SHARP: SPHERICAL-HARMONIC-INSPIRED DISTRIBUTION MODELING AND PROTOTYPE RECALIBRATION FOR TEST-TIME ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Online test-time adaptation (TTA) adapts pre-trained vision-language models (VLMs) to distribution shifts using only an unlabeled test stream. Cache-based methods retain past test samples as instance-level evidence, but a finite cache covers only a limited portion of the target distribution. Class-conditional distributions provide a complementary global view, yet their statistics are hard to estimate from few observations in the high-dimensional CLIP embedding space, especially covariance estimation. We propose SHARP, which estimates class-wise distributions in a *compact angular feature space*. Motivated by the hyperspherical geometry of normalized CLIP embeddings, SHARP builds a spherical-harmonic-inspired representation from learnable directional projections and low-order angular functions, reducing the number of covariance parameters and improving estimation reliability. The resulting class statistics drive both distribution-aware scoring and variance-adaptive prototype recalibration. SHARP further combines these statistics with instance-level evidence from multiple caches and applies lightweight residual adaptation to the classifier and cache features while keeping the CLIP encoders frozen. Across 10 cross-dataset benchmarks, SHARP improves average accuracy over the strongest baseline by 0.97% and 1.16% with ResNet-50 and ViT-B/16, respectively. On ImageNet and its four distribution-shift variants, the corresponding gains are 0.94% and 0.95%.

1 INTRODUCTION

Pre-trained vision-language models (VLMs), such as CLIP (Radford et al., 2021), enable zero-shot recognition by aligning image and text representations in a shared embedding space. Although these representations transfer well across a range of downstream tasks (Zhou et al., 2022b; Khattak et al., 2023b), their performance can degrade when the target distribution differs from the pre-training distribution (Radford et al., 2021; Shu et al., 2022). Test-time adaptation (TTA) mitigates this degradation by exploiting unlabeled target data at inference time (Shu et al., 2022; Karmanov et al., 2024). We focus on source-free online TTA, where test samples arrive sequentially, each must be predicted upon arrival, and neither source data nor target labels are available.

Online adaptation depends strongly on how information from the observed test stream is retained. Cache-based methods store representative test features and refine predictions through instance-level retrieval (Karmanov et al., 2024; Chen et al., 2025). Such local evidence is useful, but a finite cache covers only part of the target distribution. Distribution-based methods instead summarize the stream with class-wise statistics (Guan et al., 2025; Han et al., 2025), capturing variation beyond individual cached samples. Estimating these statistics, however, is demanding in the high-dimensional CLIP space: for a d-dimensional embedding, a full covariance contains $d(d+1)/2$ free entries, while only a small number of observations may be available for each class, particularly early in the stream. This creates a substantial online estimation burden and makes covariance estimation sensitive to limited and noisy pseudo-labeled samples.

We propose **SHARP**, an online TTA framework that estimates class-wise distributions in a compact angular space and uses the resulting covariance statistics for prototype adaptation. Unlike DOTA (Han et al., 2025), which models class distributions directly in the original embedding space

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

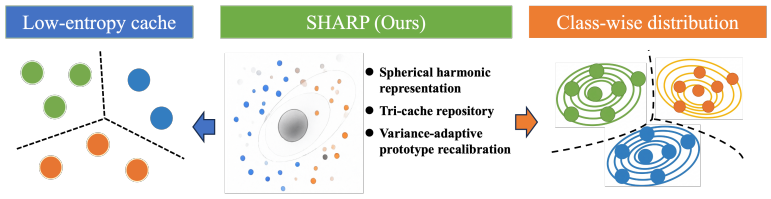

Figure 1: Motivation of SHARP. Caches provide local evidence with limited coverage, whereas class-wise statistics summarize broader variation but are costly to estimate in the original CLIP space. SHARP models class distributions in a compact angular space and uses the resulting statistics for both distribution-aware scoring and variance-adaptive prototype recalibration.

and uses them primarily for scoring, SHARP couples compact distribution modeling with variance-adaptive prototype recalibration (Figure 1).

Motivated by the hyperspherical geometry of normalized CLIP embeddings, SHARP applies learnable directional projections followed by low-order angular functions to construct a spherical-harmonic-inspired representation of dimension $L < d$. This reduces the number of covariance entries estimated from the test stream from $d(d+1)/2$ to $L(L+1)/2$. An isotropic prior and covariance shrinkage (Section 3.4) further regularize the online statistics and provide a positive eigenvalue floor for stable covariance inversion. The resulting class distributions yield a distribution-aware score that measures how well each test feature fits each class.

Beyond scoring, variance-adaptive prototype recalibration (VAPR) uses the class-wise variance and correlation structure to adjust the corresponding prototype. VAPR treats coordinates with large estimated variance or strong correlation with other coordinates as less reliable under noisy online updates and attenuates their contribution to the prototype.

To retain instance-level evidence, SHARP builds on the multi-cache framework of MCP (Chen et al., 2025) and adapts its entropy and negative-cache construction into a confidence and a hard-negative cache that emphasizes ambiguous samples near class boundaries. Following the residual adaptation strategy of MCP++ (Chen et al., 2025), lightweight residuals update the classifier and cache features over the test stream, while both CLIP encoders remain frozen.

Our contributions are summarized as follows:

- We introduce a compact spherical-harmonic-inspired representation for online class-conditional distribution modeling. It reduces the number of covariance entries estimated from the test stream from $d(d+1)/2$ to $L(L+1)/2$, while an isotropic prior and covariance shrinkage regularize the resulting online statistics.
- We propose variance-adaptive prototype recalibration (VAPR), which uses class-wise variance and correlation to attenuate unreliable prototype coordinates, allowing the same distribution model to support both prediction and prototype adaptation.
- SHARP improves average accuracy over MCP, on which it builds, by 0.97%/1.16% (ResNet-50/ViT-B/16) on 10 cross-dataset benchmarks and by 0.94%/0.95% on ImageNet and four distribution-shift variants.

2 Related Work

Vision-language models and prompt adaptation. Vision-language models learn transferable representations from large-scale image–text data (Vasu et al., 2025; Li et al., 2025), with CLIP (Radford et al., 2021) providing a widely used foundation for recognition. Prompt learning adapts these models through learnable text prompts (Zhou et al., 2022b), image-conditioned prompts (Zhou et al., 2022a), and multimodal prompts (Khattak et al., 2023a), with self-regulation and consistency constraints used to improve generalization (Khattak et al., 2023b; Roy & Etemad, 2024). These approaches typically rely on labeled adaptation data. For unlabeled test inputs, TPT (Shu et al., 2022) optimizes prompts using augmented views, while DiffTPT (Feng et al., 2023) incorporates diffusion-based augmentation. SHARP adapts classifier and cache-feature residuals while keeping the CLIP encoders frozen, and retains cached examples and class statistics across the test stream.

Cache- and prototype-based test-time adaptation. Cache-based methods retain representative target features to refine subsequent predictions (Karmanov et al., 2024; Zhang et al., 2025a), while prototype adaptation summarizes target evidence in class representatives, as in DPE's dual-modality prototypes (Zhang et al., 2024a). Related approaches organize target knowledge through task-specific and shared components in HKPT (Zhang et al., 2025b), or shape-sensitive, style-insensitive factors in SSG (Zhou et al., 2025). SCA (Guan et al., 2025) summarizes feature statistics to reduce storage requirements. MCP (Chen et al., 2025) combines entropy, alignment, and negative caches; its MCP++ extension further optimizes visual and textual prototypes through residual learning and cross-modal alignment. Finite caches provide direct access to target examples, but their coverage depends on capacity, sample selection, and pseudo-label quality. SHARP builds on MCP's multi-cache framework, with additions centered on a spherical-harmonic-inspired representation, class-conditional distribution scoring, and covariance-guided prototype recalibration.

Distribution-based test-time adaptation. Distribution-based methods summarize target observations through statistical models, providing class-level information beyond individual cached examples. StatA (Zanella et al., 2025) considers test streams with varying numbers of effective classes, while DOTA (Han et al., 2025) estimates category-specific distributions online. SHARP focuses on the representation in which class statistics are estimated and how those statistics are used: it maintains class-conditional means and covariances in the proposed feature space, then uses the same covariance estimates for distribution-aware scoring and prototype recalibration alongside instance-level cache evidence. Class covariance also appears in semantic data augmentation (Wang et al., 2019b), where it models semantic variation for supervised training; SHARP estimates covariance from unlabeled target observations for adaptation at inference time.

3 METHOD

3.1 PROBLEM SETTING AND OVERVIEW

We adapt a pre-trained CLIP model online to an unlabeled test stream with K candidate classes, without source data or target labels. Let $\mathbf{x} \in \mathbb{R}^d$ be the ℓ_2-normalized image embedding of a test view and $\mathbf{t}_k \in \mathbb{R}^d$ the normalized embedding of the class-k prompt. Frozen CLIP gives logits $u_k(\mathbf{x}) = s_{\text{clip}}\, \mathbf{x}^\top \mathbf{t}_k$ with its learned scale s_{clip} and probabilities $p(\mathbf{x}) = \mathrm{softmax}(\mathbf{u}(\mathbf{x}))$; all pseudo-labels, cache decisions, and statistical weights use these frozen probabilities. For each sample, $y = \arg\max_k p_k(\mathbf{x})$ and $b = \arg\max_{k \neq y} p_k(\mathbf{x})$ denote the top two pseudo-classes, and $\mathrm{Norm}(\cdot) = (\cdot)/\max(\|\cdot\|_2, \delta)$ with $\delta = 10^{-8}$. SHARP keeps both encoders frozen and maintains two forms of target memory (Figure 2): three instance caches and per-class Gaussian statistics in a compact feature space. The estimated covariances drive distribution-aware scoring and VAPR, while residual parameters on the classifier and cache features are updated online. Algorithm 1 in Appendix A.4 lists one complete iteration, and Table 7 maps the defined parameters to their equations and values.

3.2 SPHERICAL-HARMONIC-INSPIRED REPRESENTATION

Normalized CLIP embeddings lie on the unit sphere, so class structure is angular. We map each embedding to L coordinates, each computed as a low-order angular function of the angle to a learned direction. For component $i \in \{1, \ldots, L\}$, a learnable projection $\mathbf{B}_i \in \mathbb{R}^{d \times d}$ gives

$$\mathbf{q}_i = \mathrm{Norm}(\mathbf{B}_i^\top \mathbf{x}), \qquad a_i = \mathbf{x}^\top \mathbf{q}_i \in [-1, 1]. \tag{1}$$

Each angular coefficient passes through one of three low-order angular functions and a learnable affine calibration, and the calibrated responses form the representation:

$$g_1(a) = |a|, \quad g_2(a) = a, \qquad g_3(a) = \tfrac{1}{2}\big(3a^2 - 1\big), \quad \tilde{a}_i = \omega_i\, g_{\ell_i}(a_i) + \varphi_i, \tag{2}$$

$$\mathbf{f}_{\text{sh}}(\mathbf{x}) = [\tilde{a}_1, \ldots, \tilde{a}_L]^\top \in \mathbb{R}^L, \tag{3}$$

where ω_i and φ_i are initialized to 1 and 0 so that calibration starts from the raw responses, and $\ell_i = 1 + ((i-1) \bmod 3)$ assigns the three functions cyclically. g_2 and g_3 are the Legendre polynomials of degree one and two, the zonal spherical harmonics on $\mathbb{S}^2$, and g_1 adds an even magnitude response (Appendix A.5). Before the stream, $\{\mathbf{B}_i, \omega_i, \varphi_i\}$ are fitted on the class-text embeddings alone by minimizing $K^{-2} \sum_{k,l} (\mathbf{c}_k^\top \mathbf{c}_l - \mathbf{t}_k^\top \mathbf{t}_l)^2$, where $\mathbf{c}_k = \mathrm{Norm}(\mathbf{f}_{\text{sh}}(\mathbf{t}_k))$ are the text anchors,

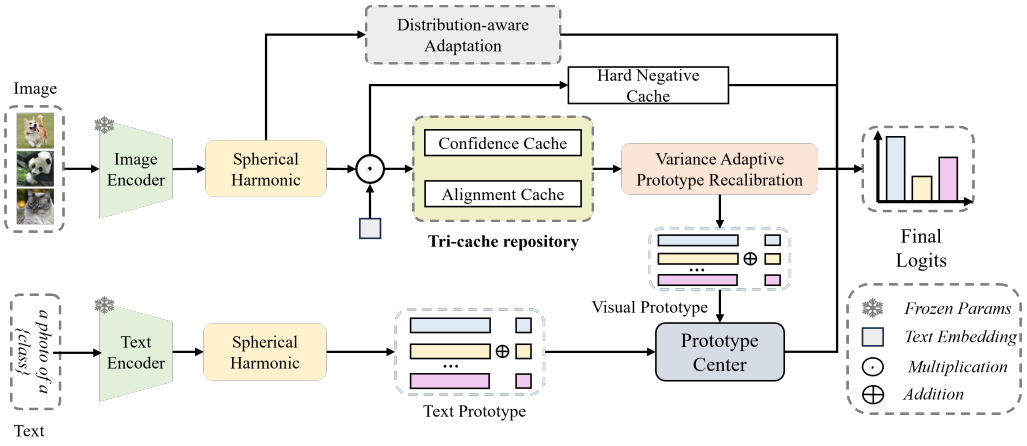

Figure 2: Overview of SHARP. Frozen CLIP encoders produce image and text features for a spherical-harmonic-inspired representation. Confidence, alignment, and hard-negative caches retain complementary target examples. Online class statistics guide distribution-aware scoring and variance-adaptive prototype recalibration; residual parameters are updated during adaptation.

and are then frozen. Because $\mathbf{q}_i$ depends on $\mathbf{x}$, the map is harmonic-inspired rather than a harmonic expansion, and it is even, $\mathbf{f}_{\mathrm{sh}}(-\mathbf{x}) = \mathbf{f}_{\mathrm{sh}}(\mathbf{x})$; the final prediction therefore also retains the CLIP logits (Section 3.6). Class means and covariances then have L and $L \times L$ entries instead of d and $d \times d$. Compactness refers to the output space and its statistics; the mapping itself uses full learnable matrices. Below we write $\mathbf{f} = \mathrm{Norm}(\mathbf{f}_{\mathrm{sh}}(\mathbf{x}))$ for the normalized representation, so that all inner products below are cosines, and use $\mathbf{c}_k$ as fixed class anchors.

3.3 Tri-cache target memory

Following MCP (Chen et al., 2025), SHARP keeps three class-specific caches $\mathcal{M}_C$, $\mathcal{M}_A$, and $\mathcal{M}_H$ for confident samples, anchor-aligned samples, and hard negatives, with a superscript indexing the class (e.g., $\mathcal{M}_C^k$); the first two (the positive caches) supply positive instance-level evidence and the third records competing classes. After a sample is predicted, its original-view feature $\mathbf{f}$ is offered to the caches of its pseudo-classes with the priorities below; a full cache keeps the entries with the smallest e, the largest s_A, or the largest h, respectively, and an image stored in several caches shares one feature and one residual. The positive-cache features of class k used below are the deduplicated union of $\mathcal{M}_C^k$ and $\mathcal{M}_A^k$, so a sample stored in both contributes once.

Confidence cache. Entries of $\mathcal{M}_C^y$ are ranked by low prediction entropy $H(p) = -\sum_k p_k \log p_k$:

$$e(\mathbf{x}) = \frac{H(\mathbf{p}(\mathbf{x}))}{\log K} + \min\left(\frac{\|\mathbf{f} - \mathbf{v}_k\|_2}{2}, 1\right). \tag{4}$$

Alignment cache. Entries of $\mathcal{M}_A^y$ are ranked by proximity to the fixed pseudo-class anchor,

$$s_A(\mathbf{x}) = -\|\mathbf{f} - \mathbf{c}_y\|_2, \tag{5}$$

so that the cache holds reference features close to the class anchor.

Hard-negative cache. $\mathcal{M}_H^b$ stores ambiguous samples for the competing pseudo-class b, ranked by the hardness score with mixing coefficient α,

$$h(\mathbf{x}) = \alpha(1 - \Delta p) + (1 - \alpha)p_{\mathrm{wrong}}, \qquad \Delta p = p_y - p_b, \quad p_{\mathrm{wrong}} = p_b. \tag{6}$$

Both terms use only frozen-CLIP probabilities: b is the runner-up pseudo-class, not a class known to be wrong, and the pair (y, b) is stored with the entry. Each hard-negative entry i receives the fixed retrieval weight

$$w_i = \exp(-\gamma e_i)\exp(-\lambda d_i)(1 + \zeta h_i), \tag{7}$$

where e_i, $d_i = \|\mathbf{f}_i - \mathbf{c}_{b_i}\|_2$, and h_i are its entropy, anchor distance, and hardness, and γ, λ, ζ are their coefficients; the exponential factors favor confident entries near the anchor, and the last factor raises the weight of ambiguous ones.

3.4 ONLINE CLASS-CONDITIONAL DISTRIBUTION ESTIMATION

Caches retain individual instances; a Gaussian model $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ per class in the compact feature space summarizes variation across all observations. Class k starts from a prior with pseudo-count $n_k = n_0$, mean $\boldsymbol{\mu}_k = \mathbf{c}_k$, and scatter $\mathbf{S}_k = n_0\sigma_0^2\mathbf{I}$, where n_0 is the prior weight, σ_0^2 the isotropic prior variance, and $\mathbf{I}$ the $L \times L$ identity. After each prediction, the original-view feature $\mathbf{f}$ updates every class with soft weight $\pi = p_k(\mathbf{x})$:

$$n_k^+ = n_k + \pi, \qquad \boldsymbol{\psi}_k = \mathbf{f} - \boldsymbol{\mu}_k, \qquad \boldsymbol{\mu}_k^+ = \boldsymbol{\mu}_k + \frac{\pi}{n_k^+}\boldsymbol{\psi}_k, \tag{8}$$

$$\mathbf{S}_k^+ = \mathbf{S}_k + \frac{n_k\pi}{n_k^+}\boldsymbol{\psi}_k\boldsymbol{\psi}_k^\top, \qquad \boldsymbol{\Sigma}_k = \mathbf{S}_k/n_k, \tag{9}$$

with right-hand sides evaluated at the pre-update state; the superscript $+$ denotes the state after the current sample is included, so the updated covariance is $\boldsymbol{\Sigma}_k^+ = \mathbf{S}_k^+/n_k^+$. Equations (8)–(9) give the exact weighted population statistics of the prior and all observations; the factor $n_k\pi/n_k^+$ is the mean-shift correction that a centered update with either the old or the new mean alone omits (proof in Appendix A.2). While only $J \ll L$ samples have been processed, the data part of the scatter, $\mathbf{S}_k - n_0\sigma_0^2\mathbf{I}$, has rank at most J; the variance assigned to the at least $L - J$ remaining directions is the diluted prior $n_0\sigma_0^2/n_k$, which decays as mass accumulates, so $\log|\boldsymbol{\Sigma}_k|$ can be dominated by these directions and the quadratic term amplifies any component outside the covered subspace (Appendix A.3); we therefore use one shrunk covariance, with shrinkage coefficient ϵ, in both the quadratic and the log-determinant term:

$$\tilde{\boldsymbol{\Sigma}}_k = (1-\epsilon)\boldsymbol{\Sigma}_k + \epsilon\mathbf{I}, \qquad \boldsymbol{\Lambda}_k = \tilde{\boldsymbol{\Sigma}}_k^{-1}, \tag{10}$$

$$s_k^D(\mathbf{f}) = -\frac{1}{2}(\mathbf{f} - \boldsymbol{\mu}_k)^\top\boldsymbol{\Lambda}_k(\mathbf{f} - \boldsymbol{\mu}_k) - \frac{1}{2}\log|\tilde{\boldsymbol{\Sigma}}_k|. \tag{11}$$

The quadratic term measures covariance-weighted deviation from the class mean, and the log-determinant term keeps classes with different spreads comparable; with the same matrix in both terms, s_k^D is the log-density of $\mathcal{N}(\boldsymbol{\mu}_k, \tilde{\boldsymbol{\Sigma}}_k)$ up to a constant (numerical details in Appendix A.3).

3.5 VARIANCE-ADAPTIVE PROTOTYPE RECALIBRATION

The distribution score uses the class covariance to evaluate a feature; VAPR uses the same estimate to reshape the class prototype. The prototype is $\mathbf{v}_k = \mathrm{Norm}(T_k^* + \mathbf{m}_k)$, with T_k^* the adapted text feature of Section 3.6 and $\mathbf{m}_k$ the normalized mean of the adapted positive-cache features of class k ($\mathbf{v}_k = T_k^*$ when they are empty). For each coordinate j, we use a variance-based reliability factor to downweight dimensions with large target variance $\sigma_{kj}^2 = (\boldsymbol{\Sigma}_k)_{jj}$, and a correlation gate, also a per-coordinate vector, with temperature τ_c attenuates coordinates that co-vary strongly with the rest, using the mean absolute correlation $\bar{r}_{kj} = \frac{1}{L-1}\sum_{l\neq j}|(\tilde{\boldsymbol{\Sigma}}_k)_{jl}|/\sqrt{(\tilde{\boldsymbol{\Sigma}}_k)_{jj}(\tilde{\boldsymbol{\Sigma}}_k)_{ll}}$:

$$[g_d(\boldsymbol{\Sigma}_k)]_j = \frac{1}{1+\sigma_{kj}^2}, \qquad [g_c(\tilde{\boldsymbol{\Sigma}}_k)]_j = \exp\Big(-\frac{\bar{r}_{kj}}{\tau_c}\Big). \tag{12}$$

The recalibrated prototype applies both gates and is normalized,

$$\mathbf{z}_k = \mathrm{Norm}\Big(\mathbf{v}_k \odot g_d(\boldsymbol{\Sigma}_k) \odot g_c(\tilde{\boldsymbol{\Sigma}}_k)\Big), \tag{13}$$

where $\odot$ denotes element-wise multiplication. Both gates can change the prototype direction before normalization and reduce to a constant for an isotropic covariance. The gates use variance and correlation as proxies for coordinate reliability. The statistics $\boldsymbol{\Sigma}_k$ are estimated from the raw features $\mathbf{f}$ and receive no gradients; they remain fixed during each residual optimization step and are updated after prediction. They therefore provide a reference for prototype recalibration, rather than estimating the covariance of the adapted features.

3.6 PREDICTION AND RESIDUAL TEST-TIME OPTIMIZATION

SHARP adapts the text classifier and the cached features through additive residuals,

$$T^* = \mathrm{Norm}(T + \Delta_T), \qquad M^* = \mathrm{Norm}(M + \Delta_M), \tag{14}$$

where the rows of T are the anchors $\mathbf{c}_k$, the rows of M are stored cache features, each residual matches its array in shape, and normalization is row-wise. Here $T, \Delta_T \in \mathbb{R}^{K\times L}$, $M, \Delta_M \in \mathbb{R}^{N_t\times L}$, and N_t counts the distinct cached samples; $\mathbf{c}_k$, T_k^*, $\mathbf{m}_k$, $\mathbf{v}_k$, $\mathbf{z}_k$ denote the fixed anchor, adapted text feature, positive-cache mean, fused prototype, and recalibrated prototype.

Prediction. For a query $\mathbf{f}$, let $V_k = \mathbf{f}^\top \mathbf{z}_k$, and let R_k^+ and R_k^- be the mean and w_i-weighted mean of the kernel $\kappa(\mathbf{f}, \mathbf{m}) = \exp[-\beta_r(1 - \mathbf{f}^\top \mathbf{m})]$ over the adapted positive-cache and hard-negative features of class k (zero when empty), where β_r sets the kernel sharpness; both averages are written out in Appendix A.1. With $[\mathcal{S}(\mathbf{a})]_k = (a_k - \bar{a})/\max(\sigma_\mathbf{a}, \delta_s)$ standardizing a K-vector, where $\bar{a} = K^{-1}\sum_j a_j$, $\sigma_\mathbf{a}^2 = K^{-1}\sum_j (a_j - \bar{a})^2$, and the floor $\delta_s > 0$ maps a constant vector (e.g., $\mathbf{R}^\pm = \mathbf{0}$ before any entry is cached) to $\mathbf{0}$, and $\mathbf{V}$, $\mathbf{R}^\pm$, $\mathbf{s}^D$ collecting the per-class scores, the logits for the t-th test sample are

$$\boldsymbol{\ell}_t = \mathcal{S}(\mathbf{u}) + \mathcal{S}(\mathbf{V}) + \lambda_{\text{cache}}\big[\mathcal{S}(\mathbf{R}^+) - \mathcal{S}(\mathbf{R}^-)\big] + \lambda_{\text{dist}}(t)\,\mathcal{S}(\mathbf{s}^D), \qquad \lambda_{\text{dist}}(t) = \eta\big(1 - \exp(-\rho t)\big), \tag{15}$$

where $\mathbf{u}$ are the frozen CLIP logits, λ_{cache} weights the cache evidence, and the schedule $\lambda_{\text{dist}}(t)$, with amplitude η and rate ρ, raises the distribution weight as statistics accumulate (Section 4.5). Equation (15) is evaluated for each view v of the sample, giving $Q_t^{(v)} = \text{softmax}(\boldsymbol{\ell}_t^{(v)})$; the output $\hat{y} = \arg\max_k \ell_{t,k}$ is taken on the original view.

Losses. Let $\mathcal{A}_t$ be the set of the $|\mathcal{A}_t|$ views with the lowest frozen-CLIP entropy $e(\mathbf{x})$ among all 32 views of the current sample, including the original view, $\bar{Q}_t = |\mathcal{A}_t|^{-1}\sum_{v\in\mathcal{A}_t} Q_t^{(v)}$ their mean prediction, $\mathcal{K}_+$ the classes with nonempty positive caches (the second term of (17) is omitted while $\mathcal{K}_+$ is empty), and $\mathcal{E}_t$ the hard-negative records $(\mathbf{m}, y, b)$, with $\mathbf{m}$ the adapted cache feature (a row of M^*), plus the record $(\mathbf{f}, y, b)$ of the current sample, formed from its original-view feature without a residual. The three terms are

$$\mathcal{L}_{\text{ent}} = -\sum\nolimits_k \bar{Q}_{t,k} \log \bar{Q}_{t,k}, \tag{16}$$

$$\mathcal{L}_{\text{align}} = \frac{1}{K}\sum\nolimits_k \big(1 - T_k^{*\top}\mathbf{c}_k\big) + \frac{1}{|\mathcal{K}_+|}\sum\nolimits_{k\in\mathcal{K}_+} \big(1 - \mathbf{m}_k^\top T_k^*\big), \tag{17}$$

$$\mathcal{L}_{\text{sep}} = \frac{1}{|\mathcal{E}_t|}\sum\nolimits_{(\mathbf{m},y,b)\in\mathcal{E}_t} \text{softplus}\Big(\frac{\mathbf{m}^\top\mathbf{z}_b - \mathbf{m}^\top\mathbf{z}_y + m_s}{\tau_s}\Big), \tag{18}$$

which sharpen predictions, anchor adapted features to CLIP semantics, and separate hard negatives from their competing prototype by margin m_s at temperature τ_s, where $\text{softplus}(a) = \log(1 + e^a)$. The residuals are optimized with weights λ_1 and λ_2,

$$\mathcal{L} = \mathcal{L}_{\text{ent}} + \lambda_1\mathcal{L}_{\text{align}} + \lambda_2\mathcal{L}_{\text{sep}}. \tag{19}$$

Online procedure. For each test sample we (i) compute frozen CLIP features and probabilities of its augmented views, (ii) build prototypes, gates, and scores from the memory of previous samples, (iii) take one AdamW step on the residuals with $\mathcal{L}$, (iv) predict the original view with the updated residuals, and (v) only then update the class statistics and caches with that sample. Residuals and optimizer state persist across samples; the encoders, projection, cache assignments, and statistics receive no gradients (Algorithm 1).

4 Experiments

4.1 Experimental setup

Implementation. We evaluate CLIP with ResNet-50 and ViT-B/16 image encoders (Radford et al., 2021), using the same adaptation configuration for both backbones. Following MCP (Chen et al., 2025), we generate 31 random-resized-crop augmentations per test image, giving 32 views including the original. One AdamW step per test sample updates the residuals, with representation dimension $L = 256$ and covariance shrinkage coefficient $\epsilon = 0.01$. Parameter values are listed in Appendix A.1; qualitative analysis is in Appendix B.

Datasets and metric. We report top-1 accuracy for cross-dataset adaptation and robustness to natural distribution shifts, following the evaluation setting of MCP (Chen et al., 2025). The cross-dataset suite contains 10 datasets spanning several recognition tasks. Caltech101 evaluates generic object recognition (Fei-Fei et al., 2004), while Oxford-Pets (Parkhi et al., 2012), StanfordCars (Krause et al., 2013), Flowers102 (Nilsback & Zisserman, 2008), Food101 (Bossard et al., 2014), and FGVCAircraft (Maji et al., 2013) evaluate fine-grained recognition. The remaining datasets are SUN397 for scenes (Xiao et al., 2010), UCF101 for actions (Soomro, 2012), DTD for textures (Cimpoi et al., 2014), and EuroSAT for satellite imagery (Helber et al., 2019). For distribution-shift evaluation, we use ImageNet (Deng et al., 2009) and four variants: ImageNet-V2 (Recht et al., 2019), ImageNet-Sketch (Wang et al., 2019a), ImageNet-A (Hendrycks et al., 2021b), and ImageNet-R (Hendrycks et al., 2021a). We report the unweighted cross-dataset average and, for ImageNet, the five-dataset and OOD (four shifted variants) averages.

Comparisons. We compare SHARP with zero-shot CLIP, prompt-learning, and recent TTA approaches. Prompt-learning baselines use labeled training data and serve as reference alongside the source-free TTA methods. For MCP, the main comparison tables use the published results, whereas the ablation study uses our re-implementation under the SHARP experimental configuration.

Table 1: Cross-dataset top-1 accuracy (%) with ResNet-50 and ViT-B/16. Bold indicates the best result within each backbone.

Method	Aircraft	Caltech	Cars	DTD	EuroSAT	Flowers	Food101	Pets	SUN397	UCF101	Average
CLIP-ResNet-50 (Radford et al., 2021)	15.66	85.88	55.70	40.37	23.69	61.75	73.97	83.57	58.80	58.84	55.82
CoOp (Zhou et al., 2022b)	15.12	86.53	55.32	37.29	26.20	61.55	75.59	87.00	58.15	59.05	56.18
TPT (Shu et al., 2022)	17.58	87.02	58.46	40.84	28.33	62.69	74.88	84.49	61.46	60.82	57.66
DiffTPT (Feng et al., 2023)	17.60	86.89	60.71	40.72	41.04	63.53	**79.21**	83.40	62.72	62.67	59.85
TDA (Karmanov et al., 2024)	17.61	89.70	57.78	43.74	42.11	68.74	77.75	86.18	62.53	64.18	61.03
BoostAdapter (Zhang et al., 2024b)	18.93	88.48	59.67	43.85	44.40	68.25	78.78	85.75	62.83	64.42	61.54
DPE (Zhang et al., 2024a)	19.80	90.83	59.26	50.18	41.67	67.60	77.83	85.97	64.23	61.98	61.93
SSG (Zhou et al., 2025)	21.63	90.67	60.75	52.19	43.26	69.02	77.73	86.64	64.23	62.15	62.83
SCA (Guan et al., 2025)	20.82	91.03	59.73	53.31	43.33	**71.09**	76.51	86.15	64.50	66.40	63.29
MCP (Chen et al., 2025)	23.04	90.99	61.27	53.19	55.60	68.49	78.37	87.35	65.39	67.14	65.08
SHARP (Ours)	**24.21**	**91.43**	**62.41**	**54.46**	**57.02**	70.31	78.71	**87.63**	**66.31**	**68.02**	**66.05**
CLIP-ViT-B/16 (Radford et al., 2021)	23.67	93.35	65.48	44.27	42.01	67.44	83.65	88.25	62.59	65.13	63.58
CoOp (Zhou et al., 2022b)	18.47	93.70	64.51	41.92	46.39	68.71	85.30	89.14	64.15	66.55	63.88
PromptSRC (Khattak et al., 2023b)	23.90	93.60	65.70	46.87	45.50	70.25	86.15	90.25	67.10	68.75	65.81
MaPLe (Khattak et al., 2023a)	24.74	93.53	65.57	46.49	48.06	72.23	86.20	90.49	67.01	68.69	66.30
TPT (Shu et al., 2022)	24.78	94.16	66.87	47.75	42.44	68.98	84.67	87.79	65.50	68.04	65.10
DiffTPT (Feng et al., 2023)	25.60	92.49	67.01	47.00	43.13	70.10	87.23	88.22	65.74	62.67	64.92
PromptAlign (Abdul Samadh et al., 2023)	24.80	94.01	68.50	47.24	47.86	72.39	86.65	90.76	67.54	69.47	66.92
TDA (Karmanov et al., 2024)	23.91	94.24	67.28	47.40	58.00	71.42	86.14	88.63	67.62	70.66	67.53
BoostAdapter (Zhang et al., 2024b)	27.45	94.77	69.30	45.69	61.22	71.66	87.17	89.51	68.09	71.93	68.68
DPE (Zhang et al., 2024a)	28.95	94.81	67.31	54.20	55.79	75.07	86.17	91.14	70.07	70.44	69.40
DOTA (Han et al., 2025)	26.25	94.16	69.56	47.64	62.78	75.23	87.08	92.01	69.80	72.54	69.71
SSG (Zhou et al., 2025)	30.21	94.93	68.65	56.15	59.91	74.54	86.40	91.58	70.10	69.73	70.22
DMN (Zhang et al., 2024c)	30.03	95.38	67.96	55.85	59.43	74.49	85.08	92.04	70.18	72.51	70.30
SCA (Guan et al., 2025)	28.50	94.85	68.49	57.09	57.16	76.09	86.09	91.44	70.27	73.43	70.34
MCP (Chen et al., 2025)	30.18	95.33	69.99	55.91	**68.42**	75.88	86.85	91.88	71.04	74.36	71.98
SHARP (Ours)	**31.84**	**95.89**	**71.32**	**58.46**	67.91	**77.42**	**87.31**	**92.11**	**72.18**	**76.95**	**73.14**

4.2 Cross-dataset evaluation

Across the recognition tasks in Table 1, SHARP achieves the highest average accuracy among the listed methods: 66.05% with ResNet-50 and 73.14% with ViT-B/16. These exceed MCP by 0.97% and 1.16%; SHARP ranks first on 8 of 10 datasets with ResNet-50 and 9 of 10 with ViT-B/16, including Aircraft, Cars, DTD, SUN397, and UCF101 under both backbones. On UCF101, SHARP reaches 76.95% with ViT-B/16, exceeding MCP by 2.59%. SHARP trails the best ResNet-50 results on Flowers102 and Food101; EuroSAT is its only non-leading ViT-B/16 result (0.51% below MCP, versus 1.42% above MCP with ResNet-50).

4.3 Robustness to natural distribution shifts

Table 2 evaluates four natural shifts of ImageNet. With ResNet-50, SHARP achieves a 5-dataset average of 52.14%, 0.94% above MCP, and an OOD average of 48.89%. SHARP improves over MCP on all four shifted variants, with gains of 1.64% on ImageNet-A, 0.75% on ImageNet-V2,

Table 2: Top-1 accuracy (%) on ImageNet and four distribution-shift variants. ImageNet-S denotes ImageNet-Sketch. Average includes all five datasets; OOD Avg. includes the four shifted variants.

Method	Reference	Shifted dataset				Summary	
	ImageNet	ImageNet-A	ImageNet-V2	ImageNet-R	ImageNet-S	Average	OOD Avg.
CLIP-ResNet-50 (Radford et al., 2021)	58.16	21.83	51.41	56.15	33.37	44.18	40.69
CoOp (Zhou et al., 2022b)	63.33	23.06	55.40	56.60	34.67	46.61	42.43
CoCoOp (Zhou et al., 2022a)	62.81	23.32	55.72	57.74	34.48	46.81	42.82
TPT (Shu et al., 2022)	60.74	26.67	54.70	59.11	35.09	47.26	43.89
DiffTPT (Feng et al., 2023)	60.80	31.06	55.80	58.80	37.10	48.71	45.69
TDA (Karmanov et al., 2024)	61.35	30.29	55.54	62.58	38.12	49.58	46.63
DMN (Zhang et al., 2024c)	63.87	28.57	56.12	61.44	39.84	49.97	46.49
DPE (Zhang et al., 2024a)	63.41	30.15	56.72	63.72	40.03	50.81	47.66
SCA (Guan et al., 2025)	63.11	30.64	56.51	62.85	39.33	50.49	47.33
MCP (Chen et al., 2025)	64.19	29.30	56.93	64.40	41.16	51.20	47.95
SHARP (Ours)	65.13	30.94	57.68	65.10	41.85	52.14	48.89
CLIP-ViT-B/16 (Radford et al., 2021)	66.73	47.87	60.86	73.98	46.09	59.11	57.20
CoCoOp (Zhou et al., 2022a)	71.51	49.71	64.20	75.21	47.99	61.72	59.28
TPT (Shu et al., 2022)	68.98	54.77	63.45	77.06	47.94	62.44	60.81
DiffTPT (Feng et al., 2023)	70.30	55.68	65.10	75.00	46.80	62.58	60.65
TDA (Karmanov et al., 2024)	69.51	60.11	64.67	80.24	50.54	65.01	63.89
DPE (Zhang et al., 2024a)	71.91	59.63	65.44	80.40	52.26	65.93	64.43
SCA (Guan et al., 2025)	71.75	60.33	65.38	80.85	52.50	66.16	64.77
MCP (Chen et al., 2025)	72.37	59.71	65.16	81.59	53.89	66.54	65.09
DOTA (Han et al., 2025)	70.69	61.50	-	81.21	51.84	-	-
SHARP (Ours)	72.85	62.23	65.79	82.41	54.16	67.49	66.15

0.70% on ImageNet-R, and 0.69% on ImageNet-Sketch. However, DiffTPT achieves a slightly higher ImageNet-A accuracy.

With ViT-B/16, SHARP reaches a 5-dataset average of 67.49% and an OOD average of 66.15%, improving over MCP by 0.95% and 1.06%. Its ImageNet-A accuracy is 62.23%, compared with 61.50% for DOTA, the strongest baseline on that variant.

4.4 Ablation study

Baseline and comparison protocol. Table 3 measures incremental component gains through cumulative configurations on the 10 benchmarks with ViT-B/16. The baseline re-implements MCP (entropy, alignment, and negative caches) under the SHARP augmentation and optimization settings and reaches 71.57%, versus the published 71.98% (largest gap on EuroSAT); ablation gains are measured against this re-implementation.

Cache variants. The confidence-cache variant (CA) replaces the baseline entropy cache of Eq. (4), and the hard-negative cache (HNC) replaces the baseline negative cache with the weighted hard-negative cache of Eqs. (6)–(7). CA increases average accuracy from 71.57% to 71.79%, and HNC to 71.92% (0.35% above Baseline). Their effects vary by dataset: adding HNC improves UCF101 from 74.14% to 76.18%, while DTD decreases from 56.97% to 56.09%.

Distribution modeling and prototype recalibration. Distribution-aware adaptation (DA) adds the distribution score of Eq. (11) to the logits with statistics estimated in the original CLIP space; SH moves features, anchors, and statistics to the low-dimensional representation of Eqs. (1)–(3) with other settings unchanged. DA adds 0.57% over +HNC, reaching 72.49%. Adding SH raises accuracy by 0.33% to 72.82%, and adding VAPR gives a further 0.32% to reach 73.14%. The full model exceeds the re-implemented baseline by 1.57% and leads on every dataset in this ablation. Together, SH and VAPR add 0.65% over +DA; these gains are conditional on the preceding components in the cumulative pipeline.

4.5 Sensitivity to cache capacity and distribution-weight schedule

Larger caches do not consistently improve accuracy in Table 4. With ViT-B/16, OOD accuracy ranges from 65.42% to 66.15%, peaking at 10 confidence entries and 3 hard-negative entries per class. At this hard-negative capacity, increasing the confidence capacity from 5 to 10 adds 0.73%,

Table 3: Sequential ablation on ViT-B/16 (top-1 accuracy, %). Configurations are cumulative top to bottom from re-implemented MCP; CA, HNC, DA, SH, and VAPR are defined in the text.

Config.	Aircraft	Caltech	Cars	DTD	EuroSAT	Flowers	Food101	Pets	SUN397	UCF101	Average
Baseline	30.24	95.50	69.78	55.61	63.74	76.98	86.91	91.88	70.93	74.15	71.57
+CA	30.51	95.53	69.97	56.97	63.70	77.18	86.92	92.07	70.94	74.14	71.79
+HNC	30.27	95.70	70.09	56.09	64.01	76.98	86.91	91.96	71.01	76.18	71.92
+DA	30.66	95.70	70.44	57.87	66.50	77.06	87.01	91.96	71.41	76.26	72.49
+SH	31.17	95.82	70.69	58.31	67.03	77.38	87.24	92.04	71.76	76.73	72.82
+VAPR	**31.84**	**95.89**	**71.32**	**58.46**	**67.91**	**77.42**	**87.31**	**92.11**	**72.18**	**76.95**	**73.14**

Table 4: OOD average accuracy (%) for per-class cache capacities, ViT-B/16.

$\lvert\mathcal{M}_C^k\rvert$	$\lvert\mathcal{M}_H^k\rvert = 3$	$\lvert\mathcal{M}_H^k\rvert = 5$
5	65.42	65.74
10	**66.15**	65.81
15	65.68	65.83

Table 5: Runtime and accuracy on ImageNet-A, ViT-B/16; gains over CLIP in points.

Method	Time	Acc. (%)	Gain (%)
CLIP	5 min	47.87	–
TPT	3 h	54.77	+6.90
DiffTPT	>10 h	55.68	+7.81
MCP	30 min	59.71	+11.84
DOTA	40 min	61.50	+13.63
SHARP (Ours)	35 min	62.23	+14.36

whereas increasing it to 15 loses 0.47%. Adding hard-negative entries lowers accuracy at a confidence capacity of 10 but raises it at 5 and 15, indicating interaction between the two capacities.

For $\lambda_{\text{dist}}(t)$ in Eq. 15, 24 of 25 settings in Table 6 exceed 72%. Accuracy ranges from 71.81% to 73.14%, peaking at $(\eta, \rho) = (0.2, 0.005)$. Several settings approach this maximum, without a monotonic trend in either parameter.

Table 6: Cross-dataset average accuracy (%) for schedule parameters (η, ρ), ViT-B/16.

η	$\rho = 0.0025$	$\rho = 0.005$	$\rho = 0.001$	$\rho = 0.002$	$\rho = 0.003$
0.1	72.52	72.35	72.84	72.92	73.07
0.2	72.68	**73.14**	73.05	71.81	72.59
0.3	72.93	72.38	72.26	72.62	72.51
0.5	73.11	72.89	72.96	72.77	72.64
0.8	73.08	72.29	72.75	72.43	72.91

4.6 Computational efficiency

Table 5 compares ImageNet-A runtime and accuracy with ViT-B/16 on an NVIDIA RTX 3060 (12 GB). SHARP reaches 62.23% in 35 minutes, gaining 2.52% over MCP for 5 extra minutes and 0.73% over DOTA in 5 fewer minutes, and is about $5.1\times$ and over $17\times$ faster than TPT and DiffTPT.

5 Conclusion

We presented SHARP, an online TTA method that estimates class-conditional distributions in a compact angular space rather than directly in the original CLIP embedding space. With an isotropic prior and covariance shrinkage, these statistics can be estimated more stably from limited test samples and are used for both distribution-aware scoring and variance-adaptive prototype recalibration. Combined with multi-cache instance evidence, SHARP consistently improves cross-dataset and distribution-shift accuracy with two CLIP backbones. Limitations remain: pseudo-label errors early in the stream may affect later estimates, a single Gaussian per class may be restrictive for multimodal targets, and full class covariances and residual optimization add memory and test-time cost as the label space grows. These motivate more robust online statistics, mixture-based class models, and low-rank covariance approximations.

AI USE STATEMENT

Generative AI tools were used to improve the language, grammar, readability, and organization of the manuscript, and to assist with literature verification, numerical consistency checks of reported tables. The tools were not used to conduct experiments or generate experimental results. The authors are responsible for reviewing, editing, and verifying all AI-assisted text and for the final content of this publication.

REPRODUCIBILITY STATEMENT

The adaptation framework is described in Section 3, with full pseudocode in Algorithm 1 and the defined method parameters in Table 7; the datasets, evaluation metric, and implementation settings are summarized in Section 4.1. Section 4.4 reports component ablations, while Sections 4.5 and 4.6 analyze cache capacity, the distribution-weight schedule, and runtime. Appendix A derives the covariance recursion and the shrinkage choice, and Appendix B provides qualitative analysis.

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.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European Conference on Computer Vision*, pp. 446–461, 2014.

Xinyu Chen, Haotian Zhai, Can Zhang, Xiupeng Shi, and Ruirui Li. Multi-cache enhanced prototype learning for test-time generalization of vision-language models. In *International Conference on Computer Vision*, pp. 2281–2291, 2025.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Computer Vision and Pattern Recognition*, pp. 3606–3613, 2014.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

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 *Computer Vision and Pattern Recognition Workshop*, pp. 178–178, 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 *International Conference on Computer Vision*, pp. 2704–2714, 2023.

Zenghao Guan, Zhou Yucan, Wu Liu, and Xiaoyan Gu. Statistics caching test-time adaptation for vision-language models. In *Advances in Neural Information Processing Systems*, pp. 1–24, 2025.

Zongbo Han, Jialong Yang, Guangyu Wang, Junfan Li, Qianli Xu, Mike Zheng Shou, and Changqing Zhang. Dota: Distributional test-time adaptation of vision-language models. In *Advances in Neural Information Processing Systems*, 2025.

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 *International Conference on Computer Vision*, pp. 8340–8349, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Computer Vision and Pattern Recognition*, pp. 15262–15271, 2021b.

Adilbek Karmanov, Dayan Guan, Shijian Lu, Abdulmotaleb El Saddik, and Eric Xing. Efficient test-time adaptation of vision-language models. In *Computer Vision and Pattern Recognition*, pp. 14162–14171, 2024.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Computer Vision and Pattern Recognition*, pp. 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 *International Conference on Computer Vision*, pp. 15190–15200, 2023b.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *International Conference on Computer Vision Workshops*, pp. 554–561, 2013.

Haoyang Li, Liang Wang, Chao Wang, Jing Jiang, Yan Peng, and Guodong Long. Dpc: Dual-prompt collaboration for tuning vision-language models. In *Computer Vision and Pattern Recognition*, pp. 25623–25632, 2025.

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

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Computer Vision, Graphics & Image Processing*, pp. 722–729, 2008.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *Computer Vision and Pattern Recognition*, pp. 3498–3505, 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*, pp. 8748–8763, 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pp. 5389–5400, 2019.

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

K Soomro. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Pavan Kumar Anasosalu Vasu, Fartash Faghri, Chun-Liang Li, Cem Koc, Nate True, Albert Antony, Gokula Santhanam, James Gabriel, Peter Grasch, Oncel Tuzel, et al. Fastvlm: Efficient vision encoding for vision language models. In *Computer Vision and Pattern Recognition*, pp. 19769–19780, 2025.

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, 2019a.

Yulin Wang, Xuran Pan, Shiji Song, Hong Zhang, Gao Huang, and Cheng Wu. Implicit semantic data augmentation for deep networks. *Advances in Neural Information Processing Systems*, 32, 2019b.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer Vision and Pattern Recognition*, pp. 3485–3492, 2010.

Maxime Zanella, Clément Fuchs, Christophe De Vleeschouwer, and Ismail Ben Ayed. Realistic test-time adaptation of vision-language models. In *Computer Vision and Pattern Recognition*, pp. 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 *Computer Vision and Pattern Recognition*, pp. 30032–30041, 2025a.

Qiang Zhang, Mengsheng Zhao, Jiawei Liu, Fanrui Zhang, Yongchao Xu, and Zheng-Jun Zha. Hierarchical knowledge prompt tuning for multi-task test-time adaptation. In *Computer Vision and Pattern Recognition*, pp. 30524–30533, 2025b.

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 *Computer Vision and Pattern Recognition*, pp. 28718–28728, 2024c.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Computer Vision and Pattern Recognition*, pp. 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.

Shenglong Zhou, Manjiang Yin, Leiyu Sun, Shicai Yang, Di Xie, and Jiang Zhu. Training-free test-time adaptation via shape and style guidance for vision-language models. In *Advances in Neural Information Processing Systems*, 2025.