T-VSS: TEST-TIME VISUAL SUBSPACE STEERING FOR ADVERSARIAL ROBUSTNESS OF VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Contrastive vision-language models (VLMs), exemplified by CLIP, enable strong zero-shot visual recognition but remain highly vulnerable to adversarial perturbations. Recent test-time adaptation methods improve robustness without retraining, but they do not directly adapt the corrupted visual representation itself. Prompt-based methods adapt the learnable text prompts, while input-space methods optimize pixels or padding at test time. These approaches can improve predictions, but they do so through an indirect and expensive optimization path. We propose **Test-time Visual Subspace Steering (T-VSS)**, a lightweight defense that performs test-time adaptation directly in the visual feature space. T-VSS first builds a sample-specific low-rank subspace from multi-view feature residuals anchored at the attacked image. It then learns a shared feature correction within this subspace using reliability-weighted entropy minimization. By constraining adaptation to a compact visual geometry, T-VSS steers attacked features toward more stable and discriminative predictions while avoiding noisy full-space updates. Experiments on fine-grained, ImageNet, and ImageNet-OOD benchmarks show that T-VSS improves adversarial robustness while maintaining competitive clean accuracy and better efficiency than prior test-time adaptation methods.

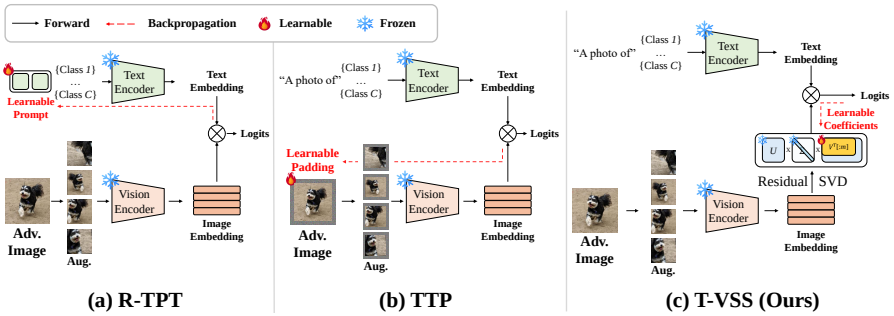
1 INTRODUCTION

Vision-language models (VLMs) (Radford et al., 2021; Chen et al., 2023; Zhang et al., 2024b; Sun et al., 2023) have become a strong foundation for zero-shot visual recognition. By aligning images and text in a shared embedding space, they can recognize unseen categories and transfer to downstream tasks without task-specific fine-tuning. This flexibility makes VLMs attractive in realistic settings where labeled data are scarce and rapid deployment is important.

However, their zero-shot predictions are highly sensitive to adversarial perturbations, where even small and nearly imperceptible input changes can lead to incorrect predictions (Goodfellow et al., 2015; Madry et al., 2018; Fang et al., 2024; Li et al., 2024b; Schlarmann et al., 2024). This vulnerability is particularly concerning in safety-critical applications such as medical AI, autonomous driving, and public security surveillance, where small perceptual failures can lead to high-stakes downstream consequences (Finlayson et al., 2019; Eykholt et al., 2018; Bai et al., 2021). Classical approaches such as adversarial training (Madry et al., 2018; Zhang et al., 2019; Cui et al., 2026), robust fine-tuning (Mao et al., 2023; Schlarmann et al., 2024; Wang et al., 2024; Zhang et al., 2026), diffusion-based purification (Nie et al., 2022), and adversarial prompt tuning (Li et al., 2024a; Zhang et al., 2024a; Zhou et al., 2024) can improve robustness, but they typically require additional data, label access, expensive retraining, or auxiliary models. To avoid these costs, recent work has explored test-time adaptation, which seeks to improve robustness using only unlabeled inputs at inference.

In the VLM setting, representative methods adapt each test sample either through the text branch or in the input space at test time as illustrated in Fig. 1: R-TPT (Sheng et al., 2025) improves adversarial robustness by optimizing learnable prompts, whereas TTC (Xing et al., 2025) and TTP (Li et al., 2026) operate in the input space by optimizing a learnable counterattack perturbation and instance-specific image padding, respectively. Despite their promise, these methods improve predictions only through indirect and often computationally expensive optimization.

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.



(a) R-TPT **(b) TTP** **(c) T-VSS (Ours)**

Figure 1: Comparison of test-time adaptation strategies for adversarially robust vision-language models. (a) **R-TPT** (Sheng et al., 2025) adapts learnable text prompts through backpropagation in the text branch. (b) **TTP** (Li et al., 2026) optimizes learnable input padding in the pixel space via backpropagation through the vision encoder. (c) In contrast, **T-VSS** adapts the visual feature space by learning a small set of steering coefficients in a low-rank visual subspace. Compared with prior methods, T-VSS provides a more direct and lightweight correction mechanism.

This indirection is problematic because adversarial attacks ultimately impair zero-shot recognition by corrupting the visual feature that is matched against text prototypes. A more direct defense should therefore adapt the attacked visual representation itself. However, unconstrained feature-space adaptation can be unstable. If the update is not guided by the local structure of the test sample, it may move the feature away from its semantic content and reinforce an incorrect prediction. The key challenge is therefore to steer the attacked representation at test time toward a more discriminative region while restricting the update to sample-specific, geometrically plausible directions.

In this paper, we address this challenge with **Test-time Visual Subspace Steering (T-VSS)**, a lightweight defense that adapts each sample within a compact, sample-specific visual subspace. Given a test image and its augmented views, the method extracts frozen visual features once and forms anchor-based residuals by subtracting the original test-image feature from each augmented-view feature. Our design is motivated by two complementary properties of attacked multi-view features. First, perturbation-induced feature shifts remain partially aligned across views, suggesting that a shared correction can benefit multiple transformations of the same attacked image. Second, after subtracting the original attacked view as an anchor, the remaining cross-view residuals exhibit a compact, sample-specific structure. These observations play distinct roles: cross-view alignment motivates a shared correction, whereas residual compactness motivates restricting that correction to a low-dimensional, locally supported search space. Accordingly, T-VSS applies singular value decomposition to the anchor-relative residuals and optimizes a shared correction only within the resulting subspace. Because augmented views can vary in reliability under attack, it estimates view reliability from feature-level agreement across views. The resulting weights are used during both adaptation and final aggregation, reducing the influence of unstable views on the final prediction. By constraining entropy minimization to these dominant residual directions, T-VSS reframes adversarial test-time adaptation as structured feature correction rather than prompt tuning or dense pixel-space search. This change in adaptation space has a direct empirical payoff: the proposed approach consistently improves the robustness–efficiency trade-off over prior test-time defenses.

Across eight fine-grained datasets and multiple CLIP (Radford et al., 2021) backbones, it achieves the best average adversarial robustness with competitive clean accuracy. The same trend holds on ImageNet and four ImageNet-OOD benchmarks, showing that the benefit extends beyond fine-grained recognition. Since optimization is performed only over a small set of subspace coefficients rather than text prompts or dense input variables, the method also reduces inference overhead. These results position compact feature-space steering as a simple and practical alternative to prompt- and pixel-space adaptation for robust zero-shot VLM inference.

2 Related Work

2.1 Adversarial Attacks and Defenses

Adversarial examples reveal the vulnerability of deep neural networks by introducing small, often imperceptible perturbations that lead to incorrect predictions (Goodfellow et al., 2015; Kurakin

et al., 2018; Madry et al., 2018; Carlini & Wagner, 2017). Representative attacks include single-step methods such as FGSM (Goodfellow et al., 2015), iterative optimization-based methods such as BIM and PGD (Kurakin et al., 2018; Madry et al., 2018), and stronger transfer-based or black-box attacks that do not require direct gradient access. Universal perturbations further demonstrate that a shared perturbation can fool a model across many inputs (Moosavi-Dezfooli et al., 2017). Recent studies show that vision-language models (VLMs), despite their strong zero-shot generalization, are also highly vulnerable to such attacks, which poses a major obstacle to reliable deployment.

To mitigate this issue, prior defenses have mainly focused on training-time robustness improvement. Adversarial training and its variants improve robustness by explicitly optimizing models on adversarially perturbed examples (Madry et al., 2018; Zhang et al., 2019; Rice et al., 2020; Cui et al., 2026), while robust fine-tuning (Mao et al., 2023; Schlarmann et al., 2024; Wang et al., 2024; Zhang et al., 2026), diffusion-based purification (Nie et al., 2022), and adversarial prompt tuning (Li et al., 2024a; Zhang et al., 2024a; Zhou et al., 2024) extend this paradigm to VLMs. Although effective, these methods typically require labeled data, repeated adversarial example generation, and costly retraining or fine-tuning of large pretrained models. Such requirements are especially burdensome for large frozen VLMs, motivating lightweight defense mechanisms that can operate directly at inference time.

2.2 Test-Time Adaptation and Defense for VLMs

Test-time adaptation (TTA) (Nado et al., 2020; Seo et al., 2026; Liu et al., 2021; Sun et al., 2020) aims to improve model generalization on unseen test distributions using only unlabeled test inputs. For VLMs, Test-Time Prompt Tuning (TPT) (Shu et al., 2022) tunes learnable text prompts (Zhou et al., 2022a;b) using multiple augmented views of each test image, and follow-up methods such as MTA (Zanella & Ayed, 2024) further improve adaptation stability through multi-view or multi-prompt aggregation. STS (Dafnis & Metaxas, 2025) instead performs spectrum-aware latent steering in the text embedding space, enabling efficient adaptation. However, these methods are primarily designed for natural distribution shifts and generally assume clean test inputs, which limits their effectiveness under adversarial perturbations.

Recent work has extended TTA to adversarially robust inference for VLMs. R-TPT (Sheng et al., 2025) revisits TPT under adversarial attack by optimizing learnable text prompts with pointwise entropy over selected low-entropy views at test time. A different line of work instead adapts the input itself at test time. TTC (Xing et al., 2025) performs test-time counterattacks by optimizing an additive perturbation in the image space, while TTP (Li et al., 2026) optimizes instance-specific padding parameters. Separately, recent training-free defenses bypass optimization-based adaptation, leveraging textual descriptions generated by large language models (Zhu et al., 2025), or applying calibration-dependent thresholded feature reconstruction (Liu et al., 2026).

In this paper, we focus on optimization-based test-time adaptation. Our T-VSS operates directly on attacked visual representations by learning a shared correction in a sample-specific low-rank visual subspace, providing a geometry-aware alternative to both prompt-space adaptation and padding-based input optimization without auxiliary models or datasets.

3 Method

3.1 Preliminaries

CLIP for zero-shot classification. We build on CLIP (Radford et al., 2021), a dual-encoder vision-language model consisting of an image encoder $F(\cdot)$ and a text encoder $G(\cdot)$. Given a C-way classification task with class names $\{t_c\}_{c=1}^{C}$, CLIP constructs a text prototype for each class as $g_c = G(\mathrm{prompt}(t_c)) \in \mathbb{R}^d$, where $\mathrm{prompt}(t_c)$ denotes a hand-crafted prompt template (e.g., "a photo of a [CLASS]") instantiated with class name t_c, and g_c is the resulting textual embedding of class c. For an input image x_i, the image encoder produces a visual feature $f_i = F(x_i) \in \mathbb{R}^d$. The zero-shot prediction probability of class c is then computed by the cosine similarity between the visual feature and the text prototypes:

$$p_c(x_i) = \frac{\exp(\cos(f_i, g_c)/\tau)}{\sum_{j=1}^{C} \exp(\cos(f_i, g_j)/\tau)}, \tag{1}$$

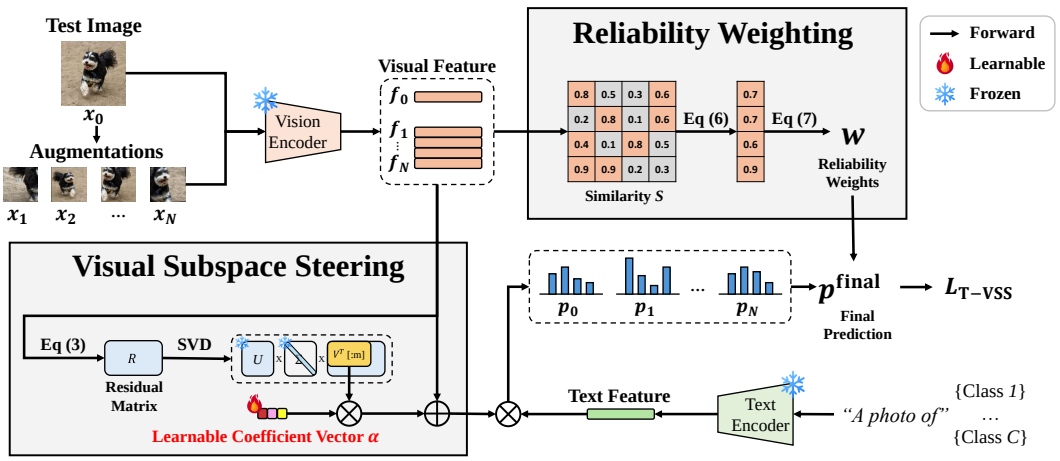


Figure 2: Overview of **T-VSS**. From multi-view CLIP visual features, T-VSS applies Singular Value Decomposition (SVD) to anchor-based residuals to extract a compact visual subspace that captures local cross-view geometry, then learns a shared low-rank correction within that subspace via reliability-weighted entropy minimization. Predictions from the adapted views are finally aggregated using reliability-aware ensembling.

where τ is the temperature parameter.

Adversarial test-time adaptation. Following prior test-time defenses (Sheng et al., 2025; Li et al., 2026), we construct $N+1$ stochastic views from a potentially adversarial test image x: $\mathcal{X}(x) = \{x_0, x_1, \ldots, x_N\}$, where $x_0 = x$ and $\{x_i\}_{i=1}^{N}$ are augmented views of x. In adversarial evaluation, all views are generated from the attacked image and therefore share the same perturbation source.

Let $p(x_i) \in \mathbb{R}^C$ denote the class probability vector in Eq. 1. We measure its prediction uncertainty with Shannon entropy:

$$\mathcal{H}(p(x_i)) = -\sum_{c=1}^{C} p_c(x_i) \log p_c(x_i). \tag{2}$$

3.2 Test-time Visual Subspace Steering

Unlike prior test-time defenses that adapt text prompts or optimize pixel-space variables, T-VSS adapts entirely in the visual feature space after a single frozen encoder pass. Given multi-view CLIP features, it first estimates a compact sample-specific visual subspace from anchor-based residual geometry, then learns a shared low-rank correction inside that subspace, and finally uses reliability-aware weighting to emphasize stable views during optimization and prediction. The image encoder and text prototypes remain fixed throughout; only a low-dimensional coefficient vector is optimized at test time.

Local visual subspace estimation. We begin by extracting normalized CLIP visual features $\{f_i = F(x_i)\}_{i=0}^{N}$ for all views. Instead of optimizing an unconstrained shift in the full d-dimensional embedding space, T-VSS first estimates a compact subspace that captures the local variation of the current sample. We use the original attacked view x_0 as an anchor and build the residual matrix

$$R = \begin{bmatrix} (f_1 - f_0)^\top \\ (f_2 - f_0)^\top \\ \vdots \\ (f_N - f_0)^\top \end{bmatrix} \in \mathbb{R}^{N \times d}. \tag{3}$$

We then compute the singular value decomposition $R = U\Sigma V^\top$, where the right singular vectors in V define orthogonal directions in the visual embedding space. Rather than fixing the rank manually,

we choose the smallest active rank m whose cumulative singular-value energy exceeds a threshold ρ:

$$m = \min \left\{ q \;\middle|\; \frac{\sum_{j=1}^{q} \sigma_j^2}{\sum_{j=1}^{\mathrm{rank}(R)} \sigma_j^2} \geq \rho \right\}, \tag{4}$$

where $\{\sigma_j\}$ are the singular values and $\rho \in (0, 1]$ is the explained-variance threshold. The residual matrix is not intended to estimate the absolute adversarial displacement, which is unobserved at test time because the corresponding clean feature is unavailable. In particular, subtracting the anchor feature f_0 removes any feature offset shared across all views. The remaining residuals instead describe the relative variation induced by stochastic transformations around the current attacked sample. Our analysis in Appendix B shows that this anchor-relative variation has a substantially lower effective rank under attack than in the corresponding clean case. We therefore interpret the leading right singular vectors of R as a compact, sample-specific basis of augmentation-supported feature variation. This basis defines where test-time adaptation is allowed to move, rather than directly specifying the attack direction or the final correction.

Shared low-rank feature steering. Given the basis $V_m \in \mathbb{R}^{d \times m}$, which collects the top-m right singular vectors, T-VSS performs adaptation by learning a shared low-rank correction inside this subspace. We initialize a learnable coefficient vector $\alpha \in \mathbb{R}^m$ at zero and generate a shared shift $\Delta = V_m \alpha$. The same shift is then applied to every view as follows:

$$\tilde{f}_i = \frac{f_i + \Delta}{\|f_i + \Delta\|_2}, \quad i = 0, \ldots, N. \tag{5}$$

Importantly, the residual-SVD basis determines only the admissible search space; the direction and magnitude of the correction are selected by reliability-weighted entropy minimization. Parameterizing $\Delta = V_m \alpha$ therefore restricts the update to directions supported by the local multi-view geometry and avoids unconstrained optimization in the full embedding space. Applying the same correction to every view does not make their adapted representations identical, since each $\tilde{f}_i$ retains its view-specific feature f_i. Instead, shared steering acts as a cross-view regularizer that prevents individual views from independently fitting augmentation artifacts or spurious low-entropy directions. Since the only learnable variable is the m-dimensional coefficient vector, the number of sample-wise trainable parameters is exactly the selected rank.

Reliability-aware optimization and aggregation. Not all stochastic views are equally informative under attack. T-VSS therefore assigns each view a reliability score (Sheng et al., 2025) as a practical proxy for how well that view agrees with the local feature neighborhood. Because CLIP already outputs ℓ_2-normalized image features, this agreement can be computed directly from pairwise similarities $S_{ij} = f_i^\top f_j$. For each view, we average its top-K nearest neighbor similarities:

$$r_i = \frac{1}{K} \sum_{j \in \mathcal{N}_K(i)} S_{ij}, \tag{6}$$

where $\mathcal{N}_K(i)$ denotes the indices of the top-K most similar views excluding itself. We then convert these scores into reliability weights with a temperature-scaled softmax:

$$w_i = \frac{\exp(r_i/\tau_r)}{\sum_{j=0}^{N} \exp(r_j/\tau_r)}, \tag{7}$$

where τ_r is a reliability temperature. Views that remain in a dense and mutually consistent feature neighborhood receive larger weights, while unstable outliers are suppressed.

Given the adapted features $\{\tilde{f}_i\}$, we compute logits with the frozen CLIP text prototypes and denote the resulting class probabilities by $p(\tilde{f}_i)$. We then optimize α by minimizing reliability-weighted pointwise entropy:

$$\mathcal{L}_{\mathrm{T-VSS}} = \sum_{i=0}^{N} w_i \, \mathcal{H}\big(p(\tilde{f}_i)\big), \tag{8}$$

where w_i is the reliability weight defined in Eq. 7. In contrast to hard view selection, the loss softly uses all views while reducing the influence of unreliable ones.

Final prediction and efficiency. After optimization, the final prediction is obtained by reliability-weighted averaging of the adapted view probabilities:

$$p^{\text{final}} = \sum_{i=0}^{N} w_i \, p(\tilde{f}_i), \qquad \hat{y} = \arg\max_c \; p_c^{\text{final}}. \tag{9}$$

T-VSS encodes the image views only once and performs all subsequent optimization directly on cached visual features. Unlike prompt- or pixel-space methods, it avoids repeated backpropagation through large model components or dense input variables. Overall, T-VSS remains lightweight and easy to interpret as a sample-wise low-rank correction of the attacked visual representation guided by multi-view agreement.

4 Experiments

4.1 Setup

Datasets and Models. We evaluate T-VSS on both fine-grained recognition benchmarks and large-scale ImageNet-style benchmarks. For fine-grained evaluation, we use eight datasets spanning diverse visual domains: Caltech101 (Fei-Fei et al., 2004), Pets (Parkhi et al., 2012), Flower102 (Nilsback & Zisserman, 2008), Stanford Cars (Krause et al., 2013), FGVC Aircraft (Maji et al., 2013), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019), and UCF101 (Soomro et al., 2012). We further evaluate on ImageNet (Deng et al., 2009) and four ImageNet out-of-distribution benchmarks: ImageNet-A (Hendrycks et al., 2021b), ImageNet-V2 (Recht et al., 2019), ImageNet-R (Hendrycks et al., 2021a), and ImageNet-S (Wang et al., 2019). As the underlying VLM, we adopt official CLIP checkpoints and consider three widely used backbones: ResNet-50, ViT-B/16, and ViT-L/14.

Evaluation and Baselines. We report both clean top-1 accuracy (**Acc.**) and adversarial top-1 accuracy (**Rob.**). Following prior work on adversarial test-time defense for CLIP, adversarial examples are generated under a white-box but defense-oblivious threat model against the original CLIP model using PGD, while the test-time defense remains hidden from the attacker. We compare T-VSS with vanilla CLIP, a simple multi-view **Ensemble**, standard VLM test-time adaptation baselines, and recent adversarial test-time defenses. Depending on the benchmark and backbone, the comparison set includes **TPT** (Shu et al., 2022), **C-TPT** (Yoon et al., 2024), **MTA** (Zanella & Ayed, 2024), **TTC** (Xing et al., 2025), **R-TPT** (Sheng et al., 2025), and **TTP** (Li et al., 2026). For fair comparison, all test-time methods use the same CLIP backbone and the same AugMix augmentation (Hendrycks et al., 2020), without relying on additional foundation models or external knowledge.

Implementation Details. For adversarial evaluation, we generate ℓ_∞-bounded PGD (Madry et al., 2018) examples with backbone-specific settings following standard CLIP robustness benchmarks. For ResNet-50, we use a perturbation budget of $\epsilon = 1/255$ and 7 attack steps. For ViT-B/16 and ViT-L/14, we use a stronger setting with $\epsilon = 4/255$ and 100 attack steps. In all cases, the attack step size is set to $\epsilon/4$. For zero-shot classification, we use the default hand-crafted prompt template "a photo of a [CLASS]" to construct text prototypes. We perform a single update step using AdamW with a learning rate of 0.1. We set the explained-variance threshold ρ to 0.9, construct the visual subspace with anchor-based residuals, and compute reliability weights with temperature 0.05 using a default top-K neighbor count of $K = 5$. Each test sample is processed with 64 views in total, including the original image and 63 augmented views.

4.2 Experimental Results

Results on Fine-grained Datasets. Table 1 reports clean and adversarial accuracy on eight fine-grained datasets across three CLIP backbones. T-VSS achieves the best average robust accuracy on all three backbones, reaching 48.5% on ResNet-50, 45.8% on ViT-B/16, and 53.8% on ViT-L/14. These scores exceed those of the strongest prior defense by 3.3, 2.9, and 2.2 points, respectively, while keeping clean accuracy competitive. Overall, the results show that the benefit of T-VSS is not confined to a particular backbone or dataset, but extends consistently across architectures while improving the robustness–accuracy trade-off. Notably, the margin is especially clear on CLIP-ResNet-50, where padding-based TTP is less competitive than on ViT backbones. This pattern suggests that

Table 1: Clean (Acc.) and adversarial (Rob.) top-1 accuracy (%) on eight fine-grained datasets across three CLIP backbones. Best clean accuracy and best adversarial accuracy are highlighted in **bold** and **bold**, respectively. † indicates reproduced results.

Method	Caltech101		Pets		Cars		Flower102		Aircraft		DTD		EuroSAT		UCF101		Avg.	
	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.
CLIP-ResNet-50 ($\epsilon = 1/255$)																		
CLIP	85.9	2.6	83.5	0.0	55.7	0.0	61.7	0.0	15.7	0.0	40.4	0.8	23.7	0.0	58.9	0.0	53.2	0.4
Ensemble	83.5	74.8	82.3	69.9	57.1	36.2	58.0	46.6	16.4	9.8	37.1	29.5	16.7	13.7	53.9	43.0	50.6	40.4
TPT	**87.9**	7.0	84.7	0.1	58.4	0.0	62.1	0.0	17.3	0.0	**42.4**	4.3	**28.4**	0.0	**60.6**	0.3	**55.2**	1.5
C-TPT	87.7	3.7	83.6	0.0	56.6	0.0	**64.8**	0.0	16.7	0.0	41.5	1.3	27.0	0.0	60.1	0.1	54.8	0.6
MTA	87.3	65.9	**84.8**	59.8	**58.7**	17.8	61.0	31.5	**18.1**	3.7	40.3	18.8	22.5	1.6	**60.6**	31.3	54.1	28.8
R-TPT	86.7	**79.8**	84.6	74.2	58.1	42.9	60.6	51.9	17.5	12.6	41.3	33.5	21.2	15.9	59.7	50.9	53.7	45.2
TTP†	85.9	73.7	83.8	46.9	55.6	38.9	61.0	35.2	15.6	11.6	40.1	25.9	24.0	**23.4**	58.5	49.0	53.1	38.1
T-VSS (Ours)	84.4	78.1	84.4	**75.5**	55.6	**54.7**	58.8	**54.0**	18.0	**20.3**	38.9	**35.5**	18.5	17.1	58.9	**52.5**	52.2	**48.5**
CLIP-ViT-B/16 ($\epsilon = 4/255$)																		
CLIP	94.0	0.0	**88.3**	0.0	65.5	0.0	67.4	0.0	23.9	0.0	44.4	0.0	42.2	0.0	65.2	0.0	61.4	0.0
TTC	87.6	8.4	82.3	10.4	55.0	2.9	**69.0**	7.4	23.3	0.5	41.0	4.5	**47.4**	0.4	65.8	1.6	58.9	4.5
Ensemble	91.9	74.7	86.2	51.2	65.7	26.0	65.9	36.3	23.4	8.7	43.2	25.1	28.2	2.2	63.0	30.6	58.4	31.8
MTA	**94.3**	72.1	88.0	51.8	**67.7**	18.5	67.4	27.9	**25.0**	4.3	**46.5**	16.2	42.5	1.2	**67.5**	27.5	**62.3**	27.4
R-TPT	93.7	82.0	87.2	60.2	67.0	34.7	68.7	44.6	23.9	13.2	46.4	32.8	34.7	8.5	67.2	43.2	61.1	39.9
TTP	93.5	**82.3**	**88.3**	64.7	65.4	37.4	67.3	47.2	23.9	14.8	44.1	36.0	42.0	**14.5**	65.0	**47.2**	61.2	42.9
T-VSS (Ours)	93.4	81.5	87.3	**64.9**	65.8	**54.5**	65.9	**50.6**	24.3	**24.4**	45.9	**37.4**	34.8	7.5	66.0	45.2	60.4	**45.8**
CLIP-ViT-L/14 ($\epsilon = 4/255$)																		
CLIP	95.2	0.1	93.1	0.0	76.8	0.0	76.2	0.0	30.0	0.0	52.4	0.0	55.1	0.0	73.7	0.0	**69.1**	0.0
TTC	88.7	7.7	92.2	7.6	67.8	2.2	**76.5**	7.5	31.7	0.5	49.7	6.2	**64.1**	0.2	**75.0**	2.2	68.2	4.3
Ensemble	94.9	83.6	93.4	63.5	76.3	40.5	75.0	48.6	31.7	12.7	51.3	31.3	38.7	11.1	71.7	48.3	66.6	42.5
MTA	**95.8**	83.1	**93.7**	64.9	**78.4**	36.6	76.1	44.2	**32.7**	8.0	53.4	27.2	47.8	7.5	74.7	47.5	**69.1**	39.9
R-TPT	95.7	88.2	**93.7**	72.9	77.2	49.1	76.2	55.6	31.7	17.2	**54.0**	38.0	44.3	20.4	74.3	55.6	68.4	49.6
TTP	95.1	**88.6**	93.1	**76.3**	76.8	51.1	76.1	58.7	29.2	17.7	52.3	41.3	55.0	**21.6**	73.6	**57.4**	68.9	51.6
T-VSS (Ours)	94.8	87.5	**93.7**	73.8	76.3	**63.2**	75.2	**60.9**	**32.7**	**26.9**	53.4	**41.7**	45.1	20.3	73.5	55.9	68.1	**53.8**

Table 2: Clean (Acc.) and adversarial (Rob.) top-1 accuracy (%) on ImageNet and four ImageNet-OOD benchmarks with CLIP-ResNet-50. † indicates reproduced results.

Method	ImageNet		ImageNet-A		ImageNet-V2		ImageNet-R		ImageNet-S		Avg.	
	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.
CLIP	58.2	0.1	21.8	0.0	51.5	0.1	56.1	0.8	33.3	0.5	44.2	0.3
Ensemble	58.0	40.1	22.6	10.1	52.0	37.2	51.3	39.3	29.5	20.7	42.7	29.5
TPT	60.7	0.3	26.5	0.0	54.8	0.3	**58.9**	1.8	35.0	1.4	**47.2**	0.7
C-TPT	60.4	0.1	24.1	0.0	54.3	0.1	57.7	1.0	34.7	0.9	46.2	0.4
MTA	60.4	30.0	27.5	5.6	54.2	24.6	58.4	29.8	**35.2**	11.3	47.1	20.3
R-TPT	**60.9**	47.7	**28.4**	14.4	**54.9**	41.6	57.6	46.9	34.0	26.2	47.1	35.4
TTP†	58.2	43.0	21.9	12.6	51.3	38.1	56.1	38.3	33.4	17.3	44.2	29.9
T-VSS (Ours)	59.7	**50.1**	27.8	**16.0**	53.6	**44.6**	57.2	**47.6**	33.6	**29.8**	46.4	**37.6**

input-space padding may transfer less reliably across backbone families than feature-space correction, since padding operates at the pixel-level while T-VSS adapts visual representations directly.

The per-dataset results further clarify where direct feature correction is most beneficial. T-VSS is especially strong on challenging fine-grained datasets such as Cars, Flower102, Aircraft, and DTD, where it achieves the best robust accuracy on all three backbones. It also remains competitive on UCF101, although TTP is slightly stronger on the two larger ViT backbones. EuroSAT is the clearest exception, where TTP attains the best robust accuracy. This result suggests that T-VSS may be less effective when the local residual geometry is less informative, as observed on EuroSAT. Despite this exception, T-VSS remains the strongest method on average in terms of robust accuracy across all three backbones. Additional results are in Tables 8 and 9.

Results on ImageNet and ImageNet-OOD Datasets. Table 2 shows that the gains of T-VSS extend beyond fine-grained recognition to large-scale and out-of-distribution evaluation on CLIP-ResNet-50. T-VSS obtains the best robust accuracy on ImageNet and on every OOD benchmark, improving the average robust accuracy to 37.6% and surpassing the previous best defense, R-TPT, by 2.2 points. The gain is consistent across ImageNet-A, ImageNet-V2, ImageNet-R, and ImageNet-S, which suggests that the proposed feature-space correction is not tied to a specific dataset bias or shift type. T-VSS also preserves competitive clean accuracy. Although some prompt-based methods obtain slightly higher clean scores, their adversarial robustness remains substantially lower. This comparison highlights the central advantage of T-VSS: by adapting the attacked visual representa-

Table 3: Adversarial accuracy (%) under additional attacks on Flower102 and DTD using CLIP-ViT-B/16. DF denotes DeepFool.

Method	Flower102				DTD			
	CW	DF	FGSM	Avg.	CW	DF	FGSM	Avg.
CLIP	0.8	0.4	4.8	2.0	2.3	7.6	13.4	7.8
Ensemble	50.1	52.2	46.6	49.7	31.1	32.9	29.7	31.2
TPT	13.8	10.8	14.2	12.9	21.3	24.4	22.2	22.6
C-TPT	6.6	5.5	6.2	6.1	11.9	15.8	17.5	15.1
MTA	34.5	35.4	36.6	35.5	23.6	23.5	23.9	23.7
R-TPT	51.6	54.7	49.2	51.8	34.2	35.9	32.5	34.2
TTP	54.1	56.4	51.8	54.1	38.9	40.1	37.1	38.7
T-VSS (Ours)	**54.8**	**60.5**	**53.7**	**56.3**	**39.1**	**42.3**	**37.4**	**39.6**

Table 4: Per-image latency and adversarial accuracy (%) on UCF101 with CLIP-ResNet-50 under different view budgets.

Method	Running time (s/image)	Rob.
R-TPT (64 views)	0.533	50.9
TTP (64 views)	0.408	49.0
T-VSS (64 views)	0.383	**52.5**
T-VSS (32 views)	0.193	52.2
T-VSS (16 views)	0.092	51.1
T-VSS (8 views)	0.046	49.0

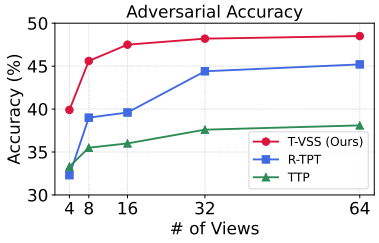


Figure 3: Ablation of the number of views.

Table 5: Ablation of adaptive rank selection and reliability weighting.

Adaptive Rank	Reliability Weighting	ResNet-50		ViT-B/16	
		Acc.	Rob.	Acc.	Rob.
✗	✗	50.6	44.1	59.2	38.4
✗	✓	51.2	44.5	59.6	38.5
✓	✗	51.5	48.1	59.9	45.7
✓	✓	**52.2**	**48.5**	**60.4**	**45.8**

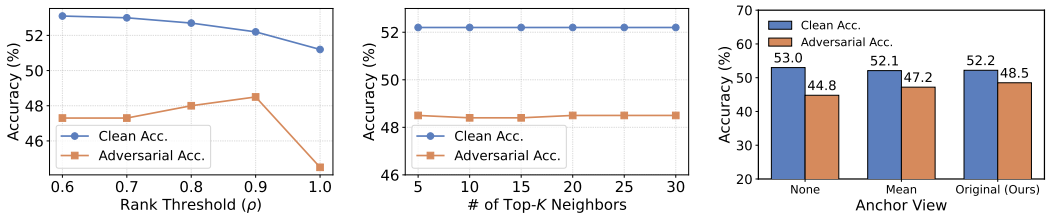
tion directly, it improves robustness without paying the large clean-accuracy penalty often associated with aggressive test-time correction. Taken together with the fine-grained results, these experiments support T-VSS as a robust and scalable test-time defense for zero-shot VLM inference.

5 Analysis and Ablation

Robustness under Various Attacks. Table 3 evaluates T-VSS under three additional attacks: ℓ_2-based CW (Carlini & Wagner, 2017), ℓ_2-based DeepFool (Moosavi-Dezfooli et al., 2016), and ℓ_∞-bounded FGSM (Goodfellow et al., 2015), on Flower102 and DTD. T-VSS achieves the best adversarial accuracy across all attack settings and both datasets, reaching 56.3% average robustness on Flower102 and 39.6% on DTD. The consistent improvements across the evaluated ℓ_∞- and ℓ_2-based attacks suggest that T-VSS is not narrowly tied to the primary ℓ_∞-PGD setting. Additional robustness results under stronger attacks are in Table 10.

Analysis of Inference Efficiency. Table 4 compares per-image latency and adversarial accuracy on UCF101 with CLIP-ResNet-50. Under the same 64-view budget, T-VSS is both the fastest and the most robust adaptive defense, achieving 52.5% robust accuracy at 0.383 seconds per image. This advantage follows directly from the design of T-VSS: image views are encoded once, and test-time optimization is performed only in a low-dimensional visual subspace rather than through prompt parameters or dense input variables. The latency benefit becomes even clearer with fewer views. With only 16 views, T-VSS still reaches 51.1% robust accuracy, outperforming 64-view R-TPT while reducing latency by nearly $5.8\times$. Even with 8 views, it matches the robustness of 64-view TTP while being about $8.9\times$ faster. These results show that T-VSS can maintain strong robustness at substantially lower inference cost.

Robustness under Different View Budgets. Figure 3 reports average adversarial accuracy under different view budgets on fine-grained benchmarks with CLIP-ResNet-50. T-VSS remains consistently strongest across all view budgets, with the clearest advantage in the low-view regime. Although adversarial accuracy improves for all methods with more views, T-VSS dominates the entire curve and exploits multi-view information more effectively than prior methods. In particular, with only 8 views, T-VSS already rivals the 64-view performance of R-TPT and clearly outperforms 64-view TTP. This behavior is consistent with the shared structure in attacked multi-view features, which allows T-VSS to estimate an effective consensus low-rank correction even with few views. Such view efficiency is valuable when test-time latency or augmentation budget is limited.



(a) Rank Threshold ρ (b) Top-K Nearest Neighbor (c) Anchor View

Figure 4: Sensitivity of T-VSS to the rank threshold ρ, the number of neighbors K, and the choice of anchor view. Results are averaged over the eight fine-grained datasets on CLIP-ResNet-50.

Ablation of Core Components. Table 5 isolates the two components of T-VSS: adaptive rank selection and reliability weighting. Adaptive rank selection is the primary source of robustness gain. When it is disabled, T-VSS retains the full available rank of the multi-view residual matrix, including low-energy directions that are only weakly supported by the local cross-view geometry. Enabling adaptive rank substantially improves robust accuracy, from 44.1% to 48.1% on ResNet-50 and from 38.4% to 45.7% on ViT-B/16. This result supports our interpretation that restricting adaptation to the dominant residual subspace regularizes entropy minimization and promotes more stable feature steering. Reliability weighting provides a consistent complementary gain by suppressing unstable views during adaptation and aggregation. Across both backbones, it slightly but uniformly improves clean and robust accuracy, yielding the best overall clean–robustness balance in the full model. Overall, these results suggest that adaptive rank determines where T-VSS should steer, while reliability weighting helps determine which views should be trusted.

Ablation of Hyperparameters and Design Choices. Figure 4 shows that T-VSS is stable across a broad range of hyperparameters. For the rank threshold, $\rho = 1.0$ corresponds to using the full available rank of the residual matrix. Robustness is highest around $\rho = 0.9$, but even smaller thresholds still outperform the strongest prior average robustness on the same ResNet-50 setting (R-TPT 45.2% in Table 1), indicating that T-VSS does not require delicate tuning as long as adaptation remains in a moderately compact subspace. By contrast, setting $\rho = 1.0$ reduces robust accuracy to 44.5%, suggesting that low-energy residual directions provide additional degrees of freedom through which entropy minimization can produce less stable or spuriously confident updates. The number of top-K nearest neighbors used for reliability estimation in Eq. 6 has almost no effect on either clean or adversarial accuracy, indicating that the reliability weighting is not sensitive to precise tuning. Finally, we examine the reference construction in Eq. 3. Using the original test image as the anchor gives the best robust accuracy (48.5%), outperforming mean-centering across views (47.2%) and using raw features without reference subtraction (44.8%). This suggests that the original-view anchor best suppresses view-shared offsets while preserving the local relative geometry needed for well-constrained feature steering. Overall, these results show that the default configuration of T-VSS is effective and robust to moderate variation in hyperparameters and design choices.

6 Conclusion

This paper addressed adversarial test-time defense for zero-shot vision-language models by proposing **Test-time Visual Subspace Steering (T-VSS)**, a lightweight feature-space adaptation method that adjusts attacked visual representations at test time. The central idea is to estimate a compact sample-specific visual subspace from multi-view anchor residuals and to learn a shared, reliability-aware correction inside that subspace. By constraining adaptation to this low-rank geometry, T-VSS turns test-time entropy minimization into structured feature-space steering rather than prompt-space adjustment or dense input-space search. Experiments across eight fine-grained datasets, ImageNet, and four ImageNet-OOD benchmarks show that T-VSS consistently improves adversarial robustness while preserving competitive clean accuracy. Additional analysis under diverse attacks, low-view regimes, and component ablations further shows that this constrained feature-space adaptation yields a stronger robustness–efficiency trade-off than prior test-time defenses.

AI USE STATEMENT

In this work, we used generative AI tools to assist with language editing and translation, code cleanup and refactoring, and the implementation of the proposed method. We did not use generative AI to develop the core research ideas, design the methodology or experiments, or interpret the experimental results. All AI-assisted code and text were reviewed by the authors. The authors take full responsibility for the final manuscript, code, claims, and other artifacts produced in this work.

ETHICS STATEMENT

This work aims to improve the reliability of zero-shot vision-language models under adversarial perturbations. Stronger test-time defense can be beneficial in high-stakes settings such as medical decision support and autonomous perception, where small input corruptions may otherwise cause harmful errors. However, improved robustness is not a guarantee of safety and should not be over-interpreted, especially because defense methods may still fail under stronger adaptive attacks. In addition, robustness research is inherently dual-use, since it may also inform the design of stronger attacks. We therefore view T-VSS as a complementary safety mechanism that should be paired with rigorous evaluation and additional safeguards in real-world deployment.

REPRODUCIBILITY STATEMENT

The proposed T-VSS method is described in detail in Section 3, including the construction of the sample-specific visual subspace, reliability-aware optimization, and final prediction. The datasets, evaluation protocols, attack configurations, baselines, and implementation settings are documented in Section 4.1, with additional experimental details provided in Appendix A. We will publicly release the source code upon acceptance.

REFERENCES

Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *Proc. ICML*, pp. 284–293, 2018.

Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. Recent advances in adversarial training for adversarial robustness. In *IJCAI*, pp. 4312–4321, 2021. Survey Track.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *Proc. S&P*, 2017.

Fei-Long Chen, Du-Zhen Zhang, Ming-Lun Han, Xiu-Yi Chen, Jing Shi, Shuang Xu, and Bo Xu. Vlp: A survey on vision-language pre-training. *Machine Intelligence Research*, 20(1):38–56, 2023.

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

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proc. ICML*, pp. 2206–2216, 2020.

Yubo Cui, Xianchao Guan, Zijun Xiong, and Zheng Zhang. Agft: Alignment-guided fine-tuning for zero-shot adversarial robustness of vision-language models. *Proc. CVPR*, 2026.

Konstantinos M. Dafnis and Dimitris N. Metaxas. Test-time spectrum-aware latent steering for zero-shot generalization in vision-language models. In *Proc. NeurIPS*, 2025.

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

Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *Proc. CVPR*, pp. 1625–1634, 2018.

Hao Fang, Jiawei Kong, Bin Chen, Tao Dai, Hao Wu, and Shu-Tao Xia. Clip-guided generative networks for transferable targeted adversarial attacks. In *Proc. ECCV*, pp. 1–19, 2024.

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 *Proc. CVPR Workshops*, pp. 178–178, 2004.

Samuel G Finlayson, John D Bowers, Joichi Ito, Jonathan L Zittrain, Andrew L Beam, and Isaac S Kohane. Adversarial attacks on medical machine learning. *Science*, 363(6433):1287–1289, 2019.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *Proc. ICLR*, 2015.

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, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *Proc. ICLR*, 2020.

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 *Proc. ICCV*, pp. 8340–8349, 2021a.

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

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

Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pp. 99–112. Chapman and Hall/CRC, 2018.

Lin Li, Haoyan Guan, Jianing Qiu, and Michael Spratling. One prompt word is enough to boost adversarial robustness for pre-trained vision-language models. In *Proc. CVPR*, 2024a.

Xiao Li, Wei Zhang, Yining Liu, Zhanhao Hu, Bo Zhang, and Xiaolin Hu. Language-driven anchors for zero-shot adversarial robustness. In *Proc. CVPR*, pp. 24686–24695, 2024b.

Zhiwei Li, Yitian Pang, Weining Wang, Zhenan Sun, and Qi Li. Ttp: Test-time padding for adversarial detection and robust adaptation on vision-language models. *Proc. CVPR*, 2026.

Liangsheng Liu, Si Chen, Jiamin Wu, Weiwei Feng, Zhixin Cheng, Xiaotian Yin, Wenfei Yang, and Tianzhu Zhang. Adversarial attacks already tell the answer: Directional bias-guided test-time defense for vision-language models. In *Proc. ICLR*, 2026.

Yuejiang Liu, Parth Kothari, Bastien Germain van Delft, Baptiste Bellot-Gurlet, Taylor Mordan, and Alexandre Alahi. TTT++: When does self-supervised test-time training fail or thrive? In *Proc. NeurIPS*, 2021.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *Proc. ICLR*, 2018.

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

Chengzhi Mao, Scott Geng, Junfeng Yang, Xin Wang, and Carl Vondrick. Understanding zero-shot adversarial robustness for large-scale models. In *Proc. ICLR*, 2023.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proc. CVPR*, pp. 2574–2582, 2016.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proc. CVPR*, 2017.

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.

Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. In *Proc. ICML*, 2022.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Proc. ICVGIP*, pp. 722–729, 2008.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *Proc. CVPR*, 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 *Proc. ICML*, 2021.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *Proc. ICML*, pp. 5389–5400, 2019.

Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In *Proc. ICML*, pp. 8093–8104, 2020.

Christian Schlarmann, Naman Deep Singh, Francesco Croce, and Matthias Hein. Robust clip: Unsupervised adversarial fine-tuning of vision embeddings for robust large vision-language models. In *Proc. ICML*, 2024.

Minseok Seo, Wonjun Lee, Jaehyuk Jang, and Changick Kim. Efficient test-time optimization for depth completion via low-rank decoder adaptation. *arXiv preprint arXiv:2603.01765*, 2026.

Lijun Sheng, Jian Liang, Zilei Wang, and Ran He. R-tpt: Improving adversarial robustness of vision-language models through test-time prompt tuning. In *Proc. CVPR*, pp. 29958–29967, 2025.

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. In *Proc. NeurIPS*, volume 35, pp. 14274–14289, 2022.

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.

Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

Yu Sun, Xiaolong Wang, Liu Zhuang, John Miller, Moritz Hardt, and Alexei A. Efros. Test-time training with self-supervision for generalization under distribution shifts. In *Proc. ICML*, 2020.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Proc. NeurIPS*, 32, 2019.

Sibo Wang, Jie Zhang, Zheng Yuan, and Shiguang Shan. Pre-trained model guided fine-tuning for zero-shot adversarial robustness. In *Proc. CVPR*, pp. 24502–24511, 2024.

Songlong Xing, Zhengyu Zhao, and Nicu Sebe. Clip is strong enough to fight back: Test-time counterattacks towards zero-shot adversarial robustness of clip. In *Proc. CVPR*, pp. 15172–15182, 2025.

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

Maxime Zanella and Ismail Ben Ayed. On the test-time zero-shot generalization of vision-language models: Do we really need prompt learning? In *Proc. CVPR*, pp. 23783–23793, 2024.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *Proc. ICML*, pp. 7472–7482, 2019.

Jiacheng Zhang, Jinhao Li, Hanxun Huang, Sarah Monazam Erfani, Benjamin I. P. Rubinstein, and Feng Liu. Semantic-aware adversarial fine-tuning for CLIP. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856.

Jiaming Zhang, Xingjun Ma, Xin Wang, Lingyu Qiu, Jiaqi Wang, Yu-Gang Jiang, and Jitao Sang. Adversarial prompt tuning for vision-language models. In *Proc. ECCV*, 2024a.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024b.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proc. CVPR*, 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.

Yiwei Zhou, Xiaobo Xia, Zhiwei Lin, Bo Han, and Tongliang Liu. Few-shot adversarial prompt learning on vision-language models. In *Proc. NeurIPS*, volume 37, pp. 3122–3156, 2024.

Xingyu Zhu, Beier Zhu, Shuo Wang, Kesen Zhao, and Hanwang Zhang. Enhancing CLIP robustness via cross-modality alignment. In *Proc. NeurIPS*, 2025.

APPENDIX

A More Experimental Details **14**

A.1 Datasets 14

A.2 Augmentation Configuration 14

A.3 Source of Baseline Results 15

B Analysis of Shared Perturbation Structure and Residual Compactness **15**

C Additional Experiments and Analysis **16**

C.1 Results under Robust Pretrained Backbone 16

C.2 Additional CLIP Backbone Results 16

C.3 Additional Robustness under Stronger Attacks 17

C.4 Importance of the Residual-SVD Basis 17

C.5 Ablation of Update Step 17

C.6 Stability Across Random Seeds 18

C.7 Comparison with DBD 18

C.8 Selected Rank m and Number of Learnable Parameters 19

C.9 Shared versus Per-view Feature Steering 19

D Limitations and Future Work **20**

A MORE EXPERIMENTAL DETAILS

A.1 DATASETS

Table 6 summarizes the number of classes and test samples for all datasets used in our experiments.

Table 6: Dataset statistics used in the experiments.

Dataset	# Classes	# Test
Caltech101	100	2,465
Pets	37	3,669
Cars	196	8,041
Flower102	102	2,463
Aircraft	100	3,333
DTD	47	1,692
EuroSAT	10	8,100
UCF101	101	3,783
ImageNet	1,000	50,000
ImageNet-A	200	7,500
ImageNet-V2	1,000	10,000
ImageNet-R	200	30,000
ImageNet-S	1,000	50,889

A.2 AUGMENTATION CONFIGURATION

We use the same augmentation pipeline and hyperparameters as R-TPT (Sheng et al., 2025). For each test sample, we retain the original image as the anchor view and independently gener-

Table 7: Backbone-wise summary of paired clean/adversarial multi-view feature statistics on the $50{,}000$ ImageNet validation images under the 64-view evaluation protocol. Each row uses the backbone-specific PGD setting from the main paper.

Backbone	Cosine Similarity ↑	Shared Energy ↑	Clean Rank	Adv. Rank
ResNet-50	0.314	0.230	8.28	3.37
ViT-B/16	0.498	0.435	8.58	1.45
ViT-B/32	0.475	0.435	10.31	2.13

ate 63 augmented views. Each augmented view is produced using `RandomResizedCrop` and `RandomHorizontalFlip`, followed by AugMix (Hendrycks et al., 2020). The AugMix operation set contains nine transformations: `AutoContrast`, `Equalize`, `Posterize`, `Rotate`, `Solarize`, `ShearX`, `ShearY`, `TranslateX`, and `TranslateY`. We use severity 1 and a mixture width of three augmentation chains. The depth of each chain is sampled independently and uniformly from $\{1, 2, 3\}$, and the Dirichlet and Beta distributions use concentration parameter $\alpha = 1.0$. Together with the unmodified anchor image, this produces the 64-view input set used in our default evaluation.

A.3 Source of Baseline Results

Unless otherwise noted, many of the baseline results reported in our tables are taken directly from the original R-TPT (Sheng et al., 2025) and TTP (Li et al., 2026) papers when the evaluation setting matches ours in backbone, dataset, and attack protocol. We reformat these numbers only for presentation consistency across tables. When an exact result is not available in the original paper, we reproduce the baseline using the official implementation; such entries are marked with † in the tables.

B Analysis of Shared Perturbation Structure and Residual Compactness

This analysis asks a simple question: why can T-VSS learn one shared feature correction for many stochastic views of the same attacked image? To answer it, we analyze PGD adversarial examples on the $50{,}000$ ImageNet validation images using the same backbone-specific attack settings as in the main paper, and compare paired clean and adversarial 64-view features under identical stochastic augmentations. For each sample, we measure: (i) the pairwise cosine similarity between the adv-clean feature shifts across views, (ii) the shared-energy ratio of the mean shift, and (iii) the rank required to explain 90% of the residual variance. These statistics directly test whether the perturbation-induced changes are coordinated across views and whether the residual variation is sufficiently compact to support a low-dimensional parameterization.

Table 7 reports the resulting backbone-wise summaries. The pattern is consistent across all three backbones. First, the adv-clean feature shifts remain meaningfully aligned across views, with positive pairwise cosine similarities and substantial shared-energy ratios. This alignment indicates that stochastic views of the same adversarial image do not drift independently and motivates applying one consensus correction across views. Second, although clean multi-view features already exhibit nontrivial structure, the attacked residual rank is much smaller than the clean residual rank, decreasing from 8.28 to 3.37 on ResNet-50, from 8.58 to 1.45 on ViT-B/16, and from 10.31 to 2.13 on ViT-B/32. Importantly, anchor subtraction removes any component shared by all views; the residual-SVD basis therefore does not estimate the absolute adversarial displacement. Instead, the low attacked residual rank shows that the remaining augmentation-relative variation is concentrated in a compact, sample-specific subspace. The two observations thus support different parts of T-VSS: cross-view alignment motivates a shared correction, while residual compactness provides a low-dimensional search space in which reliability-weighted entropy minimization selects that correction. This interpretation is also consistent with the anchor-view ablation in Fig. 4c and the random-basis comparison in Table 11, which together show that T-VSS benefits from the structure of the residual basis rather than from low-dimensional restriction alone.

C Additional Experiments and Analysis

C.1 Results under Robust Pretrained Backbone

Table 8 evaluates whether T-VSS remains effective when the underlying CLIP-ViT-B/32 encoder is already robust-pretrained with TeCoA (Mao et al., 2023). The answer is affirmative: T-VSS achieves the best average robust accuracy at 22.0%, improving over the robust-pretrained baseline itself by 9.7 points and over the strongest test-time baseline, R-TPT, by 0.8 points. The gain is also consistent across individual datasets, where T-VSS attains the best robust accuracy on seven of the eight benchmarks. These results suggest that the proposed feature-space correction is complementary to training-time robustness and can further improve an already strengthened visual encoder without any additional fine-tuning or retraining.

Table 8: Clean (Acc.) and adversarial (Rob.) top-1 accuracy (%) on eight fine-grained datasets with a TeCoA-pretrained CLIP-ViT-B/32 backbone ($\epsilon = 4/255$). Best clean and robust results are highlighted in **bold** and **bold**, respectively.

Method	Caltech101		Pets		Cars		Flower102		Aircraft		DTD		EuroSAT		UCF101		Avg.	
	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.
TeCoA-CLIP-ViT-B/32 ($\epsilon = 4/255$)																		
CLIP-TeCoA	79.3	44.3	**66.9**	15.8	10.2	1.0	**30.8**	9.0	6.6	0.5	24.5	10.7	**14.5**	10.8	34.6	6.7	33.4	12.3
Ensemble	72.7	55.1	59.9	38.9	5.6	2.7	26.6	16.0	4.2	2.0	23.5	16.2	12.5	11.0	26.4	14.0	28.9	19.5
TPT	79.3	52.7	65.2	27.4	9.6	2.0	27.9	12.3	6.7	1.7	25.5	14.6	12.2	11.2	34.9	10.2	32.7	16.5
C-TPT	**79.8**	47.3	66.1	19.5	**10.6**	1.3	29.4	10.7	6.4	0.7	**26.2**	12.4	13.0	11.1	**36.4**	8.1	**33.5**	13.9
MTA	79.7	55.7	66.2	31.2	9.0	2.5	29.1	14.0	6.5	1.6	24.4	13.5	13.3	11.2	34.6	12.5	32.9	17.8
R-TPT	76.1	60.5	63.2	40.1	7.7	3.5	26.6	**16.5**	6.1	2.7	25.2	17.7	11.5	11.3	31.1	17.4	30.9	21.2
T-VSS (Ours)	77.0	**62.4**	61.0	**41.3**	8.7	**4.1**	25.7	16.4	**7.3**	**2.9**	24.5	**18.1**	11.8	**11.4**	31.5	**19.6**	30.9	**22.0**

C.2 Additional CLIP Backbone Results

Table 9 further evaluates T-VSS on CLIP-ViT-B/32 and broadens the comparison to both training-time and test-time defenses. The overall trend remains consistent: T-VSS achieves the best average robust accuracy at 41.0%, outperforming the strongest test-time baseline, TTP, by 1.3 points and the strongest training-time baseline, FARE (Schlarmann et al., 2024), by 0.8 points. The gains are particularly clear on Cars, Flower102, Aircraft, and the overall average, showing that the proposed feature-space correction transfers effectively to this additional backbone. Although some training-time defenses remain competitive on individual datasets or in clean accuracy, they require robust pretraining or adversarial fine-tuning. By contrast, T-VSS delivers the strongest overall robustness without any additional training, reinforcing the practical advantage of direct test-time feature correction.

Table 9: Comparison of training-time and test-time defenses on fine-grained classification datasets with pre-trained CLIP-ViT-B/32 ($\epsilon = 4/255$). Best clean (Acc.) and adversarial (Rob.) results are highlighted in **bold** and **bold**, respectively.

Method	Caltech101		Pets		Cars		Flower102		Aircraft		DTD		EuroSAT		UCF101		Avg.	
	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.
CLIP-ViT-B/32 ($\epsilon = 4/255$)																		
CLIP	91.4	0.2	85.1	0.0	60.1	0.0	64.0	0.0	18.1	0.0	43.0	0.0	35.8	0.0	61.6	0.0	57.4	0.0
Training-time Defense Methods																		
TeCoA	79.3	78.0	66.9	63.7	10.2	9.1	30.8	28.9	6.6	5.9	24.5	24.0	14.5	14.3	34.6	33.4	33.4	32.2
FARE	86.3	**85.4**	76.7	**73.8**	39.2	34.4	37.0	34.0	9.5	8.5	28.3	27.3	16.6	16.3	44.2	41.9	42.2	40.2
APT	10.7	0.4	10.0	0.2	1.5	0.1	0.9	0.2	2.6	0.5	9.0	0.1	7.8	6.7	3.7	0.2	5.8	1.0
APT+TeCoA	81.4	80.2	66.7	63.9	20.8	18.9	42.5	40.4	5.2	5.0	35.2	**33.7**	29.3	**29.2**	40.2	39.4	40.2	38.8
Test-time Defense Methods																		
TTC	86.5	22.7	83.5	11.8	48.1	2.3	64.3	3.2	18.2	1.0	37.3	4.7	**53.0**	3.0	62.6	6.1	56.7	6.9
Ensemble	88.2	74.9	75.0	52.5	51.7	25.9	58.1	36.1	16.4	7.9	39.8	28.6	30.8	11.9	54.9	36.9	51.9	34.3
MTA	**92.0**	76.3	**86.3**	53.6	**63.4**	26.4	**64.4**	36.5	**20.2**	8.2	**43.8**	28.8	34.6	11.3	**63.3**	39.1	**58.5**	35.0
R-TPT	90.6	76.4	84.5	55.8	63.1	28.4	62.6	37.6	19.1	9.2	42.1	29.1	32.0	5.1	62.8	41.0	57.1	35.3
TTP	90.9	81.8	84.7	61.0	59.8	29.8	63.6	42.0	18.0	10.3	42.8	32.2	35.6	14.1	61.3	**46.6**	57.1	39.7
T-VSS (Ours)	91.7	76.6	85.3	63.2	61.2	**43.8**	62.4	**45.2**	19.6	**15.8**	43.3	33.1	29.8	6.6	61.7	43.8	56.9	**41.0**

C.3 Additional Robustness under Stronger Attacks

Table 10 extends the evaluation beyond the PGD setting used in the main paper by considering stronger composite and first-order attacks on CLIP-ViT-B/16. Across both Flower102 and DTD, T-VSS remains consistently robust under AutoAttack (Croce & Hein, 2020), APGD-CE, and APGD-DLR, achieving the best adversarial accuracy in every reported setting. The gains are especially clear on Flower102, where T-VSS improves over the strongest prior baseline by 5.9 points under AutoAttack, 5.6 points under APGD-CE, and 5.1 points under APGD-DLR. On DTD, the advantage is smaller but still consistent under AutoAttack and APGD-DLR, while under APGD-CE the advantage is substantially larger: T-VSS increases robust accuracy from 32.4% for R-TPT to 45.1%. These results strengthen the main claim of the paper: the benefit of T-VSS does not depend narrowly on a single PGD configuration, but persists under diverse optimization-based attacks, suggesting that the proposed low-rank feature correction provides a more stable adaptation mechanism than prior prompt-space or input-space defenses.

Table 10: Adversarial accuracy (%) under additional attacks on Flower102 and DTD with CLIP-ViT-B/16. Best results are highlighted in **bold**.

Method	Flower102			DTD		
	AutoAttack	APGD-CE	APGD-DLR	AutoAttack	APGD-CE	APGD-DLR
CLIP	0.0	0.0	0.0	0.0	0.0	0.0
R-TPT	39.2	39.5	46.5	32.4	32.4	34.6
TTP	26.7	38.9	27.1	22.3	26.0	22.6
T-VSS (Ours)	**45.1**	**45.1**	**51.6**	**33.3**	**45.1**	**36.2**

C.4 Importance of the Residual-SVD Basis

Table 11 examines whether the gain of T-VSS comes merely from restricting adaptation to an arbitrary low-dimensional subspace, or from using the sample-specific basis estimated from anchor-based residuals. To isolate this question, we replace the residual-SVD basis with a random orthonormal basis of the same selected rank m, while keeping the rest of the adaptation pipeline unchanged. T-VSS consistently outperforms this random-basis variant on all four CLIP backbones. The robust-accuracy gain is especially clear, improving over the random basis by 4.5 points on ResNet-50, 7.6 points on ViT-B/16, 5.7 points on ViT-B/32, and 5.0 points on ViT-L/14, while also slightly improving clean accuracy. These results show that the benefit of T-VSS cannot be explained by dimensionality reduction alone: the residual-SVD basis provides an augmentation-supported local geometry that constrains entropy minimization more effectively than an arbitrary orthonormal subspace.

Table 11: Comparison with a random orthonormal basis of the same selected rank m. Results are averaged over the eight fine-grained datasets.

Method	ResNet-50		ViT-B/16		ViT-B/32		ViT-L/14	
	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.	Acc.	Rob.
Random	51.6	44.0	59.3	38.2	56.3	35.3	67.9	48.8
Residual-SVD (Ours)	**52.2**	**48.5**	**60.4**	**45.8**	**56.9**	**41.0**	**68.1**	**53.8**

C.5 Ablation of Update Step

Figure 5 shows the effect of increasing the number of test-time update steps on the average clean and adversarial accuracy over the eight fine-grained datasets with CLIP-ResNet-50. Adversarial accuracy improves steadily as more update steps are used, increasing from 48.5% with one step to 51.3% with five steps. This gain comes with a small clean-accuracy cost: clean accuracy drops from 52.2% to 51.4% and then largely saturates after three steps. The overall trend highlights a clear clean–robustness trade-off, where additional optimization steps can further strengthen low-rank feature-space adaptation under attack, while the default one-step setting remains attractive when inference efficiency and clean performance are both important.

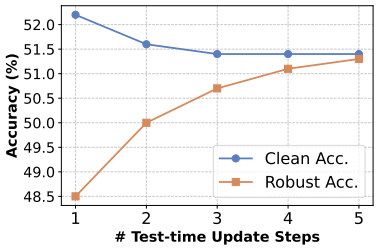


Figure 5: Effect of the number of test-time update steps on average clean (Acc.) and adversarial (Rob.) accuracy (%) over the eight fine-grained datasets with CLIP-ResNet-50.

Table 12: Mean ± standard deviation of clean and adversarial (robust) accuracy (%) averaged over the eight fine-grained datasets across three independent runs with different random seeds. T-VSS shows consistently low variance across all backbones.

Backbone	Acc.	Rob.
ResNet-50	$52.2_{\pm 0.1}$	$48.5_{\pm 0.1}$
ViT-B/16	$60.4_{\pm 0.1}$	$45.8_{\pm 0.2}$
ViT-B/32	$56.9_{\pm 0.1}$	$41.0_{\pm 0.1}$
ViT-L/14	$68.1_{\pm 0.1}$	$53.8_{\pm 0.1}$

C.6 Stability Across Random Seeds

Table 12 reports the mean and standard deviation of clean and adversarial accuracy averaged over the eight fine-grained datasets across three independent runs with different random seeds. T-VSS exhibits uniformly low variance across all backbones, with fluctuations of at most 0.2 points in both clean and adversarial accuracy. This stability indicates that the method is not sensitive to seed-specific initialization or stochastic view generation at test time. In other words, the gains reported in the main paper are not driven by a favorable run, but are reproduced consistently across independent trials.

C.7 Comparison with DBD

DBD (Liu et al., 2026) is a strong concurrent training-free defense that also reconstructs visual features at test time, but it follows a different inference regime from the optimization-based methods in our main benchmark. Specifically, DBD estimates a single defense direction from transformed views and applies a DB-score-based thresholded reconstruction rule with validation-calibrated hyperparameters, whereas T-VSS performs sample-wise test-time optimization in a low-rank visual subspace without thresholded routing. Table 13 shows a case study on Caltech101. On standard CLIP-ViT-B/32, DBD is substantially stronger than all optimization-based baselines, which is consistent with the effectiveness of its calibrated single-direction reconstruction on the standard backbone. For the TeCoA-CLIP-ViT-B/32 result, we apply DBD using the DB-score threshold reported in the original paper, without additional recalibration for the robust backbone. Under this setting, DBD still improves adversarial accuracy over the TeCoA backbone, but its advantage becomes much smaller and it achieves lower clean accuracy and slightly lower adversarial accuracy than T-VSS. We do not claim that DBD cannot be improved further with backbone-specific retuning; rather, this case

Table 13: Case-study comparison with DBD on Caltech101 under standard and robustly pretrained ViT-B/32 backbones. DBD is strongest on standard CLIP-ViT-B/32, whereas T-VSS attains better clean accuracy and slightly higher adversarial accuracy on TeCoA-CLIP-ViT-B/32. † indicates reproduced results.

Method	Acc.	Rob.
CLIP-ViT-B/32 ($\epsilon = 4/255$)		
CLIP	91.4	0.2
R-TPT	90.6	76.4
DBD†	90.4	**98.8**
T-VSS (Ours)	**91.7**	76.6
TeCoA-CLIP-ViT-B/32 ($\epsilon = 4/255$)		
CLIP-TeCoA	**79.3**	44.3
R-TPT	76.1	60.5
DBD†	70.9	60.9
T-VSS (Ours)	77.0	**62.4**

Table 14: Average selected rank m across eight fine-grained datasets under the default 64-view protocol (one original image + 63 augmented views). Adversarial rows use the same backbone-specific PGD attacks as in the main paper. Since T-VSS optimizes only the m-dimensional coefficient vector, the selected rank is also the number of sample-wise learnable parameters.

Setting	Model	Caltech101	Pets	Cars	Flower102	Aircraft	DTD	EuroSAT	UCF101	Avg.
Clean	ResNet-50	13.3	14.9	14.2	14.1	16.6	12.1	8.3	13.7	13.4
	ViT-B/16	12.3	13.8	13.4	12.6	17.0	12.6	7.9	12.6	12.8
	ViT-B/32	14.0	15.5	14.8	14.6	17.7	13.5	9.8	14.4	14.3
	ViT-L/14	14.8	15.5	14.5	14.6	16.5	15.4	11.7	14.2	14.6
Adversarial	ResNet-50	6.8	6.0	5.9	5.4	6.3	5.4	3.6	6.5	5.7
	ViT-B/16	2.0	1.5	1.9	1.6	2.2	1.5	1.4	2.2	1.8
	ViT-B/32	3.1	2.6	3.0	3.2	3.3	2.5	2.0	3.6	2.9
	ViT-L/14	2.2	1.9	2.1	1.7	2.9	2.1	2.3	2.4	2.2

study suggests that its fixed thresholded rule may transfer less directly across backbones when robust pretraining changes the underlying feature geometry. By contrast, T-VSS uses sample-specific low-rank optimization without hard thresholding, which may help it transfer more favorably to the robustly pretrained backbone considered here.

C.8 Selected Rank m and Number of Learnable Parameters

Table 14 reports the average selected rank m across the eight fine-grained datasets under the default 64-view protocol, which consists of one original image and 63 augmented views. For adversarial evaluation, the rank is measured under the same backbone-specific PGD attacks used in the main paper. Since T-VSS optimizes only the m-dimensional coefficient vector α, the selected rank is exactly the number of sample-wise learnable parameters at test time. Even though T-VSS uses 63 augmented views, the average rank on adversarial examples remains very small: 5.7 for ResNet-50, 1.8 for ViT-B/16, 2.9 for ViT-B/32, and 2.2 for ViT-L/14. In other words, T-VSS typically performs sample-wise adaptation with only a handful of learnable coefficients, far fewer than prior optimization-based defenses such as R-TPT and TTP, which optimize higher-dimensional prompt or input variables at test time. The clear reduction from the clean-image ranks to the adversarial ranks is also consistent with our main analysis that attacked multi-view residuals become markedly more compact, which makes shared low-rank steering both effective and parameter-efficient.

C.9 Shared versus Per-view Feature Steering

Although stochastic augmentations produce distinct visual features, all views originate from the same attacked image and retain a common adversarial component, as supported by the cross-view alignment statistics in Table 7. Applying a shared correction $\Delta = V_m\alpha$ does not collapse the views to an identical representation: each adapted feature preserves its view-specific component f_i in Eq. 5. Instead, sharing Δ regularizes adaptation by preventing individual views from independently fitting augmentation artifacts or spurious low-entropy directions.

To test this design choice, we replace the shared correction with independently optimized per-view corrections $\Delta_i = V_m\alpha_i$, while retaining the same residual-SVD basis and optimization objective. As shown in Table 15, the per-view variant obtains 44.9% average robust accuracy across the eight fine-grained datasets on CLIP-ViT-B/16, compared with 45.8% for shared steering. Thus, despite using parameters that scale linearly with the number of views, independent corrections perform 0.9 percentage points worse, supporting the shared correction as an effective cross-view regularizer.

Table 15: Average robust accuracy (%) of per-view and shared steering on CLIP-ViT-B/16.

Correction	Rob.
Per-view steering	44.9
Shared steering (Ours)	**45.8**

Table 16: Robust accuracy (%) under adaptive EOT-PGD with matched 64-view attack and defense budgets on CLIP-ResNet-50. Best results are highlighted in **bold**.

Method	Caltech101	Pets	Cars	Flower102	Aircraft	DTD	EuroSAT	UCF101	Avg.
R-TPT	65.6	36.8	14.0	25.8	4.6	23.1	2.5	26.2	24.8
TTP	13.9	2.1	0.7	1.5	0.1	4.4	1.1	3.1	3.4
T-VSS (Ours)	**71.7**	**65.1**	**31.3**	**40.1**	**10.0**	**26.7**	**3.8**	**40.0**	**36.1**

D Limitations and Future Work

Like other augmentation-driven test-time defenses considered in this work, T-VSS relies on stochastic multi-view augmentation at inference time. While this mechanism is effective under the defense-oblivious threat model used in our main experiments, the augmentation and adaptation pipeline can itself become an attack surface when the adversary explicitly accounts for the defense. We assess this limitation using defense-aware PGD with Expectation over Transformation (EOT) (Athalye et al., 2018), which differentiates through the stochastic augmentation distribution and the test-time optimization procedure. By granting the attacker full knowledge of the inference pipeline, this setting is more stringent than our primary benchmark. Although such access may exceed what is available in some deployment scenarios, it provides an important adaptive stress test for randomized test-time defenses. We evaluate R-TPT, TTP, and T-VSS under this setting on CLIP-ResNet-50 using the same PGD parameters as in the main experiments: 7 attack steps, $\epsilon = 1/255$, and a step size of $\epsilon/4$. We also match the attack and defense budgets at 64 views. At each PGD step, the gradient is estimated from one stochastic defended forward pass comprising the base image and 63 augmented views, and defended inference follows the same 64-view protocol.

As shown in Table 16, this stringent adaptive evaluation reduces the robustness of all three methods relative to their defense-oblivious results, confirming that differentiating through the multi-view inference mechanism poses a meaningful challenge. Nevertheless, T-VSS achieves the best robust accuracy on every dataset and an eight-dataset average of 36.1%, outperforming R-TPT by 11.3 percentage points and TTP by 32.7 percentage points. Thus, among the evaluated augmentation-driven test-time defenses, T-VSS provides the strongest robustness under the matched defense-aware setting, demonstrating that low-rank feature steering retains a substantial advantage even when the attacker has full knowledge of the inference pipeline. A gap from defense-oblivious robustness remains, and future work should investigate how to preserve the representational benefits of multi-view inference while further reducing the attack surface introduced by differentiable stochastic augmentation and test-time adaptation.