# SATTC: Structure-Aware Label-Free Test-Time Calibration for Cross-Subject EEG-to-Image Retrieval

Supplementary Material

## S1. Dataset and Protocol Details

We briefly summarize the THINGS-EEG dataset, the preprocessing pipeline, and the leave-one-subject-out (LOSO) protocol with dev packs used for hyper-parameter selection.

### S1.1. THINGS-EEG Overview

We use the public THINGS-EEG dataset under the cross-subject setup. Ten subjects participated in a rapid serial visual presentation (RSVP) task across four sessions. Stimuli are natural images from the THINGS database, each depicting a single object.

Following the official split, 1,654 object classes (10 images each) are used for training and 200 disjoint classes for zero-shot testing. This yields 16,540 training image conditions ($1{,}654 \times 10$) repeated four times and 200 test conditions repeated 80 times. EEG is recorded with 64 channels at 1,000 Hz; we retain 63 EEG channels and represent each trial as a $63 \times 250$ tensor (channels $\times$ time) after preprocessing.

### S1.2. EEG Preprocessing

We follow the official ATM preprocessing code and make implementation choices explicit for reproducibility.

**Raw data and events.** For each subject and session we load the raw 1,000 Hz, 64-channel arrays into MNE, keep 63 EEG channels (Fp1–O2), and use the trigger channel to extract events. Trials with event code `99999` are discarded.

**Epochs, baseline, resampling.** Continuous EEG is epoched from $-0.2$ s to $1.0$ s around stimulus onset. We apply baseline correction using $[-0.2, 0]$ s, then down-sample to 250 Hz. After resampling we keep the 0–1.0 s window, yielding $\approx 250$ time samples per trial ($63 \times 250$).

**Conditions and repetitions.** Event codes serve as image-condition indices. Epochs are grouped by condition. To obtain balanced tensors we cap repetitions per condition: at most 20 for test and at most 2 for train, per session. Each session thus yields arrays of shape

$$\text{conditions} \times \text{repetitions} \times 63 \times T,$$

with $T \approx 250$.

**Multivariate noise normalization.** We apply multivariate noise normalization (MVNN) independently to each session. For the training partition we compute condition-wise covariance matrices of epoched EEG, average them to obtain $\Sigma_{\text{train}}$, and derive a whitening transform $\Sigma_{\text{train}}^{-1/2}$. The same transform is applied to both train and test trials of that session; no test-only statistics are used.

Table S1. Summary of the THINGS-EEG dataset and the zero-shot split used in our experiments.

| | Train | Test |
|---|---|---|
| # object classes | 1,654 | 200 |
| Images per class | 10 | 10 |
| # subjects | 10 | |
| EEG channels | 63 | |
| Time samples per trial | 250 | |

**Merging sessions.** Whitened trials from all four sessions are merged. For test, we concatenate sessions along the repetition axis and shuffle repetitions within each image condition. For train, we regroup trials by condition across sessions and shuffle repetitions. The final EEG representation has shape

$$\text{images} \times \text{repetitions} \times 63 \times T,$$

with $T \approx 250$, shared by all EEG encoders (ATM, EEGNet, EEGConformer, ShallowFBCSPNet).

### S1.3. LOSO and Dev Packs

**LOSO folds.** We use a strict LOSO protocol. In each of 10 folds, one subject $s_{\text{test}}$ is held out for evaluation. All trials from $s_{\text{test}}$ are excluded from training and dev packs. The across-subject encoder is trained on the remaining 9 subjects using only the 1,654 training classes.

**Subject distances and dev subjects.** Among the 9 training subjects we build a $9 \times 9$ distance matrix using training EEG only. For each subject we average all training trials ($63 \times T$), flatten the mean, and compute pairwise Euclidean distances between means. Subjects are ranked by mean distance to the others. We select three dev subjects: "easy" (smallest mean distance), "hard" (largest), and "medium" (closest to the median).

**Dev packs and tuning.** For each dev subject we sample a balanced dev pack of 200 object classes from the 1,654 training classes, disjoint from the 200 test-only classes. Dev packs are used only for hyper-parameter and epoch selection (e.g., CSLS settings, PoE weights). Statistics or labels from the held-out test subject are never used. Once tuned, all hyper-parameters are fixed and reused across all LOSO folds and across all EEG encoders. Test labels are used strictly for final evaluation.

## S2. Revisiting the ATM Baseline and Standardized Inference

SATTC builds on the recent ATM framework for EEG-to-image retrieval. Our goal is to use ATM as a fair and reusable baseline rather than to weaken it. We therefore (i) clarify the original retrieval protocol, (ii) standardize the similarity computation, and (iii) remove subject-conditioning "spoof" labels that leak supervision in the across-subject setting.

### S2.1. Original ATM Retrieval vs. Our Re-Implementation

ATM trains an EEG encoder and an image encoder with a contrastive objective and performs retrieval by comparing EEG and image embeddings. In the official code, the similarity between an EEG query $q$ and an image candidate $c$ is computed as an unnormalized dot product

$$S_{\text{ATM}}(q, c) = z_q^{\text{EEG}} \cdot z_c^{\text{img}},$$

where $z_q^{\text{EEG}}$ and $z_c^{\text{img}}$ are the final encoder embeddings.

This choice makes retrieval scores sensitive to embedding norms and arbitrary global rescalings of each encoder: changing the temperature, regularization, or projection head can multiply embeddings by a constant without changing their geometry, yet directly affect rankings. This complicates cross-encoder comparison and standardized benchmarking.

We re-implemented the official ATM retrieval code and verified that we can closely match the reported THINGS-EEG results. All comparisons in the main paper use this faithful re-implementation as the "ATM (original retrieval)" baseline.

### S2.2. Cosine + $\ell_2$ and Candidate Whitening as a Standardized Baseline

To obtain a more principled and comparable evaluation protocol, we replace the unnormalized dot product with cosine similarity between $\ell_2$-normalized embeddings:

$$S_{\cos}(q, c) = \frac{z_q^{\text{EEG}} \cdot z_c^{\text{img}}}{\|z_q^{\text{EEG}}\|_2 \, \|z_c^{\text{img}}\|_2}.$$

Cosine similarity is standard in cross-modal retrieval and removes dependence on encoder-specific norms, yielding a better-conditioned similarity matrix for subsequent whitening and hubness-mitigation steps.

On top of cosine + $\ell_2$ normalization, we apply a simple candidate whitening (CW) transform to image embeddings:

$$\tilde{z}_c^{\text{img}} = W_{\text{CW}}\big(z_c^{\text{img}} - \mu_{\text{img}}\big),$$

where $\mu_{\text{img}}$ and $W_{\text{CW}}$ are estimated once from training image embeddings. CW removes global correlations on the candidate side and helps align the scale of rows (queries) and columns (classes) in the similarity matrix before any hubness-aware calibration such as CSLS.

We refer to the combination of cosine similarities, $\ell_2$-normalized embeddings, and CW as our standardized baseline. As shown in Table 1 of the main paper (first two rows), this standardized retrieval pipeline already improves the original ATM baseline on THINGS-EEG under LOSO: Top-5 accuracy increases from 20.0% to 30.5%, and Top-1 accuracy from 5.5% to 9.2%, even before introducing subject-adaptive whitening (SAW) or any test-time calibration. All SATTC variants reported in the main paper are built on top of this standardized baseline.

### S2.3. Fixing Subject-Conditioning "Spoof" Labels with an Unknown Token

A less obvious issue in the original ATM protocol is the use of subject-dependent "spoof" labels that condition the EEG encoder during both training and test, even in the across-subject configuration.

In the released training script, each batch is associated with a subject index parsed from the folder name (e.g., `sub-06` $\rightarrow$ 6). A vector of this integer ID is passed to the EEG model for every trial in the batch. Internally, a learnable subject embedding is looked up from an index table and added to the EEG token embeddings before the Transformer blocks. As a result, EEG representations are explicitly conditioned on subject-specific tokens for all trials, while evaluation is described as "unsupervised" and "zero-shot" with respect to object classes. In the across-subject setting, the branch that would fall back to an unknown ID is commented out, so real subject IDs are still used.

To remove this source of label leakage while keeping the architecture unchanged, we replace subject-specific embeddings by a single learnable "unknown" token $e_{\text{unk}}$ shared by all subjects:

- **Training.** Every EEG trial is augmented with the same token $e_{\text{unk}}$, so the encoder cannot rely on explicit subject identity.
- **Test.** We use the exact same token $e_{\text{unk}}$ for all trials; no subject IDs or other supervised cues are injected at test time.

All ATM, standardized-baseline, and SATTC results reported in the main paper use this strictly label-free variant of the subject-conditioning mechanism. This ensures that improvements come from standardized inference and SATTC calibration rather than from exploiting subject labels at test time.

## S3. Subject-Adaptive Whitening (SAW): Additional Analysis

The main paper (Sec. 3.2) introduces subject-adaptive whitening (SAW) as a per-subject centering and whiten-

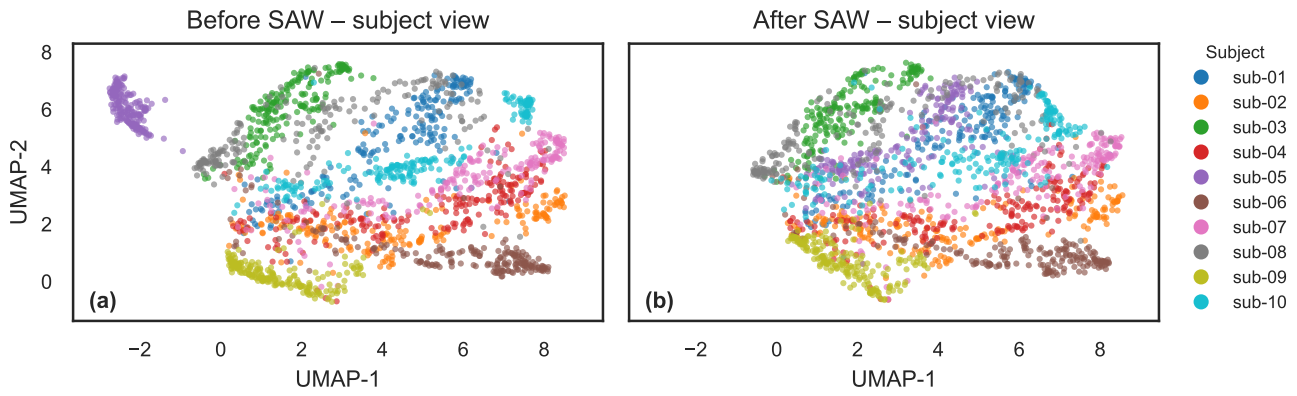


Figure S1. Subject-view UMAP of EEG embeddings before and after subject-adaptive whitening (SAW). Each point is a trial and colors denote subject identity. (a) Before SAW, embeddings form well-separated subject-specific clusters, revealing strong second-order subject bias. (b) After SAW, subject clusters collapse into a single manifold with much weaker subject separation while preserving the overall geometry, indicating that SAW maps all subjects into a shared second-order domain.

ing step on EEG embeddings before similarity computation. Here we provide additional evidence that SAW (i) mitigates cross-subject second-order bias in the EEG space and (ii) is more effective than simpler whitening schemes. Per- subject retrieval gains are already reported in Fig. 2(a) of the main paper.

## S3.1. Subject-Level Geometry Before vs. After SAW

We first inspect how SAW changes the geometry across subjects. For a given encoder, let $z_{s,i}^{\mathrm{EEG}}$ denote the EEG embedding of subject $s$ at trial $i$. Before SAW, different subjects exhibit noticeably different means and covariance structures; SAW re-centers and whitens these embeddings per subject using only unlabeled EEG from that subject (see Sec. 3.2 in the main paper).

**Subject-view UMAP.** We apply UMAP to EEG embeddings from all training subjects and color-code points by subject identity. Fig. S1 shows two views: before SAW, embeddings form multiple well-separated clusters, each corresponding to a different subject; after SAW, these clusters collapse into a single manifold with much weaker subject separation while retaining a coherent global structure. This is consistent with the interpretation that SAW maps all subjects into a shared second-order domain while preserving the overall representation geometry.

**Covariance statistics.** We also estimate per-subject covariance matrices over training embeddings and summarize their magnitudes (e.g., Frobenius norms or eigenvalue spread). After SAW, these statistics become more concentrated across subjects, indicating that second-order subject bias is substantially reduced (results omitted for brevity).

Combined with the per-subject Top-5/Top-1 improvements in Fig. 2(a), these geometric observations support SAW as a robust cross-subject normalization step rather than a subject-specific overfitting mechanism.

## S3.2. Comparison to Alternative Whitening Strategies

We do not claim that whitening itself is new in EEG processing. Rather, the contribution of SAW is to instantiate *per-subject second-order alignment* in the embedding space as a label-free test-time calibration module specifically designed for cross-subject EEG-to-image retrieval, and to demonstrate that this particular instantiation is substantially more effective than simpler global or diagonal normalization strategies in this setting.

We next compare SAW with simpler whitening strategies, all built on top of the same standardized baseline (cosine + $\ell_2$ + CW):

- **None (Std.+CW).** No additional whitening on the EEG side.
- **Global whitening.** A single covariance matrix is estimated by pooling EEG embeddings from all training subjects, and the same whitening transform is applied to every subject at test time.
- **BN-style diagonal scaling.** We estimate per- dimension means and variances across all training embeddings and apply diagonal rescaling (batch- normalization style), without modeling cross-channel correlations.
- **Subject-adaptive whitening (SAW).** The per- subject whitening defined in Sec.3.2 of the main paper, using each subject's own covariance.

Table S2 reports Top-5 and Top-1 accuracy under the LOSO protocol on THINGS-EEG for these variants.

Global whitening mildly alleviates hubness but does not fully resolve cross-subject mismatch, and BN-style diagonal scaling yields only small gains. In contrast, SAW —

Table S2. Whitening strategies on THINGS-EEG (LOSO, zero-shot retrieval). All methods extend the same standardized baseline (cosine + $\ell_2$ + CW). SAW consistently yields the best Top-5 and Top-1 accuracy.

| Whitening strategy | Top-5 (%) | Top-1 (%) |
|---|---|---|
| None (Std.+CW) | 30.5 | 9.2 |
| Global whitening | 23.0 | 6.5 |
| BN-style diagonal scaling | 31.4 | 9.7 |
| Subject-adaptive whitening (SAW) | 36.4 | 13.7 |

which models each subject's full covariance and whitens in that subject's own coordinate system — consistently achieves the best Top-5 and Top-1 accuracy and provides a cleaner geometry for the label-free calibration modules in SATTC.

## S4. Practical Deployment and Label-Free Test-Time Calibration

This section complements Secs. 3.3–3.5 of the main paper by (i) clarifying the two-phase deployment protocol and calibration data requirements of SATTC, (ii) quantifying how many unlabeled trials from a new subject are needed for effective adaptation, (iii) detailing implementation choices for adaptive CSLS and the structural PoE head, and (iv) illustrating how CSLS reshapes the similarity space to reduce hubness.

### S4.1. Two-Phase Deployment Protocol and Calibration Requirement

We explicitly acknowledge that SATTC is not a strict online single-trial adaptation method. In practice, SATTC follows a **two-phase deployment protocol**.

**Phase 1 (calibration).** A small unlabeled calibration set $D_{\text{calib}}$ from the held-out subject is used to estimate the statistics required by subject-adaptive whitening (SAW) and adaptive CSLS. Concretely, SAW requires an estimate of the subject's EEG embedding mean and covariance; adaptive CSLS requires local query density estimates derived from the resulting score matrix. Critically, *neither operation requires class labels or any trial-level ground truth*: only raw, unlabeled EEG signals from the held-out subject are used, and all calibration operations remain strictly label-free.

**Phase 2 (retrieval).** All calibration quantities (SAW whitening matrix, CSLS neighbourhood statistics) are frozen, and retrieval is performed independently for each incoming EEG trial as a simple matrix-vector query against the calibrated, frozen similarity structure. Once Phase 1 is complete, Phase 2 introduces no further dependence on the held-out subject's data.

Table S3. SATTC (ATM encoder, LOSO, THINGS-EEG) Top-5 retrieval accuracy as a function of unlabeled calibration window size $N$. For $N < 200$, $D_{\text{calib}}$ and $D_{\text{eval}}$ are disjoint splits of the held-out subject's test trials, and SAW/adaptive CSLS statistics are estimated only from $D_{\text{calib}}$. The "Full (200)" row is the standard LOSO reference used for comparison. "% of Full" $=$ Top-5$(N)$ / Top-5$(200)$. Results are averaged over LOSO folds; for $N < 200$, we additionally average over three random splits.

| Calibration size $N$ | Top-5 (%) | % of Full |
|---|---|---|
| 5 | 26.6 | 69.4 |
| 10 | 30.5 | 79.3 |
| 20 | 34.1 | 88.7 |
| 50 | 36.4 | 94.8 |
| 100 | 38.3 | 99.7 |
| Full (200) | 38.4 | 100.0 |

We therefore position SATTC as a practical *label-free test-time calibration method* rather than a single-trial covariance-estimation method. The practical question is how many unlabeled trials from the held-out subject are necessary in Phase 1 for SAW to produce a useful whitening transform. Sec. S4.2 below quantifies this requirement.

### S4.2. Disjoint $N$-Shot Calibration Windows

To determine how much unlabeled calibration data SATTC requires, we evaluate performance under disjoint $N$-shot calibration windows. For each LOSO fold and each $N \in \{5, 10, 20, 50, 100\}$, the held-out subject's test trials are randomly partitioned into a calibration subset $D_{\text{calib}}$ of size $N$ and a disjoint evaluation subset $D_{\text{eval}}$. SAW and adaptive CSLS statistics are estimated exclusively from $D_{\text{calib}}$, and retrieval accuracy is measured on $D_{\text{eval}}$. The "Full (200)" result is reported separately as the standard LOSO reference used for comparison. For $N < 200$, results are averaged over three random calibration/evaluation splits and all 10 LOSO folds.

Table S3 summarises relative and absolute Top-5 accuracy at each calibration window size ("% of Full" is defined as Top-5$(N)$ / Top-5$(N{=}200)$). Performance already recovers a substantial fraction of the full-calibration result at small $N$: with only $N{=}50$ unlabeled trials, SATTC achieves 94.8% of the full-batch Top-5, and the gap closes to within 0.3% at $N{=}100$. The calibration cost at $N{=}50$ amounts to a few minutes of passive recording, well within standard BCI session budgets. These results confirm that SATTC is practically deployable even when only a modest number of unlabeled EEG trials from a new subject are available before deployment.

### S4.3. Fixed-$k$ CSLS vs. Adaptive CSLS

Standard CSLS and our adaptive CSLS share the same functional form (Eq. (11)), but differ in how the neighborhood sizes are chosen. In the fixed-$k$ variant we use a single global value $k$ tuned once on dev subjects. In the adaptive variant, query- and class-specific neighborhood sizes $k_{\text{row}}(q)$ and $k_{\text{col}}(c)$ are obtained from local densities on $S_{\text{new}}$ via the monotonic mapping described in Eqs. (13)–(18). In practice, the adaptive neighborhood sizes remain concentrated in a narrow band around the fixed-$k$ choice, so Ada-CSLS behaves as a density-aware refinement of the fixed-$k$ baseline rather than a completely different scoring rule.

Empirically, Table 1 in the main paper shows that fixed-$k$ CSLS and Ada-CSLS achieve very similar Top-1 and Top-5 accuracy, with Ada-CSLS trading a small decrease in Top-1 for a slightly higher Top-5. The benefit of adaptivity is more visible in their hubness profiles: as seen from the class-occurrence curves $N_K(c)$ in Fig. 2(b), adaptive CSLS further flattens the popularity distribution and reduces the dominance of a few hub classes compared to fixed-$k$ CSLS, leading to a more balanced use of prototypes across ranks. This supports our choice of Ada-CSLS as the geometric expert in SATTC: it preserves retrieval accuracy while providing a more flexible and effective hubness mitigation than a single global $k$.

### S4.4. Structural Expert from Pre-CSLS Evidence

The structural expert $S_{\text{struct}}$ is built exclusively from the pre-CSLS similarity matrix $S_{\text{new}}$ and rank statistics, re-using the notation and definitions of Sec. 3.4 (Eqs. (19)–(23)). For each LOSO test subject, we compute class-popularity counts $N_K(c)$ and hubness scores $h(c) \in [0, 1]$ together with strict mutual nearest neighbours (MNN@1) and bi-directional top-$L$ pairs on unlabeled test EEG. These quantities are converted into a sparse correction matrix $S_{\text{struct}}$ that

- adds a positive bonus $\lambda_{\text{anchor}}$ to high-confidence anchor pairs (MNN@1 or symmetric top-$L$), - applies a negative penalty $-\lambda_{\text{pen}} h(c)$ to hub-like entries that are popular but not supported by mutual neighbourhood evidence, and - leaves all remaining entries at zero.

Penalties are masked for anchor pairs, so structurally consistent matches are only boosted, never suppressed. Before fusion, $S_{\text{struct}}$ is standardized per test subject to have approximately zero mean and unit variance. All decisions in this construction depend solely on $S_{\text{new}}$; neither labels nor $S_{\text{geom}}$ are used, and $S_{\text{new}}$ itself is never modified. In this way, $S_{\text{struct}}$ acts as a simple structure-aware prior that locks reliable mutual neighbours while gently down-weighting ubiquitous hubs.

### S4.5. Product-of-Experts Fusion and Case-Wise Behavior

The final calibrated scores combine the geometric and structural experts via the PoE fusion of Sec. 3.5: we interpret $S_{\text{geom}}$ (Ada-CSLS on top of SAW+CW) and $S_{\text{struct}}$ as unnormalized logits of two experts and form a weighted sum

$$S_{\text{final}}(q, c) = S_{\text{geom}}(q, c) + \beta\, S_{\text{struct}}(q, c),$$

corresponding to the log-density of a product distribution $p_{\text{geom}}(c \mid q)\, p_{\text{struct}}(c \mid q)^{\beta}$. Both experts are standardized per subject before fusion, and $S_{\text{struct}}$ is computed once from $S_{\text{new}}$ and held fixed, so the operator $T_\beta(S_{\text{new}})$ acts as a single-shot regularizer rather than an iterative re-ranking procedure.

Qualitatively, we observe that many tail classes whose true category lies at medium ranks under Ada-CSLS move into the top positions once the structural expert is added, reflecting the positive bias on consistent mutual neighbours. Conversely, obvious hub classes with very high hubness scores $h(c)$ often experience modest downward shifts in rank. These effects align with the flatter $N_K(c)$ curves obtained under SATTC in Fig. 2(b) and support the interpretation of the PoE fusion as a principled, structure-aware shortlist refiner.

### S4.6. Hyper-Parameter Sensitivity

We summarize the sensitivity of SATTC to its main hyper-parameters: the PoE weight $\beta$, the consistency threshold $L$ for bi-directional top-$L$ pairs, the popularity horizon $K$ in $N_K(c)$, and the anchor/penalty strengths $\lambda_{\text{anchor}}$ and $\lambda_{\text{pen}}$. All hyper-parameters are tuned once on a fixed dev pack constructed from training subjects and the resulting configuration is reused unchanged across all LOSO folds and across all EEG encoders.

To provide concrete evidence of robustness, Table S4 reports Top-5 accuracy under a one-dimensional sweep of the PoE weight $\beta$ around the default configuration ($\beta^\star$=1.9), with all other hyper-parameters held fixed. The results show a broad flat region: Top-5 stays within 0.8 pp across the range $\beta \in [0.5, 2.5]$, and even $\beta$=0 (Ada-CSLS only, no structural expert) achieves 38.03%, confirming that PoE provides a moderate but consistent lift rather than being the primary driver. Graceful degradation only occurs at more extreme values. A similar plateau is observed when sweeping $L$ independently.

SATTC is similarly robust to the choice of $K$ and to moderate changes in $\lambda_{\text{anchor}}$ and $\lambda_{\text{pen}}$. Taken together, these observations confirm that our label-free calibration head does not rely on fragile fine-tuning: a single configuration selected once on dev subjects generalizes well across all LOSO folds and all EEG encoders considered in the main paper. CSLS neighbourhood size $k$=12 and SAW shrinkage

Table S4. Sensitivity of SATTC to the PoE fusion weight $\beta$ (ATM encoder, LOSO, THINGS-EEG). All other hyper-parameters are fixed at their dev-tuned values. Top-5 accuracy remains stable over a wide range of $\beta$; the default is marked with $\star$.

| $\beta$ | Top-5 (%) |
|---|---|
| 0.0 (Ada-CSLS only) | 38.03 |
| 0.5 | 38.33 |
| 1.0 | 38.77 |
| $1.9^{\star}$ (default) | 38.67 |
| 2.5 | 37.97 |
| 4.0 | 38.31 |

coefficient $0.8$ are inherited defaults and require no target-domain tuning whatsoever.

### S4.7. Pre-/Post-CSLS Similarity-Space Visualization

To complement the subject-view UMAP in Fig. S1 (which visualizes the effect of SAW on EEG embeddings), we provide a candidate-view visualization that directly shows how adaptive CSLS reshapes the similarity space and mitigates hubness.

**Visualization design.** Because CSLS does not modify embedding vectors—it only changes the similarity scoring rule—we build the visualization in *score space* rather than embedding space. Each image candidate $c$ is represented by its similarity profile across all EEG queries (i.e., one row of the transposed score matrix). We then apply UMAP to these candidate representations, coloring each point by its hubness count $N_K(c)$ (number of queries for which $c$ appears in the top-$K$ shortlist).

**Interpretation.** Fig. S2 shows the candidate-view score-space UMAP before and after adaptive CSLS. Pre-CSLS, candidates with very high hubness counts (bright/yellow color) are more concentrated and form visible high-hubness hotspots in the score space, consistent with the tendency of a few popular classes to be repeatedly selected near the top of many queries. Post-CSLS, the spatial distribution of high-hubness candidates becomes less concentrated, and their $N_K(c)$ values decrease substantially, while the overall manifold structure is preserved. Quantitatively, adaptive CSLS reduces the maximum hubness count from $425$ to $216$, decreases the Gini coefficient of $N_5(c)$ from $0.615$ to $0.474$, and lowers the share of the top-1% most popular candidates from $7.8\%$ to $4.0\%$. Since the total number of top-5 assignments is fixed at $Q \times 5$, these changes reflect a redistribution of candidate popularity rather than a change in the total number of shortlist occurrences. This visualization provides a candidate-view complement to the flatter $N_K(c)$ popularity curves reported in Fig. 2(b) of the main paper.

## S5. Extended Hubness and Per-Class Fairness Analyses

The main paper analyzes hubness and shortlist quality using aggregated $N_K(c)$ popularity curves, $\Delta$Recall@$K$, and per-class Recall@5 (Fig. 2). This section provides compact distribution-level statistics that support the same conclusions from complementary angles, while avoiding additional figures and large tables.

### S5.1 Hubness Statistics

To quantify hubness beyond a single $N_K(c)$ curve, we also measure the distribution of per-class hit counts, i.e., the number of queries for which each class appears in the top-$K$ shortlist. For each method and $K \in \{5, 10\}$ we compute the skewness and kurtosis of this hit-count distribution on THINGS-EEG, averaged across LOSO folds.

Across all settings we consistently observe that applying SAW + Ada-CSLS reduces both skewness and kurtosis compared with the standardized baseline (Std.+CW), indicating fewer extreme hub classes. Adding the structural PoE head (SATTC) further lowers these measures for both $K$, meaning that shortlist mass is spread more evenly across classes and that very popular hubs are suppressed. These distribution-level observations mirror the flatter $N_K(c)$ curves obtained under SAW + Ada-CSLS and SATTC in Fig. 2(b), and confirm that our calibration head provides systematic hubness reduction rather than improvements driven by a handful of favorable classes.

### S5.2 Per-Class Recall Distribution and Tail Classes

We next analyze per-class fairness more directly via the distribution of Recall@5 across classes. For each method, we consider the histogram of per-class Recall@5 values and track how much mass lies in low-, medium-, and high-recall bins.

Across LOSO folds, SATTC consistently shifts probability mass away from the lowest-recall bins toward higher-recall bins: the fraction of classes with near-zero Recall@5 decreases, while the fraction of classes with medium and high recall increases. This suggests that SATTC not only improves overall accuracy but also makes performance more uniform across classes.

To highlight the effect on difficult categories, we define *tail classes* as the bottom-$x\%$ classes according to baseline (Std.+CW) Recall@5. We then track their recall under SAW + Ada-CSLS and under SATTC. On average, SATTC provides the largest relative gains for these tail classes, while keeping performance on head classes competitive. This reinforces our claim that structure-aware calibration mainly helps challenging classes instead of over-optimizing already easy ones.

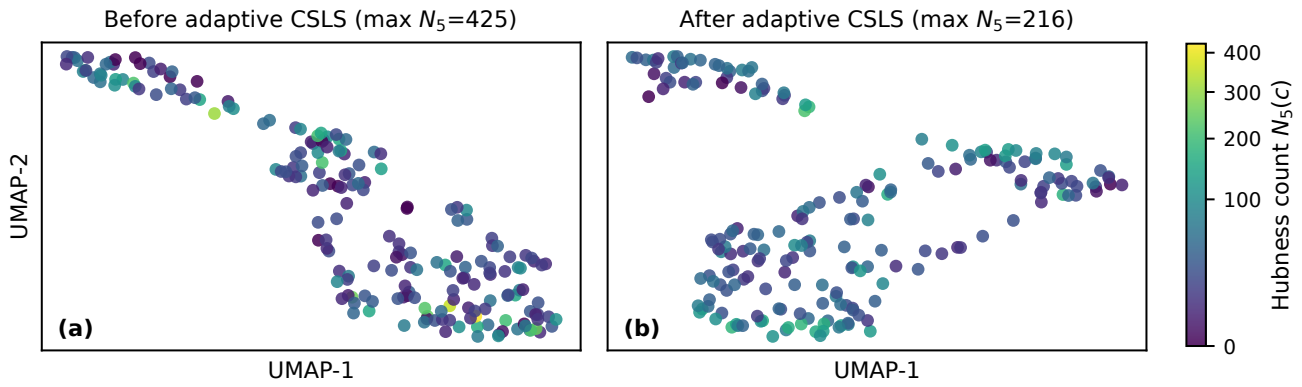


Figure S2. Candidate-view score-space UMAP before (a) and after (b) adaptive CSLS. Each point represents one image candidate, positioned by its similarity profile across all EEG queries. Point color indicates hubness count $N_5(c)$ at $K$=5 (yellow: high hubness; purple: low hubness; $\sqrt{\cdot}$-scaled colormap). Adaptive CSLS weakens and disperses high-hubness hotspots ($\max N_5(c)$: 425 $\rightarrow$ 216) while preserving the overall score-space organization, consistent with a redistribution of candidate popularity rather than a destructive restructuring of the score space.

### S5.3 Per-Encoder Hubness Behaviour

Table 2 in the main paper shows that SATTC consistently improves retrieval accuracy across multiple EEG encoders (ATM, EEGNet, EEGConformer, ShallowFBCSPNet). We briefly examine how hubness behaves for each encoder before and after calibration.

Repeating the $N_K(c)$ and per-class Recall@5 analyses encoder-wise yields a consistent pattern: for all encoders, SAW + Ada-CSLS already flattens the extreme tail of highly popular classes compared with the standardized baseline, and adding the structural PoE head further straightens the popularity curves and improves per-class Recall@5. Lower-capacity encoders (e.g., shallow EEGNet) exhibit stronger hubness under the baseline, but also benefit more from SATTC in terms of $N_K(c)$ flattening and tail-class recall gains.

Taken together, these encoder-wise analyses show that our calibration head provides systematic hubness reduction and per-class fairness improvements across encoders and LOSO folds, rather than relying on a single aggregated curve or a specific backbone.

## S6. Additional Quantitative and Qualitative Results

This section complements the main quantitative results with more fine-grained per-subject and per-encoder trends, and with qualitative retrieval examples that illustrate both the strengths and the remaining limitations of SATTC. We note that a within-subject reference belongs to a different supervision regime, as it requires subject-specific labeled data for training. Since SATTC is designed for label-free cross-subject test-time calibration, we focus the analyses below on deployment-relevant calibration behavior, hubness reduction, and encoder-level robustness rather than adding a separately supervised within-subject reference. Unless otherwise stated, all metrics follow the same LOSO evaluation protocol and three-seed averaging as in the main paper.

### S6.1. Per-Subject and Per-Encoder Results

We briefly summarize LOSO retrieval accuracy for each held-out subject and for each EEG encoder, focusing on four representative methods: the standardized baseline (*cosine* + $\ell_2$ + *CW*), baseline + SAW, baseline + SAW + Ada-CSLS, and SATTC (SAW + Ada-CSLS + PoE). Instead of listing all per-subject and per-encoder numbers in a large table, we refer the reader to our released code and logs, and highlight the main trends below.

- Adding SAW on top of the standardized baseline yields substantial Top-5 and Top-1 gains for almost all subjects, confirming that subject-adaptive whitening is the main driver of cross-subject improvements.
- Ada-CSLS further improves or at least maintains performance for most subjects while mitigating hubness (cf. Sec. 3.3 of the main paper), providing a better accuracy–fairness trade-off than fixed-$k$ CSLS.
- SATTC provides the best or near-best performance for the majority of subjects across all encoders considered (ATM, EEGNet, EEGConformer, ShallowFBCSPNet). This supports our claim that the structure-aware calibration head is plug-and-play and largely decoupled from the choice of EEG encoder.

Taken together, these per-subject and per-encoder trends show that the gains reported in Tables 1 and Tables 2 of the main paper are not driven by a handful of favorable cases, but reflect consistent improvements across subjects

with different noise levels and across encoders with different capacities.

### S6.2. Qualitative Retrieval Examples and Failure Cases

Fig. S3 shows four representative EEG queries. For each query, we visualize the Top-5 retrieved image classes for the standardized baseline (Std. + CW, left) and for SATTC (right); the true class is highlighted with a green frame whenever it appears in the shortlist.

The first two rows illustrate successful cases. In both examples, the baseline already retrieves visually related items but either misses the true class or places it at a non-top-1 position, while several hub classes (e.g., "jukebox", "jelly bean") appear repeatedly across queries. Under SATTC, the true class is moved to the top-1 or top-2 position and the most prominent hubs are slightly pushed down the ranking, yielding a cleaner and more semantically plausible shortlist.

The last two rows show challenging failure cases. In the third row, the baseline correctly retrieves the true class within the Top-5, but SATTC replaces it with other semantically reasonable candidates (e.g., other animals) and the true class drops out of the shortlist. In the fourth row, both methods struggle: visually similar but incorrect classes dominate the Top-5 and SATTC only makes small local adjustments. These cases are consistent with the limitations discussed in the main paper: some subjects exhibit extremely noisy EEG, some object classes are visually and semantically very similar to others in the test set, and in a few cases the encoder embeddings are not sufficiently separable to support fine-grained retrieval.

Overall, this compact figure illustrates SATTC's intended behavior as a shortlist refiner: it often promotes the correct class and down-weights ubiquitous hubs, while its remaining errors tend to be "reasonable mistakes" among semantically related categories.

## S7. Complexity and Practicality of the Test-Time Head

Although SATTC introduces several test-time modules (SAW, adaptive CSLS, and the structural PoE head), all of them operate on top of the already computed similarity matrix and therefore add only modest computational and memory overhead compared with the baseline retrieval pipeline. Below we summarize the complexity and practicality of the calibration head.

### S7.1. Asymptotic Complexity

Let $|Q|$ denote the number of EEG queries for a given test subject and $|C|$ the number of image candidates. Computing the baseline cosine similarity matrix $S_{\cos} \in \mathbb{R}^{|Q|\times|C|}$ has time complexity $\mathcal{O}(|Q||C|d)$ for embedding dimension $d$ and dominates the overall cost.

On top of this, the SATTC head performs:

- **SAW.** Applying the per-subject whitening transform to EEG embeddings is a single matrix multiplication per query with cost $\mathcal{O}(|Q|d^2)$, and is done once before computing $S_{\cos}$. In practice $d \ll |C|$, so this cost is negligible relative to similarity computation.
- **Adaptive CSLS.** For each query and candidate, we compute row- and column-wise neighbourhood averages based on local top-$k$ sets. Using efficient top-$k$ primitives, the overall cost is $\mathcal{O}(|Q||C|)$, i.e., linear in the size of the similarity matrix with a small constant factor.
- **Structural expert and PoE.** $N_K(c)$ counts, mutual-neighbour sets, and bi-directional top-$L$ neighbourhoods are all computed from $S_{\text{new}}$ using rank-based operations, again with total cost $\mathcal{O}(|Q||C|)$. Combining the geometric and structural experts via PoE is a simple element-wise operation, also $\mathcal{O}(|Q||C|)$.

Overall, the SATTC head adds only a constant-factor $\mathcal{O}(|Q||C|)$ overhead on top of the baseline similarity computation, without changing the asymptotic scaling in the number of queries or candidates.

### S7.2. Wall-Clock Overhead

To quantify practical overhead, we measure the end-to-end test-time cost per LOSO test subject on THINGS-EEG, including similarity computation and calibration, on a single GPU/CPU configuration (see code for details). In our implementation, the additional calibration step remains within the same order of magnitude as the baseline similarity computation and is negligible compared with the cost of running the EEG and image encoders themselves. This makes SATTC practical for large-scale batched evaluation across subjects.

### S7.3. Memory Footprint

The main extra memory requirement of SATTC is storing the similarity matrix and a small number of auxiliary statistics. For $|Q|$ queries and $|C|$ candidates, a dense float32 similarity matrix requires $4|Q||C|$ bytes. In our THINGS-EEG setting, $|Q|$ and $|C|$ are on the order of a few thousands, so $S_{\text{new}}$ and all derived matrices comfortably fit into GPU memory (tens of megabytes), even when using multiple encoders.

All calibration operations in SATTC decompose over rows and columns of the similarity matrix and can therefore be implemented with chunked processing if memory is limited. This makes the test-time head realistic to deploy in practice, not only on toy-scale setups but also in large-scale EEG-to-image retrieval pipelines.

## S8. Training and Compute Details (Summary)

All experiments are conducted on the public THINGS-EEG benchmark under the LOSO (leave-one-subject-out) proto-

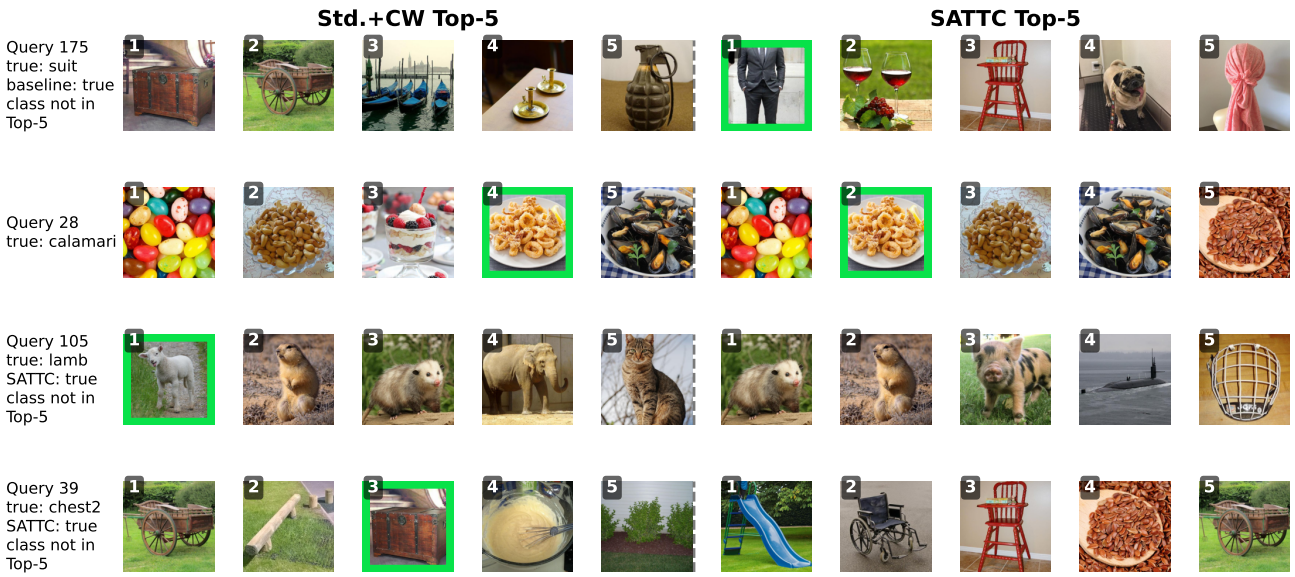


Figure S3. Qualitative Top-5 retrieval examples comparing the standardized baseline (Std. + CW, left) and SATTC (right) on THINGS-EEG. Each row corresponds to one EEG query, with the Top-5 retrieved image classes shown from rank 1 to 5. The green frame marks the true class when it appears in the shortlist. Rows 1–2 illustrate successful cases where SATTC promotes the true class into or closer to the top ranks while slightly down-weighting hub classes. Rows 3–4 show failure cases where the baseline already retrieves the true class within the Top-5, but SATTC pushes it out of the shortlist due to its structural prior. Together, these examples highlight both the strengths and remaining limitations of our structure-aware test-time calibration head.

col described in the main paper. Unless otherwise stated, we train one EEG encoder at a time and keep the image encoder fixed.

**Hardware.** All models are trained on a single NVIDIA RTX 4090 GPU with 24 GB of memory and a host with 120 GB of RAM. No distributed or multi-GPU training is used.

**Optimization.** EEG encoders are trained with the Adam optimizer using cosine learning–rate decay and weight decay in the range reported in the ATM baseline. We adopt fixed batch sizes per dataset split and train for a fixed number of epochs with early stopping on held–out development subjects from the training folds. Hyper-parameters of all encoders are tuned once on the dev subjects and then fixed across all LOSO folds and across all EEG encoder backbones; no statistics or labels from test subjects are used at any stage.

**Training cost.** For each EEG encoder, a full LOSO run (10 outer folds, 3 random seeds) requires on the order of tens of GPU hours on the single RTX 4090. SATTC itself is a test-time head: it does not introduce any additional training, and its parameters (for SAW, candidate whitening, adaptive CSLS, and structural PoE) are either closed-form or tuned on development folds with negligible extra compute. Overall, the total compute budget of this work comfortably fits within a single-GPU setting.

**Test-time cost.** At test time, SATTC only operates on precomputed EEG and image embeddings. For each query set, building the similarity matrix $S_{\text{new}}$ and applying SAW, CW, adaptive CSLS, and the structural PoE expert takes less than a few seconds on a single GPU for the THINGS-EEG scale (a few thousand candidates), and is dominated by matrix multiplications and simple statistics. This confirms that SATTC is practical as a lightweight calibration head.

## S9. Reproducibility Notes

We summarize here the main steps we took to make the experiments reproducible.

**Datasets and splits.** We use only publicly available THINGS-EEG data. The exact preprocessing pipeline (filtering, epoching, channel selection) and the LOSO subject splits are documented in Sec. S1 of the supplementary; the JSON example is included in the code package

**Code.** We publicly release the SATTC codebase at the project repository. The repository includes the evaluation pipeline for the label-free test-time calibration head, reference implementations of the preprocessing and retrieval components, and instructions for reproducing the main results and supplementary analyses. Where applicable, we also provide representative configurations, logs, and example scripts for reproducing the LOSO evaluation protocol. The repository will be maintained as the canonical imple-

mentation of the method.

**Pretrained components.** The image encoder is a standard ViT-H/14 CLIP model; we provide precomputed training and test image features for the THINGS-EEG classes used in our experiments. EEG encoders are trained from scratch on THINGS-EEG and are not initialized from any proprietary models.

**Randomness and seeds.** All reported numbers in the main paper are averages over three random seeds for each LOSO fold and each EEG encoder. For each seed, we fix the data–loader shuffling, parameter initialization, and any stochastic data augmentations.

**Metrics and evaluation protocol.** We report Top-1 and Top-5 retrieval accuracy under LOSO zero-shot evaluation: test images and their EEG responses come from held-out subjects and held-out classes, and are never used during training or hyper-parameter tuning. The evaluation script in the released code mirrors exactly the standardized inference pipeline described in Sec.3 and Sec.4 of the main paper (cosine similarities, $\ell_2$-normalized embeddings, candidate whitening, SAW, adaptive CSLS, structural PoE, and Top-$k$ computation).

**Re-running key results.** Using the provided precomputed embeddings and the evaluation script, a reader can reproduce the main SATTC ablation on a representative LOSO fold (sub-01) with a single command. The resulting Top-1/Top-5 accuracies closely match the corresponding row in Table 1, up to minor fluctuations due to numerical and environment differences.