ROSE-TTA: RELIABLE ONLINE STRUCTURAL ENHANCEMENT FOR TEST-TIME ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large-scale vision-language models like CLIP exhibit remarkable zero-shot generalization but suffer significant performance degradation under real-world distribution shifts. Although recent cache-based test-time adaptation (TTA) methods mitigate the issues, they are limited by: (i) unreliability in cache construction, as entropy-based sample selection is insufficient under distribution shifts; and (ii) incomplete cache information at inference, with both imbalanced category information caused by sequential online updates and insufficient sample-specific information for next online instance. To address these limitations, we propose ***ROSE-TTA***, a unified framework that enhances both cache construction and utilization for more reliable and stable adaptation. For construction, we introduce a noise-aware uncertainty measure that combines entropy with perturbation-based prediction stability to robustly select cache entries. To complete the cache information for utilization, we develop a graph-based structural completion strategy, which effectively mi tigates class imbalance and completes global information by transferring information between text embeddings and cached features. Additionally, we introduce a sample-specific refinement mechanism to dynamically update cache features and incorporate local information of each online test sample. Experiments on 15 datasets demonstrate the effectiveness of our method.

1 INTRODUCTION

The emergence of large-scale vision-language models (VLMs), such as CLIP (Radford et al., 2021), has profoundly reshaped the landscape of multi-modal learning. By effectively aligning visual and textual modalities through contrastive pretraining on web-scale datasets, these models demonstrate remarkable zero-shot capabilities across diverse downstream tasks. However, deploying CLIP in real world remains challenging, especially on specific data with distribution shifts between the test and pretrained data (Shu et al., 2022; Han et al., 2025; Karmanov et al., 2024).

A common approach to address distribution shift at test time is test-time adaptation (TTA) (Sun et al., 2020; Wang et al., 2020; Chen et al., 2022), which has also been leveraged to improve CLIP's zero-shot capability (Shu et al., 2022; Karmanov et al., 2024). Test-time prompt tuning (TPT) (Shu et al., 2022; Feng et al., 2023; Abdul Samadh et al., 2023; Yoon et al., 2024) fine-tunes textual prompts at inference, often guided by entropy minimization. Despite their effectiveness in addressing distribution shifts, these methods are computationally expensive due to backpropagation. To improve efficiency, cache-based TTA methods have emerged (Karmanov et al., 2024; Han et al., 2025). These approaches introduce a dynamic cache to store test features online, which are used to refine predictions without backpropagation, providing a lightweight online adaptation mechanism.

Despite their computational efficiency, existing cache-based TTA methods suffer from two fundamental limitations that compromise their robustness under distribution shifts. First, cache construction lacks reliability. Relying solely on entropy-based selection fails to effectively eliminate noisy or misclassified instances (Nguyen et al., 2023a; Han et al., 2025; Shamsi et al., 2025; Zhou et al., 2025), leading to corrupted cache. Second, cache information available during inference is inherently incomplete. At global level, sequential online updates induce class imbalance within cached features. Especially during the early adaptation stage, this random arrival leads to unequal cache capacities across classes. This imbalance causes the cache to favor the majority classes and provides insufficient guidance for the minority classes, ultimately biasing model predictions and

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.

even leading to error accumulation during continuous adaptation (Zhao et al., 2023). Additionally, cached features lack local information of each incoming sample, providing limited sample-specific guidance for the online instance. These issues hinder both the stability and generalization of cache-based adaptation. However, prior work (Karmanov et al., 2024; Han et al., 2025; Zhou et al., 2025) primarily focuses on either construction or utilization, treating them as separate problems. We recognize these are inherently coupled: unreliable cache construction cascades into biased utilization, while incomplete utilization fails to leverage even high-quality cached samples.

To systematically address these limitations, we propose ***ROSE-TTA*** (***R***eliable ***O***nline ***S***tructural ***E***nhancement for ***T***est-***T***ime ***A***daptation), a novel and unified framework that enhances the reliability and the stability in cache update and integrates complementary global (class level) and local (sample-specific) information for robust cache-based adaptation. In cache construction, we introduce an improved cache update mechanism to synergistically combine entropy with a noise-enhanced uncertainty measure, which evaluates prediction robustness under perturbations as a complementary signal to confidence. This dual-criterion approach ensures that only the most reliable and consistently stable online instances are incorporated into the cache, improving the cache reliability. Moreover, to complement global semantic information in the cache, we propose a novel graph-based structural completion strategy (Wei et al., 2024). The method reconstructs the categorical graph with the semantic relationships between classes from text embeddings and test specific information from cached features, mitigating class imbalance and strengthening the representation of underrepresented categories within the cache (Du et al., 2021). We also design a sample-specific refinement mechanism that updates cached features on-the-fly using the information of each test sample, incorporating local information and improving alignment between the cache and instance.

We evaluate ***ROSE-TTA*** on 15 widely used datasets, covering typical evaluation scenarios such as domain generalization and cross-dataset. The experimental results demonstrate the effectiveness of our method on enhancing the reliability and adaptability of cache-based TTA.

2 PRELIMINARY

CLIP (Radford et al., 2021). CLIP is known for the remarkable ability in vision-language representations learning through large-scale training in image-text data. The pretrained CLIP model consists of an image encoder $\mathcal{F}_{\theta_I}(\cdot)$ and a text encoder $\mathcal{F}_{\theta_T}(\cdot)$, with θ_I and θ_T denoting the model parameters, respectively. Based on a zero-shot C-class classification task, for each class $c \in \{1, \dots, C\}$, we generate a text prompt $\boldsymbol{t}_c$ by instantiating a template such as "*a photo of a [class]*", where "*[class]*" is replaced with the name corresponding to class c. Each text prompt $\boldsymbol{t}_c$ is then encoded as $\boldsymbol{f}_c = \mathcal{F}_{\theta_T}(\boldsymbol{t}_c)$ and the image $\boldsymbol{x}$ is encoded as $\boldsymbol{f}_{\boldsymbol{x}} = \mathcal{F}_{\theta_I}(\boldsymbol{x})$. Collecting all text embeddings as the matrix $\boldsymbol{W}_C = [\boldsymbol{f}_1, \boldsymbol{f}_2, \dots, \boldsymbol{f}_C]$, CLIP seeks to associate the image feature $\boldsymbol{f}_{\boldsymbol{x}}$ with the most semantically relevant text feature from $\boldsymbol{W}_C$. The probability of $\boldsymbol{x}$ to be classified as class c is $p(\hat{y} = c \mid \boldsymbol{x}) = \frac{\exp(\cos(\boldsymbol{f}_{\boldsymbol{x}}, \boldsymbol{f}_c)/\tau)}{\sum_{k=1}^{C} \exp(\cos(\boldsymbol{f}_{\boldsymbol{x}}, \boldsymbol{f}_k)/\tau)}$, where $\cos(\cdot,\cdot)$ denotes cosine similarity and τ is a temperature parameter. The most relevant class is obtained from CLIP by $\arg\max_c \; p(\hat{y} = c \mid \mathbf{x})$. For subsequent analysis, we denote the predicted probabilities $p(\hat{y} = c \mid \boldsymbol{x})$ over all C classes as P_{clip}.

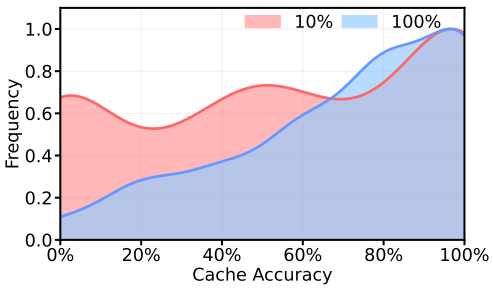
Test-time adaption based on key-value cache for CLIP. As a training-free solution that adapts pre-trained models to test data with distributional shift (Tahir et al., 2022; Zhang et al., 2024; Gao et al., 2025), test-time adaptation adjusts model predictions on-the-fly to better align with the test data. A prominent family of methods leverages a *key–value cache* that accumulates reliable test samples to refine CLIP's predictions (Han et al., 2025; Karmanov et al., 2024).

In the cache-based methods, a memory $(\boldsymbol{F}, \hat{\boldsymbol{L}})$ is introduced to store N historical features $\boldsymbol{F} \in \mathbb{R}^{N \times d}$ and their corresponding (pseudo-)labels $\hat{\boldsymbol{L}} \in \mathbb{R}^{N \times C}$. The interaction between a new feature $\boldsymbol{f}_{test}$ and the cache follows a unified paradigm:

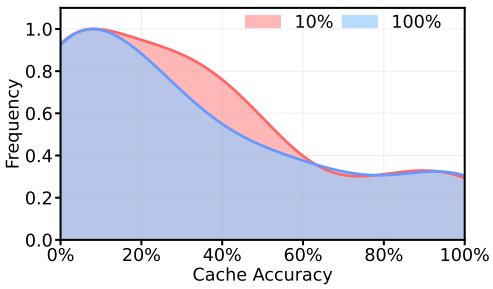
$$P_{cache}(\boldsymbol{f}_{test}) = A\big(\boldsymbol{f}_{test}\boldsymbol{F}^T\big)\ \hat{\boldsymbol{L}}, \tag{1}$$

where $\boldsymbol{f}_{test}\boldsymbol{F}^T$ denotes the affinity scores (Karmanov et al., 2024) between the new feature and cached features, and $A(\cdot)$ is an activation function that maps these affinities into weights. Finally, the complete prediction combines the original CLIP output with the cache contribution:

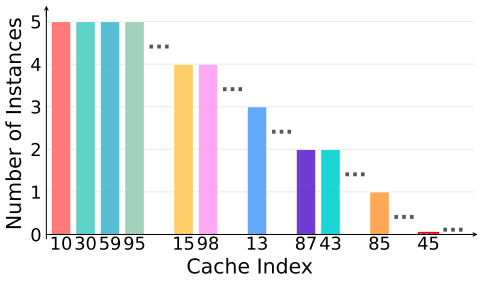
$$P_{final} = P_{clip} + \alpha \cdot P_{cache}. \tag{2}$$



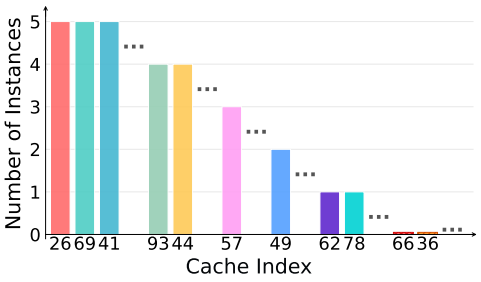
(a) Cache accuracy on ImageNet-A at 10% and 100% test stage.



(b) Cache accuracy on FGVC at 10% and 100% test stage.



(c) Distribution of cache numbers in ImageNet-A at 10% test stage.



(d) Distribution of cache numbers in FGVC at 10% test stage.

Figure 1: **Category and accuracy statistics of the entropy-based online cache.** The online updated cache can be unreliable (a and b) and imbalanced (c and d).

where α is a scaling factor to balance the influence of cache information. This formulation provides a standardized cache-based mechanism for refining the CLIP zero-shot predictions, where the effectiveness depends on how $(\boldsymbol{F}, \hat{\boldsymbol{L}})$ are updated during inference.

3 METHOD

3.1 MOTIVATION

Although the cache-based methods achieve efficient test-time adaptation for zero-shot classification of CLIP (Wang et al., 2020; Zhang et al., 2024), the entropy-based cache construction and utilization are not always reliable (Han et al., 2025; Zhou et al., 2025). In Figures 1a and 1b, we report the precision of the features stored in the entropy-based online cache (Karmanov et al., 2024) at the beginning and end of the online test stage on two datasets. Even when selecting features with minimum entropy, many of them are cached under incorrect labels. The problem is more severe at the earlier test stage (10%). These findings indicate that entropy alone is an insufficient criterion for cache construction and update.

Moreover, since online test samples arrive in random sequential order, cache updates inherently induce class imbalance, particularly at the early adaptation stage. As shown in Figures 1c and 1d, after processing only 10% of the stream, some classes have already filled their cache capacity while others hold few or even no samples. Such imbalance skews predictions toward classes with richer cache representations, and the inaccuracies can further accumulate during online learning.

Additionally, the incoming instance in the online test stream is unpredictable, which can be from a new class or an unseen style. Although the cache is updated online, it still lacks local instance information of such specific test samples, leading to unstable predictions.

Therefore, we propose an enhanced cache for online test-time adaptation via uncertainty-aware cache construction and graph-based information completion. An illustration of our method is shown in Figure 2.

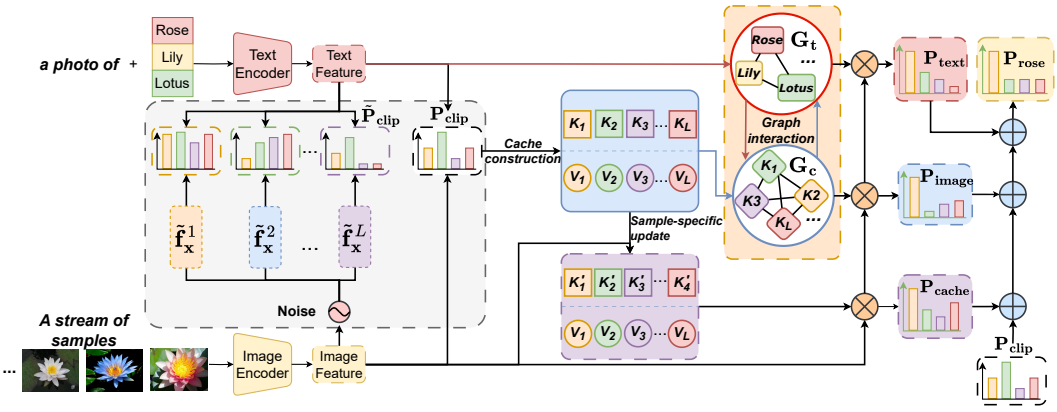


Figure 2: **Illustration of Rose-TTA.** We propose uncertainty aware cache construction to store more reliable test samples in the cache. During utilization, the cache is further enhanced by graph-based structural completion and sample-specific refinement, which inject the global category information and local instance information, respectively.

3.2 Uncertainty-aware cache construction

To combat the unreliability of cache construction, where sole reliance on entropy often leads to noisy or misclassified samples (Nguyen et al., 2023b; Han et al., 2025; Lee et al., 2024), we introduce an improved cache update mechanism. That ensures only the most reliable and consistently stable features are incorporated into the cache, improving the quality of the cache construction features.

Noise-enhanced uncertainty estimation. The noise perturbation mechanism plays a critical role in assessing prediction reliability under distribution shifts: while entropy-based selection can identify confident predictions, it fails to distinguish genuinely stable samples from overconfident but fragile ones. We therefore propose a noise-enhanced uncertainty estimation mechanism, which evaluates the prediction stability under controlled perturbations and uses it to select stable features for cache construction and update during online test-time adaptation.

Given an input test sample $\boldsymbol{x}$, we first obtain its CLIP prediction $\hat{c} = \arg\max_c P_{clip}$ to find the corresponding class-wise cache. To access the prediction stability of this sample, i.e., the prediction consistency under small perturbations, we generate n augmented features by adding calibrated Gaussian noise on the original feature:

$$\tilde{\boldsymbol{f}}_{\boldsymbol{x}}^{i} = \boldsymbol{f}_{\boldsymbol{x}} + \epsilon_i \cdot \sigma, \quad \epsilon_i \sim \mathcal{N}(0, \boldsymbol{I}), \quad i = 1, \ldots, n, \tag{3}$$

where σ controls the noise magnitude and n is the number of augmentation, which is used to balance perturbation strength and feature semantic preservation. With the predictions on the noise-perturbed features, we obtain the prediction stability for the sample $\boldsymbol{x}$ by:

$$\mathbf{d}(\boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^{n} |p(\hat{y} = \hat{c} \mid \tilde{\boldsymbol{f}}_{\boldsymbol{x}}^{i}) - p(\hat{y} = \hat{c} \mid \boldsymbol{f}_{\boldsymbol{x}})|, \tag{4}$$

where $p(\hat{y} = \hat{c} \mid \tilde{\boldsymbol{f}}_{\boldsymbol{x}}^{i})$ and $p(\hat{y} = \hat{c} \mid \boldsymbol{f}_{\boldsymbol{x}})$ denote the predicted probability on class $\hat{c}$ given the noise-perturbed feature $\tilde{\boldsymbol{f}}_{\boldsymbol{x}}^{i}$ and original feature $\boldsymbol{f}_{\boldsymbol{x}}$, respectively. The prediction stability metric measures how confidence changes under noise perturbations. Smaller $\mathbf{d}(\boldsymbol{x})$ indicates that the instance is more robust to perturbations and therefore more reliable for caching.

Doubly robust cache. To achieve a more reliable cache during online test-time adaptation, we adopt both entropy and noise-enhanced stability as our overall selection metric to construct a doubly robust cache $\mathcal{C} = (\boldsymbol{F}, \hat{\boldsymbol{L}}) = \{\mathcal{C}_c\}_{c=1}^{C} = \{(\boldsymbol{F}_c, \hat{\boldsymbol{L}}_c)\}_{c=1}^{C}$. Following TDA (Karmanov et al., 2024), $\boldsymbol{F}_c, \hat{\boldsymbol{L}}_c$ are the raw features of the historical samples and the one-hot labels of class c, respectively. When the cache $\mathcal{C}_c$ for class c has not reached its maximum capacity, the new sample is directly appended along with its entropy and stability measure. Since the cache size cannot be infinite, when the cache reaches its capacity $n_{\mathcal{C}}$, the new sample is admitted to the cache $\mathcal{C}_c$ only if it surpasses

existing cached samples on both metrics. Specifically, the replacement occurs if and only if both conditions are satisfied:

$$\begin{cases} \mathcal{H}(p(y|\boldsymbol{x})) < \mathcal{H}(p(y|\boldsymbol{x}_{j^*})) & \text{(lower entropy)} \\ \mathbf{d}(\boldsymbol{x}) < \mathbf{d}(\boldsymbol{x}_{j^*}) & \text{(higher stability)}, \end{cases} \tag{5}$$

where $\mathcal{H}$ and $\mathbf{d}$ are the entropy and prediction stability of the input sample. j^* denotes the index of the weakest cached sample with the highest combined entropy and the lowest stability score. This dual-criterion approach ensures that the cache maintains samples that are not only confident(low entropy) but also robust to input perturbations, avoiding overconfidence under distribution shifts and improving the reliability of cached features for test-time adaptation.

3.3 Global and local completion in cache utilization

Beyond improving cache reliability during construction, our method also strengthens the utilization of the test-time cache by completing both global and local information.

Graph-based structural completion. As shown in Figure 1, since the test sequence is random and unpredictable, the online cache exhibits class imbalance and incomplete global task information, which biases predictions towards majority classes while neglecting minority ones. For example, if no sample from class c has appeared, the cache provides little categorical guidance; when a new sample from c arrives, such guidance is noisy or even degrades the final prediction.

To mitigate class imbalance and complete the global information in online cache, we propose a graph-based structural completion strategy, which reconstructs the categorical graph of the task by integrating class information from both text embeddings and cache features. The graph addresses the class imbalance issue by introducing the global class information from text embeddings of all categories, while preserving the test-specific information by considering the cache features. Specifically, we construct three complementary graphs to capture different aspects of the information:

1) Cache-graph $\boldsymbol{G}_c \in \mathbb{R}^{N\times N}$: $\boldsymbol{G}_c = (\boldsymbol{F}\boldsymbol{F}^T)$ models the relationships between the N cached features in $\boldsymbol{F}$, preserving the class-level and instance-level information of the test data. Since cached features vary in reliability, we further introduce a reliability-weighted cache graph. Each cached sample is assigned a reliability weight $w_j = \sqrt{(1-\mathbf{d}_j)/\mathcal{H}_j}$, where $\mathcal{H}_j$ and $\mathbf{d}_j$ are the entropy and stability scores of the j-th sample; lower entropy and higher stability thus yield a larger w_j and greater influence in the graph. The weight matrix $\boldsymbol{W}_{jk} = \sqrt{w_j \cdot w_k}$ then refines the cache graph as $\hat{\boldsymbol{G}}_c = \boldsymbol{F}\boldsymbol{F}^T \odot \boldsymbol{W}$, where $\odot$ denotes element-wise multiplication.

2) Text-graph $\boldsymbol{G}_t \in \mathbb{R}^{C\times C}$: $\boldsymbol{G}_t = (\boldsymbol{W}_C\boldsymbol{W}_C^T)$ encodes the semantic relationships among the C classes at the text level, where $\boldsymbol{W}_C$ collects their text embeddings. By covering all classnames, $\boldsymbol{G}_t$ provides the global category information absent from the imbalanced cache, complementing underrepresented classes via inter-class semantics and thereby producing more balanced predictions.

3) Gate-graph $\boldsymbol{M} \in \{0,1\}^{C\times N}$ is a binary mapping matrix that links samples to their corresponding classes. $\boldsymbol{M}_{cj}$ is set to 1 if the cached sample $\boldsymbol{x}_j$ belongs to class c, else 0.

We integrate the three graphs into a unified categorical graph, where $\boldsymbol{M}\hat{\boldsymbol{G}}_c\boldsymbol{M}^T$ aggregates the sample-level cache affinities into class-level ones, and $\boldsymbol{G}_t$ injects the global text semantics:

$$\hat{\boldsymbol{G}} = \text{softmax}\left(\boldsymbol{G}_t\boldsymbol{M}\hat{\boldsymbol{G}}_c\boldsymbol{M}^T\boldsymbol{G}_t^T\right), \tag{6}$$

where the softmax normalizes each row of $\hat{\boldsymbol{G}} \in \mathbb{R}^{C\times C}$, yielding the affinity between every pair of classes, such that the cached features of semantically related classes contribute to the prediction of classes with few or no cached samples. For a test feature $\boldsymbol{f}_{\boldsymbol{x}}$, we derive two graph-enhanced prediction terms from the textual and visual paths, respectively:

$$P_{\text{t}} = \boldsymbol{f}_{\boldsymbol{x}}(\hat{\boldsymbol{G}}\boldsymbol{W}_C^T)^T, \qquad P_{\text{i}} = \boldsymbol{f}_{\boldsymbol{x}}(\hat{\boldsymbol{G}}\boldsymbol{M}\boldsymbol{F})^T. \tag{7}$$

The textual prediction P_t is thereby injected with the test-specific information stored in the cache, and the cache prediction P_i with the global class information from text embeddings.

Sample-specific completion. Beyond the global class information, the cache also lacks local instance information, since the next test sample is unpredictable and a sample unlike any cached one

can still yield an unreliable cache prediction. To incorporate such local information in cache utilization, we propose a sample-specific cache refinement mechanism that dynamically updates the cached features with each incoming test sample. Gradient-based TTA (Shu et al., 2022; Zhang et al., 2022) refines the model parameters per test sample via backpropagation of entropy minimization, which is computationally expensive. Instead, we update the cache features directly with a backpropagation-free pseudo-gradient inferred from the sample feature and its entropy.

To control the update of each cache feature according to the cache prediction, we first obtain the averaged cache prediction $P_n = softmax\left(\frac{1}{n}\sum_{i=1}^{n} P_{cache}(\tilde{\boldsymbol{f}}_{\boldsymbol{x}}^{i})\right)$ based on the noise-augmented features in Eq. (3), and introduce two control factors $\gamma = 1 - \mathcal{H}(P_n)$ and $\zeta = P_n - 1/C$, where $\mathcal{H}(P_n)$ denotes the normalized entropy of P_n. γ controls the update intensity (higher confidence, larger update), while ζ controls the direction, pushing probabilities above the uniform $1/C$ up and the others down, as entropy minimization tends to do; theoretical and empirical analyses of this pseudo-gradient versus the actual gradient of entropy minimization are provided in Appendix B.

Based on $\boldsymbol{f}_{\boldsymbol{x}}$ and the control factors, we refine the cache features by a pseudo-gradient update:

$$\hat{\boldsymbol{F}} = \boldsymbol{F} + \eta \cdot \gamma \cdot \zeta \cdot \boldsymbol{f}_{\boldsymbol{x}}, \tag{8}$$

where η is a pseudo-learning rate, and γ and ζ control the update intensity and direction, respectively, as defined above. Following Eq. (1), the refined cache prediction is calculated by:

$$\hat{P}_{cache} = A\left(\boldsymbol{f}_{\boldsymbol{x}}\hat{\boldsymbol{F}}^{T}\right)\hat{\boldsymbol{L}}, \tag{9}$$

where $A(\cdot)$ is the activation function and $\hat{\mathbf{L}}$ denotes the one-hot pseudo labels of the cache features. This sample-specific refinement injects local instance information into the cache in a fully forward-pass manner, without gradient computation or backpropagation.

Overall, our method integrates multiple complementary sources of information to produce robust test-time predictions. The final prediction is calculated with a balancing coefficient α:

$$P_{rose} = P_{clip} + \alpha \cdot (\hat{P}_{cache} + P_t + P_i). \tag{10}$$

4 Related Work

Vision-language models. The emergence of large-scale vision-language models has fundamentally transformed the landscape of multi-modal understanding. Early pioneering work CLIP (Radford et al., 2021) demonstrated that contrastive learning on web-scale image-text pairs enables powerful zero-shot transfer capabilities across diverse tasks. Building upon this foundation, ALIGN (Jia et al., 2021) demonstrated that scale matters significantly and FILIP (Yao et al., 2021) introduced fine-grained cross-modal alignment through token-level interactions, moving beyond global image-text matching. With ongoing research, works such as InternVL3.5 (Wang et al., 2025) further advance open-source multimodal models in versatility. Concurrently, studies like Qwen-Image (Wu et al., 2025) focus on image generation and DINOv3 (Siméoni et al., 2025) continues to push the boundaries of self-supervised learning.

Test-time adaptation for vision-language models. Test-time adaptation (TTA) (Sun et al., 2020; Nado et al., 2020; Wang et al., 2020) emerged to address distribution shift between training and test environments without requiring labeled target data, and has recently been extended to vision-language models with their dual-modal structure and rich semantics. Early works explored *gradient-based prompt tuning* (Shu et al., 2022; Yoon et al., 2024; Xiao et al., 2025; Ahamed et al., 2026), which adapt text prompts via entropy minimization. While effective, these methods incur high computational overhead. To improve efficiency, *training-free approaches* were proposed. Motivated by the limitations of entropy-based objectives, DOTA (Han et al., 2025) and Bayesian TTA (Zhou et al., 2025) provide principled alternatives through distribution estimation and uncertainty quantification respectively. Cache-based methods (Karmanov et al., 2024; Nguyen et al., 2025; Zhang et al., 2026) leverage stored representative samples refine predictions without gradient updates. For example, TDA (Karmanov et al., 2024) refines predictions by storing low-entropy samples in a positive cache and mid-entropy samples in a negative cache. Despite progress, current methods often over-rely on entropy-based selection, suffer from class imbalance where underrepresented classes receive insufficient cache guidance, and employ static feature representations that fail to adapt to evolving test distributions or utilize the semantic structure in text embeddings.

Table 1: **Comparisons on the cross-dataset setting.** The best and runner-up results are bolded and underlined, respectively.

Method	Aircraft	Caltech	Cars	DTD	EuroSAT	Flowers	Food101	Pets	SUN397	UCF101	*Mean*
CLIP-ResNet-50	16.11	87.26	55.89	40.37	25.79	62.77	74.82	82.97	60.85	59.48	56.63
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
C-TPT (Yoon et al., 2024)	17.50	87.40	57.30	43.10	29.40	65.30	76.00	84.00	62.10	60.70	58.28
DynaPrompt (Xiao et al., 2025)	16.44	88.07	57.01	41.19	23.41	61.84	75.73	83.07	62.34	59.87	56.90
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
BCA (Zhou et al., 2025)	**19.89**	89.70	58.13	**48.58**	42.12	66.30	77.19	85.58	63.38	63.51	61.44
PTA (Huang et al., 2026)	17.44	88.64	59.35	44.98	**51.63**	68.49	76.83	86.67	62.86	**66.43**	62.33
TTA-CaP (Sharafi et al., 2026)	18.02	89.68	58.86	44.82	49.24	67.88	77.53	86.42	63.01	65.92	62.14
Rose-TTA (Hand-crafted)	18.46	89.73	58.05	45.10	49.31	68.17	77.78	86.51	**63.41**	64.23	62.07
Rose-TTA (Generated)	**19.89**	**90.21**	**59.42**	47.98	51.54	**68.99**	**78.65**	**86.73**	63.38	66.29	**63.31**
CLIP-ViT-B/16	23.22	93.55	66.11	45.04	50.42	66.99	82.86	86.92	65.63	65.16	64.59
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
C-TPT (Yoon et al., 2024)	23.90	94.10	66.70	46.80	48.70	69.90	84.50	87.40	66.00	66.70	65.47
DynaPrompt (Xiao et al., 2025)	24.33	94.32	67.65	47.96	42.28	69.95	85.42	88.28	66.32	68.72	65.52
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
BCA (Zhou et al., 2025)	**28.59**	94.69	66.86	**53.49**	56.63	73.12	85.97	90.43	68.41	67.59	68.58
PTA (Huang et al., 2026)	26.10	94.81	**68.55**	47.70	61.57	75.23	86.44	91.06	**69.21**	**73.12**	69.38
TTA-CaP (Sharafi et al., 2026)	25.02	94.38	66.99	50.21	63.11	73.89	85.91	90.82	67.01	72.96	69.03
Rose-TTA (Hand-crafted)	25.86	94.76	67.06	48.00	65.64	74.17	86.20	90.95	68.00	71.34	69.20
Rose-TTA (Generated)	27.92	**95.21**	**68.55**	50.02	**66.02**	**75.28**	**86.52**	**91.15**	68.92	72.99	**70.26**

5 EXPERIMENTS

Datasets. The *cross dataset* benchmark covers ten diverse datasets, including Aircraft (Maji et al., 2013), Cars (Krause et al., 2013), Pets (Parkhi et al., 2012), Flower102 (Nilsback & Zisserman, 2008), Food101 (Bossard et al., 2014), Caltech101 (Fei-Fei, 2004), SUN397 (Xiao et al., 2010), DTD (Cimpoi et al., 2014)), EuroSAT (Helber et al., 2019), and UCF101 (Soomro et al., 2012). The *out-of-distribution(OOD)* benchmark consists of ImageNet(Deng et al., 2009) and its four variants: ImageNet-A (Hendrycks et al., 2021b), ImageNet-V2 (Recht et al., 2019), ImageNet-R (Hendrycks et al., 2021a), and ImageNet-Sketch (Wang et al., 2019).

Implementation details. Following (Karmanov et al., 2024), we build on the pre-trained CLIP (Radford et al., 2021) with ResNet-50 (He et al., 2016) and ViT-B/16 (Dosovitskiy, 2020) as image encoders, and a Transformer (Vaswani et al., 2017) as the text encoder. To reflect real-world online test-time adaptation scenarios, we set the test batch size to 1. We fix n / σ / η to 10 / 0.1 / 0.01 and the cache capacity n_C to 5 across all datasets, and tune only α individually for each dataset. We report ROSE-TTA with two types of text prompts: hand-crafted templates from prior TTA method (Karmanov et al., 2024) (***Hand-crafted***) and GPT-4o (Hurst et al., 2024) generated fine-grained descriptions (***Generated***). Top-1 accuracy is used as the evaluation metric, and all experiments run on a single NVIDIA RTX 4090 GPU.

5.1 COMPARISONS

Baselines. We compare ROSE-TTA with two kinds of methods: (1) *test-time prompt tuning* methods: TPT (Shu et al., 2022), C-TPT (Yoon et al., 2024), and DynaPrompt (Xiao et al., 2025); (2) *test-time adaptation* methods: TDA (Karmanov et al., 2024), BCA (Zhou et al., 2025), PTA (Huang et al., 2026), and TTA-CaP (Sharafi et al., 2026).

Results on the cross-dataset benchmark. We first compare ROSE-TTA with state-of-the-art methods on the *cross-dataset* benchmark (Table 1). For ResNet-50, ROSE-TTA equipped with GPT-4o generated descriptions achieves the best overall performance (63.31%), outperforming the strongest baseline (62.33%) by +0.98. Compared with gradient-based prompt tuning methods, our method performs better on all ten datasets on both backbones. On ViT-B/16, our method again delivers the best overall results (70.26%, +0.88 over PTA), with the most pronounced gain on EuroSAT (66.02%, +2.91 over the best alternative). Notably, even with hand-crafted templates, ROSE-TTA surpasses TDA and BCA on both backbones. The results demonstrate that our method effectively adapts across diverse datasets.

Table 2: **Comparisons on the out-of-distribution benchmark.** The best and runner-up results are bolded and underlined, respectively. *OOD Avg* denotes the average over the four ImageNet variants.

Method	ImageNet	IN-V2	IN-S	IN-A	IN-R	*OOD Avg*	*Mean*
CLIP-ResNet-50 (Radford et al., 2021)	59.81	52.91	35.48	23.24	60.72	43.09	46.43
TPT (Shu et al., 2022)	60.74	54.70	35.09	26.67	59.11	43.89	47.26
C-TPT (Yoon et al., 2024)	61.20	54.80	35.70	25.60	59.70	43.95	47.40
DynaPrompt (Xiao et al., 2025)	61.56	55.12	35.64	27.84	60.63	44.81	48.16
TDA (Karmanov et al., 2024)	61.35	55.54	<u>38.12</u>	30.29	62.58	46.63	49.57
BCA (Zhou et al., 2025)	61.81	<u>56.58</u>	38.08	30.35	62.89	46.98	49.94
PTA (Huang et al., 2026)	61.81	56.23	38.10	30.42	62.99	46.94	49.91
TTA-CaP (Sharafi et al., 2026)	61.74	**56.60**	38.02	30.42	62.86	46.98	49.93
Rose-TTA (Hand-crafted)	<u>62.00</u>	56.45	37.96	<u>30.93</u>	<u>63.01</u>	<u>47.09</u>	<u>50.07</u>
Rose-TTA (Generated)	**62.12**	<u>56.58</u>	**38.20**	**30.99**	**63.92**	**47.42**	**50.36**
CLIP-ViT-B/16	68.34	61.88	48.24	49.89	77.65	59.42	61.20
TPT (Shu et al., 2022)	68.98	63.45	47.94	54.77	77.06	60.81	62.44
C-TPT (Yoon et al., 2024)	69.30	63.40	48.50	52.90	78.00	60.70	62.42
DynaPrompt (Xiao et al., 2025)	69.61	64.67	48.22	56.17	78.17	61.81	63.37
TDA (Karmanov et al., 2024)	69.51	64.67	50.54	60.11	80.24	63.89	65.01
BCA (Zhou et al., 2025)	70.22	64.90	50.87	61.14	80.72	64.41	65.57
PTA (Huang et al., 2026)	70.28	64.85	<u>51.00</u>	61.15	80.79	64.45	65.61
TTA-CaP (Sharafi et al., 2026)	70.26	65.56	50.67	<u>61.20</u>	80.69	64.53	<u>65.68</u>
Rose-TTA (Hand-crafted)	<u>70.54</u>	<u>65.83</u>	50.82	**61.22**	<u>81.81</u>	<u>64.92</u>	<u>66.04</u>
Rose-TTA (Generated)	**70.78**	**66.22**	**51.03**	**61.22**	**82.01**	**65.12**	**66.25**

Results on the OOD benchmark. We also evaluate ROSE-TTA on the OOD benchmark (Table 2), with conclusions consistent with the cross-dataset results. Our method surpasses both prompt tuning methods and recent TTA approaches in the overall accuracy on both backbones, achieving mean accuracies of 50.36% and 66.25% and the best performance on most of the five datasets. The results demonstrate the effectiveness of ROSE-TTA for adaptation across different distribution shifts.

5.2 Ablation Study

Component analysis. We validate each component by removing it from ROSE-TTA on both benchmarks with ViT-B/16. As shown in Table 3, without the prediction stability $\mathbf{d}(\boldsymbol{x})$, cache selection relies solely on entropy and thus admits unreliable samples; without the graph-enhanced terms P_t and P_i, the cache loses class-level completion; and without the refined features $\hat{\boldsymbol{F}}$, the static cache cannot adapt to individual test samples. All three removals degrade the mean accuracy on both benchmarks, confirming the contribution of each component.

Table 3: **Ablations on components for cache predictions.** Removing $\mathbf{d}$ in Eq. (4), P_t and P_i in Eq. (10), or $\hat{\boldsymbol{F}}$ in Eq. (8) leads to performance degradation.

Dataset	w/o $\mathbf{d}$	w/o P_t, P_i	w/o $\hat{\boldsymbol{F}}$	Rose-TTA
Cross-Dataset	69.82	68.56	69.12	**70.26**
OOD	65.67	65.34	65.76	**66.25**

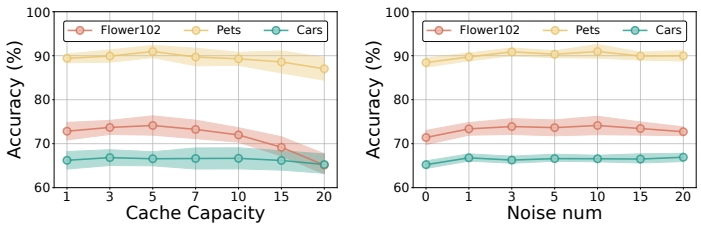
Efficiency efficiency. Table 4 compares time, memory, and accuracy on ImageNet with ViT-B/16. The training-free design keeps ROSE-TTA 26× faster than DiffTPT with +0.48% accuracy and 12× faster than TPT with +1.80%, while remaining 1.03× faster and +1.27% more accurate than TDA. The extra memory over TDA stems from the intermediate tensors for noise-augmented stability es-

Table 4: **Efficiency Analysis on ImageNet.**

Method	Time (min)	Memory (MB)	Acc (%)	Gain (%)
CLIP	9.36	808.96	68.34	—
TPT	585.00	4392.96	68.98	0.64
DiffTPT	1272.00	4710.40	70.30	1.96
TDA	50.00	**860.16**	69.51	1.17
ROSE-TTA	**48.54**	1212.98	**70.78**	**2.44**

timation and graph construction, and this moderate overhead translates directly into the +1.27% accuracy gain, which is a worthwhile trade-off. Further analysis is provided in Appendix C.1.

Effect of cache capacity n_C. We vary the per-class cache capacity from 1 to 20 on three datasets with ViT-B/16 (Fig. 3a). Accuracy peaks at a capacity of 5 and stays stable across most settings, indicating that structural completion and the doubly robust replacement jointly regulate cache content. The exception is fine-grained datasets such as Flower102, where large capacities accumulate redundant samples that blur decision boundaries among visually similar categories, and early, less reliable entries further amplify this instability under online adaptation.



(a) Varying cache capacity (n_c) performance.

(b) Varying number of noise-augmented features (n) performance.

Figure 3: Performance analysis under different hyperparameter settings. (a) Varying cache capacity (n_c) performance. (b) Varying number of noise-augmented features (n) performance.

Effect of noise augmentation number n. We ablate the number of noise augmentations used for stability estimation (Fig. 3b). Compared with $n = 0$, noise augmentation improves accuracy in most cases, and the gain grows mildly with more augmentations, peaking around 10. Even with as few as 3 augmentations, the stability estimation is already reliable, indicating that the uncertainty-aware mechanism is robust to this hyperparameter. Beyond the peak, excessive augmentations yield diminishing returns and even introduce redundant perturbations.

Progressive adaptation analysis. To trace the online behavior of ROSE-TTA, we evaluate accuracy at multiple checkpoints of the test stream on ImageNet and EuroSAT (Figure 4). ROSE-TTA stays ahead of TDA and BCA throughout the entire process, and the margin is most pronounced in the early stage (10%–30% of the stream), where TDA suffers unstable predictions and BCA, though free from entropy-based selection in its prediction, still collapses early for lacking a mechanism to admit high-quality samples. This confirms that uncertainty-aware cache construction and information completion provide reliable guidance from the beginning of adaptation, and as the stream progresses the gap narrows yet persists, showing that the early advantage does not trade off long-term effectiveness.

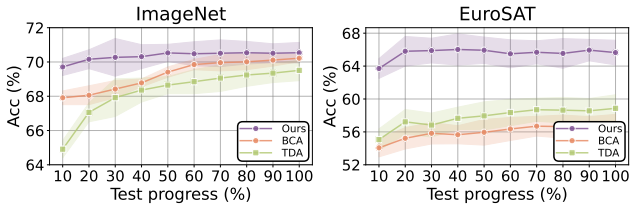


Figure 4: **Progressive adaptation analysis** on ImageNet and EuroSAT.

6 Conclusion

In this work, we propose ROSE-TTA, a reliable, test-time–enhanced caching framework that improves CLIP's zero-shot performance across datasets and distribution shifts. ROSE-TTA constructs a more reliable cache by selecting more stable test features via a noise-aware uncertainty strategy that integrates entropy minimization with noise-based stability. During cache utilization, we enhance its global class information through a graph-based structural completion strategy that mitigates class imbalance and strengthens cache representations. The method also injects local instance information with a sample-specific refinement module to adapt cached features to each incoming test sample. Extensive experiments on 15 datasets demonstrate the effectiveness of ROSE-TTA, providing a robust and efficient approach to test-time cache construction and utilization.

AI USE STATEMENT

In this work, we used the generative AI assistant Claude Opus 5 (Anthropic) for editing and polishing parts of the manuscript text, for formatting tables and references, and for drafting visualization scripts for the cache statistics and progressive adaptation figures under our detailed specifications. We have not used generative AI tools for the problem formulation, the theoretical analysis, the experimental design and implementation, or the interpretation of results. All AI-assisted text and artifacts were reviewed, verified, and revised by the authors. We take responsibility for the final content of this work, including any text or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses only publicly available benchmark datasets for test-time adaptation research and introduces no new data collection. It does not involve any human subjects, personal data, or sensitive information. The proposed method adapts open pre-trained vision-language models at test time without modifying their weights, and all experiments are conducted for standard academic benchmark evaluation. We are not aware of any direct negative societal impact of this work; as with any method that improves model performance under distribution shift, we note that its benefits depend on the downstream deployment context.

REPRODUCIBILITY STATEMENT

All experiments are built on public datasets and open-weight models: the CLIP backbones (ResNet-50 and ViT-B/16), the cross-dataset benchmark (Aircraft, Caltech101, Cars, DTD, EuroSAT, Flower102, Food101, Pets, SUN397, and UCF101), the OOD benchmark (ImageNet and its A/V2/R/S variants), and the GPT-4o generated text descriptions. The method is training-free and involves no model training, and all hyperparameters are reported in the implementation details.

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 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.

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. Springer, 2014.

Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 295–305, 2022.

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*, 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 *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Guodong Du, Jia Zhang, Min Jiang, Jinyi Long, Yaojin Lin, Shaozi Li, and Kay Chen Tan. Graph-based class-imbalance learning with label enhancement. *IEEE transactions on neural networks and learning systems*, 34(9):6081–6095, 2021.

Li Fei-Fei. Learning generative visual models from few training examples. In *Workshop on Generative-Model Based Vision, IEEE Proc. CVPR, 2004*, 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 *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2704–2714, 2023.

Huan-ang Gao, Jiayi Geng, Wenyue Hua, Mengkang Hu, Xinzhe Juan, Hongzhang Liu, Shilong Liu, Jiahao Qiu, Xuan Qi, Yiran Wu, et al. A survey of self-evolving agents: What, when, how, and where to evolve on the path to artificial super intelligence. *arXiv preprint arXiv:2507.21046*, 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. *Advances in Neural Information Processing Systems*, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

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 *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 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*, pp. 15262–15271, 2021b.

Zhaohong Huang, Yuxin Zhang, Wenjing Liu, Fei Chao, and Rongrong Ji. Prototype-based test-time adaptation of vision-language models. *arXiv preprint arXiv:2604.21360*, 2026.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pp. 4904–4916. PMLR, 2021.

Adilbek Karmanov, Dayan Guan, Shijian Lu, Abdulmotaleb El Saddik, and Eric Xing. Efficient test-time adaptation of vision-language models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

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*, pp. 554–561, 2013.

Jonghyun Lee, Dahuin Jung, Saehyung Lee, Junsung Park, Juhyeon Shin, Uiwon Hwang, and Sungroh Yoon. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In *International Conference on Learning Representations*, 2024.

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

Zachary Nado, Shreyas Padhy, D Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift. *arXiv preprint arXiv:2006.10963*, 2020.

A Tuan Nguyen, Thanh Nguyen-Tang, Ser-Nam Lim, and Philip HS Torr. Tipi: Test time adaptation with transformation invariance. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24162–24171, 2023a.

A Tuan Nguyen, Thanh Nguyen-Tang, Ser-Nam Lim, and Philip HS Torr. Tipi: Test time adaptation with transformation invariance. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24162–24171. IEEE, 2023b.

Khanh-Binh Nguyen, Phuoc-Nguyen Bui, Hyunseung Choo, and Duc Thanh Nguyen. Adaptive cache enhancement for test-time adaptation of vision-language models. *arXiv preprint arXiv:2508.07570*, 2025.

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*, pp. 722–729. IEEE, 2008.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 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*, pp. 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*, pp. 5389–5400. PMLR, 2019.

Afshar Shamsi, Rejisa Becirovic, Ahmadreza Argha, Ehsan Abbasnejad, Hamid Alinejad-Rokny, and Arash Mohammadi. Etage: enhanced test time adaptation with integrated entropy and gradient norms for robust model performance. In *2025 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 5907–5911. IEEE, 2025.

Masoumeh Sharafi, Muhammad Osama Zeeshan, Soufiane Belharbi, Alessandro Lameiras Koerich, Marco Pedersoli, and Eric Granger. Test-time adaptation via cache personalization for facial expression recognition in videos. *arXiv preprint arXiv:2603.21309*, 2026.

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.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

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.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pp. 9229–9248. PMLR, 2020.

Anique Tahir, Lu Cheng, Ruocheng Guo, and Huan Liu. Distributional shift adaptation using domain-specific features. In *2022 IEEE International Conference on Big Data (Big Data)*, pp. 5593–5597. IEEE, 2022.

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

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.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Yanbin Wei, Shuai Fu, Weisen Jiang, Zejian Zhang, Zhixiong Zeng, Qi Wu, James Kwok, and Yu Zhang. Gita: Graph to visual and textual integration for vision-language graph reasoning. *Advances in neural information processing systems*, 37:44–72, 2024.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025.

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*, pp. 3485–3492. IEEE, 2010.

Zehao Xiao, Shilin Yan, Jack Hong, Jiayin Cai, Xiaolong Jiang, Yao Hu, Jiayi Shen, Qi Wang, and Cees G Snoek. Dynaprompt: Dynamic test-time prompt tuning. In *International Conference on Learning Representations*, 2025.

Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*, 2021.

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

Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. *Advances in neural information processing systems*, 35:38629–38642, 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*, 2024.

Yuchao Zhang, Hao Wang, Fan Zhang, Qirui Mi, Mengyue Yang, Yisen Wang, Jun Wang, Haoxuan Li, and Zhouchen Lin. Learning to memorize with attributive and associative memory for online test-time adaptation of vision-language models. In *Forty-third International Conference on Machine Learning*, 2026.

Bowen Zhao, Chen Chen, and Shu-Tao Xia. Delta: degradation-free fully test-time adaptation. *arXiv preprint arXiv:2301.13018*, 2023.

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 *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29999–30009. IEEE, 2025.

Algorithm 1 ROSE-TTA: online test-time adaptation for each test sample $\boldsymbol{x}$

Require: Test stream $\{\boldsymbol{x}_t\}$; frozen CLIP encoders $\mathcal{F}_{\theta_I}, \mathcal{F}_{\theta_T}$; text embeddings $\boldsymbol{W}_C$; per-class cache $\mathcal{C}$ with capacity $n_{\mathcal{C}}$; hyperparameters n, σ, η, α
Ensure: Predictions $\{\hat{y}_t\}$; updated cache $\mathcal{C}$

for each test sample $\boldsymbol{x}$ in the stream **do**
  $\boldsymbol{f}_{\boldsymbol{x}} \leftarrow \mathcal{F}_{\theta_I}(\boldsymbol{x})$; $P_{clip} \leftarrow p(\hat{y} \mid \boldsymbol{x})$; $\hat{c} \leftarrow \arg\max_c P_{clip}$
  // Uncertainty-aware cache construction
  $\tilde{\boldsymbol{f}}_{\boldsymbol{x}}^i \leftarrow \boldsymbol{f}_{\boldsymbol{x}} + \epsilon_i \cdot \sigma, \quad i = 1, \ldots, n$ ▷ Eq. (3): noise-augmented features
  $\mathbf{d}(\boldsymbol{x}) \leftarrow \frac{1}{n} \sum_{i=1}^{n} \left| p(\hat{y} = \hat{c} \mid \tilde{\boldsymbol{f}}_{\boldsymbol{x}}^i) - p(\hat{y} = \hat{c} \mid \boldsymbol{f}_{\boldsymbol{x}}) \right|$ ▷ Eq. (4): prediction stability
  if $|\mathcal{C}_{\hat{c}}| < n_{\mathcal{C}}$: append $\boldsymbol{x}$ to $\mathcal{C}_{\hat{c}}$
  else if $\mathcal{H}(p(y|\boldsymbol{x})) < \mathcal{H}(p(y|\boldsymbol{x}_{j^*}))$ **and** $\mathbf{d}(\boldsymbol{x}) < \mathbf{d}(\boldsymbol{x}_{j^*})$:
    replace $\boldsymbol{x}_{j^*}$ with $\boldsymbol{x}$ ▷ Eq. (5): doubly robust replacement
  // Graph-based structural completion
  $w_j \leftarrow \sqrt{(1 - \mathbf{d}_j)/\mathcal{H}_j}$; $\boldsymbol{W}_{jk} \leftarrow \sqrt{w_j \cdot w_k}$ ▷ reliability weights
  $\hat{\boldsymbol{G}}_c \leftarrow \boldsymbol{F}\boldsymbol{F}^T \odot \boldsymbol{W}$; $\boldsymbol{G}_t \leftarrow \boldsymbol{W}_C \boldsymbol{W}_C^T$; $\boldsymbol{M} \leftarrow$ binary mapping matrix ▷ cache-, text-, gate-graph
  $\hat{\boldsymbol{G}} \leftarrow \text{softmax}(\boldsymbol{G}_t \boldsymbol{M} \hat{\boldsymbol{G}}_c \boldsymbol{M}^T \boldsymbol{G}_t^T)$ ▷ Eq. (6): categorical graph
  $P_{\text{t}} \leftarrow \boldsymbol{f}_{\boldsymbol{x}} (\hat{\boldsymbol{G}} \boldsymbol{W}_C^T)^T$; $P_{\text{i}} \leftarrow \boldsymbol{f}_{\boldsymbol{x}} (\hat{\boldsymbol{G}} \boldsymbol{M} \boldsymbol{F})^T$ ▷ Eq. (7): textual and visual paths
  // Sample-specific completion
  $P_n \leftarrow \text{softmax}\big(\frac{1}{n} \sum_{i=1}^{n} P_{cache}(\tilde{\boldsymbol{f}}_{\boldsymbol{x}}^i)\big)$ ▷ averaged cache prediction
  $\gamma \leftarrow 1 - \mathcal{H}(P_n)$; $\zeta \leftarrow P_n - 1/C$ ▷ control factors: intensity, direction
  $\hat{\boldsymbol{F}} \leftarrow \boldsymbol{F} + \eta \cdot \gamma \cdot \zeta \cdot \boldsymbol{f}_{\boldsymbol{x}}$ ▷ Eq. (8): pseudo-gradient update
  $\hat{P}_{cache} \leftarrow A(\boldsymbol{f}_{\boldsymbol{x}} \hat{\boldsymbol{F}}^T)\, \hat{\boldsymbol{L}}$ ▷ Eq. (9): refined cache prediction
  // Final prediction (Sec. 3.2)
  $P_{rose} \leftarrow P_{clip} + \alpha \cdot (\hat{P}_{cache} + P_{\text{t}} + P_{\text{i}})$ ▷ Eq. (10): final prediction
  yield $\hat{y} = \arg\max_c P_{rose}$
end for

APPENDIX

A ALGORITHM OF ROSE-TTA

Algorithm 1 summarizes the complete online procedure of ROSE-TTA for each incoming test sample.

B COMPARISON OF THE PSEUDO-GRADIENT WITH ACTUAL GRADIENT

B.1 THEORETICAL ANALYSIS: PSEUDO-GRADIENT APPROXIMATION

Connection to entropy minimization. The common gradient-based method for test-time adaptation calculates gradients based on entropy minimization $H(p) = -\sum_{k=1}^{C} p_k \log p_k$, where p_k denotes the predicted probability for class k and C is the number of classes. In our sample-specific cache refinement (Eq. 8), the pseudo gradient consists of two components: $\gamma = 1 - \mathcal{H}(P_n)$ and $\zeta = P_n - \frac{1}{C}$, where $P_n = \text{softmax}\left(\frac{1}{n} \sum_{i=1}^{n} P_{cache}(\tilde{\boldsymbol{f}}_{\boldsymbol{x}}^i)\right)$ is the averaged cache prediction based on noise-augmented features and $\mathcal{H}(\cdot)$ denotes the normalized entropy. Here, γ controls the update intensity (higher confidence yields stronger updates), and ζ controls the update direction. This design is motivated by the behavior of entropy minimization, which pushes probabilities above the uniform distribution higher and pulls those below uniform lower.

In cache-based TTA, the prediction for class c is computed via the similarity between the test feature $\boldsymbol{f}_{\boldsymbol{x}}$ and cached features $\{f_i\}_{i=1}^{N}$ in $\boldsymbol{F}$. For a cached feature f_c belonging to class c, the corresponding

logit is $z_c = \boldsymbol{f}_{\boldsymbol{x}}^T f_c$, and p_c is the probability after softmax. The gradient of entropy with respect to f_c can be decomposed as:

$$\frac{\partial H}{\partial f_c} = \frac{\partial H}{\partial p_c}\frac{\partial p_c}{\partial z_c}\frac{\partial z_c}{\partial f_c} = \frac{\partial H}{\partial p_c}\frac{\partial p_c}{\partial z_c}\boldsymbol{f}_{\boldsymbol{x}}, \tag{11}$$

where $\frac{\partial z_c}{\partial f_c} = \boldsymbol{f}_{\boldsymbol{x}}$ follows from the inner product $z_c = \boldsymbol{f}_{\boldsymbol{x}}^T f_c$.

The first term is $\frac{\partial H}{\partial p_c} = \frac{\partial}{\partial p_c}\left(-\sum_{k=1}^{C} p_k \log p_k\right) = -\log p_c - 1$, serving as the directional component that determines how each probability should be adjusted. The second term stems from the softmax Jacobian: for softmax $p_c = \frac{e^{z_c}}{\sum_k e^{z_k}}$, we have

$$\frac{\partial p_c}{\partial z_c} = p_c(1 - p_c). \tag{12}$$

Critically, since $p_c \in (0, 1)$, this term is always positive and only modulates the update magnitude without affecting its direction. Therefore, the directional behavior of entropy minimization is determined by $\frac{\partial H}{\partial p_c} = -\log p_c - 1$, while the update naturally includes the test feature $\boldsymbol{f}_{\boldsymbol{x}}$ as a multiplicative factor.

Linear approximation via Taylor expansion. The nonlinearity in $-\log p_c - 1$ can be numerically unstable when probabilities approach 0 or 1, especially under distribution shifts. We derive a stable linear approximation by performing first-order Taylor expansion around the uniform distribution $p_c = \frac{1}{C}$:

$$-\log p_c - 1 \approx -\log\frac{1}{C} - 1 + \frac{d}{dp_c}(-\log p_c - 1)\Big|_{p_c=\frac{1}{C}}\left(p_c - \frac{1}{C}\right). \tag{13}$$

Computing the derivative: $\frac{d}{dp_c}(-\log p_c - 1) = -\frac{1}{p_c}$, which evaluated at $p_c = \frac{1}{C}$ gives $-C$. The constant term $-\log\frac{1}{C} - 1 = \log C - 1$ does not affect the gradient direction and can be absorbed into the pseudo learning rate η. Thus:

$$-\log p_c - 1 \approx -C\left(p_c - \frac{1}{C}\right) + \text{const}. \tag{14}$$

Combining with the test feature factor, the gradient direction is approximated as $-C(p_c - \frac{1}{C}) \cdot \boldsymbol{f}_{\boldsymbol{x}}$. Therefore, **our cache update in Eq. (8): $\hat{\boldsymbol{F}} = \boldsymbol{F} + \eta \cdot \gamma \cdot \zeta \cdot \boldsymbol{f}_{\boldsymbol{x}}$ directly implements this approximation**.

Table 5: Comparison of pseudo-gradient vs true gradient

Method	Aircraft	Caltech101	Cars	DTD	EuroSAT	Flower102	Food101	Pets	SUN397	UCF101	ImageNet	ImageNet-A	ImageNet-V2	ImageNet-R	ImageNet-S	Mean
Pseudo-gradient	25.86	94.76	67.06	48.00	65.64	74.17	86.20	90.95	68.00	71.34	70.54	61.22	65.83	81.81	50.82	68.15
True gradient	26.73	94.93	66.92	48.78	66.01	74.25	88.13	90.46	67.43	71.23	71.11	61.11	65.9	81.33	51.32	68.38

B.2 Empirical Results

We also conducted experiments using true-gradient refinement based on entropy minimization. The results in Table 5 are provided in the following table: Across all datasets tested, we found that the performance of the pseudo-gradient closely matches that of the true entropy-minimization gradient with less computational cost, empirically demonstrating the proposed pseudo-gradient.

C More Efficiency Details

C.1 Computational Efficiency Analysis Of Different Components

We profile the runtime and memory footprint of each module during inference, as shown in Table 6.

Table 6: Breakdown of computational costs for each component during test-time adaptation.

Component	**Time (min)**	**Memory (MB)**
Noise Augmentation	1.55	209.13
Graph Construction	5.81	309.01
Sample-Specific Adapt.	1.44	22.84
Overall	**48.54**	**1212.98**

As Table 6 reveals, the three proposed components take 8.80 min and 541 MB , i.e., 18.1% of the total runtime and 44.6% of the total memory on ImageNet. Unlike TPT (Shu et al., 2022) and DiffTPT (Feng et al., 2023), which require iterative gradient descent over prompts at test time, our method operates entirely in the forward pass, replacing backpropagation with the pseudo-gradient update. The noise augmentation module costs only 1.55 min (3.2% of the runtime), as the n perturbed features are generated by a single matrix addition per sample and require no extra encoder forward passes. The graph construction module is the dominant overhead at 5.81 min (12.0%), as it involves matrix products over N cached features and C classes; these operations are highly parallelizable on modern GPUs, so the structural completion remains affordable. The sample-specific refinement costs the least (1.44 min, 3.0%), since it only updates the cached features with a scaled addition of the test feature.

C.2 Computational Efficiency with Varying Noise Budgets

Table 7 demonstrates the scalability of our noise-enhanced uncertainty estimation on ImageNet. Increasing the noise budget from $n = 1$ to $n = 20$ raises the per-sample cost from 58.36 to 59.69 ms, i.e., merely 1.11 min (2.28%) over the entire test set, and the memory from 21.40 to 29.64 KB/sample. This low overhead stems from the design that each perturbed feature is generated by one matrix addition without extra encoder forward passes. Our default choice $n = 10$ follows the accuracy analysis in Fig. 3, and more conservative ($n = 3$) or aggressive ($n = 20$) settings remain viable, as the accuracy variation is small while the overhead stays marginal.

Table 7: Scalability with varying noise budgets

Noise Num	Testing Time (ms/sample)	Memory Usage (KB/sample)
1	58.36	21.40
3	58.51	22.29
5	58.68	23.17
10	58.75	25.28
15	59.17	27.53
20	59.69	29.64

D Extra Experiment Results

D.1 More Experiments on Standard OOD Corruption Benchmarks

To show the effectiveness of the proposed method on more benchmarks, we have incorporated additional experiments on CIFAR-10-C, CIFAR-100-C, and ImageNet-C. We reproduced zero-shot CLIP and TDA on the datasets for comparisons. The results are in the following Table 8. Our **ROSE-TTA** achieves better overall performance on these three datasets compared with CLIP, TDA and BCA, demonstrating the effectiveness of our method on these challenging distribution shifts.

Table 8: Mean accuracy (%) on CIFAR-10C, CIFAR-100C, and ImageNet-C - TTA mean accuracy of the 15 corruptions at severity level 5.

Method	Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Brightness	Contrast	Elastic	Pixelate	JPEG	Mean
ImageNet-C																
CLIP	18.56	12.11	19.45	31.22	23.12	32.15	26.25	34.59	34.17	45.35	57.11	30.94	48.93	45.01	35.14	32.94
TDA	19.54	13.46	19.90	32.76	26.35	34.32	27.79	36.42	35.47	47.79	58.16	31.92	49.22	45.05	35.80	34.26
BCA	19.76	13.44	20.38	33.08	27.84	34.62	27.88	38.12	35.96	52.26	59.02	32.38	49.20	45.08	35.94	34.99
Ours	**20.18**	**13.57**	**21.56**	**33.67**	**29.78**	**35.19**	**28.12**	**40.17**	**36.66**	**60.89**	**59.94**	**33.26**	**49.38**	**45.18**	**36.21**	**36.25**
CIFAR-10C																
CLIP	46.22	50.17	45.23	65.45	43.33	60.14	68.54	69.12	71.11	55.23	80.45	44.53	55.25	52.17	60.31	57.82
TDA	48.66	53.24	46.26	67.09	43.55	63.17	69.97	70.02	72.57	56.13	81.90	45.59	56.84	53.71	60.91	59.31
BCA	49.08	53.72	46.18	68.00	44.55	65.92	70.82	71.38	72.98	56.75	81.98	47.05	56.35	54.02	61.52	60.03
Ours	**50.14**	**54.76**	**46.31**	**69.73**	**46.17**	**69.56**	**72.45**	**73.52**	**74.23**	**57.86**	**81.96**	**49.23**	**56.97**	**54.75**	**62.69**	**61.36**
CIFAR-100C																
CLIP	28.15	28.18	20.17	38.22	20.85	35.24	42.56	40.14	42.88	30.13	52.19	20.89	29.66	24.15	33.11	32.43
TDA	29.18	29.84	24.68	39.44	20.99	37.44	43.84	42.86	44.24	30.72	53.72	22.85	30.78	25.13	33.81	33.97
BCA	29.66	30.10	25.96	40.36	21.64	37.70	43.62	42.92	46.28	30.79	54.90	23.74	32.16	25.18	33.88	34.51
Ours	**30.22**	**30.46**	**27.89**	**41.58**	**22.45**	**38.11**	**44.19**	**43.00**	**48.69**	**30.87**	**56.24**	**24.78**	**33.94**	**25.22**	**33.96**	**35.44**

D.2 Ablation Study of α

To further investigate the impact of the hyperparameter α, we have conducted additional experiments across all datasets to evaluate the performance under different values of α varying in the range of $\{0, 1, 3, 5, 10\}$. Our results in Table 9 show that the performance remains relatively stable with changes in α and significantly outperforms the case where α is set to 0. Notably, across the majority of α values tested, our method consistently outperforms BCA, demonstrating that even with randomly selected α, our approach maintains its effectiveness. This robustness validates the effectiveness of our cache-based refinement strategy and reduces the need for extensive hyperparameter tuning.This confirms that our cache-based refined cache prediction approach is effective.

Table 9: Performance under different α values

Dataset	Aircraft	Caltech101	Cars	DTD	EuroSAT	Flower102	Food101	Pets	SUN397	UCF101	ImageNet	ImageNet-A	ImageNet-V2	ImageNet-R	ImageNet-S	Mean
$\alpha = 0$	24.70	94.16	65.80	44.50	47.67	71.42	86.10	89.15	66.62	66.77	70.04	59.43	64.35	80.33	49.40	65.36
BCA	**28.59**	94.69	66.86	53.59	56.63	73.12	85.97	90.43	**68.41**	67.59	70.22	61.14	64.90	80.72	50.87	67.58
$\alpha = 1$	24.77	94.45	66.92	47.78	60.12	73.42	86.12	**90.95**	67.43	**71.34**	70.11	60.11	64.90	81.33	50.32	67.40
$\alpha = 3$	**25.86**	94.50	**67.06**	**48.00**	**65.64**	73.67	86.15	90.42	**68.00**	70.33	**70.54**	**61.22**	65.68	81.56	50.45	**67.94**
$\alpha = 5$	25.23	**94.76**	66.77	47.90	65.50	**74.17**	86.16	90.56	67.56	69.57	70.33	60.58	65.77	**81.681**	**50.82**	67.83
$\alpha = 10$	25.69	94.68	67.02	47.22	63.25	74.11	**86.16**	90.66	67.78	70.12	70.42	61.01	**65.83**	81.23	50.64	67.73

D.3 Empirical Results of σ

As shown in Table 10, we conducted a sensitivity analysis on $\sigma \in \{0, 0.05, 0.1, 0.15, 0.2\}$ across all datasets, where $\sigma = 0$ corresponds to using entropy only. The performance is relatively the best around $\sigma = 0.1$. If σ is too small, the perturbations become negligible, failing to meaningfully test the sample's robustness. If σ is too large, excessive noise can corrupt the semantic information in the features, causing even truly reliable samples to exhibit unstable predictions. This validates our choice of $\sigma = 0.1$ as a robust default.

D.4 Performance Gains Comparison with Varying Noise Level

In Table 11, we have added the comparisons with the no-noise-augmentation baseline ($n = 0$) to more clearly demonstrate the benefits of noise augmentation. Below, we present the performance of each dataset at different noise levels, where "Noise Gains" represents the average performance gain

Table 10: Performance under different σ values

σ	Cross Dataset Avg	OOD Avg
0	68.21	65.13
0.05	69.17	66.01
0.1	**69.20**	**66.04**
0.15	69.11	65.98
0.2	68.88	65.56

Table 11: Impact of noise augmentation budget on accuracy. We report accuracy ± std and gain ± std over $n = 0$ baseline.

Dataset	$n=0$	$n=1$		$n=3$		$n=5$		$n=10$		$n=15$		$n=20$	
	Acc	Acc	Gain	Acc	Gain	Acc	Gain	Acc	Gain	Acc	Gain	Acc	Gain
Aircraft	$23.51_{\pm.41}$	$24.78_{\pm.82}$	$1.27_{\pm.23}$	$25.14_{\pm.77}$	$1.63_{\pm.24}$	$25.16_{\pm.12}$	$1.65_{\pm.11}$	$25.86_{\pm.14}$	$\mathbf{2.35}_{\pm.11}$	$25.14_{\pm.79}$	$1.63_{\pm.22}$	$25.11_{\pm.78}$	$1.60_{\pm.25}$
Caltech101	$94.16_{\pm.01}$	$94.52_{\pm.03}$	$0.36_{\pm.01}$	$94.62_{\pm.05}$	$0.46_{\pm.02}$	$94.63_{\pm.01}$	$0.47_{\pm.01}$	$94.76_{\pm.05}$	$\mathbf{0.60}_{\pm.03}$	$94.61_{\pm.02}$	$0.45_{\pm.01}$	$94.59_{\pm.02}$	$0.43_{\pm.01}$
Cars	$65.73_{\pm.23}$	$66.78_{\pm.76}$	$1.05_{\pm.33}$	$66.27_{\pm.74}$	$0.54_{\pm.43}$	$66.61_{\pm.66}$	$0.88_{\pm.56}$	$67.06_{\pm.54}$	$\mathbf{1.33}_{\pm.64}$	$66.57_{\pm.45}$	$0.84_{\pm.35}$	$66.91_{\pm.44}$	$1.18_{\pm.52}$
DTD	$44.61_{\pm.39}$	$47.34_{\pm.76}$	$2.73_{\pm.63}$	$47.38_{\pm.89}$	$2.77_{\pm.77}$	$47.56_{\pm.98}$	$2.95_{\pm.82}$	$48.00_{\pm.78}$	$\mathbf{3.39}_{\pm.67}$	$47.64_{\pm.33}$	$3.03_{\pm.23}$	$47.79_{\pm.56}$	$3.18_{\pm.55}$
EuroSAT	$58.67_{\pm.24}$	$64.25_{1.01}$	$5.58_{\pm.78}$	$64.86_{\pm.74}$	$6.19_{\pm.69}$	$64.34_{\pm.56}$	$5.67_{\pm.73}$	$65.64_{\pm.62}$	$\mathbf{6.97}_{\pm.54}$	$65.12_{\pm.29}$	$6.45_{\pm.13}$	$65.32_{\pm.27}$	$6.65_{\pm.09}$
Flower102	$71.41_{\pm.23}$	$73.37_{1.45}$	$1.96_{\pm.79}$	$73.89_{1.32}$	$2.48_{1.23}$	$73.65_{1.57}$	$2.24_{1.21}$	$74.17_{1.02}$	$\mathbf{2.76}_{\pm.98}$	$73.44_{1.23}$	$2.03_{1.01}$	$72.71_{\pm.99}$	$1.30_{\pm.45}$
Food101	$86.11_{\pm.01}$	$86.18_{\pm.00}$	$0.07_{\pm.01}$	$86.17_{\pm.01}$	$0.06_{\pm.01}$	$86.15_{\pm.03}$	$0.04_{\pm.02}$	$86.21_{\pm.00}$	$\mathbf{0.10}_{\pm.01}$	$86.20_{\pm.01}$	$0.09_{\pm.01}$	$86.19_{\pm.03}$	$0.08_{\pm.02}$
Pets	$89.17_{\pm.43}$	$89.72_{\pm.89}$	$0.55_{\pm.42}$	$90.87_{\pm.67}$	$1.70_{\pm.78}$	$90.32_{\pm.79}$	$1.15_{\pm.72}$	$90.95_{1.01}$	$\mathbf{1.78}_{\pm.92}$	$89.92_{\pm.65}$	$0.75_{\pm.52}$	$89.97_{\pm.78}$	$0.80_{\pm.62}$
SUN397	$66.62_{\pm.12}$	$67.88_{\pm.13}$	$1.26_{\pm.08}$	$67.76_{\pm.24}$	$1.14_{\pm.15}$	$67.54_{\pm.03}$	$0.92_{\pm.07}$	$68.00_{\pm.15}$	$\mathbf{1.38}_{\pm.14}$	$67.36_{\pm.11}$	$0.74_{\pm.08}$	$67.68_{\pm.15}$	$1.06_{\pm.21}$
UCF101	$68.12_{\pm.25}$	$70.23_{\pm.67}$	$2.11_{\pm.42}$	$70.68_{\pm.56}$	$2.56_{\pm.55}$	$71.29_{\pm.78}$	$3.17_{\pm.74}$	$71.34_{\pm.89}$	$\mathbf{3.22}_{\pm.92}$	$71.11_{\pm.85}$	$2.99_{\pm.93}$	$70.98_{1.01}$	$2.86_{\pm.97}$
ImageNet	$70.05_{\pm.01}$	$70.24_{\pm.04}$	$0.19_{\pm.04}$	$70.48_{\pm.07}$	$0.43_{\pm.06}$	$70.29_{\pm.04}$	$0.24_{\pm.03}$	$70.54_{\pm.06}$	$\mathbf{0.49}_{\pm.05}$	$70.31_{\pm.05}$	$0.26_{\pm.05}$	$70.44_{\pm.04}$	$0.39_{\pm.03}$
ImageNet-A	$59.43_{\pm.06}$	$60.87_{\pm.06}$	$1.44_{\pm.01}$	$61.15_{\pm.02}$	$1.72_{\pm.03}$	$60.96_{\pm.03}$	$1.53_{\pm.03}$	$61.22_{\pm.05}$	$\mathbf{1.79}_{\pm.06}$	$61.21_{\pm.02}$	$1.78_{\pm.05}$	$61.03_{\pm.04}$	$1.60_{\pm.03}$
ImageNet-V2	$64.35_{\pm.21}$	$65.11_{\pm.25}$	$0.76_{\pm.11}$	$65.25_{\pm.31}$	$0.90_{\pm.31}$	$65.76_{\pm.09}$	$1.41_{\pm.15}$	$65.83_{\pm.32}$	$\mathbf{1.48}_{\pm.25}$	$65.81_{\pm.13}$	$1.46_{\pm.11}$	$65.72_{\pm.23}$	$1.37_{\pm.22}$
ImageNet-R	$80.37_{\pm.13}$	$81.54_{\pm.11}$	$1.17_{\pm.04}$	$81.64_{\pm.13}$	$1.27_{\pm.05}$	$81.79_{\pm.45}$	$1.42_{\pm.25}$	$81.81_{\pm.03}$	$\mathbf{1.44}_{\pm.14}$	$81.77_{\pm.14}$	$1.40_{\pm.13}$	$81.57_{\pm.09}$	$1.20_{\pm.11}$
ImageNet-S	$49.41_{\pm.02}$	$50.67_{\pm.05}$	$1.26_{\pm.02}$	$50.74_{\pm.05}$	$1.33_{\pm.03}$	$50.72_{\pm.07}$	$1.31_{\pm.04}$	$50.82_{\pm.07}$	$\mathbf{1.41}_{\pm.05}$	$50.81_{\pm.04}$	$1.40_{\pm.01}$	$50.69_{\pm.03}$	$1.28_{\pm.02}$

Table 12: Cache replacement statistics across different datasets. We report the replacement number (Num) and replacement ratio (Ratio) for each dataset.

	ImageNet Variants										*Cross-Dataset Benchmarks*															
	ImageNet		ImageNet-A		ImageNet-V2		ImageNet-R		ImageNet-S		Aircraft		Caltech101		Cars		DTD		EuroSAT		Flower102		Food101		Pets		SUN397		UCF101	
Method	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio	Num	Ratio
TDA	15762	0.31	1456	0.19	2384	0.24	7825	0.27	13184	0.26	528	0.16	443	0.18	1339	0.16	258	0.14	118	0.02	513	0.29	8546	0.28	314	0.09	2178	0.11	236	0.05
Ours	8846	**0.18**	801	**0.11**	1325	**0.13**	4375	**0.15**	7346	**0.15**	300	**0.09**	246	**0.10**	744	**0.09**	143	**0.08**	65	**0.01**	285	**0.16**	4753	**0.16**	174	**0.05**	1211	**0.06**	131	**0.03**

relative to the no-noise baseline. As shown, noise augmentation obviously improves classification performance in most cases, and our method exhibits good stability across different noise levels.

D.5 Cache Dynamics Analysis

Table 12 reveals the effectiveness of our doubly robust cache construction. Across all 15 datasets, the average replacement ratio is only **12.7%**, indicating that cached samples remain highly stable throughout the online testing process. This low replacement rate validates that our noise-enhanced uncertainty estimation effectively identifies truly representative prototypes that persist under distribution shifts. Notably, the replacement behavior exhibits dataset-dependent adaptivity. ImageNet variants (A/V2/R/S) show moderate replacement ratios of 11-18%, reflecting our cache's ability to adapt appropriately to different types and severities of natural corruptions while maintaining core prototypical features. In contrast, cross-dataset benchmarks display more diverse replacement patterns: fine-grained datasets like EuroSAT (0.01) and UCF101 (0.03) exhibit minimal replacement, suggesting that their test distributions closely align with initial cache entries, whereas coarse-grained datasets like Flower102 (0.16) and Food101 (0.16) require more frequent updates to capture intra-class variability. The consistently low replacement ratios across diverse benchmarks validate that our dual-criterion selection (entropy + stability) successfully filters noisy samples while allowing necessary adaptation, leading to robust test-time performance.

D.6 Robustness to random sample ordering.

As our method updates the cache online, the performance can be influenced by the order of the test samples. To show the robustness of our method regarding the test sample orders, we conducted

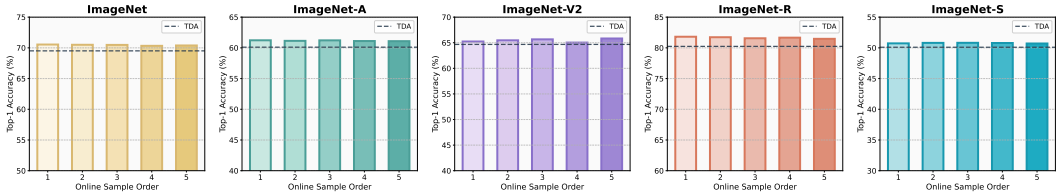


Figure 5: **Sensitivity to test-time sample order on five OOD datasets.** The performance is stable across datasets, demonstrating the robustness of the algorithm to different test example orders.

Table 13: Comparison of cache eviction strategies.

Metric	Entropy only	Similarity	Random	FIFO	LRU	Ours
Cross Dataset Avg	68.21	68.13	67.11	64.68	63.21	**69.02**
OOD Avg	65.13	64.99	63.64	62.25	60.02	**66.04**

multiple experiments on the OOD dataset with different sample orders for 5 rounds. We observe that there are fluctuations in the performance of the five datasets. The experimental results in Figure 5 demonstrate that our method performs stably under random order conditions, confirming the robustness of the algorithm to the random permutation of test examples. Moreover, independent of the test order, the proposed method surpasses TDA consistently.

D.7 Cache Replacement Policy Analysis

To validate our eviction design, we conducted ablation studies comparing ROSE-TTA's noise-aware joint scoring against common cache replacement policies: (1) Entropy-only, (2) Similarity, (3) Random, (4) FIFO (First-In-First-Out), (5) LRU (Least Recently Used). The results in Table 13 reveal critical insights. The poor performance of FIFO (64.68%/62.25%) and LRU (63.21%/60.02%) validates that early-arriving samples are not necessarily more representative, especially when test data arrives in random order. Entropy-only (68.21%/65.13%) performs competitively but still accumulates misclassified confident samples. Our dual-criterion approach achieves consistent improvements across both settings, with notably larger gains on OOD data (+0.91%). This validates our hypothesis: under severe distribution shifts, combining entropy-based confidence with noise-enhanced stability provides complementary signals that effectively filter unreliable samples while retaining genuinely robust prototypes.

D.8 Effect of text embeddings alleviate biased logits at early stage

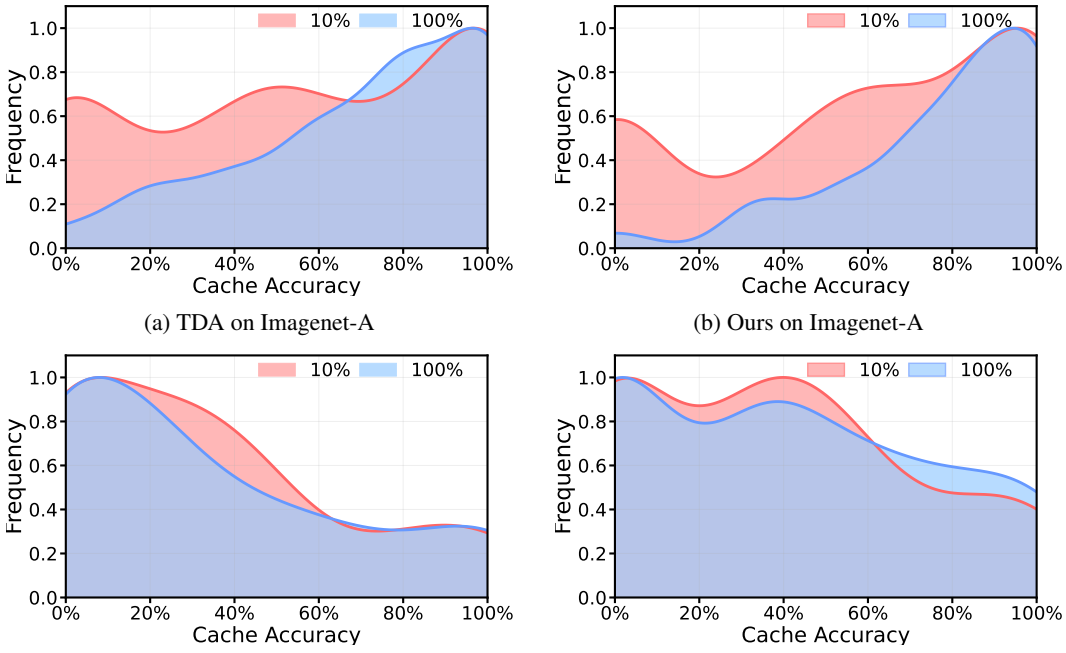
To further investigate the effect of $\boldsymbol{W}_C$ for mitigating class imbalance in the early stage, we also provide an ablation study on the predictive cache logits with and without the textual graph at 10% progress. We average all cache logits of categories with each cache capacity. As shown in the Table 14 , without the textual graph, classes with fewer samples in the cache contribute very small logits (even 0). This leads to the early inference process being heavily influenced by a few dominant classes. Additionally, classes that have not yet appeared in the cache fail to provide any useful information, exacerbating the class imbalance. By contrast, with the textual graph, our method alleviates the bias with more balanced predictive logits.

D.9 Comparisons of Cache Accuracy

As shown in Figure 6, on both FGVC and ImageNet-A datasets, our proposed ROSE-TTA consistently achieves higher cache accuracy than the baseline TDA method across both early and final stages of testing. Specifically, the distribution of cache accuracy is noticeably shifted toward higher values under ROSE-TTA, indicating that our method effectively enhances the reliability of cached samples. This improvement not only mitigates early-stage performance collapse but also maintains

Table 14: Predictive cache logits with and without textual graph.

Dataset	Method	Cache Capacity					
		0	1	2	3	4	5
ImageNet	w/o textual graph	0.0000	0.0006	0.0007	0.0008	0.0009	0.0011
	w/ textual graph	0.0007	0.0008	0.0009	0.0009	0.0010	0.0010
Caltech101	w/o textual graph	0.0000	0.0081	0.0090	0.0103	0.0118	0.0221
	w/ textual graph	0.0064	0.0093	0.0095	0.0101	0.0109	0.0142
Flowers102	w/o textual graph	0.0000	0.0060	0.0074	0.0106	0.0154	0.0211
	w/ textual graph	0.0073	0.0082	0.0095	0.0096	0.0101	0.0132
Stanford Cars	w/o textual graph	0.0000	0.0029	0.0042	0.0039	0.0046	0.0054
	w/ textual graph	0.0039	0.0041	0.0045	0.0044	0.0049	0.0052



(a) TDA on Imagenet-A

(b) Ours on Imagenet-A

(c) TDA on FGVC-Aircraft

(d) Ours on FGVC-Aircraft

Figure 6: Cache accuracy comparisons across datasets and methods.

strong performance throughout the adaptation process, demonstrating the robustness and practical efficacy of ROSE-TTA in scenarios with class imbalance.