Batch-Conditioned Semantic Anchors for Robust Transductive Adaptation of Vision–Language Models

Mohammed Rahman Sherif Khan Mohammad
Edge Hill University, UK
Mohamm@edgehill.ac.uk

Ardhendu Behera *
Edge Hill University, UK
Beheraa@edgehill.ac.uk

Sandip Pradhan
Edge Hill University, UK
Pradhans@edgehill.ac.uk

Swagat Kumar
Edge Hill University, UK
Kumars@edgehill.ac.uk

Amr Ahmed
Edge Hill University, UK
Ahmeda@edgehill.ac.uk

Abstract

Transductive adaptation improves vision–language models at test time by refining predictions on an unlabeled batch, but current methods face a key trade-off. Aggressive approaches exploit batch structure but can drift when batches are sparse, have prior shift, or cover little of the label space. Conservative methods better preserve CLIP text semantics, but fixed semantic references can under-adapt even when the batch provides strong corrective evidence. We propose **JPTA** (*Joint Prior and Transport Adaptation*), a prior-aware framework that casts transductive VLM adaptation as *batch-conditioned semantic reference estimation*. Instead of treating CLIP text prototypes as fixed priors or freely replaceable, JPTA views them as semantic references that shift toward image-side batch structure only under calibrated evidence. It estimates batch priors and soft image-side prototypes to form transported semantic references whose impact is scaled by batch reliability. Across standard benchmarks, low effective-class regimes, all-class evaluation, and online streams, JPTA consistently outperforms TransCLIP and StatA, with the largest gains on sparse and prior-shifted batches. Anchor-drift, active-support, and absent-class-mass diagnostics show that JPTA fixes effective-class mismatch without unconstrained transductive drift, indicating that robust transductive VLM adaptation should recalibrate semantic references using reliable batch evidence rather than choose between unrestricted transduction and fixed prototypes. Code is available at `https://github.com/MR-Sherif/JPTA.git`.

1 Introduction

Vision–language models (VLMs) such as CLIP have transformed visual recognition by transferring natural-language supervision to diverse downstream tasks Radford et al. [2021]. Many methods adapt VLMs with prompt learning, cache models, and lightweight adapters Zhou et al. [2022b,a], Zhang et al. [2022], Song et al. [2023], but typically assume an *inductive* setting with labeled support examples. In practice, labels are often unavailable at test time: the model receives only unlabeled batches from unknown environments, with domain shift, class imbalance, and varying batch composition. This motivates *transductive* adaptation, where predictions are refined jointly over the current unlabeled batch rather than per sample.

Transductive inference is attractive because the test batch reveals useful structure: nearby samples in the VLM feature space should share predictions, and the batch can indicate which classes are present. This idea has a long history in few-shot learning via graph propagation, manifold regularization,

*Corresponding author.

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

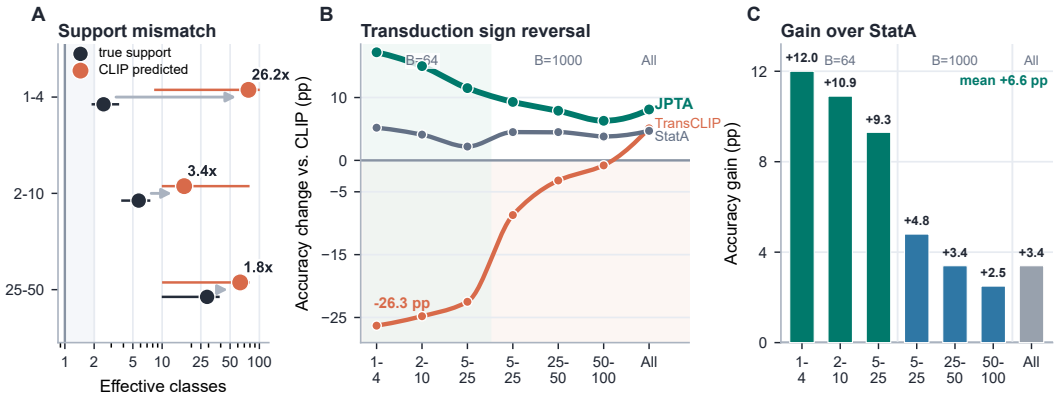


Figure 1: **Low-coverage batches cause class mismatch and semantic drift; JPTA fixes both. A:** In sparse batches, the true effective class support (black) is much smaller than that estimated by zero-shot CLIP predictions (orange), revealing strong class-support inflation; labels show the median inflation factor. **B:** This mismatch alters transductive adaptation: TransCLIP Zanella et al. [2024] (orange) hurts performance in low-coverage regimes but recovers as coverage grows, whereas JPTA (green) remains consistently beneficial. **C:** JPTA outperforms StatA Zanella et al. [2025] in both sparse-batch and all-class regimes, with results grouped by $B = 64$ (green), $B = 1000$ (blue), and all-class (gray).

prototype refinement, and information-theoretic objectives Satorras and Estrach [2018], Rodríguez et al. [2020], Ziko et al. [2020], Boudiaf et al. [2020], Kang et al. [2021], Zhu and Koniusz [2023]. Recently, it has been applied to CLIP through transductive zero-shot and few-shot inference Martin et al. [2024], realistic test-time adaptation Zanella et al. [2025], and language-aware transductive objectives Baklouti et al. [2026]. These works show that unlabeled test batches can strongly guide adaptation even without labeled support or gradient-based fine-tuning.

Despite this promise, transductive VLM adaptation remains brittle in realistic settings. Batch usefulness depends heavily on its composition: large, representative batches yield reliable image-side geometry, while batches with few labels, class-prior shift, or noisy neighborhoods make transductive dynamics misleading. In such *low effective-class* regimes, locally coherent but semantically incorrect groups can dominate refinement, pushing assignments and class statistics away from the VLM's pretrained language-aligned structure Veilleux et al. [2021], Zanella et al. [2025].

Existing methods face a sharp trade-off between *adaptability* and *stability*. Aggressive methods like TransCLIP Zanella et al. [2024] jointly refine assignments, class statistics, and neighborhood consistency to exploit batch structure, but can drift when batches are sparse, imbalanced, or semantically misleading. Conservative methods like StatA Zanella et al. [2025] gain robustness by anchoring refinement to the original CLIP text semantics, but this anchor is fixed. We identify the core failure as *semantic-reference mismatch*: low-coverage batches inflate the predicted class support, reverse the benefit of aggressive transduction, and create the largest gap between fixed anchoring and JPTA. In low-coverage batches, image-side geometry can reveal which classes are active, yet unconstrained transduction turns this signal into semantic drift. Conversely, fixed text anchoring prevents drift but also prevents the semantic reference from correcting systematic zero-shot mismatch. Thus, the key object to adapt is not only the assignment matrix or class statistics, but the semantic reference itself.

We propose **JPTA** (*Joint Prior and Transport Adaptation*), a transductive VLM adaptation framework based on batch-conditioned semantic anchoring. JPTA keeps the usual transductive inference setting but changes what is adapted: instead of fixing CLIP text prototypes, it estimates a batch-conditioned semantic anchor via reliability-controlled transport. Concretely, JPTA estimates the batch class prior, builds soft image-side class prototypes from evolving assignments, and transports text prototypes toward the batch geometry only when the evidence is reliable. The transported anchor is then shrunk toward the original CLIP geometry and used in a prior-aware refinement that couples semantic anchoring, graph consistency, class likelihood, and batch-marginal regularization. This lets the model stay conservative on unreliable batches while adapting when batches consistently support a corrected semantic configuration.

Contributions. Our contributions are threefold. First, we identify *semantic-reference mismatch* as a key failure mode of transductive VLM adaptation under sparse, prior-shifted batches: free

transduction drifts toward hallucinated class support, while fixed anchoring cannot repair misaligned zero-shot references. Second, we introduce *batch-conditioned semantic anchoring*, a principle that treats CLIP text prototypes as movable yet constrained semantic references, implemented via reliability-weighted orthogonal transport and shrinkage toward the original CLIP text geometry. Third, we design a prior-aware transductive solver that couples transported anchors, graph consistency, class-likelihood refinement, and batch-marginal regularization. Across sparse, high-coverage, all-class, and online regimes, JPTA outperforms fixed-anchor and free-transduction baselines, with diagnostics showing reduced absent-class mass and controlled semantic drift.

2 Related Work

Inductive few-shot adaptation of vision–language models. Following CLIP Radford et al. [2021], many works study adapting VLMs with limited labeled data. Prompt-learning methods such as CoOp Zhou et al. [2022b], CoCoOp Zhou et al. [2022a], MaPLe Khattak et al. [2023a], and self-regularized or consistency-guided prompting Khattak et al. [2023b], Roy and Etemad [2024] improve zero-shot transfer by learning task-specific contexts around frozen text and image encoders. Complementary work proposes lightweight feature adaptation and cache-based refinement, including CLIP-Adapter Gao et al. [2024], TIP-Adapter Zhang et al. [2022], Meta-Adapter Song et al. [2023], TaskRes Yu et al. [2023], low-rank adaptation Zanella and Ben Ayed [2024a], graph-based tuning Li et al. [2023], training-only graph supervision Mohammad et al. [2026a], structural distillation into LoRA Mohammad et al. [2026b], and dual-memory mechanisms Zhang et al. [2024c]. Recent studies also question evaluation practices and hidden design choices in few-shot VLM adaptation Silva-Rodriguez et al. [2024], Kravets et al. [2025], Farina et al. [2025]. Our setting is fundamentally different: these methods mainly address the *inductive* regime with labeled support sets, whereas we study *transductive* adaptation on an unlabeled test batch.

Transductive prediction and batch-wise inference. Transductive inference has a long history in few-shot learning, where unlabeled query batches are exploited via graph propagation, manifold smoothing, iterative prototype correction, and information-theoretic objectives Satorras and Estrach [2018], Rodríguez et al. [2020], Ziko et al. [2020], Boudiaf et al. [2020], Liu et al. [2020], Hu et al. [2021], Kang et al. [2021], Zhu and Koniusz [2023], Lazarou et al. [2021]. This work established that query samples should be inferred jointly whenever batch geometry carries useful structure, and that realistic class composition is crucial, since transductive gains depend strongly on how query batches are formed Veilleux et al. [2021]. This principle was extended to CLIP in Transductive Zero-Shot and Few-Shot Martin et al. [2024] and strengthened by LIMO Baklouti et al. [2026], which showed that language-aware transductive objectives improve few-shot CLIP adaptation. Our work adopts this transductive view but targets a fully unlabeled test-time setting, where the batch alone must guide inference.

Test-time adaptation of vision–language models. Recent work extends adaptation from few-shot, support-based settings to unlabeled test time. One direction is *test-time prompt tuning*, which adapts textual context at inference to improve zero-shot robustness, including TPT Shu et al. [2022] and its calibration-focused variants Yoon et al. [2024], Sharifdeen et al. [2025], Ahamed et al. [2026], Sheng et al. [2025]. Another direction performs single-image or instance-level adaptation in feature or parameter space, such as CLIP Reward Zhao et al. [2024], UMFC Liang et al. [2024], WATT Osowiechi et al. [2024], CLIPTTA Lafon et al. [2025], and ROSITA Sreenivas and Biswas [2025]. These methods improve robustness to domain shift, calibration error, or open-set conditions but do not explicitly treat the problem as batch-wise transductive inference over a current unlabeled set. By contrast, SCAP Zhang et al. [2025] and TransCLIP Zanella et al. [2024] directly exploit batch structure, showing that neighborhood agreement and joint refinement can significantly outperform per-sample adaptation when the batch is informative. This distinction is central for our setting: JPTA is a *batch-transductive* VLM adaptation method, rather than a generic per-sample test-time adaptation approach.

Robust transductive adaptation under realistic batch regimes. The closest prior work is StatA Zanella et al. [2025], which shows that aggressive transduction is brittle when the batch has few effective classes, a strong prior shift, or a correlated structure. StatA stabilizes adaptation by anchoring transductive refinement to the original text-side CLIP semantics, improving robustness over free transduction, but its fixed anchor cannot adjust when the batch offers reliable evidence that the semantic reference should shift. Our method differs from both TransCLIP Zanella et al. [2024] and StatA Zanella et al. [2025] by making the anchor itself *batch-aware*: JPTA estimates the batch

prior, transports text prototypes toward the image-side batch structure, and uses this transported anchor in prior-aware transductive refinement. This situates JPTA between free transduction and rigid anchoring, retaining the robustness of anchoring while regaining the flexibility to exploit reliable batch evidence.

Comparison scope. We focus on methods with the same access pattern: frozen CLIP features, no labeled support set, and batch-wise transductive inference. Under this setting, TransCLIP and StatA are the closest baselines. Broader test-time adaptation methods (TPT, WATT, CLIPTTA, and related prompt/feature adaptation approaches) address a different problem because they update prompts, features, or model parameters, or adapt per image instead of using a batch-level transductive solver. We still report compatible representatives with an access-pattern audit in the Appendix (Table 8).

3 Batch-Conditioned Semantic Anchoring

Given an unlabeled test batch and a fixed CLIP label space, JPTA estimates a batch-conditioned semantic reference. It starts from zero-shot predictions to estimate the batch prior, builds soft image prototypes, and transports CLIP text prototypes toward reliable batch geometry with trust-controlled shrinkage. These anchors then guide prior-aware assignment refinement and adaptive class-statistic updates, avoiding both unconstrained transductive drift and rigid anchoring.

3.1 Batch-transductive setting

We adopt the transductive VLM setting of TransCLIP and StatA Zanella et al. [2024, 2025]. Let $f(\cdot)$ and $g(\cdot)$ be the frozen image and text encoders of a pretrained VLM. For label set $\mathcal{Y} = \{1, \dots, K\}$, we encode one normalized text prototype per class, $T = [t_1, \dots, t_K] \in \mathbb{R}^{d \times K}$. At test time, we observe an unlabeled batch of normalized image features $\mathcal{B} = \{x_i\}_{i=1}^N,\ x_i = f(I_i) \in \mathbb{R}^d$, with no labeled support set. The goal is to jointly infer labels for all samples in $\mathcal{B}$ rather than independently.

As in prior work, the full label space may be large, while a given batch may contain only a small subset of active classes. We call this a *low effective-class regime*, where the batch prior can differ substantially from the global label prior Zanella et al. [2025]. In this regime, free transduction may drift, and fixed anchors may become overly conservative. We initialize inference from the zero-shot VLM predictions

$$\ell_{ik}^{(0)} = \tau x_i^\top t_k, \qquad A_{ik}^{(0)} = \frac{\exp(\ell_{ik}^{(0)})}{\sum_{c=1}^K \exp(\ell_{ic}^{(0)})}, \tag{1}$$

where τ is the logit scale and $A^{(0)} \in \mathbb{R}^{N \times K}$ is the zero-shot anchor. We build a batch k-nearest-neighbor graph with affinity matrix $W \in \mathbb{R}^{N \times N}$ and soft assignment matrix $Z \in \mathbb{R}^{N \times K}$ whose rows lie on the simplex. This transductive scaffold fixes the inference protocol but leaves one key choice: *what should adapt at test time?* Existing solvers mainly update assignments and class statistics, either from image-side geometry Zanella et al. [2024] or under a fixed text reference Zanella et al. [2025]. JPTA instead adapts the semantic reference itself: CLIP text prototypes become batch-conditioned anchors estimated via reliability-controlled transport, and assignments are refined relative to these calibrated references.

3.2 Regime statistics and batch prior

The current batch provides more than local geometry: it shows how many classes appear active and how reliable the transductive signal is. From the anchor probabilities $A^{(0)}$, we compute hard pseudo-labels $\hat{y}_i = \arg\max_k A_{ik}^{(0)}$, and estimate several regime statistics. We first measure the *neighbor agreement* ρ on the batch graph, then compute the *anchor ambiguity* $\bar{m}$ as the mean margin between the top-1 and top-2 probabilities.

$$\rho = \frac{1}{|\mathcal{E}|} \sum_{(i,j) \in \mathcal{E}} \mathbf{1}[\hat{y}_i = \hat{y}_j],\ \bar{m} = \frac{1}{N} \sum_{i=1}^N \left(A_{i,(1)}^{(0)} - A_{i,(2)}^{(0)} \right), \tag{2}$$

where $A_{i,(1)}^{(0)}$ and $A_{i,(2)}^{(0)}$ are the largest and second-largest entries of row i. Third, we estimate the effective class coverage of the current batch through the number of predicted classes, maintaining a soft batch prior by averaging the current class posteriors initialized with $Z = A^{(0)}$.

$$K_{\text{pred}} = |\{\hat{y}_i : i = 1, \dots, N\}|, \quad C_{\text{hard}} = \frac{K_{\text{pred}}}{K}, \quad \pi_k = \frac{\frac{1}{N} \sum_{i=1}^N z_{ik} + \alpha_\pi / K}{1 + \alpha_\pi}, \tag{3}$$

where α_π is a fixed symmetric prior strength. C_{hard} indicates whether the batch covers most of the label space. High hard coverage triggers conservative fallback and adaptive iteration control, while low hard coverage uses the full refinement budget, since sparse batches may still contain reliable evidence for their active classes. From π_k, we derive a soft effective-class estimate C_{soft} to compute the transport trust coefficient:

$$\eta = 1 - \frac{C_{\text{soft}}}{K}, \qquad C_{\text{soft}} = \exp(H(\pi)), \qquad H(\pi) = -\sum_{k=1}^{K} \pi_k \log \pi_k. \tag{4}$$

Thus, ρ, $\bar{m}$, and samples per predicted class N/K_{pred} control *whether and how aggressively to refine*, C_{hard} controls the *batch regime*, and C_{soft} controls *how much the semantic anchor is allowed to move*. In the next section, η is employed to calculate the transported semantic anchors.

3.3 Transported semantic anchors

The core idea of JPTA is to adapt the semantic anchor itself. Given current assignments Z, we form soft image-side class prototypes by averaging the image features weighted by their class posteriors: $\bar{x}_k = \frac{\sum_{i=1}^{N} z_{ik} x_i}{\sum_{i=1}^{N} z_{ik} + \varepsilon}$, $n_k = \sum_{i=1}^{N} z_{ik}$, where n_k is the effective support of class k. We seek an orthogonal transport that aligns the original text prototypes with the image-side prototypes while preserving the text space geometry. We solve the weighted Procrustes problem

$$R^\star = \arg\min_{R \in \mathcal{O}(d)} \sum_{k=1}^{K} w_k \|R t_k - \bar{x}_k\|_2^2, \quad \hat{t}_k = R^\star t_k, \quad \tilde{t}_k = \text{norm}\big((1-\eta_k) t_k + \eta_k \hat{t}_k\big), \tag{5}$$

where Rt_k is the transported text prototype of class k, obtained by applying the global orthogonal alignment R to t_k. Here $\mathcal{O}(d)$ is the orthogonal group and w_k are confidence weights based on class support and batch reliability. Let $M = \sum_{k=1}^{K} w_k \bar{x}_k t_k^\top$, and if $M = U\Sigma V^\top$ is the singular value decomposition, then the optimal transport is $R^\star = UV^\top$. The transported text prototypes are $\hat{t}_k$. To reduce sensitivity to unreliable batches, we shrink the transported prototypes toward the original text prototypes t_k. $\eta_k \in [0,1]$ is a transport-trust coefficient: it increases when class k is well supported and the batch regime is reliable, and decreases when the batch is ambiguous or poorly covered. This yields a *transported semantic anchor*: unlike StatA Zanella et al. [2025], which keeps t_k fixed, JPTA lets the anchor move toward the image-side batch geometry while still shrinking toward the original CLIP semantics.

Using $\tilde{T} = [\tilde{t}_1, \ldots, \tilde{t}_K]$, we obtain transported anchor probabilities $A_{ik}^{(\text{tr})}$. Instead of discarding the original zero-shot anchor, we combine both probability sets geometrically in log space,

$$A_{ik}^{(\text{tr})} = \frac{\exp(\tau x_i^\top \tilde{t}_k)}{\sum_{c=1}^{K} \exp(\tau x_i^\top \tilde{t}_c)}, \quad \log A_{ik}^{(\text{anc})} \propto (1-\eta) \log A_{ik}^{(0)} + \eta \log A_{ik}^{(\text{tr})}, \tag{6}$$

where η is a batch-level trust coefficient controlled by the regime statistics. Eq. (6) is central to JPTA: the anchor is no longer fixed, but it is also not free to drift arbitrarily.

Why shrinkage controls semantic drift? The transported prototype can improve semantic reference when the batch estimate is reliable, but should not move freely. The following analysis formalizes this reliability–drift trade-off.

Proposition 1. Let $t_k, \hat{t}_k, u_k \in \mathbb{S}^{d-1}$ denote the original CLIP text prototype, the transported prototype, and the unknown image-side semantic center for class k. Define

$$\tilde{t}_k(\eta_k) = \text{norm}\big((1-\eta_k) t_k + \eta_k \hat{t}_k\big), \qquad \eta_k \in [0,1],$$

and let

$$\alpha_k = \angle(t_k, u_k), \quad \beta_k = \angle(\hat{t}_k, u_k), \quad \theta_k = \angle(t_k, \hat{t}_k).$$

If the transported estimate is better aligned with the image-side centre, $\beta_k < \alpha_k < \pi/2$, then there exists $\eta_k > 0$ such that $\angle(\tilde{t}_k(\eta_k), u_k) < \angle(t_k, u_k)$. Moreover, the semantic drift from the original CLIP anchor is exactly

$$\delta_k(\eta_k) = \angle(\tilde{t}_k(\eta_k), t_k) = \arctan \frac{\eta_k \sin \theta_k}{1 - \eta_k + \eta_k \cos \theta_k},$$

equivalently,

$$\delta_k(\eta_k) = \arccos \frac{1 - \eta_k + \eta_k \cos\theta_k}{\sqrt{(1-\eta_k)^2 + \eta_k^2 + 2\eta_k(1-\eta_k)\cos\theta_k}}.$$

Thus, shrinkage deterministically bounds semantic movement: $\delta_k(\eta_k) = O(\eta_k \sin\theta_k)$ for small η_k. Reducing η_k under weak support or poor alignment limits worst-case anchor drift (proof in Appx A).

3.4 Prior-aware transductive refinement

We refine the assignment matrix $Z \in \Delta_K^N$ using four evidence sources: class likelihood, graph consistency, semantic anchoring, and batch-prior regularization. Class-conditional image features follow diagonal Gaussians $q(x_i \mid k) = \mathcal{N}(x_i; \mu_k, \Sigma_k)$, with (μ_k, Σ_k) updated from current soft assignments. We take ℓ_{ik} from Eq. (1), the anchor distribution $A_{ik}^{(\mathrm{anc})}$ from Eq. (6), and let L_G be the graph Laplacian of affinity matrix W. For fixed anchor and prior, JPTA solves

$$\min_{Z \in \Delta_K^N} -\left\langle Z, \ell + \lambda_a \log A^{(\mathrm{anc})} \right\rangle + \lambda_g \operatorname{Tr}(Z^\top L_G Z) + \lambda_\pi \operatorname{KL}\left(\frac{1}{N} Z^\top \mathbf{1} \,\middle\|\, \pi \right), \tag{7}$$

where λ_a, λ_g, and λ_π control the strength of semantic anchoring, graph smoothness, and batch-marginal consistency, respectively. We optimize Eq. (7) by alternating assignment refinement with updates of the class statistics and prior. At iteration t, we fix the current likelihood $\ell^{(t)}$, anchor $A_{\mathrm{anc}}^{(t)}$, prior $\pi^{(t)}$, and graph message $G^{(t)} = -L_G Z^{(t)}$. We then apply an exponentiated-gradient update

$$\tilde{Z}_{ik}^{(t+1)} = \exp\left(\ell_{ik}^{(t)} + \lambda_a \log A_{\mathrm{anc},ik}^{(t)} + \lambda_g G_{ik}^{(t)} - \lambda_\pi \log \frac{\bar{z}_k^{(t)} + \epsilon}{\pi_k^{(t)} + \epsilon} \right), \tag{8}$$

where $\bar{z}_k^{(t)} = \frac{1}{N}\sum_i z_{ik}^{(t)}$. We normalize the rows of $\tilde{Z}^{(t+1)}$ to obtain $Z^{(t+1)}$, and absorb constant graph and KL factors into λ_g and λ_π. Eq. (7) highlights each term: likelihood updates class statistics, graph smoothing enforces local consistency, semantic anchoring links assignments to the batch-conditioned reference, and prior matching prevents drift toward hallucinated classes. We adapt the prior: in reliable regimes we use Sinkhorn-style normalization for stronger marginal matching; in weak or ambiguous regimes we keep a softer KL penalty.

3.5 Adaptive anchored class statistics

Given updated assignments Z, we compute the empirical batch moments,

$$\bar{\mu}_k = \frac{\sum_i z_{ik} x_i}{n_k + \varepsilon}, \qquad \bar{\Sigma}_k = \frac{\sum_i z_{ik}(x_i - \bar{\mu}_k)^{\odot 2}}{n_k + \varepsilon}, \tag{9}$$

where $\odot 2$ denotes element-wise squaring. Following StatA's statistical anchoring principle, we update class moments via shrinkage instead of direct replacement, anchoring to the *transported* prototypes $\tilde{t}_k$ rather than the original text prototypes. Let $\alpha_k > 0$ be an adaptive pseudocount and $\beta_k = \frac{n_k}{n_k + \alpha_k}$. The anchored updates are

$$\mu_k \leftarrow \operatorname{norm}\big(\beta_k \bar{\mu}_k + (1-\beta_k)\tilde{t}_k\big), \qquad \Sigma_k \leftarrow \beta_k \bar{\Sigma}_k + (1-\beta_k)\Sigma_k^{(0)} + \beta_k(1-\beta_k)(\bar{\mu}_k - \tilde{t}_k)^{\odot 2}, \tag{10}$$

where $\Sigma_k^{(0)}$ is the initial covariance anchor. The pseudocount α_k increases in ambiguous or low-coverage settings and decreases when the batch provides stronger support. JPTA thus ranges from conservative anchored adaptation to flexible batch-driven refinement.

Relation to previous work. JPTA strictly generalizes the TransCLIP–StatA continuum Zanella et al. [2024, 2025]. With zero transport trust and a weakened prior, it behaves like an anchored solver similar to StatA; removing anchor shrinkage and letting the transported anchor dominate yields a more aggressive transductive regime. The contribution is not separate controls but a single organizing principle: *batch-conditioned semantic anchoring*. Regime statistics only determine when the batch adjusts the anchor and how strongly the prior guides inference.

4 Evaluation Under Realistic Batch Regimes

We follow the transductive VLM protocol of TransCLIP Zanella et al. [2024] and StatA Zanella et al. [2025]; unless otherwise stated, all main experiments use the same frozen CLIP ViT-B/16

Table 1: **Small-batch low effective-class evaluation ($B = 64$).** Very Low, Low, and Medium correspond to 1–4, 2–10, and 5–25 active classes per transductive task, respectively. Each result is top-1 accuracy (%) averaged over 1,000 sampled tasks. CLIP and MTA are regime-independent shared baselines. For JPTA, the smaller value below each score reports the signed margin Δ_{best} relative to the strongest non-JPTA baseline in the same column, including the shared baselines.

K_{eff}	Method	**Avg**	ImageNet	SUN397	Aircraft	EuroSAT	Cars	Food101	Pets	Flowers	Caltech	DTD	UCF101
	CLIP	65.2	66.6	62.5	24.7	48.3	65.6	85.9	89.1	70.7	93.2	43.5	67.5
	MTA	66.6	69.3	64.8	27.4	46.9	68.0	87.2	89.4	71.7	94.0	44.4	69.0
Very Low (1–4)	Dirichlet	68.5	79.2	75.7	28.2	47.2	68.2	88.1	87.5	71.2	88.8	50.3	69.0
	ZLaP	27.5	14.5	13.0	8.4	36.6	23.7	31.9	57.0	22.4	52.4	13.0	29.2
	TransCLIP	38.9	21.6	21.1	11.6	45.1	34.7	59.2	72.4	36.4	62.3	26.1	37.7
	StatA	70.4	72.9	66.0	29.3	56.8	76.2	90.3	95.5	77.6	93.0	46.1	70.2
	JPTA (Ours)	**82.4** Δ+12.1	**91.5** Δ+12.3	**90.0** Δ+14.3	**40.1** Δ+10.8	**70.6** Δ+13.8	**84.2** Δ+8.0	**99.5** Δ+9.2	**99.5** Δ+4.0	**82.9** Δ+5.3	**97.7** Δ+3.7	**69.3** Δ+19.0	**81.6** Δ+11.4
Low (2–10)	Dirichlet	70.3	80.1	78.0	28.1	43.5	71.5	92.3	92.7	74.7	93.0	48.9	70.9
	ZLaP	35.2	19.1	19.0	12.0	46.4	27.9	43.5	66.6	31.3	60.8	22.4	38.7
	TransCLIP	40.4	20.3	22.4	14.3	53.9	30.8	55.6	69.4	40.9	64.6	31.6	40.9
	StatA	69.3	72.8	66.9	27.7	51.3	73.5	89.5	93.7	76.6	93.6	46.9	69.6
	JPTA (Ours)	**80.2** Δ+9.9	**90.9** Δ+10.8	**85.5** Δ+7.5	**36.2** Δ+8.1	**66.6** Δ+12.7	**82.5** Δ+9.0	**98.8** Δ+6.5	**98.8** Δ+5.1	**80.6** Δ+4.0	**97.2** Δ+3.2	**64.4** Δ+15.5	**80.6** Δ+9.7
Medium (5–25)	Dirichlet	67.5	77.7	72.9	26.1	38.6	71.6	90.8	88.4	71.5	93.7	42.9	67.8
	ZLaP	44.6	29.0	27.9	16.5	49.0	36.0	59.1	76.4	42.9	72.0	32.0	50.3
	TransCLIP	42.7	15.5	22.8	17.0	58.2	32.9	56.3	72.6	45.0	65.6	37.5	46.5
	StatA	67.4	70.7	65.3	26.0	45.0	71.1	88.2	90.8	73.7	93.9	47.5	69.1
	JPTA (Ours)	**76.7** Δ+9.3	**87.6** Δ+9.9	**82.0** Δ+9.1	**32.9** Δ+5.5	**58.9** Δ+0.7	**80.4** Δ+8.8	**96.2** Δ+5.4	**97.0** Δ+6.2	**78.6** Δ+4.9	**96.7** Δ+2.7	**57.4** Δ+9.9	**76.3** Δ+7.2

backbone, text prompts, precomputed image features, datasets, and batch-generation procedure as the baselines. We evaluate on ImageNet and ten transfer benchmarks (listed in the Appendix). Baselines include shared non-adaptive references, CLIP Radford et al. [2021] and MTA Zanella and Ben Ayed [2024b]; frozen-feature transductive methods, Dirichlet Martin et al. [2024], ZLaP Kalantidis et al. [2024], TransCLIP Zanella et al. [2024], and StatA Zanella et al. [2025]; and online stream methods, TENT Wang et al. [2020], TDA Karmanov et al. [2024], and DMN Zhang et al. [2024c]. We focus on TransCLIP and StatA in the main text as the most comparable VLM-specific batch-transductive baselines under the same frozen-feature, no-support-label protocol. We report top-1 accuracy for independent batch adaptation, low effective-class batches, and realistic online streams. For batch evaluation, we vary batch size B and effective active classes K_{eff} as in Table 7, covering sparse, low-coverage to all-class regimes to test robustness when batch structure is informative but not fully representative. Full implementation details, baseline access patterns, hyperparameters, validation grids, backbone sensitivity, and adaptation-cost measurements appear in Appendix B and Tables 8–17.

Small-Batch Low-Coverage Regimes: Table 1 examines the hardest case: small batches with low effective class coverage, where transductive methods are prone to misinterpreting batch structure. Here, TransCLIP is highly brittle, with average accuracies of 38.9, 40.4, and 42.7 for Very Low, Low, and Medium coverage—well below zero-shot CLIP. This supports StatA's claim that aggressive transduction can amplify class-support errors in sparse batches.

JPTA performs best in these adverse regimes, reaching 82.4, 80.2, and 76.7 average accuracy across the three $B = 64$ settings—margins of +12.1, +9.9, and +9.3 over the strongest baseline. The gains are consistent: JPTA surpasses the strongest baseline in every column and effective-class regime. The largest margins occur on datasets with ambiguous zero-shot predictions and local batch structure, such as DTD, EuroSAT, SUN397, and ImageNet. These results support the central claim that *low-coverage batches require controlled semantic anchoring rather than free batch transduction*; the sparse EuroSAT case in Figure 2 further shows that JPTA improves by suppressing absent-class mass and concentrating predictions on the realized batch support.

Large-Batch and Higher-Coverage Regimes: Table 2 tests JPTA as batch size and class coverage grow. As expected, method gaps shrink because batch statistics improve and class-support inflation declines. Still, JPTA attains the best average performance in all three regimes—74.5 (Medium), 73.1 (High), and 71.5 (Very High)—with margins of +4.8, +3.3, and +2.5 over the strongest baseline.

Per-dataset results show nuance. JPTA is not always best: StatA slightly outperforms it on some datasets (e.g., Aircraft, Pets, UCF101, Cars), depending on the regime. These gaps are small and mostly occur where fixed anchoring is already strong. In contrast, JPTA delivers its largest and most consistent gains on ImageNet and EuroSAT, with further gains on SUN397, Flowers, Caltech,

Table 2: **Large-batch effective-class evaluation ($B = 1000$).** Medium, High, and Very High correspond to 5–25, 25–50, and 50–100 active classes per transductive task, respectively. Each result is top-1 accuracy (%) averaged over 1,000 sampled tasks. For JPTA, the smaller value below each score reports the signed margin Δ_{best} relative to the strongest non-JPTA baseline in the same column.

K_{eff}	Method	Avg	ImageNet	SUN397	Aircraft	EuroSAT	Cars	Food101	Pets	Flowers	Caltech	DTD	UCF101
Medium (5–25)	Dirichlet	64.4	60.9	75.4	26.7	38.8	74.1	76.2	91.0	71.6	92.4	36.2	65.4
	ZLaP	41.5	16.6	20.1	16.4	49.0	32.2	55.5	76.4	40.6	67.7	34.2	48.1
	TransCLIP	56.5	39.9	42.7	22.0	63.1	49.9	80.6	87.9	58.7	79.1	42.9	55.0
	StatA	69.7	70.8	64.5	**28.4**	60.4	74.0	87.5	**93.1**	77.5	92.8	47.1	**70.2**
	JPTA (Ours)	**74.5** Δ+4.8	**88.7** Δ+17.9	**79.6** Δ+4.2	26.0 Δ−2.4	**68.4** Δ+5.3	**77.1** Δ+3.0	**94.4** Δ+6.9	92.6 Δ−0.5	**79.8** Δ+2.3	**94.0** Δ+1.2	**50.1** Δ+3.0	68.8 Δ−1.4
High (25–50)	Dirichlet	45.3	17.3	37.3	21.0	37.9	65.4	46.3	81.3	46.3	80.5	21.1	43.6
	ZLaP	52.2	23.8	32.2	22.2	49.3	45.4	74.9	86.5	56.2	79.7	43.6	60.8
	TransCLIP	62.0	43.9	49.6	24.8	64.0	57.3	83.0	91.4	69.1	85.5	47.5	65.4
	StatA	69.7	71.9	66.4	25.9	60.7	73.6	88.0	91.4	76.7	93.2	47.9	**71.5**
	JPTA (Ours)	**73.1** Δ+3.3	**84.6** Δ+12.7	**71.6** Δ+5.2	**26.6** Δ+0.7	**69.1** Δ+5.1	**73.9** Δ+0.3	**90.6** Δ+2.6	**92.6** Δ+1.2	**79.0** Δ+2.3	**95.2** Δ+2.0	**50.2** Δ+2.3	70.5 Δ−1.0
Very High (50–100)	Dirichlet	33.6	10.8	15.7	17.5	37.8	51.2	29.1	79.3	24.3	59.1	19.0	26.1
	ZLaP	58.4	32.7	44.0	25.4	49.3	55.2	83.3	87.3	64.8	87.9	45.2	67.8
	TransCLIP	64.4	44.5	53.0	25.6	64.1	60.9	85.2	91.9	74.3	90.5	48.1	70.7
	StatA	69.0	71.8	67.1	23.9	60.7	**70.2**	87.1	91.1	74.3	93.7	48.0	70.7
	JPTA (Ours)	**71.5** Δ+2.5	**79.1** Δ+7.3	**69.5** Δ+2.4	**26.9** Δ+1.3	**69.1** Δ+5.0	69.3 Δ−0.9	**87.3** Δ+0.2	**92.5** Δ+0.6	**76.4** Δ+2.1	**94.5** Δ+0.8	**50.3** Δ+2.2	**71.6** Δ+0.9

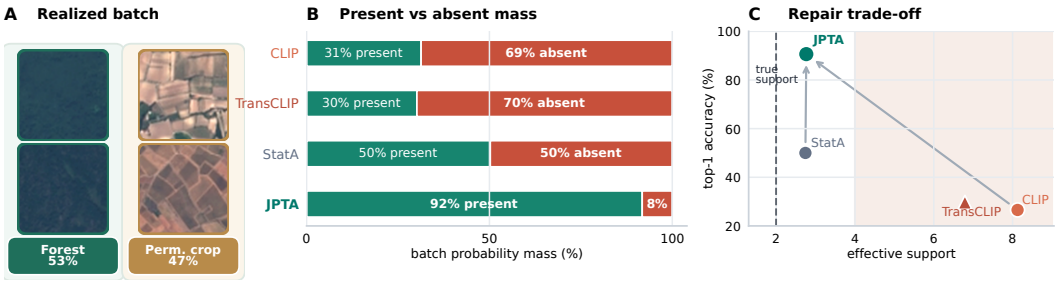


Figure 2: **Sparse EuroSAT case study: JPTA suppresses absent-class mass.** The batch contains only two active classes, *Forest* and *Permanent crop*, while inference is performed over the full label set. **A:** Realized batch composition. **B:** Batch-averaged posterior mass: CLIP and TransCLIP assign high mass to absent classes (red), StatA partially reduces this drift, and JPTA concentrates mass on present classes (green). **C:** JPTA achieves compact predicted support with higher accuracy.

and DTD. Food101 shows a strong gain in the Medium regime but narrows as class coverage increases, where batch-aware anchor transport improves class separation without the severe drift seen in TransCLIP. Thus, JPTA is not only effective in the hardest sparse setting; it remains beneficial as batches become more informative.

Favorable All-Class Evaluation: Table 3 reports results in the full-dataset all-class setting, where the batch label distribution closely matches the full label space and favors standard transductive refinement. This setting evaluates whether JPTA's robustness in sparse batches harms performance when transduction is reliable. JPTA achieves the best average accuracy of 73.3, exceeding TransCLIP by 3.0 points and StatA by 3.4 points. These gains over StatA are important because both methods use semantic anchoring, but only JPTA adapts the anchor itself; the improvement therefore isolates the value of transported, prior-aware anchoring rather than merely stronger regularization.

JPTA outperforms the strongest baseline on all datasets in the all-class setting, with especially large gains on DTD, UCF101, Flowers, Cars, EuroSAT, and SUN397. This indicates that batch-aware anchoring does not just make the method conservative; it also enables stronger adaptation when batches provide reliable class evidence. Thus, the results address a key robustness concern: JPTA improves sparse-regime stability without reducing accuracy under favorable transductive conditions.

Online Test-Time Adaptation: Table 4 evaluates online test-time adaptation with correlated streams and class-sequential batches, a more realistic setting than independent batches because consecutive batches can have biased, non-stationary class priors. TENT, TDA, and DMN provide modest gains over zero-shot CLIP, while StatA is more stable due to its fixed semantic anchor.

Table 3: **All-class transductive evaluation.** All classes appear in the evaluation batch, i.e., the favorable full-label-space setting. Results are reported as top-1 accuracy (%). CLIP and MTA serve as shared baselines. For JPTA, the smaller value under each score is the signed margin Δ_{best} relative to the strongest non-JPTA baseline in the same column, including the shared baselines.

K_{eff}	Method	**Avg**	ImageNet	SUN397	Aircraft	EuroSAT	Cars	Food101	Pets	Flowers	Caltech	DTD	UCF101
	CLIP	65.2	66.6	62.5	24.7	48.3	65.6	85.9	89.1	70.7	93.2	43.5	67.5
	MTA	66.6	69.3	64.8	27.4	46.9	68.0	87.2	89.4	71.7	94.0	44.4	69.0
All	Dirichlet	29.5	66.6	0.7	15.5	39.9	21.2	8.5	76.3	14.3	44.0	19.4	18.4
	ZLaP	66.4	68.8	67.2	26.9	49.1	67.1	86.9	87.8	68.7	90.8	45.5	71.4
	TransCLIP	70.3	70.4	68.9	26.9	66.1	69.5	87.1	92.5	76.5	92.7	48.6	74.1
	StatA	69.9	69.9	68.7	24.7	67.3	68.0	87.1	92.4	75.2	94.2	48.4	73.5
	JPTA (Ours)	**73.3** Δ+3.0	**72.3** Δ+1.9	**72.1** Δ+3.2	**28.2** Δ+0.8	**70.6** Δ+3.3	**72.9** Δ+3.4	**87.5** Δ+0.3	**93.4** Δ+0.9	**80.0** Δ+3.5	**95.1** Δ+0.9	**54.2** Δ+5.6	**79.6** Δ+5.5

Table 4: **Online test-time adaptation under correlated streams ($B = 128$).** Low, Medium, and High denote stream correlation settings with $\gamma = 0.1$, $\gamma = 0.01$, and $\gamma = 0.001$, respectively; Separate denotes class-sequential streams. CLIP and MTA are shared, scenario-independent baselines. Each value is top-1 accuracy (%) averaged over 100 sampled tasks. For JPTA, the smaller value below each score is the signed margin Δ_{best} relative to the strongest non-JPTA baseline in that column.

Scenario	Method	**Avg**	ImageNet	SUN397	Aircraft	EuroSAT	Cars	Food101	Pets	Flowers	Caltech	DTD	UCF101
Shared	CLIP	65.2	66.6	62.5	24.7	48.3	65.6	85.9	89.1	70.7	93.2	43.5	67.5
	MTA	66.6	69.3	64.8	27.4	46.9	68.0	87.2	89.4	71.7	94.0	44.4	69.0
Low ($\gamma = 0.1$)	TENT	65.8	66.6	64.5	24.6	51.8	65.7	85.9	89.3	70.6	93.4	44.0	67.8
	TDA	67.7	68.3	66.0	25.4	60.6	66.9	86.1	89.6	72.5	93.4	45.5	71.0
	DMN	67.2	68.0	64.8	24.9	59.8	67.0	84.2	89.9	73.3	92.2	44.8	70.3
	StatA	67.0	66.2	63.6	24.3	52.3	67.4	88.0	92.5	72.7	94.2	46.8	68.8
	JPTA (Ours)	**72.6** Δ+4.9	**73.6** Δ+4.3	**69.2** Δ+3.2	**28.7** Δ+1.3	**69.4** Δ+8.8	**73.1** Δ+5.1	**92.9** Δ+4.9	**94.3** Δ+1.8	**76.9** Δ+3.6	**96.4** Δ+2.2	**49.7** Δ+2.9	**74.7** Δ+3.7
Medium ($\gamma = 0.01$)	TENT	65.5	66.7	64.3	24.6	47.9	65.6	85.9	89.4	70.6	93.3	44.0	67.8
	TDA	67.1	68.2	65.6	25.2	56.5	66.5	85.8	89.3	72.6	93.5	45.2	70.1
	DMN	66.5	68.0	64.8	24.9	56.2	66.8	81.9	89.0	73.0	92.1	44.9	69.6
	StatA	68.9	69.6	65.9	27.3	52.3	73.2	89.1	94.6	75.6	94.3	46.8	69.7
	JPTA (Ours)	**74.3** Δ+5.4	**75.3** Δ+5.7	**71.2** Δ+5.3	**29.8** Δ+2.4	**66.1** Δ+9.6	**78.4** Δ+5.2	**94.1** Δ+5.0	**96.5** Δ+1.9	**78.4** Δ+2.8	**97.9** Δ+3.6	**52.5** Δ+5.7	**76.9** Δ+6.8
High ($\gamma = 0.001$)	TENT	65.3	66.8	64.3	24.8	45.6	65.6	86.1	89.4	70.5	93.4	44.0	67.9
	TDA	66.8	67.9	65.1	25.1	55.3	66.3	85.5	89.0	72.5	93.6	45.1	69.7
	DMN	66.3	67.9	64.8	24.9	56.3	66.8	79.9	88.9	72.9	92.1	44.8	69.4
	StatA	69.5	71.9	66.0	27.9	51.8	74.7	89.3	94.8	76.4	94.4	47.0	69.8
	JPTA (Ours)	**75.2** Δ+5.7	**76.5** Δ+4.6	**72.4** Δ+6.4	**31.0** Δ+3.1	**66.7** Δ+10.4	**79.2** Δ+4.5	**95.3** Δ+6.0	**96.9** Δ+2.1	**80.2** Δ+3.8	**98.4** Δ+4.0	**53.8** Δ+6.8	**77.1** Δ+7.3
Separate	TENT	64.5	66.7	64.2	24.7	37.0	65.6	86.1	89.3	70.8	93.4	43.9	67.9
	TDA	66.6	67.4	64.6	24.9	55.3	65.9	85.2	88.9	72.3	93.6	45.0	69.6
	DMN	65.8	67.7	64.7	24.9	55.1	66.7	78.5	88.0	72.8	91.9	44.8	69.0
	StatA	69.1	71.7	64.9	28.9	48.2	75.2	88.9	95.2	77.6	94.3	45.8	69.0
	JPTA (Ours)	**75.1** Δ+6.0	**76.3** Δ+4.6	**71.8** Δ+6.9	**30.8** Δ+1.9	**66.9** Δ+11.6	**80.4** Δ+5.2	**92.1** Δ+3.2	**97.2** Δ+2.0	**81.4** Δ+3.8	**97.7** Δ+3.4	**53.4** Δ+7.6	**77.8** Δ+8.2

JPTA outperforms all baselines in every online scenario, exceeding the strongest competitor by $+4.9$, $+5.4$, $+5.7$, and $+6.0$ points for Low, Medium, High, and Separate streams. Gains are especially large on EuroSAT, DTD, UCF101, SUN397, Food101, and Cars, indicating JPTA remains effective under changing batch priors. The Separate scenario is most revealing: classes arrive sequentially, so methods must avoid over-committing to transient evidence. JPTA performs best here, validating regime-adaptive trust and prior-aware refinement for non-i.i.d. test streams.

Dissecting Batch-Aware Anchoring (Ablation Studies): Table 5 ablates JPTA's main components. Replacing the batch-aware anchor with the original fixed one causes the largest drop: 10.7 on average and over 13 in the very-low coverage regime, indicating that fixed semantic anchoring is too rigid when the batch provides reliable corrective evidence. Transported-only anchoring also underperforms Full JPTA, indicating that transport must be blended with the original semantic reference rather than replacing it. Removing prior-aware refinement or disabling regime-adaptive control causes large losses in sparse regimes, confirming that class-marginal regularization and evidence-dependent trust are needed to avoid class-support inflation. Fixed pseudocounts yield smaller but consistent drops, showing adaptive anchored statistics mainly improve cross-regime stability.

Table 5: **Compact ablation summary.** Results are regime-level means over Aircraft, EuroSAT, and Oxford Flowers. Entries are top-1 accuracy (%); parentheses show change relative to Full JPTA. StatA-style uses a fixed CLIP text anchor, Transported-only removes adaptive blending, w/o prior removes batch-marginal refinement, w/o control disables regime-adaptive trust, and Fixed pseudocount replaces adaptive anchored-statistic updates with a fixed schedule.

Regime	Full JPTA	StatA-style	Transported-only	w/o prior	w/o control	Fixed pseudo.
Very low (1–4), B=64	**64.56**	51.47 (−13.09)	59.82 (−4.74)	61.86 (−2.70)	56.46 (−8.10)	61.04 (−3.52)
Low (2–10), B=64	**61.15**	51.51 (−9.64)	57.59 (−3.56)	49.27 (−11.88)	50.41 (−10.74)	57.14 (−4.01)
High (25–50), B=1000	**58.28**	49.21 (−9.07)	55.34 (−2.94)	54.48 (−3.80)	56.11 (−2.17)	56.63 (−1.65)
All classes	**59.62**	48.72 (−10.90)	56.28 (−3.34)	53.01 (−6.61)	55.02 (−4.60)	56.35 (−3.27)
Overall mean	**60.90**	50.23 (−10.67)	57.26 (−3.64)	54.66 (−6.24)	54.50 (−6.40)	57.79 (−3.11)

Additional ablations and diagnostics are provided in Appx. C.1 and Appx. C.2, including the full per-dataset component study in Table 19, the anchor/statistic-update audit in Table 20, the oracle diagnostic study in Table 21, and the transport-geometry audit in Table 22. Qualitative and diagnostic analyses in Appx. C.3 and Appx. C.4 further show how JPTA corrects fine-grained support dilution and applies different adaptive components under different evidence regimes; see Figs 3 and 4. We discuss scope, failure modes, and extensions to open-set or feature-adaptive settings in Appx. D.

Generalization without target-domain retuning: We transfer the original CLIP-selected JPTA configuration unchanged to OpenCLIP ViT-B/16 (DataComp-1B) Gadre et al. [2023] and SigLIP2 So400m/14-384 Tschannen et al. [2025]. Table 6 reports gains over each backbone's own zero-shot baseline across 11 datasets and three regimes. Stronger pretraining reduces headroom but retains adaptation gains; SigLIP2 improves in 29/33 conditions, with four regressions. With ImageNet-selected hyperparameters frozen on ImageNet-A/R/Sketch/V2, JPTA reaches 79.46% versus StatA's 63.46% in Very-Low coverage and 69.33% versus 62.01% in High coverage (Appx. B.9).

Table 6: **Transfer to modern VLMs without backbone-specific retuning.** Entries are mean accuracy gains (pp) over the same-backbone zero-shot model; each regime averages 11 datasets, and Overall includes Very Low, Very High, and All-Class.

Backbone	Very Low	All-Class	Overall	Improved
OpenCLIP DataComp-1B	+10.46	+4.06	+6.13	33/33
SigLIP2 So400m/14-384	+5.96	+2.89	+3.71	29/33

Twenty hyperparameter perturbations incur at most 1.33 pp macro loss (Appx. B.6). Paired uncertainty, initialization robustness, and large-batch scaling are detailed in Appx. B.10, C.7, and B.12; these analyses also expose negative-adaptation cases.

5 Conclusion

We study a key failure mode of transductive adaptation in vision–language models: the tension between exploiting batch structure and preserving reliable semantic priors. Existing methods make this trade-off explicit: aggressive transduction can hallucinate class support in sparse or prior-shifted batches, while fixed anchoring can ignore useful batch evidence. JPTA resolves this by making the semantic anchor batch-aware. It estimates the batch prior, transports text prototypes toward trustworthy image structure, and refines assignments through prior-aware, regime-adaptive control. Across standard benchmarks, low effective-class regimes, all-class evaluation, and online streams, JPTA consistently outperforms TransCLIP and StatA, with the largest gains in challenging regimes while remaining competitive when batches are favorable. This supports a broader principle for test-time VLM adaptation: robustness should come not from choosing between frozen semantics and free transduction, but from adapting the semantic reference to calibrated batch evidence.

Acknowledgments and Disclosure of Funding

This work was supported by UK Research and Innovation (UKRI) through the Engineering and Physical Sciences Research Council (EPSRC) grant ATRACT (`EP/X028631/1`), the Economic and Social Research Council (ESRC) grant SCAnDi (`ES/Y010655/1`), and Cancer Research UK (CRUK) grants `EDDISA-May21\100001` and `EDDPMA-May22\100028`.

References

Jameel Abdul Samadh, Mohammad Hanan Gani, Noor Hussein, Muhammad Uzair Khattak, Muhammad Muzammal Naseer, Fahad Shahbaz Khan, and Salman H Khan. Align your prompts: Test-time prompting with distribution alignment for zero-shot generalization. *Advances in Neural Information Processing Systems*, 36: 80396–80413, 2023.

Shihab Aaqil Ahamed, Udaya Sampath K. Perera Miriya Thanthrige, Ranga Rodrigo, and Muhammad Haris Khan. A-tpt: Angular diversity calibration properties for test-time prompt tuning of vision-language models. In *International Conference on Learning Representations*, 2026.

Ghassen Baklouti, Maxime Zanella, and Ismail Ben Ayed. Language-aware information maximization for transductive few-shot clip. *Transactions on Machine Learning Research*, 2026.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer, 2014.

Malik Boudiaf, Imtiaz Ziko, Jérôme Rony, José Dolz, Pablo Piantanida, and Ismail Ben Ayed. Information maximization for few-shot learning. *Advances in Neural Information Processing Systems*, 33:2445–2457, 2020.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

Qiyuan Dai and Sibei Yang. Free on the fly: Enhancing flexibility in test-time adaptation with online em. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9538–9548, 2025.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Matteo Farina, Massimiliano Mancini, Giovanni Iacca, and Elisa Ricci. Rethinking few-shot adaptation of vision-language models in two stages. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 29989–29998, 2025.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

Chun-Mei Feng, Kai Yu, Yong Liu, Salman Khan, and Wangmeng Zuo. Diverse data augmentation with diffusions for effective test-time prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2704–2714, 2023.

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, Eyal Orgad, Rahim Entezari, Giannis Daras, Sarah Pratt, Vivek Ramanujan, Yonatan Bitton, Kalyani Marathe, Stephen Mussmann, Richard Vencu, Mehdi Cherti, Ranjay Krishna, Pang Wei Koh, Olga Saukh, Alexander Ratner, Shuran Song, Hannaneh Hajishirzi, Ali Farhadi, Romain Beaumont, Sewoong Oh, Alex Dimakis, Jenia Jitsev, Yair Carmon, Vaishaal Shankar, and Ludwig Schmidt. DataComp: In search of the next generation of multimodal datasets. *arXiv preprint arXiv:2304.14108*, 2023.

Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2):581–595, 2024.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8340–8349, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15262–15271, 2021b.

Yuqing Hu, Vincent Gripon, and Stéphane Pateux. Leveraging the feature distribution in transfer-based few-shot learning. In *International conference on artificial neural networks*, pages 487–499. Springer, 2021.

Fanding Huang, Jingyan Jiang, Qinting Jiang, Hebei Li, Faisal Nadeem Khan, and Zhi Wang. Cosmic: Clique-oriented semantic multi-space integration for robust clip test-time adaptation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9772–9781, 2025.

Yannis Kalantidis, Giorgos Tolias, et al. Label propagation for zero-shot classification with vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23209–23218, 2024.

Dahyun Kang, Heeseung Kwon, Juhong Min, and Minsu Cho. Relational embedding for few-shot classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8822–8833, 2021.

Adilbek Karmanov, Dayan Guan, Shijian Lu, Abdulmotaleb El Saddik, and Eric Xing. Efficient test-time adaptation of vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14162–14171, 2024.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19113–19122, 2023a.

Muhammad Uzair Khattak, Syed Talal Wasim, Muzammal Naseer, Salman Khan, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Self-regulating prompts: Foundational model adaptation without forgetting. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15190–15200, 2023b.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013.

Alexey Kravets, Da Chen, and Vinay P Namboodiri. Rethinking few shot clip benchmarks: A critical analysis in the inductive setting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1902–1911, 2025.

Marc Lafon, Gustavo Adolfo Vargas Hakim, Clément Rambour, Christian Desrosiers, and Nicolas Thome. CLIPTTA: Robust contrastive vision-language test-time adaptation. *Advances in neural information processing systems*, 2025.

Michalis Lazarou, Tania Stathaki, and Yannis Avrithis. Iterative label cleaning for transductive and semi-supervised few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8751–8760, 2021.

Xin Li, Dongze Lian, Zhihe Lu, Jiawang Bai, Zhibo Chen, and Xinchao Wang. Graphadapter: Tuning vision-language models with dual knowledge graph. *Advances in Neural Information Processing Systems*, 36: 13448–13466, 2023.

Jiachen Liang, Ruibing Hou, Minyang Hu, Hong Chang, Shiguang Shan, and Xilin Chen. UMFC: Unsupervised multi-domain feature calibration for vision-language models. *Advances in Neural Information Processing Systems*, 37:114072–114093, 2024.

Jinlu Liu, Liang Song, and Yongqiang Qin. Prototype rectification for few-shot learning. In *European conference on computer vision*, pages 741–756. Springer, 2020.

Sarthak Maharana, Baoming Zhang, Leonid Karlinsky, Rogerio Feris, and Yunhui Guo. Batclip: Bimodal online test-time adaptation for clip. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1569–1579, 2025.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

Ségolène Martin, Yunshi Huang, Fereshteh Shakeri, Jean-Christophe Pesquet, and Ismail Ben Ayed. Transductive zero-shot and few-shot clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28816–28826, 2024.

Mohammed Rahman Sherif Khan Mohammad, Ardhendu Behera, Sandip Pradhan, Swagat Kumar, and Amr Ahmed. Training-only heterogeneous image-patch-text graph supervision for advancing few-shot learning adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19613–19622, June 2026a.

Mohammed Rahman Sherif Khan Mohammad, Ardhendu Behera, Sandip Pradhan, Swagat Kumar, and Yonghuai Liu. Sd-lora: Training-time structural distillation into lora for few-shot vision-language adaptation, 2026b. URL https://arxiv.org/abs/2611.07063.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008.

David Osowiechi, Mehrdad Noori, Gustavo A Hakim, Moslem Yazdanpanah, Ali Bahri, Milad Cheraghalikhani, Sahar Dastani, Farzad Beizaee, Ismail B Ayed, and Christian Desrosiers. WATT: Weight average test-time adaptation of CLIP. *Advances in neural information processing systems*, 37:48015–48044, 2024.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

Pau Rodríguez, Issam Laradji, Alexandre Drouin, and Alexandre Lacoste. Embedding propagation: Smoother manifold for few-shot classification. In *European conference on computer vision*, pages 121–138. Springer, 2020.

Shuvendu Roy and Ali Etemad. Consistency-guided prompt learning for vision-language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Oindrila Saha, Logan Lawrence, Grant Van Horn, and Subhransu Maji. Generate, transduct, adapt: Iterative transduction with vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1369–1379, 2025.

Victor Garcia Satorras and Joan Bruna Estrach. Few-shot learning with graph neural networks. In *International conference on learning representations*, 2018.

Ashshak Sharifdeen, Muhammad Akhtar Munir, Sanoojan Baliah, Salman Khan, and Muhammad Haris Khan. O-tpt: Orthogonality constraints for calibrating test-time prompt tuning in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19942–19951, 2025.

Lijun Sheng, Jian Liang, Zilei Wang, and Ran He. R-tpt: Improving adversarial robustness of vision-language models through test-time prompt tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 29958–29967, 2025.

Changho Shin, Jitian Zhao, Sonia Cromp, Harit Vishwakarma, and Frederic Sala. Otter: effortless label distribution adaptation of zero-shot models. *Advances in Neural Information Processing Systems*, 37:44064–44101, 2024.

Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems*, 35:14274–14289, 2022.

Julio Silva-Rodriguez, Sina Hajimiri, Ismail Ben Ayed, and Jose Dolz. A closer look at the few-shot adaptation of large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23681–23690, 2024.

Lin Song, Ruoyi Xue, Hang Wang, Hongbin Sun, Yixiao Ge, Ying Shan, et al. Meta-adapter: An online few-shot learner for vision-language model. *Advances in Neural Information Processing Systems*, 36:55361–55374, 2023.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Manogna Sreenivas and Soma Biswas. Efficient open set single image test time adaptation of vision language models. *Transactions on Machine Learning Research (TMLR)*, 2025.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Olivier Veilleux, Malik Boudiaf, Pablo Piantanida, and Ismail Ben Ayed. Realistic evaluation of transductive few-shot learning. *Advances in Neural Information Processing Systems*, 34:9290–9302, 2021.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Advances in neural information processing systems*, 32, 2019.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

Hee Suk Yoon, Eunseop Yoon, Joshua Tian Jin Tee, Mark Hasegawa-Johnson, Yingzhen Li, and Chang D. Yoo. C-tpt: Calibrated test-time prompt tuning for vision-language models via text feature dispersion. In *International Conference on Learning Representations*, 2024.

Tao Yu, Zhihe Lu, Xin Jin, Zhibo Chen, and Xinchao Wang. Task residual for tuning vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10899–10909, 2023.

Maxime Zanella and Ismail Ben Ayed. Low-rank few-shot adaptation of vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 1593–1603, 2024a.

Maxime Zanella and Ismail Ben Ayed. On the test-time zero-shot generalization of vision-language models: Do we really need prompt learning? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23783–23793, 2024b.

Maxime Zanella, Benoît Gérin, and Ismail Ben Ayed. Boosting vision-language models with transduction. *Advances in Neural Information Processing Systems*, 37:62223–62256, 2024.

Maxime Zanella, Clément Fuchs, Christophe De Vleeschouwer, and Ismail Ben Ayed. Realistic test-time adaptation of vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25103–25112, 2025.

Ce Zhang, Simon Stepputtis, Katia Sycara, and Yaqi Xie. Dual prototype evolving for test-time generalization of vision-language models. *Advances in Neural Information Processing Systems*, 37:32111–32136, 2024a.

Chenyu Zhang, Kunlun Xu, Zichen Liu, Yuxin Peng, and Jiahuan Zhou. Scap: Transductive test-time adaptation via supportive clique-based attribute prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 30032–30041, 2025.

Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free adaption of clip for few-shot classification. In *European conference on computer vision*, pages 493–510. Springer, 2022.

Taolin Zhang, Jinpeng Wang, Hang Guo, Tao Dai, Bin Chen, and Shu-Tao Xia. Boostadapter: Improving vision-language test-time adaptation via regional bootstrapping. *Advances in Neural Information Processing Systems*, 37:67795–67825, 2024b.

Yabin Zhang, Wenjie Zhu, Hui Tang, Zhiyuan Ma, Kaiyang Zhou, and Lei Zhang. Dual memory networks: A versatile adaptation approach for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 28718–28728, 2024c.

Shuai Zhao, Xiaohan Wang, Linchao Zhu, and Yi Yang. Test-time adaptation with CLIP reward for zero-shot generalization in vision-language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16816–16825, 2022a.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022b.

Lihua Zhou, Mao Ye, Shuaifeng Li, Nianxin Li, Xiatian Zhu, Lei Deng, Hongbin Liu, and Zhen Lei. Bayesian test-time adaptation for vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29999–30009, 2025.

Hao Zhu and Piotr Koniusz. Transductive few-shot learning with prototype-based label propagation by iterative graph refinement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23996–24006, 2023.

Imtiaz Ziko, Jose Dolz, Eric Granger, and Ismail Ben Ayed. Laplacian regularized few-shot learning. In *International conference on machine learning*, pages 11660–11670. PMLR, 2020.

A Shrinkage-Controlled Transport and Semantic Drift

Proof of Proposition 1: Let $s(\eta_k) = \tilde{t}_k(\eta_k)^\top u_k$. Since $\angle(\tilde{t}_k(\eta_k), u_k)$ decreases when $s(\eta_k)$ increases, it is sufficient to show that $s'(0) > 0$. Writing $A = \cos\alpha_k$, $B = \cos\beta_k$, and $C = \cos\theta_k$, direct differentiation gives

$$s'(0) = B - AC.$$

Because $\beta_k < \alpha_k < \pi/2$, we have $B > A > 0$, and since $C \leq 1$, $B - AC > 0$. Hence, for sufficiently small $\eta_k > 0$, $\tilde{t}_k(\eta_k)$ is closer to u_k than t_k. The drift expression follows by computing the dot product

$$\tilde{t}_k(\eta)^\top t_k = \frac{1 - \eta_k + \eta_k \cos\theta_k}{\|(1-\eta_k)t_k + \eta_k \hat{t}_k\|},$$

and using the two-dimensional geometry of the span of t_k and $\hat{t}_k$.

In the implementation, η_k is calculated from the aggregate soft transport support and the orthogonal alignment score, rather than being learned from labels. Therefore, Proposition 1 explains why the transported anchor can improve alignment when the batch estimate is reliable, while shrinkage limits semantic drift when the transport estimate is weak.

B Experimental Protocol and Implementation Details

This appendix provides the full experimental protocol, baseline coverage, implementation details, and hyperparameter selection strategy used in Sec. 4. Unless stated otherwise, all methods are evaluated under the same frozen CLIP feature space, text prompts, batch construction, and random seeds.

B.1 Datasets and Feature Extraction

We follow the transductive VLM evaluation protocol of TransCLIP Zanella et al. [2024] and StatA Zanella et al. [2025]. Experiments are run on ImageNet Deng et al. [2009] and ten transfer benchmarks: Caltech101 Fei-Fei et al. [2004], OxfordPets Parkhi et al. [2012], StanfordCars Krause et al. [2013], Flowers102 Nilsback and Zisserman [2008], Food101 Bossard et al. [2014], FGVC-Aircraft Maji et al. [2013], SUN397 Xiao et al. [2010], DTD Cimpoi et al. [2014], EuroSAT Helber et al. [2019], and UCF101 Soomro et al. [2012], spanning object recognition, fine-grained classification, textures, scenes, actions, food, and remote sensing.

All main experiments use the frozen CLIP ViT-B/16 image encoder and its corresponding frozen text encoder; all methods share the same cached image features and normalized text prototypes. Image features are extracted once and cached before transductive inference. Text prototypes are obtained by encoding class-specific prompts with the frozen text encoder and ℓ_2-normalizing the embeddings. Unless stated otherwise, we use the TransCLIP/StatA prompt templates and class-name processing. No method sees test labels during adaptation.

For fair comparison, all transductive methods use the same normalized image features, normalized text prototypes, and unlabeled batches. Performance differences therefore stem only from the transductive solver, not feature extraction, prompt engineering, or batching.

B.2 Batch Construction and Effective-Class Regimes

We evaluate methods under three regimes: independent batch adaptation, low effective-class adaptation, and online adaptation.

Independent batch adaptation. Each test batch is adapted independently. At the start of each batch, the transductive solver is reset and given only that batch's unlabeled features. Predictions are made after batch-level refinement. This measures how well each method exploits the current batch without using information from other batches.

Low effective-class adaptation. To test robustness under realistic class-prior shift, we vary the effective number of active classes per batch. For a dataset with K classes, we sample K_{eff} active classes and form a batch of size B using only those classes. The global label space is fixed, but each

batch may cover only a small subset. This reveals the failure mode highlighted in StatA Zanella et al. [2025]: transductive methods may perform well when all classes appear, but degrade when batches are sparse or prior-shifted.

Online adaptation. We also evaluate realistic online streams when applicable. Test batches arrive sequentially and may have correlated class composition. Each method processes the stream without labels. Stateless methods operate on each batch independently, while online-capable methods update internal statistics according to their protocol. All methods see the same batch order.

Table 7 lists the batch sizes and active-class ranges used in the batch evaluations.

Table 7: Effective-class regimes used for realistic transductive evaluation. For datasets with fewer than the upper-bound number of classes, K_{eff} is clipped by the dataset class count K.

Regime	Batch size B	Effective classes K_{eff}
Very Low	64	1–4
Low	64	2–10
Medium	64	5–25
Medium	1000	5–25
High	1000	25–50
Very High	1000	50–100
All Classes	full test set / all-class batch	K

All reported results are averaged over repeated random batch samplings and random seeds. The same sampled batches are reused across methods. We report top-1 accuracy as the primary metric.

B.3 Baselines

We compare JPTA against four groups of baselines.

Zero-shot reference. **Zero-shot CLIP** classifies each image via cosine similarity between frozen image embeddings and frozen text prototypes. It is the non-adaptive reference, indicating whether test-time adaptation improves or harms the pretrained VLM.

Transductive VLM baselines. The main baselines are **TransCLIP** Zanella et al. [2024] and **StatA** Zanella et al. [2025]. TransCLIP performs strong batch-level transductive refinement by exploiting unlabeled batch structure and the text prior. StatA improves robustness by anchoring batch adaptation to the original text prototypes. They illustrate the core trade-off in our work: free transduction adapts strongly but can drift on sparse batches, while fixed anchoring is stable but may under-adapt when the batch provides strong evidence.

Classical transductive and feature-space baselines. When compatible with our protocol, we include representative transductive baselines from prior work, such as Dirichlet-based refinement, ZLaP, LaplacianShot Ziko et al. [2020], TIM Boudiaf et al. [2020], and PT-MAP-style prototype correction Hu et al. [2021]. These methods mainly operate in visual feature space and do not explicitly model the language-side semantic anchor as VLM-specific transductive methods do.

Adjacent test-time VLM adaptation baselines. We also evaluate broader test-time VLM adaptation methods whose access patterns match our setting. These include prompt-tuning methods such as TPT Shu et al. [2022], C-TPT Yoon et al. [2024], O-TPT Sharifdeen et al. [2025], and variants, as well as unlabeled adaptation methods such as CLIP Reward Zhao et al. [2024], UMFC Liang et al. [2024], WATT Osowiechi et al. [2024], CLIPTTA Lafon et al. [2025], and ROSITA Sreenivas and Biswas [2025]. These methods contextualize JPTA within test-time adaptation, but the main paper centers on TransCLIP and StatA, which share JPTA's batch-transductive access pattern and frozen CLIP feature space and thus provide the most direct comparisons.

Baseline access pattern. Table 8 summarizes the access patterns of recent VLM test-time and transductive adaptation methods.

Table 8: **Baseline coverage and access-pattern audit for VLM test-time and transductive adaptation.** A method is considered a direct protocol match only if it uses an unlabeled test batch, requires no labeled support examples, keeps CLIP encoders/prompts frozen, and performs adaptation through a frozen-feature batch-level transductive solver.

Method	Adaptation family	Batch eval.	Online eval.	No support labels	Frozen CLIP	Direct match
Shared reference baselines						
CLIP Radford et al. [2021]	Zero-shot reference	✓	✓	✓	✓	✗
MTA Zanella and Ben Ayed [2024b]	Static reference baseline	✓	✓	✓	✓	✗
Frozen-feature transductive baselines used in the main batch evaluation						
Dirichlet Martin et al. [2024]	Prior-based transductive refinement	✓	✗	✓	✓	✓
ZLaP Kalantidis et al. [2024]	Graph label propagation	✓	✗	✓	✓	✓
TransCLIP Zanella et al. [2024]	Free batch transduction	✓	✗	✓	✓	✓
StatA Zanella et al. [2025]	Fixed-anchor transduction	✓	✓	✓	✓	✓
Online or memory-based test-time adaptation baselines						
TENT Wang et al. [2020]	Entropy-minimization TTA	✗	✓	✓	✗	✗
TDA Karmanov et al. [2024]	Dynamic cache adaptation	✗	✓	✓	✓	✗
DMN Zhang et al. [2024c]	Dual-memory adaptation	✗	✓	✓	✓	✗
FreeTTA Dai and Yang [2025]	Online distribution adaptation	✗	✗	✓	✓	✗
BCA Zhou et al. [2025]	Bayesian class/prior adaptation	✗	✗	✓	✗	✗
DPE Zhang et al. [2024a]	Dual prototype evolving	✗	✗	✓	✓	✗
BoostAdapter Zhang et al. [2024b]	Regional bootstrapping and memory adaptation	✗	✗	✓	✓	✗
COSMIC Huang et al. [2025]	Cross-modal semantic cache adaptation	✗	✗	✓	✗	✗
BATCLIP Maharana et al. [2025]	Bimodal online test-time adaptation	✗	✗	✓	✗	✗
CLIPTTA Lafon et al. [2025]	Contrastive feature/model adaptation	✗	✗	✓	✗	✗
WATT Osowiechi et al. [2024]	Weight-averaged test-time adaptation	✗	✗	✓	✗	✗
Prompt-, attribute-, or language-augmented test-time adaptation methods						
TPT Shu et al. [2022]	Test-time prompt tuning	✗	✗	✓	✗	✗
DiffTPT Feng et al. [2023]	Diffusion-augmented prompt tuning	✗	✗	✓	✗	✗
PromptAlign Abdul Samadh et al. [2023]	Distribution-aligned test-time prompting	✗	✗	✓	✗	✗
C-TPT Yoon et al. [2024]	Calibrated test-time prompt tuning	✗	✗	✓	✗	✗
O-TPT Sharifdeen et al. [2025]	Orthogonality-constrained prompt tuning	✗	✗	✓	✗	✗
A-TPT Ahamed et al. [2026]	Angular-diversity prompt calibration	✗	✗	✓	✗	✗
R-TPT Sheng et al. [2025]	Robust test-time prompt tuning	✗	✗	✓	✗	✗
SCAP Zhang et al. [2025]	Clique-based attribute prompting	✗	✗	✓	✗	✗
GTA-CLIP Saha et al. [2025]	Generate–transduct–adapt with attributes	✗	✗	✓	✗	✗
LIMO Baklouti et al. [2026]	Language-aware transductive few-shot CLIP	✗	✗	✗	✗	✗
OTTER Shin et al. [2024]	Label-distribution adaptation	✗	✗	✓	✓	✗
JPTA	**Batch-conditioned semantic anchoring**	✓	✓	✓	✓	✓

Notes. "Direct match" assumes unlabeled test-batch access, no labeled support data, frozen CLIP encoders/prompts, and adaptation only via a frozen-feature, batch-level transductive solver. CLIP and MTA serve as non-adaptive baselines. TENT, TDA, and DMN are used in the online stream setting as test-time adaptation baselines. Other approaches—prompt-, feature-, weight-, memory-, or few-shot adaptation—are related but do not match the exact access pattern considered by JPTA.

Our main batch evaluation targets frozen-feature transductive baselines that follow the same unlabeled, no-support-label protocol as JPTA, including Dirichlet, ZLaP, TransCLIP, and StatA. The online evaluation adds stream-based test-time adaptation baselines (TENT, TDA, DMN) to assess robustness under non-i.i.d. sequential batches. Several recent methods are related but do not exactly match our protocol: prompt-based methods update textual prompts at inference; memory/cache methods accumulate information across test streams; Bayesian or label-distribution methods adjust priors or class embeddings; and attribute/LLM-based methods add extra language-side adaptation. We therefore cite and audit these methods for completeness, but restrict primary comparisons to methods with the same frozen-feature, unlabeled-batch transductive access pattern.

Equal-budget paired baseline comparison. For the additional Low and Very-High comparisons analyzed in Appx. B.10, TransCLIP, StatA, and JPTA use the original TransCLIP/StatA data loaders and share validation splits, sampled images, class subsets, seeds, batch sizes, effective-class ranges, and task manifests. Each method is tuned on the same held-out tasks with the same selection metric and the same number of configurations per regime. Selected settings are frozen before test evaluation;

no test labels or per-task hyperparameter selection are used. Test comparisons share frozen CLIP features and prompts, and graph-based methods share the same k-NN construction. TransCLIP's sparse-regime degradation therefore persists after equal-budget low-coverage tuning, rather than reflecting only an inherited All-Class configuration. This paired baseline study is distinct from the ImageNet-to-shifted-domain transfer protocol in Appx. B.9.

B.4 Implementation Details

JPTA is a gradient-free transductive solver on frozen CLIP features. It does not fine-tune the image encoder, text encoder, prompts, adapters, or any VLM parameters at test time. For each batch, JPTA initializes from the zero-shot CLIP posterior, estimates batch-level class marginals, builds soft image-side class prototypes, transports text prototypes toward the batch geometry, and iteratively refines assignments and anchored class statistics.

Feature normalization. All image and text embeddings are ℓ_2-normalized before inference, so cosine similarity reduces to an inner product. The CLIP logit scale is fixed and shared across zero-shot CLIP, TransCLIP, StatA, and JPTA.

Graph construction. For each batch, we build a k-nearest-neighbor graph in normalized image-feature space, used only for transductive inference and reconstructed per batch unless the online protocol states otherwise. The same graph construction is used for all graph-based baselines.

Anchor transport. JPTA forms soft image-side class prototypes from the current assignment matrix and transports text prototypes toward them via a weighted alignment step. The transported anchor is blended with the original text anchor using a trust coefficient set by batch evidence: when the batch is ambiguous, sparse, or shows low neighborhood agreement, JPTA stays close to original CLIP semantics; when the batch is coherent, the transported anchor has a stronger influence.

Prior-aware refinement. The estimated batch class prior regularizes assignment updates. It limits probability mass on many inactive classes in low-coverage batches, while staying flexible when the batch prior is uncertain. The same estimate also drives the adaptive trust mechanism for anchor transport and anchored statistics.

Numerical stability. All probability vectors are clipped by a small constant ϵ before logarithms or normalization. Covariances use diagonal statistics with variance flooring. Classes with very small effective mass are updated conservatively and remain near their semantic anchor.

Algorithm overview. Algorithm 1 summarizes the order of the existing updates, distinguishing quantities computed from the initial batch from those updated during refinement.

Algorithm 1 JPTA: Batch-Conditioned Semantic Anchoring

Require: Frozen normalized image features $X = \{x_i\}_{i=1}^{N}$, text prototypes $T = \{t_k\}_{k=1}^{K}$, pretrained logit scale τ, graph size k, fixed solver hyperparameters, iteration budget $T_{\max}$
Ensure: Predicted labels $\{\hat{y}_i\}_{i=1}^{N}$

Compute the zero-shot posterior $A^{(0)}$; set $Z \leftarrow A^{(0)}$ ▷ Eq. 1
Construct the batch k-NN graph W and its Laplacian L_G
Compute initial batch evidence and initialize the smoothed prior π ▷ Eqs. 2–4
Initialize anchored class statistics
for $t = 0, \ldots, T_{\max} - 1$ **do**
    Compute effective supports $n_k \leftarrow \sum_i z_{ik}$ and soft image prototypes $\bar{x}_k$
    Set confidence weights w_k and evidence-dependent transport trust
    $M \leftarrow \sum_k w_k \bar{x}_k t_k^\top$; compute $M = USV^\top$
    $R^\star \leftarrow UV^\top$; $\hat{t}_k \leftarrow R^\star t_k$ ▷ Eq. 5
    $\tilde{t}_k \leftarrow \mathrm{norm}((1-\eta_k)t_k + \eta_k \hat{t}_k)$ ▷ Eq. 5
    Compute $A^{(\mathrm{tr})}$ and the blended anchor $A^{(\mathrm{anc})}$ ▷ Eq. 6
    Hold the current likelihood, anchor, prior, and graph message fixed
    Refine assignments Z using the evidence-selected marginal branch ▷ Eq. 8
    Compute empirical moments $\bar{\mu}_k, \bar{\Sigma}_k$ ▷ Eq. 9
    Update anchored class statistics μ_k, Σ_k and refresh π ▷ Eqs. 10, 3
    if the solver's early-stop condition holds **then**
        break
    end if
end for
return $\hat{y}_i \leftarrow \arg\max_k z_{ik}$ for each image i

The original posterior, graph, and initial pseudo-label evidence are batch-initial quantities; assignments, soft prototypes, the prior, transport, and anchored moments belong to the alternating refinement. Independent batches reset the solver. Label-informed oracle variants in Appx. C are diagnostic interventions and are not used by this inference procedure.

B.5 Hyperparameter Selection

All hyperparameters are chosen without using test labels. When a validation split is available, we tune them once on that split and fix them for all test batches. When following a prior benchmark, we keep its TransCLIP/StatA settings for shared components (feature extraction, graph construction, batching). We do not perform any per-dataset test-time tuning. Table 9 summarizes each hyperparameter and its selection rule.

Table 9: JPTA hyperparameters. Shared protocol parameters follow TransCLIP/StatA where applicable. JPTA-specific parameters are selected once on validation data and then fixed for test evaluation.

Symbol / parameter	Role	Selection protocol
B	batch size	fixed by regime, e.g., 64 or 1000
K_{eff}	active classes per batch	fixed by regime in Table 7
k	nearest neighbors for graph	shared with TransCLIP/StatA protocol
$T_{\max}$	refinement iterations	fixed on validation split
τ	CLIP logit scale	frozen CLIP value
λ_a	anchor weight	validation-selected
λ_g	graph consistency weight	validation-selected or inherited from protocol
λ_π	batch-prior regularization weight	validation-selected

Table 10 lists the fixed settings by regime, and Table 11 gives the candidate values for JPTA-specific parameters. Unless an ablation states otherwise, we use the same selected settings across datasets to avoid hidden per-dataset tuning.

Table 10: **Fixed JPTA hyperparameters.** We report the method-level hyperparameters used in the evaluation protocol. The configuration is selected on validation tasks and then held fixed for test-set evaluation. Controller thresholds are fixed to default values in all experiments and are not tuned per dataset.

Regime	k	$T_{\max}$	λ_a	λ_g	λ_π	α_π
Small-batch low coverage	3	10	1	1	1	10
Large-batch high coverage	3	10	1	1	1	10
All-class	3	10	1	1	1	10
Online streams	3	10	1	1	1	10

Table 11: **Validation grid for JPTA method-level hyperparameters.** The final configuration in Table 10 is selected from this compact validation grid.

Hyperparameter	Search space
k	$\{3, 5, 10\}$
$T_{\max}$	$\{5, 10, 20\}$
λ_a	$\{0.25, 0.5, 1, 2\}$
λ_g	$\{0.01, 0.05, 0.1, 0.5, 1\}$
λ_π	$\{0.01, 0.05, 0.1, 0.5, 1\}$
α_π	$\{1, 5, 10, 20, 50\}$

B.6 Random Hyperparameter Perturbations

To test robustness away from the selected configuration, we evaluate 20 predeclared Latin-hypercube perturbations of the active regularization weights over a $[0.5\times, 2\times]$ interval. The same perturbation settings are applied to ten datasets in Very-Low, High, and All-Class regimes, with no validation-based selection among the 20 settings. All other settings, including controller thresholds, prior strength, and iteration budget, remain frozen. This gives 600 perturbed evaluations and 30 frozen-reference evaluations. The pretrained logit scale is fixed for each backbone and shared with its zero-shot baseline; it is not tuned by JPTA.

Table 12 shows an average loss below one percentage point in every regime and a worst macro loss of 1.33 pp across the 20 settings.

Table 12: **Hyperparameter perturbation sensitivity.** Impact is the macro accuracy difference (pp) from the corresponding frozen JPTA reference. SD, 10th percentile, and worst impact summarize the 20 settings, rather than uncertainty over sampled test tasks.

Regime	Mean impact $\pm$ SD	10th percentile	Worst impact
Very Low	-0.53 ± 0.35	-0.98	-1.24
High	-0.41 ± 0.25	-0.82	-0.85
All-Class	-0.76 ± 0.37	-1.26	-1.33

These results provide evidence of local robustness around the frozen configuration, rather than a guarantee for arbitrary hyperparameters. Individual datasets, particularly EuroSAT, exhibit greater variation than these macro summaries. Source-to-target transfer in Appx. B.9 separately tests a configuration selected without target-domain validation.

B.7 Backbone sensitivity.

Table 13 tests whether JPTA's gains depend on a specific CLIP encoder or hold across backbone capacities. Under the same $B = 64$ transductive protocol, JPTA outperforms the strongest baseline for both RN50 and ViT-L. Improvements are most pronounced in the sparse Very Low regime, where limited class coverage weakens fixed semantic anchors and batch-aware transported anchors yield a stronger adaptation signal. As expected, ViT-L achieves higher absolute accuracy than RN50, but the relative pattern is consistent: JPTA improves average performance across effective-class regimes, showing that the adaptation mechanism is not tied to a single backbone architecture.

Table 13: Comparison of various CLIP encoders for the batch test-time adaptation setting. Each reported performance is averaged over 1,000 tasks.

(a) RN50

K_{eff}	Method	**Avg**	ImageNet	SUN397	Aircraft	EuroSAT	Cars	Food101	Pets	Flowers	Caltech	DTD	UCF101
	CLIP	58.7	58.2	58.9	17.0	36.2	55.8	77.4	85.7	66.1	85.7	42.8	61.8
Very Low	StatA	65.8	68.2	63.7	21.1	43.3	71.3	87.1	93.1	74.1	90.1	45.3	66.2
	JPTA (Ours)	**73.3** Δ+7.5	**79.1** Δ+10.9	**76.3** Δ+12.6	**25.5** Δ+4.4	**45.0** Δ+1.7	**80.6** Δ+9.3	**96.8** Δ+9.7	**95.5** Δ+2.4	**75.2** Δ+1.1	**95.2** Δ+5.1	**54.6** Δ+9.3	**82.3** Δ+16.1
Low	StatA	63.5	65.4	63.1	16.5	51.7	65.4	81.0	84.4	70.0	88.3	47.2	66.0
	JPTA (Ours)	**65.8** Δ+2.3	**68.9** Δ+3.5	**66.3** Δ+3.2	**17.5** Δ+1.0	**52.9** Δ+1.2	**67.6** Δ+2.2	**85.3** Δ+4.3	**88.9** Δ+4.5	**72.2** Δ+2.2	**89.3** Δ+1.0	**48.0** Δ+0.8	**67.3** Δ+1.3
Medium	StatA	62.4	60.4	64.3	16.0	50.5	58.2	77.9	87.7	67.7	87.3	48.5	67.5
	JPTA (Ours)	**66.6** Δ+4.2	**61.8** Δ+1.4	**65.6** Δ+1.3	**17.5** Δ+1.5	**70.8** Δ+20.3	**60.6** Δ+2.4	**79.6** Δ+1.7	**89.7** Δ+2.0	**75.3** Δ+7.6	**90.4** Δ+3.1	**51.7** Δ+3.2	**69.5** Δ+2.0

(b) ViT-L/14

K_{eff}	Method	**Avg**	ImageNet	SUN397	Aircraft	EuroSAT	Cars	Food101	Pets	Flowers	Caltech	DTD	UCF101
	CLIP	72.6	73.5	67.7	32.5	60.3	76.9	90.9	93.5	79.5	95.2	53.5	74.9
Very Low	StatA	77.3	78.9	71.3	40.4	71.4	84.4	94.2	97.1	**82.9**	97.0	55.3	77.1
	JPTA (Ours)	**83.1** Δ+5.8	**88.5** Δ+9.6	**84.5** Δ+13.2	**42.8** Δ+2.4	**72.9** Δ+1.5	**86.7** Δ+2.3	**99.3** Δ+5.1	**98.8** Δ+1.7	81.1 Δ−1.8	**99.5** Δ+2.5	**72.4** Δ+17.1	**87.4** Δ+10.3
Low	StatA	76.3	77.2	70.9	36.8	71.2	82.0	92.3	94.3	81.9	95.3	58.7	78.8
	JPTA (Ours)	**79.8** Δ+3.5	**84.2** Δ+7.0	**80.1** Δ+9.2	**37.9** Δ+1.1	**81.1** Δ+9.9	**85.3** Δ+3.3	**94.3** Δ+2.0	**95.1** Δ+0.8	**82.8** Δ+0.9	**97.5** Δ+2.2	**59.6** Δ+0.9	**79.8** Δ+1.0
Medium	StatA	76.7	76.9	**72.8**	35.4	76.7	78.0	91.8	94.6	81.8	95.0	59.7	81.3
	JPTA (Ours)	**79.7** Δ+3.0	**77.9** Δ+1.0	71.0 Δ−1.8	**37.3** Δ+1.9	**86.9** Δ+10.2	**81.4** Δ+3.4	**92.2** Δ+0.4	**95.9** Δ+1.3	**88.2** Δ+6.4	**97.5** Δ+2.5	**65.2** Δ+5.5	**83.2** Δ+1.9

B.8 Transfer to OpenCLIP and SigLIP2

We extend the RN50 and ViT-L/14 analysis to OpenCLIP ViT-B/16 trained on DataComp-1B Gadre et al. [2023] and SigLIP2 So400m/14-384 Tschannen et al. [2025]. Each backbone is evaluated on all 11 benchmarks in Very-Low, Very-High, and All-Class regimes, giving 33 dataset–regime conditions per backbone and 66 additional evaluations. JPTA uses the configuration selected with the original CLIP ViT-B/16, without backbone-specific retuning. Every gain compares JPTA with the corresponding backbone's own zero-shot predictions under the same evaluation condition; it is not a comparison against zero-shot OpenAI CLIP.

Table 6 (main paper) summarizes the macro gains. OpenCLIP DataComp-1B improves in 33/33 conditions by 6.13 pp overall; SigLIP2 improves in 29/33 by 3.71 pp. The Very-Low gains are 10.46 and 5.96 pp, respectively. In All-Class evaluation, each backbone improves on all 11 datasets, by 4.06 and 2.89 pp on average. SigLIP2's mean zero-shot accuracy across the 33 conditions is already 81.95%, yet useful adaptation headroom remains. The four SigLIP2 regressions occur in sampled regimes, and gains tend to narrow when the initial accuracy is already high. These comparisons establish benefit over the same-backbone zero-shot reference, without asserting superiority over untested modern-backbone adaptation baselines or predicting the behavior of future VLMs.

In the Very-Low regime, both backbones improve over their respective zero-shot baselines on FGVC-Aircraft, Stanford Cars, and Oxford Pets (Table 14).

Table 14: **Fine-grained Very-Low gains with modern backbones.** Accuracy improvements (pp) are relative to the same-backbone zero-shot model, using the transferred JPTA configuration.

Dataset	OpenCLIP DataComp-1B	SigLIP2 So400m/14-384
FGVC-Aircraft	+9.22	+8.57
Stanford Cars	+7.67	+2.71
Oxford Pets	+5.47	+2.12

These fine-grained results complement the Stanford Cars case study in Appx. C.3: visually similar but absent classes can dilute support even when the backbone is strong. They do not imply recovery when the initial semantic correspondence is uninformative; Appx. C.7 documents that boundary.

B.9 Source-to-Target Transfer under Natural Domain Shift

We select the complete configuration on ImageNet and transfer it unchanged to ImageNet-A Hendrycks et al. [2021b], ImageNet-R Hendrycks et al. [2021a], ImageNet-Sketch Wang et al. [2019], and ImageNet-V2 Recht et al. [2019], without using target-domain validation labels. Table 15 reports macro accuracy across the four target datasets. JPTA outperforms CLIP and StatA on every shifted dataset in both Very-Low and High coverage; the paired 95% confidence intervals for these comparisons exclude zero. It improves over CLIP on 87.1% of Very-Low tasks and 95.9% of High-coverage tasks.

Table 15: **Natural domain shift with frozen ImageNet-selected hyperparameters.** Entries are macro top-1 accuracy (%) over ImageNet-A/R/Sketch/V2. The best method in each row is bold.

Regime	Zero-shot CLIP	StatA	JPTA
Very Low	59.51	63.46	**79.46**
High	59.71	62.01	**69.33**

The shifted All-Class result is more modest: JPTA improves over CLIP by 0.67 pp on average but trails TransCLIP by 0.70 pp. Thus, the strongest gains occur under partial class coverage; favorable all-class conditions do not guarantee that JPTA is the best adaptation method on every domain.

B.10 Paired Statistical Analysis

We rerun CLIP, StatA, TransCLIP, and JPTA on identical task manifests and compute paired statistics from task-level or image-level predictions. These are additional paired reruns; their precise gains need not equal differences computed from rounded entries in the original main tables. We distinguish a fixed comparator from a dataset-wise strongest comparator. In Table 16, the latter is the best of CLIP, StatA, and TransCLIP by mean accuracy within each dataset and regime; it is not a different comparator selected for each test task.

For regime-level macro inference, each of the 11 datasets contributes one paired mean accuracy difference. The confidence intervals, paired t-tests, and Wilcoxon signed-rank tests in Table 16 use these 11 dataset-level differences. We do not pool 11,000 sampled tasks as independent observations for this macro analysis. Separately, within each sampled dataset we analyze its 1,000 paired task differences and apply Holm correction across the 11 dataset-wise primary comparisons. All-Class within-dataset comparisons use paired per-image predictions and exact McNemar tests, again with Holm correction across datasets.

Table 16: **Paired macro gains over the dataset-wise strongest comparator.** Gains and 95% CIs are in pp. Macro tests use 11 dataset-level differences; the final column counts datasets with positive mean gains.

Regime	Gain	95% CI	Paired t: p	Wilcoxon: p	Positive
Low	+9.871	[5.775, 13.967]	0.0003	0.0010	11/11
Very High	+2.306	[0.448, 4.164]	0.0199	0.0049	10/11
All-Class	+2.714	[1.546, 3.882]	0.0004	0.0010	11/11

Using a single comparator for the regime, the paired reruns give +2.92 pp over StatA in Very High (95% CI [0.82, 5.02]) and +2.98 pp over TransCLIP in All-Class ([1.83, 4.12]). These differ from the stricter dataset-wise comparison above, which allows the strongest comparator to vary by dataset.

In Low coverage, all 11 dataset-wise gains remain significant after Holm correction, ranging from +3.50 pp on Oxford Pets to +19.58 pp on SUN397. In Very High, 10/11 datasets improve after correction; Stanford Cars is 0.472 pp below StatA, with paired effect size $d_z = -0.254$. In All-Class, all 11 improvements remain significant after Holm-corrected exact McNemar tests, including Food101 (+0.406 pp) and Caltech101 (+0.771 pp). Statistical significance does not imply uniform benefit on individual tasks; task-level harm is examined in Appx. C.7.

B.11 Reproducibility

All experiments use fixed random seeds for batch sampling and solver initialization. For each dataset and regime, all methods evaluate the same batches. Image features and text prototypes are computed once and reused across baselines. For each run, we log the CLIP backbone, prompt template, dataset split, batch size, effective-class range, random seed, and solver hyperparameters.

JPTA adds no trainable parameters to the frozen VLM and needs no test-time gradient updates. Its extra cost comes from batch-level graph construction, soft prototype computation, anchor transport, and iterative assignment refinement. Because all operations occur in the cached CLIP feature space, this overhead is small compared to feature extraction. Runtime and memory results appear in Table 17.

Adaptation cost. JPTA runs after the shared CLIP image and text encoders. To isolate the adaptation cost, we time JPTA on cached ViT-B/16 image features and cached text anchors, using the same sampled-batch protocol as the main experiments. As shown in Table 17, JPTA takes about 6.6–7.5 ms/image for sparse B=64 batches and amortizes to about 0.54 ms/image at B=1000 (about 1.8k adapted images/second). Thus, the transported-anchor and controller computations add only a small post-feature cost and scale well with batch size.

Table 17: **JPTA adaptation cost across batch regimes.** Timing uses cached ViT-B/16 image features and cached text anchors, excludes the shared CLIP forward pass, and reports median wall-clock time over sampled batches on a single RTX 4090 Laptop GPU. Peak memory reports the maximum GPU memory allocated during adaptation.

Regime	Batch time (ms)	ms/image	Throughput	Peak mem.
1–4, B=64	419.9	6.561	152 img/s	115.1 MB
2–10, B=64	445.2	6.956	144 img/s	115.1 MB
5–25, B=64	482.2	7.534	133 img/s	115.1 MB
5–25, B=1000	535.2	0.535	1868 img/s	358.0 MB
25–50, B=1000	545.8	0.546	1832 img/s	412.7 MB
50–100, B=1000	543.1	0.543	1841 img/s	414.5 MB

Existing assets. All experiments use publicly available datasets and pretrained CLIP models. We cite the original papers for ImageNet, Caltech101, OxfordPets, StanfordCars, Flowers102, Food101, FGVC-Aircraft, SUN397, DTD, EuroSAT, UCF101, and CLIP. We do not redistribute these datasets; users must obtain them and the pretrained models from their official sources.

B.12 Large-Batch Sinkhorn and Graph-Update Scaling

We profile cached ImageNet-1K CLIP features while retaining all $K = 1000$ prototypes, $k = 3$, and $T_{\max} = 10$. To isolate the expensive branch, we enable exact Sinkhorn normalization at every assignment update and exercise the full iteration budget. Each run performs 55 Sinkhorn normalizations and 55 sparse graph-propagation updates. Table 18 is therefore a stress test of the fully activated branch, distinct from the evidence-routed timing in Table 17.

Table 18: **Forced exact-Sinkhorn scaling on ImageNet-1K features.** Times are medians of five measured runs after three warm-ups on an NVIDIA RTX 4090 Laptop GPU. Shared feature extraction is excluded; all 1,000 class prototypes are retained.

Batch size	Sinkhorn (s)	Graph updates (s)	Total JPTA (s)	ms/image
1,000	0.349	0.046	0.742	0.742
2,000	0.364	0.052	0.756	0.378
4,000	0.415	0.073	0.848	0.212
8,000	0.628	0.115	1.184	0.148
16,000	2.279	0.241	3.129	0.196

At $B = 16{,}000$, Sinkhorn and iterative graph propagation together account for 80.6% of total adaptation time. One Sinkhorn normalization of the $B \times K$ assignment matrix costs $O(T_{\mathrm{SK}}BK)$ time and $O(BK)$ working memory, with $T_{\mathrm{SK}} = 25$. Given a sparse k-NN graph with $O(Bk)$

edges, one propagation update costs $O(BkK)$. This propagation bound assumes the graph is already constructed and is not a bound on neighbor search or the complete solver. Normal execution may use the softer KL marginal penalty or terminate earlier, so these measurements characterize the forced branch on the stated hardware rather than universal end-to-end latency.

C Additional Ablations and Qualitative Analysis

This appendix provides full ablation results supporting Sec. 4. The main paper gives only a compact regime-level summary for space. Here, we present per-dataset component ablations, a mechanism audit of the anchor and statistic-update design, and qualitative case studies showing how JPTA corrects effective-class mismatch in sparse and fine-grained batches. Unless noted, all ablations reuse the hyperparameters of Full JPTA, and batch-regime comparisons are paired over the same sampled transductive tasks.

C.1 Full Component Ablations

Table 19 presents per-dataset component ablations on FGVC-Aircraft, EuroSAT, and Oxford Flowers. These datasets stress distinct failure modes: fine-grained ambiguity, domain-specific label structure, and visually confusable classes. We test each ablation in sparse small-batch, higher-coverage large-batch, and all-class regimes.

The results show that the transported semantic anchor and regime-adaptive control are essential in sparse regimes. Removing the transported anchor reduces JPTA to a fixed-anchor variant and sharply degrades performance on all datasets, especially EuroSAT and Oxford Flowers. Disabling adaptive control is similarly harmful in low-coverage batches, indicating that fixed transport and anchor weights fail when batch evidence varies across tasks. In contrast, prior-aware refinement has regime-dependent impact: it is crucial on EuroSAT, where class-support inflation is severe, but less important in some high-coverage fine-grained settings. Overall, the pattern supports JPTA as a selective adaptation framework rather than a uniformly stronger update rule.

Table 19: **Full component ablations for JPTA.** We isolate the contribution of transported semantic anchoring, prior-aware refinement, and regime-adaptive control on FGVC-Aircraft, EuroSAT, and Oxford Flowers across sparse, high-coverage, and all-class regimes. Each entry reports top-1 accuracy (%); values in parentheses denote the change relative to Full JPTA under the same dataset and regime. All ablations reuse the hyperparameters selected for the full model. Batch-regime comparisons are paired over the same 1,000 sampled tasks, while all-class rows use the full test split.

Dataset	Regime	Full JPTA	Fixed anchor (StatA-style)	w/o prior	w/o control
FGVC-Aircraft	Very low (1–4), B=64	**40.10**	29.36 (-10.74)	38.45 (-1.65)	27.40 (-12.70)
FGVC-Aircraft	Low (2–10), B=64	**36.24**	28.57 (-7.67)	32.72 (-3.52)	26.32 (-9.92)
FGVC-Aircraft	High (25–50), B=1000	**26.67**	21.54 (-5.13)	27.10 (+0.43)	23.97 (-2.70)
FGVC-Aircraft	All classes	**28.23**	18.23 (-10.00)	27.06 (-1.17)	25.68 (-2.55)
EuroSAT	Very low (1–4), B=64	**70.63**	57.47 (-13.16)	66.87 (-3.76)	64.40 (-6.23)
EuroSAT	Low (2–10), B=64	**66.61**	57.74 (-8.87)	37.87 (-28.74)	54.35 (-12.26)
EuroSAT	High (25–50), B=1000	**69.14**	59.10 (-10.04)	56.91 (-12.23)	67.65 (-1.49)
EuroSAT	All classes	**70.64**	61.69 (-8.95)	55.07 (-15.57)	65.12 (-5.52)
Oxford Flowers	Very low (1–4), B=64	**82.94**	67.58 (-15.36)	80.27 (-2.67)	77.59 (-5.35)
Oxford Flowers	Low (2–10), B=64	**80.61**	68.22 (-12.39)	77.21 (-3.40)	70.57 (-10.04)
Oxford Flowers	High (25–50), B=1000	**79.04**	66.98 (-12.06)	79.43 (+0.39)	76.72 (-2.32)
Oxford Flowers	All classes	**79.98**	66.23 (-13.75)	76.90 (-3.08)	74.26 (-5.72)

C.2 Anchor Blending and Anchored-Statistic Updates

Table 20 analyzes the mechanisms used to construct and stabilize the batch-aware anchor. We compare Full JPTA with three variants: *Original-only*, which keeps the fixed CLIP text anchor; *Transported-only*, which uses the transported anchor without adaptive blending; and *Fixed pseudocount*, which replaces the adaptive anchored-statistic schedule with a fixed one.

Original-only consistently underperforms Full JPTA, showing that fixed text semantics are too rigid when the batch provides reliable corrections. Transported-only performs substantially better than original-only but still lags Full JPTA, indicating that transport helps but unregulated replacement of

the pretrained anchor is less robust than controlled interpolation. Fixed-pseudocount yields smaller but consistent losses, suggesting that adaptive anchored statistics mainly improve stability across regimes. Overall, JPTA's benefit arises from combining transport, semantic preservation, and adaptive statistical trust, not from transport alone.

Table 20: **Anchor and statistic-update audit.** We compare Full JPTA with three mechanism variants: Original-only, which uses the fixed CLIP text anchor; Transported-only, which uses the transported anchor without adaptive blending; and Fixed pseudocount, which removes adaptive anchored-statistic scheduling. Each entry reports top-1 accuracy (%); values in parentheses denote the change relative to Full JPTA. The final rows report regime-level means over FGVC-Aircraft, EuroSAT, and Oxford Flowers.

Dataset	Regime	Full JPTA	Original-only (StatA-style)	Transported-only	Fixed pseudocount
FGVC-Aircraft	Very low (1–4), B=64	**40.10**	29.36 (-10.74)	36.42 (-3.68)	37.88 (-2.22)
FGVC-Aircraft	Low (2–10), B=64	**36.24**	28.57 (-7.67)	33.41 (-2.83)	34.52 (-1.72)
FGVC-Aircraft	High (25–50), B=1000	**26.67**	21.54 (-5.13)	25.42 (-1.25)	25.86 (-0.81)
FGVC-Aircraft	All classes	**28.23**	18.23 (-10.00)	25.86 (-2.37)	26.91 (-1.32)
EuroSAT	Very low (1–4), B=64	**70.63**	57.47 (-13.16)	66.12 (-4.51)	66.74 (-3.89)
EuroSAT	Low (2–10), B=64	**66.61**	57.74 (-8.87)	62.93 (-3.68)	59.82 (-6.79)
EuroSAT	High (25–50), B=1000	**69.14**	59.10 (-10.04)	65.41 (-3.73)	67.12 (-2.02)
EuroSAT	All classes	**70.64**	61.69 (-8.95)	66.88 (-3.76)	65.31 (-5.33)
Oxford Flowers	Very low (1–4), B=64	**82.94**	67.58 (-15.36)	76.93 (-6.01)	78.49 (-4.45)
Oxford Flowers	Low (2–10), B=64	**80.61**	68.22 (-12.39)	76.44 (-4.17)	77.09 (-3.52)
Oxford Flowers	High (25–50), B=1000	**79.04**	66.98 (-12.06)	75.20 (-3.84)	76.90 (-2.14)
Oxford Flowers	All classes	**79.98**	66.23 (-13.75)	76.10 (-3.88)	76.84 (-3.14)
Mean	Very low (1–4), B=64	**64.56**	51.47 (-13.09)	59.82 (-4.74)	61.04 (-3.52)
Mean	Low (2–10), B=64	**61.15**	51.51 (-9.64)	57.59 (-3.56)	57.14 (-4.01)
Mean	High (25–50), B=1000	**58.28**	49.21 (-9.07)	55.34 (-2.94)	56.63 (-1.65)
Mean	All classes	**59.62**	48.72 (-10.90)	56.28 (-3.34)	56.35 (-3.27)

Note: Original-only uses the fixed CLIP text anchor. Transported-only uses the transported anchor without adaptive blending. Fixed pseudocount replaces the adaptive anchored-statistic schedule with a fixed one.

C.3 Fine-Grained Case Study on Stanford Cars

Figure 3 shows a low-coverage Stanford Cars batch with two fine-grained subtypes. This is challenging because visually similar but absent classes act as plausible distractors. CLIP spreads probability mass over these distractors, and StatA only partly corrects this. JPTA instead reallocates mass to the realized classes—especially the weaker true subtype—while suppressing absent alternatives. This shows that statistic-only adaptation is insufficient in fine-grained regimes: the semantic anchor must move toward the observed batch geometry, but under controlled trust.

C.4 Evidence-Conditioned Mechanism Audit

Figure 4 examines when each adaptive component matters by grouping paired transductive tasks into diagnostic quintiles and measuring accuracy loss from ablating each component. Prior-aware refinement helps most when the estimated batch marginal is reliable and class coverage is broad, while adaptive control and adaptive pseudocounts matter most for low-evidence batches where fixed schedules are unstable. This shows that JPTA does not merely strengthen transduction, but adapts selectively to batch evidence.

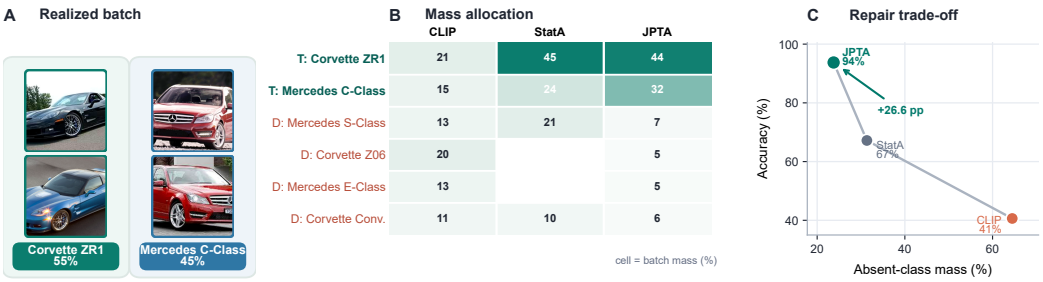


Figure 3: **Fine-grained support dilution explains why statistic-only adaptation is insufficient.** **A:** A low-coverage Stanford Cars batch contains only two realized subtypes: Corvette ZR1 (green) and Mercedes C-Class (blue). **B:** Heatmap entries show batch-averaged class probability mass. True batch labels are marked in green and absent distractor labels in orange. CLIP spreads probability over visually plausible but absent fine-grained labels; StatA increases mass on one true class but still assigns substantial mass to distractors. JPTA reallocates mass toward the realized support, especially the second true subtype, while suppressing absent alternatives. **C:** This yields the desired repair trade-off: JPTA moves to lower absent-class mass and higher accuracy, improving over StatA by 26.6 percentage points on the same batch.

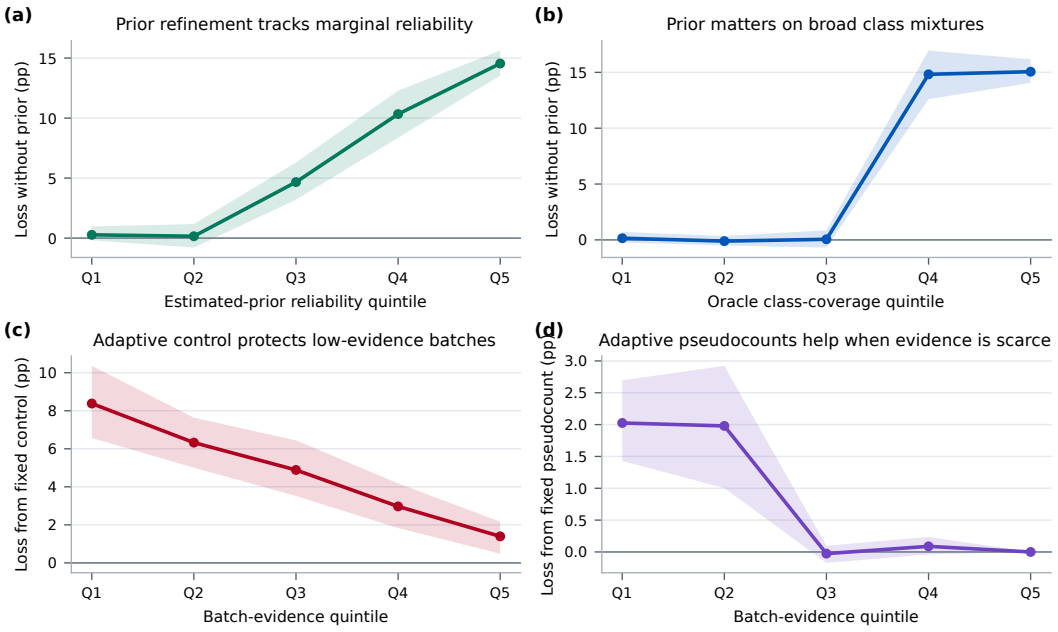


Figure 4: **Evidence-conditioned ablation analysis.** We sort paired transductive tasks into quintiles using diagnostics computed by Full JPTA and report the accuracy loss caused by each ablation within each quintile. Prior-aware refinement is most valuable when the estimated batch marginal is reliable and class coverage is broader. Adaptive control and adaptive pseudocounts matter most in low-evidence batches, where fixed schedules are more prone to drift. Bands show bootstrap 95% confidence intervals over paired task batches.

C.5 Oracle diagnostics.

Table 21 uses label-informed oracles only as diagnostics, with all other hyperparameters fixed. Oracle support and oracle prior yield substantial gains, especially in low-coverage regimes, while oracle trust is essentially identical to Full JPTA. Thus, remaining headroom is not due to a poorly tuned controller: the learned trust schedule already matches its label-informed counterpart. The main bottleneck is instead estimating batch support and class marginals from unlabeled test features—the support-mismatch problem JPTA targets.

Table 21: **Oracle diagnostics for JPTA.** Results are regime-level means over FGVC-Aircraft, EuroSAT, and Oxford Flowers. Accuracies are percentages; values in parentheses denote the change relative to Full JPTA in the same regime. Oracle variants use label information only for analysis, with all other hyperparameters fixed: oracle prior gives the true batch class marginal, oracle support gives the set of present classes without per-sample labels, and oracle trust uses labels only to choose the batch-level trust schedule. The large oracle-prior and oracle-support gains show that JPTA is mainly limited by estimating batch support and marginals, while the near-zero oracle-trust gap indicates that the learned controller is already close to its label-informed diagnostic schedule.

Regime	ZS	Full JPTA	+ oracle prior	+ oracle support	+ oracle trust
Very low (1–4), B=64	50.21	**64.19**	72.66 (+8.47)	81.90 (+17.71)	64.15 (-0.04)
High (25–50), B=1000	49.48	**58.24**	65.01 (+6.77)	68.44 (+10.20)	58.22 (-0.02)
All classes	49.68	**60.13**	65.89 (+5.76)	60.13 (0.00)	60.11 (-0.02)
Overall mean	49.79	**60.85**	67.85 (+7.00)	73.49 (+12.64)	60.83 (-0.03)

C.6 Transport geometry.

Table 22 shows that transported anchors help only when combined with JPTA's reliability controls. Fixed text anchors remove semantic transport and perform well below Full JPTA, while direct image prototypes, unweighted Procrustes, and unconstrained linear maps often over-transport and hurt accuracy. Weighted Procrustes is the strongest geometry-only variant but still lags Full JPTA, especially in sparse batches. This supports the central design: JPTA does not simply replace CLIP text anchors with batch-derived prototypes, but uses transported semantic anchors with weighting, shrinkage, prior refinement, and adaptive control.

Table 22: **Transport-geometry audit for JPTA.** Results are regime-level means over FGVC-Aircraft, EuroSAT, and Oxford Flowers. Accuracies are percentages; values in parentheses denote the change relative to Full JPTA. The audit separates transport geometry from JPTA's weighting, shrinkage, prior refinement, and adaptive controller. Fixed text anchors underperform Full JPTA, while direct image prototypes, unweighted Procrustes, and unconstrained maps often over-transport and degrade accuracy. Full JPTA is strongest because transported semantic anchors are used through a weighted, shrunk, and prior-controlled update rather than as an ungated replacement for the text geometry.

Regime	Full JPTA	Fixed text	Image proto.	Unweighted Proc.	Weighted Proc.	Linear map
Very low (1–4), B=64	**64.19**	45.19 (-19.00)	44.23 (-19.96)	49.16 (-15.03)	54.47 (-9.72)	48.95 (-15.24)
High (25–50), B=1000	**58.24**	48.20 (-10.04)	54.82 (-3.42)	53.66 (-4.58)	56.02 (-2.22)	55.14 (-3.10)
All classes	**60.13**	50.15 (-9.98)	58.43 (-1.70)	58.64 (-1.49)	58.86 (-1.27)	58.42 (-1.71)
Overall mean	**60.85**	47.85 (-13.01)	52.49 (-8.36)	53.82 (-7.03)	56.45 (-4.40)	54.17 (-6.68)

C.7 Initialization Robustness and Negative Adaptation

We test dependence on initial predictions using controlled affinity corruption and post-hoc initial-quality stratification. These complement natural domain shift in Appx. B.9. The studies measure average robustness and observed failures; neither supplies a guarantee of safe adaptation.

Controlled affinity corruption. For FGVC-Aircraft, EuroSAT, and ImageNet, we use 1,000 paired tasks per dataset. For nested fractions $\rho \in \{0.1, 0.2, 0.4\}$ of samples, we exchange the initial top-1 probability with the probability of the most semantically similar incorrect class. This intervenes on

initialization, rather than adding image-space adversarial perturbations. Degradation is the paired within-method difference $\Delta_{\text{noise}}(\rho) = \text{Acc}(0) - \text{Acc}(\rho)$. Table 23 reports degradation and the fraction of tasks on which JPTA underperforms the corrupted zero-shot baseline.

Table 23: **Degradation under controlled initialization noise.** Values are mean accuracy losses (pp) from each method's own clean run over the three datasets; lower is better. The final column reports the percentage of tasks where JPTA is strictly worse than the corresponding corrupted zero-shot initialization.

Corrupted fraction	CLIP loss	StatA loss	JPTA loss	JPTA harm rate
10%	3.60	2.20	**0.22**	15.4%
20%	7.15	5.32	**0.76**	14.1%
40%	14.81	14.51	**4.51**	15.0%

JPTA has the smallest degradation in all nine dataset–noise comparisons. At 40% corruption, its mean loss is 10.30 pp smaller than CLIP's and 10.00 pp smaller than StatA's. On Aircraft, the corresponding losses are 2.69, 5.28, and 5.91 pp for JPTA, CLIP, and StatA, respectively. The approximately stable task-level harm rates indicate no monotonic increase under this intervention, but remain nonzero. Losses relative to each method's clean run measure sensitivity, not absolute rankings of corrupted accuracy. Batch-level soft evidence, neighborhood consistency, prior refinement, and evidence-dependent shrinkage provide a plausible explanation for the observed mitigation, rather than an adversarial-robustness guarantee.

Initial-quality stratification. We divide each dataset's 1,000 Very-Low tasks into quartiles of initial zero-shot accuracy, giving 11,000 tasks across 11 datasets. Q1 denotes the lowest-quality quartile and Q4 the highest. The ground-truth labels used to form these quartiles are available only for post-hoc diagnosis, never to JPTA at inference. Macro gains over zero-shot CLIP are +15.16, +15.67, +14.47, and +9.04 pp from Q1 to Q4. In Q1, macro accuracy increases from 44.18% to 59.34%, with positive mean gains on 10/11 datasets. Table 24 compares Q1 and Q4 in terms of gain, transport trust, anchor displacement, and harm rate.

Table 24: **Observed reliability control and failure boundary.** Q1/Q4 are dataset-specific initial-accuracy quartiles. Gains are macro accuracy changes (pp); harm is the strict task-level rate of underperforming zero-shot CLIP. Displacement is the reported weighted angular anchor diagnostic.

Initial quality	Gain over CLIP	Transport trust	Displacement	Harm rate
Q1 (lowest)	+15.16	0.477	0.029°	22.38%
Q4 (highest)	+9.04	0.619	0.041°	3.87%

Lower-quality initialization is accompanied by lower transport trust and smaller weighted anchor movement, consistent with more conservative adaptation. Nevertheless, harm increases substantially in Q1. The extreme FGVC-Aircraft Q1 stratum starts at only 5.01% zero-shot accuracy and loses another 1.47 pp under JPTA. Thus, positive macro gains can coexist with harmful individual tasks or a negative dataset stratum. JPTA limits error amplification on average but requires minimally informative semantic features; trust and shrinkage mitigate risk without explicitly detecting or eliminating negative adaptation.

D Scope and Limitations

JPTA targets closed-set transductive adaptation, where the label space is fixed and each unlabeled batch is informative enough to reveal local class structure. While the regime-adaptive trust mechanism limits drift on sparse or ambiguous batches, gains remain small when batches have very weak visual evidence, strong domain shift, or many visually similar classes. In these cases, JPTA stays close to the original CLIP anchor instead of aggressively adapting, improving stability but limiting correction.

The method also assumes that class names and text prototypes offer a meaningful semantic prior. If prompts are poorly specified, labels overlap semantically, or the test distribution includes unknown

classes, the transported anchor may not resolve ambiguity. Extending JPTA to open-set, open-vocabulary, or growing label spaces is an important future direction.

Finally, JPTA uses cached frozen VLM features and no gradient-based adaptation. This yields efficient inference and avoids test-time fine-tuning, but representation failures of the frozen backbone cannot be fixed directly. Future work could pair batch-aware semantic anchoring with lightweight feature adaptation, better uncertainty estimation, or privacy-preserving batch statistics for sensitive deployments.

Missing labels and negative adaptation. The fixed candidate vocabulary must contain the true class. If a label is absent, the present closed-set assignment cannot output it; unknown detection and dynamic label expansion require additional mechanisms and are not evaluated here. JPTA could follow a candidate-retrieval stage, but that extension is not established by the present experiments. Appx. C.7 additionally shows that conservative transport is not a guarantee of non-negative adaptation: nearly uninformative semantic evidence can still lead to accuracy loss. Conversely, stronger initial models reduce headroom without removing the gains observed for the tested OpenCLIP and SigLIP2 checkpoints (Appx. B.8).

Broader impact. This work introduces a method for robust closed-set test-time adaptation of vision–language classifiers. Its main benefit is improved robustness when labels are unavailable and test batches have shifted class composition. Potential harms stem from wider use of stronger visual recognition, including unvalidated deployment in sensitive classification or surveillance contexts. We do not introduce new datasets, collect human-subject data, or deploy the method. Any practical use should include domain-specific validation, privacy-preserving handling of batch statistics, and checks for bias and open-set failures.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes].

   Justification: The abstract and introduction state the scope as closed-set, unlabeled batch-transductive adaptation for VLMs, and the main claims are evaluated in Sec. 4 across sparse, high-coverage, all-class, and online regimes. The limitations in Appendix D further qualify the expected scope of generalization.

2. **Limitations**

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

   Answer: [Yes].

   Justification: Appendix D discusses the closed-set assumption, dependence on informative batches and meaningful text prototypes, limited gains under severe domain shift or weak visual evidence, and the inability to correct representation-level failures of the frozen backbone.

3. **Theory assumptions and proofs**

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

   Answer: [Yes].

   Justification: Proposition 1 states the required angular assumptions for shrinkage-controlled transport in Sec. 3.3, and Appendix A provides the proof.

4. **Experimental result reproducibility**

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

   Answer: [Yes].

   Justification: Sec. 4 defines the evaluation setting, datasets, baselines, and batch regimes, while Appendix B gives the experimental protocol, feature extraction, batch construction, implementation details, hyperparameter selection, and reproducibility controls.

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

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

   Answer: [No].

   Justification: The paper uses publicly available datasets and pretrained CLIP models and provides detailed experimental protocols in Appendix B, but an anonymized code release is not included with the initial submission.

6. **Experimental setting/details**

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

   Answer: [Yes].

   Justification: Sec. 4 and Appendix B specify datasets, frozen CLIP features, prompt construction, batch regimes, baselines, random seeds, and hyperparameter selection. JPTA is a gradient-free test-time solver and therefore does not use a training optimizer.

7. **Experiment statistical significance**

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

   Answer: [No].

Justification: The main tables report averages over repeated sampled tasks using paired batches across methods, but most main-table entries do not include error bars due to space. Figure 4 reports bootstrap 95% confidence intervals for the evidence-conditioned ablation analysis.

8. **Experiments compute resources**

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

   Answer: [Yes].

   Justification: Appendix B.11 and Table 17 report the cached-feature evaluation setting and adaptation runtime on an RTX 4090 Laptop GPU. The method operates on cached CLIP features without gradient updates, and the reported timing isolates the post-feature adaptation cost.

9. **Code of ethics**

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

   Answer: [Yes].

   Justification: The work uses public benchmark datasets and pretrained models, does not collect human-subject data, and does not introduce deployed systems or high-risk released models. The authors have reviewed the NeurIPS Code of Ethics and identify no deviations.

10. **Broader impacts**

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

    Answer: [Yes].

    Justification: Appendix D discusses the scope of closed-set VLM adaptation and its possible deployment risks, including use in sensitive visual recognition settings, privacy concerns around batch statistics, and the need for validation under bias, domain shift, and open-set failures.

11. **Safeguards**

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

    Answer: [N/A].

    Justification: The paper does not release a new pretrained model, scraped dataset, image generator, or other high-risk asset. The method is evaluated on existing public classification benchmarks under a research protocol.

12. **Licenses for existing assets**

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

    Answer: [Yes].

    Justification: The paper cites the original sources for CLIP and all benchmark datasets and states that datasets and pretrained models are not redistributed. Users are directed to obtain each asset from its official source and comply with the corresponding license and terms of use.

13. **New assets**

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

    Answer: [N/A].

Justification: The paper does not introduce a new dataset, benchmark, pretrained model, or human-subject asset. The contribution is an algorithmic test-time adaptation method evaluated on existing public assets.

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

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

    Answer: [N/A].

    Justification: The paper does not involve crowdsourcing, user studies, annotation tasks, or other research with human subjects.

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

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

    Answer: [N/A].

    Justification: The paper does not involve human-subject experiments, crowdsourcing, or collection of participant data; therefore IRB approval is not applicable.

16. **Declaration of LLM usage**

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

    Answer: [N/A].

    Justification: LLMs are not used as an important, original, or non-standard component of the proposed method, experiments, baselines, or evaluation protocol.