are trained with cross-entropy alone, with gaps of 1.487 and 0.593 points on HAPT and MotionSense, respectively (Table 13). The interface advantage therefore extends to the simpler cross-entropy-only objective as well as the full Stage-II objective. The table also reports removal of each QSF regularizer and the corresponding repair–harm counts.

Table 13: Objective ablations on the full target split. W-F1 is pooled-query percentage, mean ± sample SD over five seeds. Repair (R) and harm (H) are mean events per 100 queries relative to DIRECT.

Head / objective	HAPT			MotionSense		
	W-F1	R	H	W-F1	R	H
QSF	96.978 ± 0.365	0.044	0.184	96.801 ± 0.598	0.257	0.377
Joint	96.491 ± 0.684	0.241	0.868	96.417 ± 0.698	0.290	0.800
Logit	96.989 ± 0.295	0.038	0.165	96.870 ± 0.581	0.156	0.207
QSF, CE only	96.891 ± 0.420	0.051	0.279	96.792 ± 0.540	0.198	0.326
Joint, CE only	95.404 ± 2.415	0.234	1.939	96.199 ± 0.799	0.368	1.098
Logit, CE only	96.892 ± 0.398	0.063	0.292	96.792 ± 0.503	0.184	0.312
QSF, $\lambda_a = 0$	96.909 ± 0.399	0.051	0.260	96.788 ± 0.551	0.207	0.340
QSF, $\lambda_m = 0$	96.977 ± 0.356	0.038	0.177	96.925 ± 0.582	0.253	0.248
QSF, $\lambda_p = 0$	96.984 ± 0.352	0.038	0.171	96.796 ± 0.609	0.271	0.395

C.4 DSADS DECISION-LEVEL INTERVENTION ANALYSIS

This analysis connects the choice of read input to the repairs and harms that determine net accuracy change. We evaluate $2{,}280$ queries from DSADS users 7–8 ($1{,}140$ each, 19 classes) across five seeds. The common upstream and source-trained heads remain frozen, with the same query and reference identities across each intervention. Logit matches the source input energy of the learned correction; both heads are evaluated at their trained output scales. Table 14 reports W-F1 differences and decision events. At the trained scales, the learned-correction input yields a larger mean net accuracy gain than Logit: $+0.1754$ versus $+0.0877$ pp. The additional repairs outweigh the additional harmful changes on average. Event rates coincide under pooled and equal-user aggregation because the two users have equal query counts.

Table 14: DSADS decision changes at the trained scales, mean ± sample SD over five seeds. (a) Paired W-F1 differences (pp). (b) Repair and harm per 100 queries and their net accuracy change from DIRECT (pp).

(a) Recognition differences

Contrast	Pooled queries	Equal users
QSF–DIRECT	$+0.1536 \pm 0.1862$	$+0.1946 \pm 0.2316$
QSF–Logit	$+0.0620 \pm 0.2642$	$+0.0664 \pm 0.3469$

(b) Decision events

Head	Repair	Harm	Net accuracy gain (pp)
QSF	0.2719 ± 0.3688	0.0965 ± 0.1682	$+0.1754 \pm 0.2034$
Logit	0.1053 ± 0.0961	0.0175 ± 0.0240	$+0.0877 \pm 0.0821$

Correction scaling and query-level changes. Figure 4a evaluates $\boldsymbol{z}_{M,\beta} = \boldsymbol{z}_D + \beta(\boldsymbol{z}_M - \boldsymbol{z}_D)$ for $M \in \{\mathrm{QSF}, \mathrm{Logit}\}$ and $\beta = 0, 0.05, \ldots, 1.5$. Zero recovers DIRECT and $\beta = 1$ gives each head's trained output scale. The multiplier acts on each head's own output increment; it does not match correction norms between the two heads.

For Figure 4b, the true-class margin is $m_M = z_{M,y} - \max_{k \neq y} z_{M,k}$, with the strongest competitor recomputed for each predictor; the margin change from DIRECT is $\Delta m_M = m_M - m_D$. For each query, margin changes and binary repair/harm indicators are computed separately for each model seed and then averaged across the five seeds. Within each user, queries are ordered by decreasing mean DIRECT entropy divided by $\log 19$, with query ID breaking ties, and partitioned into 128 contiguous bins containing eight or nine queries each. Each bin displays the arithmetic mean of its constituent query values, with equal weight assigned to every query. Both heads share query order, bins, and color scales. Ordering and binning are display steps; predictions use the individual trained heads. Table 14 and Figure 4a use all queries directly, without binning.

C.5 REFERENCE SUBSTITUTION AND CONSTANT-STATE CORRECTION ON WESAD

The WESAD controls test the predictive value of inferring a variable state from the references. Own-reference QSF improves mean-user W-F1 over both DIRECT and a constant-state control with the same number of trainable parameters (Table 15). The class-matched reference intervention uses the QSF model from Table 11, with $R = 8$, $q = 2$, and $\alpha = 2$, on 1,334 query windows from users 15–17. Each of the other two users supplies a donor set. Own-user and donor sets contain 64 references with identical true-class counts. Labels prescribe this diagnostic composition and do not enter state inference. Own-user pools exclude the query index but may overlap its raw interval.

Matched constant-state control. The control C_S matches QSF's 28 trainable parameters, initialization, source supervision, objective, and 120-epoch fitting budget. Its state is $\boldsymbol{s}_\star = \sqrt{\mathrm{tr}(\boldsymbol{M})}\, \boldsymbol{v}_{\max}(\boldsymbol{M})$, where $\boldsymbol{M}$ is the source-state second moment, averaging queries within users and then users equally. The leading unit eigenvector has its largest-magnitude coordinate positive. The attenuation is the equal-user source mean. These constants replace the variable inputs without further standardization; the support mask still recovers DIRECT for empty references. Training uses 5,239 source queries from 12 users and the shared optimization settings; QSF and C_S are evaluated in CPU FP32.

Table 15: WESAD with diagnostic class-matched references and a matched constant-state control. Scores are equal-user W-F1 (%); differences are percentage points. Entries are mean ± sample SD over five seeds, averaging donor scores within each of three users first.

Predictor	WF1	Paired difference from DIRECT (pp)
DIRECT	80.627 ± 2.413	—
Constant state C_S	80.355 ± 2.790	-0.272 ± 0.462
Own-user QSF	80.736 ± 2.425	$+0.108 \pm 0.150$
Donor-reference QSF	80.144 ± 1.610	-0.483 ± 1.683

	Own QSF–C_S	Own QSF–Donor QSF
Paired mean ± SD	$+0.380 \pm 0.492$	$+0.591 \pm 1.735$

The variable reference state adds value under both reference constructions: own-reference QSF exceeds C_S by 0.380 ± 0.492 W-F1 points with class-matched references and by 0.619 ± 1.002 points with full-session pools. In the latter setting, the same QSF model obtains 80.974 ± 2.202 mean-user W-F1; the constant head has identical predictions under both pool constructions. These comparisons support inferring context from the available references rather than using a fixed source-derived state.

The class-matched donor comparison further links reference content to decision preservation. Own references produce 0.279 harmful changes per 100 queries versus 0.738 for donors, while repair rates are 0.316 and 0.391. The lower harm rate is the main source of the own-reference accuracy advantage.

C.6 REFERENCE CONDITIONING UNDER SHARED SENSOR VARIATION

Shared sensor variation tests whether references remain useful when the query's measurement conditions change. We apply the same channel-wise affine transform to each wearer's queries and references. The matched heads on HAPT, MotionSense, and WESAD use the 17 target users and five model seeds of the matched comparison. In source-standardized sensor coordinates, wearer u, channel c, and perturbation b receive

$$x'_{u,b,c}(t) = g_{u,b,c} x_{u,c}(t) + a_{u,b,c}, \qquad \log g_{u,b,c} \sim \mathcal{N}(0, 0.1^2), \quad a_{u,b,c} \sim \mathcal{N}(0, 0.1^2). \tag{43}$$

Four fixed realizations per wearer are shared across model seeds. Source normalization and reference indices stay fixed; features, corrections, reference states, and attenuation are recomputed from the transformed signals. Scores average each user's complete W-F1 across realizations, then equally across users and datasets within seed.

For this experiment, C_S uses a coordinatewise RMS source state: standardized state coordinates are squared, averaged within users and then across users, and square-rooted. Attenuation is the corresponding arithmetic mean. This source-fitted constant receives the same clean-source supervision and training budget as QSF. Other-user reference sets have equal counts and are matched by predicted class composition; each donor is scored separately. Different wearers also have different imposed transforms, so own–other replacement changes both wearer and measurement condition.

Table 16: Frozen QSF under shared sensor-affine inputs. Paired W-F1 differences (pp), mean ± sample SD over five seeds. Scores average complete per-user W-F1 over transforms and users; the last row weights the three datasets equally. C_S uses a clean-source RMS constant state.

Dataset	QSF–DIRECT	QSF–C_S	Own–other references
Shared sensor-affine inputs			
HAPT	-0.126 ± 0.178	$+0.041 \pm 0.193$	-0.052 ± 0.185
MotionSense	$+0.706 \pm 1.342$	$+0.746 \pm 1.379$	$+0.515 \pm 1.265$
WESAD	$+0.138 \pm 0.560$	$+0.352 \pm 0.442$	-0.070 ± 0.262
Three-task mean	$+0.239 \pm 0.466$	$+0.380 \pm 0.550$	$+0.131 \pm 0.418$

Under the prescribed shared affine transforms, QSF exceeds DIRECT by 0.239 ± 0.466 W-F1 points on the equal-task mean, with the largest improvement on MotionSense (Table 16). It also exceeds the constant-state control in mean W-F1 on all three tasks, with an equal-task difference of 0.380 ± 0.550 points. The variable reference state thus retains predictive value under these changes in sensing conditions.

Within-wearer measurement matching. We keep wearer identity and reference windows fixed and vary only the reference transform. For query transform b, matched references use b; each of the other three transforms supplies a mismatch. Scores are computed for each complete wearer query set, then averaged over mismatches, query transforms, users, and tasks. DIRECT and the constant control remain fixed for each transformed query.

Table 17: Within-wearer measurement matching for frozen QSF on 17 wearers. Paired W-F1 differences (pp), mean $\pm$ sample SD over five seeds. Δ_{match} compares matched references with the mean score of three same-wearer transform mismatches. C_S is the clean-source RMS control.

Dataset	Matched$-$DIRECT	Matched$-C_S$	Δ_{match}
HAPT	-0.126 ± 0.178	$+0.041 \pm 0.193$	$+0.011 \pm 0.034$
MotionSense	$+0.706 \pm 1.342$	$+0.746 \pm 1.379$	$+0.740 \pm 1.035$
WESAD	$+0.138 \pm 0.560$	$+0.352 \pm 0.442$	-0.131 ± 0.162
Equal-task mean	$+0.239 \pm 0.466$	$+0.380 \pm 0.550$	$+0.206 \pm 0.349$

Matching measurement conditions gives the largest mean benefit on MotionSense; HAPT changes little and WESAD has the opposite response (Table 17). The equal-task matching effect is $+0.206 \pm 0.349$ points for these imposed transforms. Reference compatibility therefore involves measurement conditions even when wearer identity and reference windows are fixed.

C.7 REFERENCE COMPOSITION, SOURCE SUBSTITUTION, AND ACQUISITION BUDGET

We separately vary activity composition, the wearer supplying the references, and reference-set size, keeping queries fixed within each comparison (Figure 6). Table 18 lists the reference budgets and eligible cohorts. QSF uses the matched head; empty references recover DIRECT.

Table 18: Reference-intervention settings. Composition and source substitution use five reference draws and ground-truth class counts to define diagnostic sets; labels do not enter reference-state inference.

Intervention	HAPT	MotionSense
Composition	$M = 8$, five users; nominal 0.8 realizes $6/8 = 0.75$	$M = 32$, five users
Source substitution	$M = 16$, five users; exact class-count matching	$M = 32$, five users; exact class-count matching
Budget	Nine users through $M = 32$; separate seven-user curve through $M = 64$	Five users through $M = 128$

Composition within the same wearer. Three source-chosen activities each serve as the dominant class at nominal fractions $\{1/3, 0.5, 0.8, 1\}$. The balanced condition uses the most even feasible integer allocation. Frequency weighting redistributes the same total evidence mass, while the global-center residual control replaces activity-specific prototypes with the source global mean. Each alternative state encoder receives a refitted source normalizer and a newly trained conditioning head.

QSF stays near its balanced-reference W-F1 across the HAPT sweep. With single-class references on MotionSense, global centering yields a larger mean drop than QSF (Figure 6, left). This suggests that subtracting class-specific feature offsets helps retain useful context when the reference mix is concentrated.

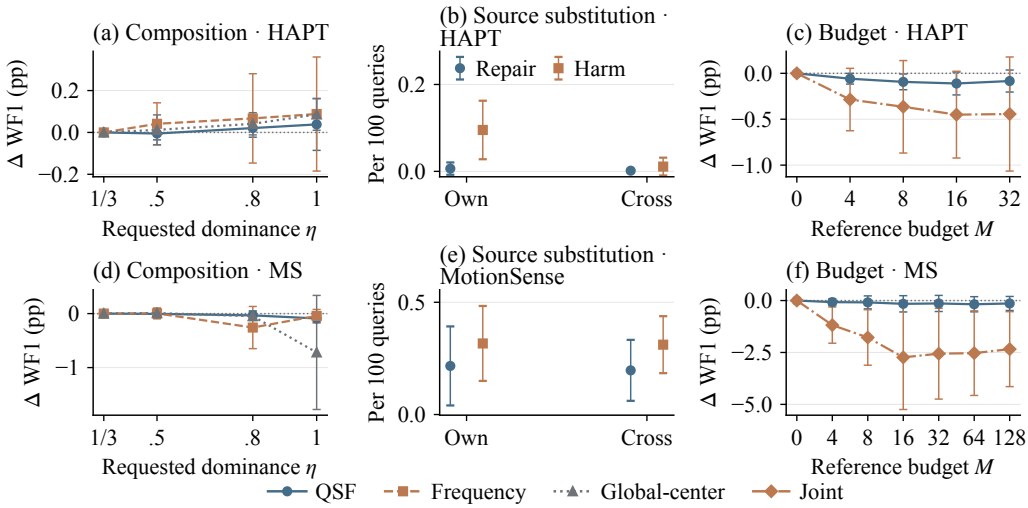


Figure 6: Reference interventions on HAPT (top) and MotionSense (bottom). Left: W-F1 change from balanced references as class concentration increases. Middle: repairs and harms per 100 queries with own-user or class-matched donor references. Right: QSF/Joint minus DIRECT W-F1 along nested acquired reference budgets. Points are equal-user means and bars are sample SDs over five seeds after averaging draws. Conditions are in Table 18.

Reference-source substitution. We compare empty references, own-user references, another own-user draw, and cross-user donors with exactly matched class counts. Class-matched donor substitution has task-dependent effects. HAPT and MotionSense show no consistent preference for own-user references, whereas the WESAD comparison in Appendix C.5 associates own-user references with fewer harmful changes. This suggests that correction-relevant context need not be unique to a wearer: class-matched donor recordings may yield states that interact similarly with the query's active modes.

Acquired references. Budgets are nested and encode exactly M selected raw windows; $M = 0$ recovers DIRECT. HAPT's nine-user and seven-user curves use separate eligible cohorts, with $M = 128$ infeasible. MotionSense keeps five users through $M = 128$. QSF stays close to DIRECT across these budgets, whereas Joint shows larger mean degradation on the same query sets (Figure 6). QSF's small performance changes across budgets are consistent with its conditioning rule: per-class evidence saturates, and reference gains act through query activations, so reference quantity does not directly set correction magnitude. MotionSense establishes sample-index disjointness; its metadata do not establish raw-interval nonoverlap or past-only eligibility.

C.8 RANK AND RIDGE SENSITIVITY

The compact interface maintains similar recognition performance across the rank and ridge settings in Table 19: the full spread of mean W-F1 is about 0.12 points on HAPT and 0.19 on MotionSense. One or two modes already attain scores close to the best in the sweep. A plausible explanation is that the frozen recognizer already supplies most of the class-discriminative structure, leaving a low-dimensional set of adjustments for reference conditioning. Each setting changes one factor from $q = 2$, $R = 8$, and $\lambda_s = 0.1$. Changing q rebuilds the head; changing R rebuilds source dictionaries and normalization; changing λ_s recomputes source states and normalization. Heads train for 120 epochs. Dictionary construction follows Appendix B.3.

Table 19: One-factor rank and ridge sensitivity. Entries are Stage-II parameter counts and pooled-query W-F1 (%), mean ± sample SD over five seeds. The default is $q = 2$, $R = 8$, $\lambda_s = 0.1$; other factors stay fixed. The shared default appears in each block. $q = 8$ is evaluated only on HAPT.

Factor	HAPT		MotionSense	
	Parameters	WF1	Parameters	WF1
$q = 1$	32	97.003 ± 0.271	20	96.921 ± 0.486
$q = 2$	64	96.978 ± 0.365	40	96.801 ± 0.598
$q = 4$	128	96.964 ± 0.296	80	96.801 ± 0.660
$q = 8$	256	96.970 ± 0.299	–	–
$R = 2$	52	97.084 ± 0.246	28	96.894 ± 0.475
$R = 4$	56	97.067 ± 0.249	32	96.733 ± 0.447
$R = 8$	64	96.978 ± 0.365	40	96.801 ± 0.598
$\lambda_s = 0.03$	64	97.034 ± 0.357	40	96.810 ± 0.637
$\lambda_s = 0.1$	64	96.978 ± 0.365	40	96.801 ± 0.598
$\lambda_s = 0.3$	64	96.996 ± 0.326	40	96.898 ± 0.552

C.9 Temporal reference availability

Table 20 compares policies at $M = 32$ on the intersection of queries eligible for every policy and gap. Calibration uses the first 32 nonoverlapping reference windows; past-only excludes future windows; full-session uses the complete session. HAPT and WESAD provide raw timing metadata for gaps $\{0, 2L, 10L\}$, where L is the window length. Every common query starts at least $10L$ after the calibration pool ends; the tabulated gap is the additional exclusion threshold.

QSF preserves the same HAPT W-F1 across the frozen-head policies. On WESAD, it achieves higher mean W-F1 than Joint under every policy, with its highest frozen-head scores using full-session references. Matching source training to the calibration policy raises QSF's mean W-F1 from 76.969 to 77.294 on WESAD and from 99.763 to 99.782 on HAPT. This row retrains QSF and Joint using source calibration episodes and their corresponding normalizers, illustrating the value of aligning training references with deployment availability.

Table 20: Temporal reference policies on fixed eligible query sets ($M = 32$): nine HAPT users / 533 queries and three WESAD users / 521 queries. Equal-user W-F1 (%), mean ± sample SD over five seeds. The matched row uses heads trained on source calibration episodes; other rows use frozen matched heads.

Protocol	Gap	HAPT		WESAD	
		QSF	Joint	QSF	Joint
Calibration	$0L$	99.763 ± 0.082	99.732 ± 0.140	76.969 ± 1.548	76.955 ± 3.413
Past only	$0L$	99.763 ± 0.082	99.742 ± 0.126	76.720 ± 2.275	76.570 ± 2.552
Past only	$2L$	99.763 ± 0.082	99.752 ± 0.115	76.772 ± 2.146	76.544 ± 2.813
Past only	$10L$	99.763 ± 0.082	99.752 ± 0.115	76.726 ± 2.111	76.541 ± 2.816
Full session	$0L$	99.763 ± 0.082	99.732 ± 0.140	77.401 ± 2.204	76.921 ± 2.831
Full session	$2L$	99.763 ± 0.082	99.752 ± 0.115	77.414 ± 2.481	76.837 ± 2.807
Full session	$10L$	99.763 ± 0.082	99.769 ± 0.106	77.452 ± 2.159	76.921 ± 2.829
Calibration, matched	$0L$	99.782 ± 0.107	99.768 ± 0.106	77.294 ± 2.801	76.653 ± 3.589

DIRECT is 99.763 ± 0.082 on HAPT and 77.342 ± 2.583 on WESAD for every row. These temporally eligible query subsets differ from the complete benchmark splits.

C.10 Query-dependent influence and numerical consistency

Measured change and bound. To relate the structural bound to observed reference influence, we fix the query and model and compare the final logits obtained from two reference sets:

$$O(\boldsymbol{x}, \mathcal{R}, \mathcal{R}') = \|\boldsymbol{z}_S(\boldsymbol{x}, \mathcal{R}) - \boldsymbol{z}_S(\boldsymbol{x}, \mathcal{R}')\|_2 ,$$
$$B(\boldsymbol{x}, \mathcal{R}, \mathcal{R}') = \|\boldsymbol{P}\boldsymbol{U}\|_2 \, \|\boldsymbol{a}(\boldsymbol{x})\|_2 \, \|\boldsymbol{g}(\mathcal{R}) - \boldsymbol{g}(\mathcal{R}')\|_\infty .$$

Here $\|\boldsymbol{PU}\|_2$ is the spectral norm of the actual centered write matrix, $\boldsymbol{a}(\boldsymbol{x}) = \boldsymbol{V}^\top \boldsymbol{d}(\boldsymbol{x})$, and the gains already include the modulation scale and attenuation in Equation (3).

Class-matched reference substitution. For each query wearer, $\mathcal{R}$ contains same-wearer references and $\mathcal{R}'$ contains references from another target wearer with identical ground-truth class counts. Labels specify this diagnostic composition and are not supplied to reference-state inference. HAPT uses 16 references per set, source-chosen classes 5, 4, and 3, and query wearers 2, 4, 10, 18, and 20, comprising 773 distinct queries. MotionSense uses 32 references per set, source-chosen classes 2, 4, and 5, and query wearers 20–24, comprising 2,176 distinct queries. The class-count patterns are $(6, 5, 5)$ and $(11, 11, 10)$, respectively, with remainder assignments following the fixed draw plan. The three classes are chosen using source-training sample counts. HAPT separates reference and query pools by original temporal blocks; MotionSense uses a fixed sample-ID hash partition with disjoint sample identities.

Draw seeds 1001–1005 sample without replacement within each reference class. Every query wearer is paired with each of the other four wearers in every draw, giving $5 \times 5 \times 4 = 100$ ordered reference-pair configurations per dataset. The same index plans are reused across five model seeds. References may recur across different draws or pairings. Across the two datasets, 200 configurations therefore give 1,000 model–reference-pair evaluations. Each query participates in 20 pairings per model seed, yielding 77,300 query–pair occurrences for HAPT and 217,600 for MotionSense: 294,900 occurrences from 2,949 distinct queries.

Activation bins and aggregation. For each dataset and model seed, the 20th, 40th, 60th, and 80th percentiles of $\|\boldsymbol{a}(\boldsymbol{x})\|_2$ on source-training queries define five activation bins. Within each bin and model seed, we take the arithmetic mean of O and B over query–pair occurrences. We then average the five seed-level bin means and report their sample standard deviation. Because every query participates in 20 pairings within a model seed, this equals first averaging over each query's pairings and then averaging over queries in the bin; no equal-user aggregation is applied. All evaluated changes satisfy their individual bounds. The largest observed-to-bound ratio is 0.99904, and Figure 7 reports its empirical distribution. The source-defined activation bins in Figure 5a also show larger mean changes at higher activation strength, connecting the query-dependent bound to the observed pattern of reference sensitivity.

Numerical checks cover reference-permutation invariance, empty-reference and zero-activation recovery, and consistency of cached and uncached computation. Repeating references by factors $\{1, 2, 4, 8\}$ preserves the precision upper bound in Equation (23).

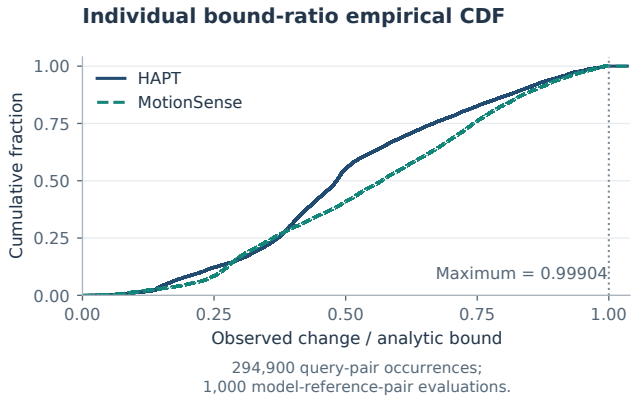


Figure 7: Observed-to-bound ratio distributions for HAPT and MotionSense. The analytic limit is one and the maximum observed ratio is 0.99904. The 200 ordered reference-pair configurations are evaluated across five model seeds, yielding 1,000 model–reference-pair evaluations and 294,900 query–pair occurrences from 2,949 distinct queries. Queries recur across reference comparisons.

C.11 ACTIVATION-DIRECTION INTERVENTIONS

Frozen-model intervention. We use the final epoch-120 checkpoints of five trained two-mode ($q = 2$) QSF models per task. Within each model, query logits $\boldsymbol{z}_D$, reference gains $\boldsymbol{g}_u$, read/write factors, and the bounded output map remain fixed across intervention angles.

Write $\boldsymbol{c}_0 = \boldsymbol{a}(\boldsymbol{x}) \odot \boldsymbol{g}_u$ for the native pre-write interaction. For the two-dimensional rotation matrix R_θ, set $\widetilde{\boldsymbol{c}}_\theta = (R_\theta \boldsymbol{a}(\boldsymbol{x})) \odot \boldsymbol{g}_u$ and define

$$\boldsymbol{c}_\theta^{\mathrm{rot}} = \begin{cases} \mathbf{0}, & \|\boldsymbol{c}_0\|_2 = 0, \\ \dfrac{\|\boldsymbol{c}_0\|_2}{\|\widetilde{\boldsymbol{c}}_\theta\|_2} \widetilde{\boldsymbol{c}}_\theta, & \|\boldsymbol{c}_0\|_2 > 0 \text{ and } \|\widetilde{\boldsymbol{c}}_\theta\|_2 > 0, \end{cases} \qquad \boldsymbol{c}_\theta^{\mathrm{rad}} = \cos(\theta)\boldsymbol{c}_0.$$

For a nonzero native interaction, the normalized rotation is defined only when $\|\widetilde{\boldsymbol{c}}_\theta\|_2 > 0$; this condition holds in the reported evaluations. On this domain it preserves the native interaction norm before the write map. For $\boldsymbol{c}_0 \neq \mathbf{0}$ and $|\theta| < 90°$, radial attenuation scales the magnitude by $\cos\theta$ without changing direction; at $|\theta| = 90°$ it removes the interaction and recovers DIRECT. The radial multiplier is explicitly set to zero at $|\theta| = 90°$. Both interventions use $\boldsymbol{z}_D + \mathcal{B}_{\tau_m}(\boldsymbol{U}\boldsymbol{c}_\theta)$ for prediction. Norm matching is applied before $\boldsymbol{U}$ and the bounded nonlinearity, so the final class-logit correction norm may change.

Scoring and aggregation. For each model seed, we score all target queries together at $|\theta| \in \{0, 15, 30, 60, 90\}°$. Positive and negative rotations are scored separately and their accuracies are averaged. Accuracy loss is the native accuracy minus this mean intervened accuracy, in percentage points. We summarize each loss curve by its trapezoidal area from 0° to 90° divided by 90°, denoted AUL. The paired excess is the rotation AUL minus the radial-control AUL within each seed. Means and sample SDs are taken across five model seeds, using the same queries at every angle.

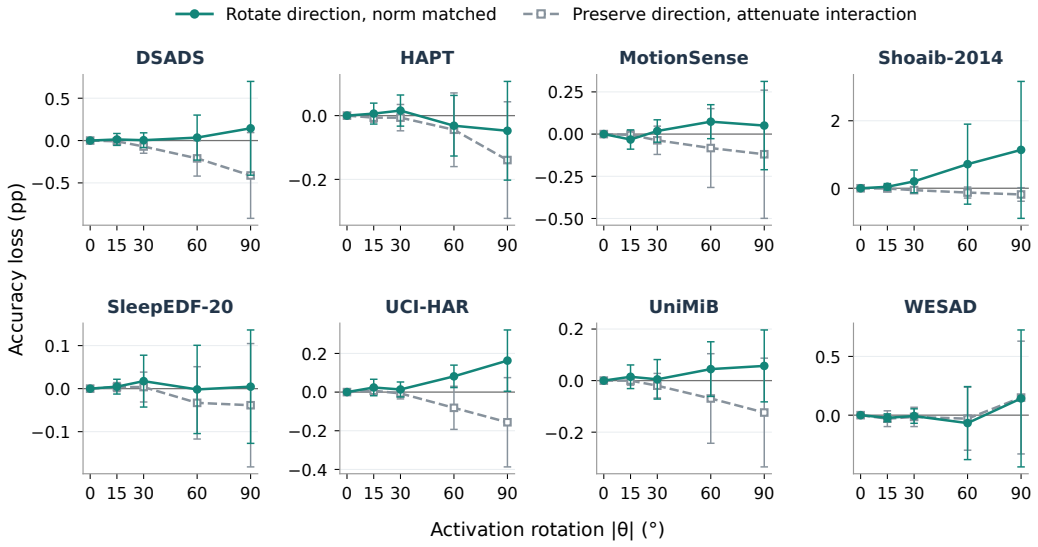


Figure 8: Activation-direction interventions on all eight tasks. Rotation matches the native pre-write interaction norm; radial attenuation preserves the native direction. Positive accuracy loss favors the native model. Points and error bars show five-seed means and sample SDs after separately scoring and averaging the two rotation signs. Vertical scales differ across datasets.

The mean paired excess is positive on seven of eight tasks (Figure 8, Table 21), with task-dependent magnitudes. DSADS and UCI-HAR have positive paired excess in all five seeds, while Shoaib-2014 has a larger mean effect and greater seed variability. On six tasks, the mean rotation AUL is positive while the mean radial AUL is negative. Rotation reallocates gain-weighted activation across learned write modes, whereas attenuation scales their native mixture uniformly. This suggests that the relative mode contributions carry task-relevant structure: reducing their strength can preserve this structure, while redirecting them can disrupt it at the same pre-write norm.

Table 21: Normalized area under the accuracy-loss curve (AUL), in percentage points. Values are five-seed mean ± sample SD. The last column is the within-seed difference between the rotation and radial AULs, summarized across seeds.

Dataset	Rotation	Radial attenuation	Paired excess
DSADS	$+0.039 \pm 0.194$	-0.158 ± 0.155	$+0.197 \pm 0.140$
HAPT	-0.013 ± 0.061	-0.041 ± 0.077	$+0.027 \pm 0.040$
MotionSense	$+0.032 \pm 0.093$	-0.057 ± 0.165	$+0.090 \pm 0.111$
Shoaib-2014	$+0.486 \pm 0.823$	-0.088 ± 0.105	$+0.574 \pm 0.899$
SleepEDF-20	$+0.005 \pm 0.071$	-0.016 ± 0.057	$+0.021 \pm 0.058$
UCI-HAR	$+0.062 \pm 0.048$	-0.054 ± 0.078	$+0.115 \pm 0.063$
UniMiB	$+0.028 \pm 0.080$	-0.049 ± 0.105	$+0.077 \pm 0.080$
WESAD	-0.004 ± 0.190	$+0.006 \pm 0.185$	-0.011 ± 0.027

C.12 Deployment measurements and amortization

Warm inference reuses the reference state and gains; the frozen upstream predictor runs for every query. Preparation includes candidate encoding, reference selection, and state/gain inference using 64 selected references. GPU measurements use an RTX 4090; CPU measurements use one pinned physical core of a Xeon Gold 6330 with one numerical thread. Models use seed 1. We apply 100 warm-ups and synchronize CUDA before and after timing. Light operations and cached CPU single-query calls (batch size 1) use 1,000 repetitions; candidate encoding, complete preparation, cold calls, and batch-512 calls use 100. Disk/model loading, source normalization, and initial device transfer are excluded; the GPU-to-CPU feature transfer for reference selection is included.

Table 3 gives single-query GPU latency and complete GPU preparation. Table 22 adds CPU preparation and cached single-query latency, and Table 23 reports throughput and memory. Fixed-input throughput uses batches of 512 copies of one eligible source query. QSF's GPU throughput differs from DIRECT by less than 0.3%. Storage includes model parameters, buffers, prototypes, dictionaries, and reusable reference tensors.

Table 22: CPU reference preparation and warm single-query timing on one pinned physical core. Preparation includes candidate encoding, selection, and state inference. Warm latency covers the complete query call; QSF reuses cached reference state and excludes preparation. Differences are QSF minus DIRECT, with percentages relative to DIRECT; both use independently timed medians before display rounding.

Task	Prep. (s)	DIRECT (ms)	QSF (ms)	Δ (ms)	Change (%)
HAPT	2.20	9.959	9.979	$+0.020$	$+0.20$
MotionSense	5.94	11.607	11.562	-0.045	-0.39
WESAD	15.98	60.749	61.224	$+0.474$	$+0.78$

Across these three CPU single-query measurements, QSF's median latency changes by -0.39% to $+0.78\%$ relative to DIRECT, with a maximum increase of approximately 0.474 ms. The reusable reference tensors occupy 93–237 bytes (Table 23), providing a compact cached representation of the context used by subsequent queries.

Table 23: Warm batch-512 throughput from median latency (windows/s), and storage. Each batch repeats one source query. Model MiB includes all parameters and buffers; Ref. B counts the saved reusable reference-object tensors. Peak GPU MiB is total allocated memory during the batch-512 call.

Dataset	Head	CPU win/s	GPU win/s	Model MiB	Ref. B	GPU peak
HAPT	DIRECT	166.4	39589.7	10.833	0	407.0
HAPT	QSF	168.9	39500.4	10.940	237	407.4
HAPT	Joint	168.0	39275.7	10.940	37	407.4
MotionSense	DIRECT	170.1	39312.4	10.809	0	409.3
MotionSense	QSF	167.1	39243.3	10.863	141	409.6
MotionSense	Joint	160.2	39201.2	10.863	37	409.6
WESAD	DIRECT	28.5	6258.9	10.422	0	2541.9
WESAD	QSF	26.5	6261.3	10.450	93	2542.2
WESAD	Joint	27.3	6258.1	10.450	37	2542.2

QSF CPU peak process RSS is 1,102.6/1,132.4/3,820.1 MiB on HAPT/MotionSense/WESAD. The separately encoded candidate caches contain 411,040/1,018,400/440,160 bytes; they are used during reference preparation and are not required by DIRECT. Memory depends on batch size and this measured implementation.

For Q queries sharing an eligible reference set, estimated per-query cost is $t_{\text{prep}}/Q + t_{\text{warm}}$, using the measured medians (Figure 9). Changing the reference set or its eligibility requires preparation again. On the profiled single-query and batch fixtures, cached and uncached computation preserves predicted classes; the maximum absolute logit difference is 4.77×10^{-7}.

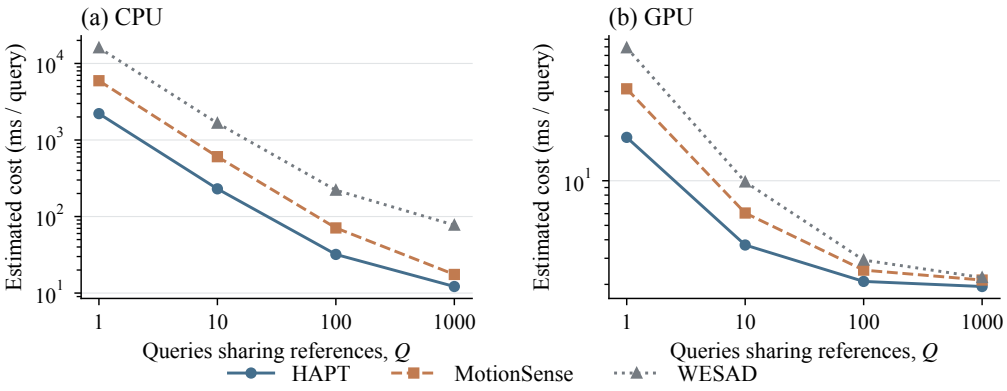


Figure 9: Estimated QSF amortization on CPU and GPU. Per-query cost is $t_{\text{prep}}/Q + t_{\text{warm}}$ for Q queries sharing one eligible reference set, using the measured preparation and warm-inference medians.

C.13 CROSS-BACKBONE TRAINING AND EVALUATION

We compare conditioning interfaces on InvConvNet (Germain et al., 2025) and DeepConvLSTM (Ordóñez & Roggen, 2016) using DSADS, MotionSense, and Shoaib-2014. For each backbone, dataset, and seed, we first train Population and then DIRECT. The resulting upstream pair is frozen and shared by QSF, compact Joint, and Logit. These three heads share source supervision, reference sets, target queries, and a reference budget of 64. Each training stage runs for 120 epochs; its checkpoint maximizes source-validation ACC, retaining the earliest epoch in a tie. Each backbone–dataset combination uses five seeds. ACC differences are computed within each seed and summarized as the mean and sample standard deviation across seeds. Section 4.4 and Table 2 report the results.

D UPSTREAM PREDICTOR CONTROL

Training recipe and parameter matching. We compare Population with the baseline CNN, the same CNN trained with the Population recipe, and a wider CNN matched to Population's parameter count. The four predictors use HAPT and UniMiB with three training seeds, batch size 64, training-only normalization, and a 120-epoch budget. Four source users per dataset provide validation, leaving 17 HAPT and 16 UniMiB users for fitting. Checkpoints maximize equal-user validation W-F1, with the earliest epoch breaking ties; the final epoch provides a second endpoint from the same runs. The base CNN uses Adam at 10^{-3}, weight decay 10^{-4}, validation-driven plateau scheduling, and noise, scaling, and masking. Recipe transfer applies the Population training settings in Appendix A.2, including AdamW, cosine decay, mixup, and label smoothing. The transferred CNNs and Population share sample order and augmentation streams. Parameter matching changes only the CNN's channel widths, to $(196, 392, 784)$ on HAPT and $(151, 302, 453)$ on UniMiB. Table 24 reports parameter counts and Conv/Linear MAC alongside recognition scores.

Table 24: Upstream predictor controls. W-F1 (%) is averaged equally across users; entries are mean $\pm$ sample SD over three seeds. Selected denotes the source-validation checkpoint. Parameters and Conv/Linear MAC are in millions. CNN + recipe and Wide CNN use the Population training recipe.

Dataset	Predictor	Params.	MAC	Selected W-F1	Epoch 120 W-F1
HAPT	CNN	0.395	16.10	92.96 ± 0.72	93.66 ± 0.35
	CNN + recipe	0.395	16.10	93.77 ± 1.56	95.11 ± 0.15
	Wide CNN	2.762	149.29	93.61 ± 2.07	94.92 ± 0.12
	Population	2.763	286.74	95.90 ± 0.63	96.33 ± 0.11
UniMiB	CNN	1.818	68.53	73.32 ± 0.13	73.99 ± 0.21
	CNN + recipe	1.818	68.53	77.66 ± 0.81	77.97 ± 0.70
	Wide CNN	2.485	95.29	76.58 ± 0.44	77.51 ± 0.13
	Population	2.470	170.31	81.80 ± 0.73	82.53 ± 0.24
Two-task mean	CNN	—	—	83.14 ± 0.40	83.82 ± 0.28
	CNN + recipe	—	—	85.71 ± 1.12	86.54 ± 0.39
	Wide CNN	—	—	85.10 ± 1.25	86.22 ± 0.03
	Population	—	—	88.85 ± 0.68	89.43 ± 0.09

Population achieves higher mean W-F1 than the parameter-matched CNN on both tasks at both the source-validation-selected and final epoch-120 endpoints. Transferring the Population training recipe also improves the base CNN on both tasks. These controls identify contributions from the training recipe and the upstream architecture to recognition performance; Table 24 reports their corresponding Conv/Linear computation alongside parameter counts.