NAP-Tuning: Neural Augmented Prompt Tuning for Adversarially Robust Vision-Language Models

Jiaming Zhang, Xin Wang, Xingjun Ma, Lingyu Qiu, Yu-Gang Jiang, *Fellow, IEEE*, and Jitao Sang

***Abstract*—Vision-Language Models (VLMs) such as CLIP have demonstrated remarkable capabilities in understanding relationships between visual and textual data through joint embedding spaces. Despite their effectiveness, these models remain vulnerable to adversarial attacks, particularly in the image modality, posing significant security concerns. Building upon our previous work on Adversarial Prompt Tuning (AdvPT), which introduced learnable text prompts to enhance adversarial robustness in VLMs without extensive parameter training, we present a significant extension by introducing the Neural Augmentor framework for Multi-modal Adversarial Prompt Tuning (*NAP-Tuning*). Our key innovations include: (1) extending AdvPT from text-only to multi-modal prompting across both text and visual modalities, (2) expanding from single-layer to multi-layer prompt architectures, and (3) proposing a novel architecture-level redesign through our Neural Augmentor approach, which implements feature purification to directly address the distortions introduced by adversarial attacks in feature space. Our NAP-Tuning approach incorporates token refiners that learn to reconstruct purified features through residual connections, allowing for modality-specific and layer-specific feature correction. Comprehensive experiments demonstrate that NAP-Tuning significantly outperforms existing methods across various datasets and attack types. Notably, our approach shows significant improvements over the strongest baselines under the challenging AutoAttack benchmark, outperforming them by *33.5%* on ViT-B16 and *33.0%* on ViT-B32 architectures while maintaining competitive clean accuracy. This work establishes the importance of architecture redesign in prompt tuning for adversarial robustness, moving beyond loss-focused approaches to create an adaptive defense mechanism that can identify and rectify adversarial perturbations across embedding spaces.**

***Index Terms*—Adversarial robustness, Vision-Language models, Prompt tuning, Feature purification, Neural augmentation**

I. Introduction

Large-scale pre-trained Vision-Language Models (VLMs) have demonstrated remarkable capabilities in understanding and connecting visual and textual information, enabling impressive performance on a wide range of downstream tasks. Models like CLIP [1] and ALIGN [2] have shown unprecedented zero-shot transfer abilities by learning from web-scale image-text pairs. As these models gain widespread adoption in real-world applications, ensuring their robustness against adversarial attacks becomes increasingly critical.

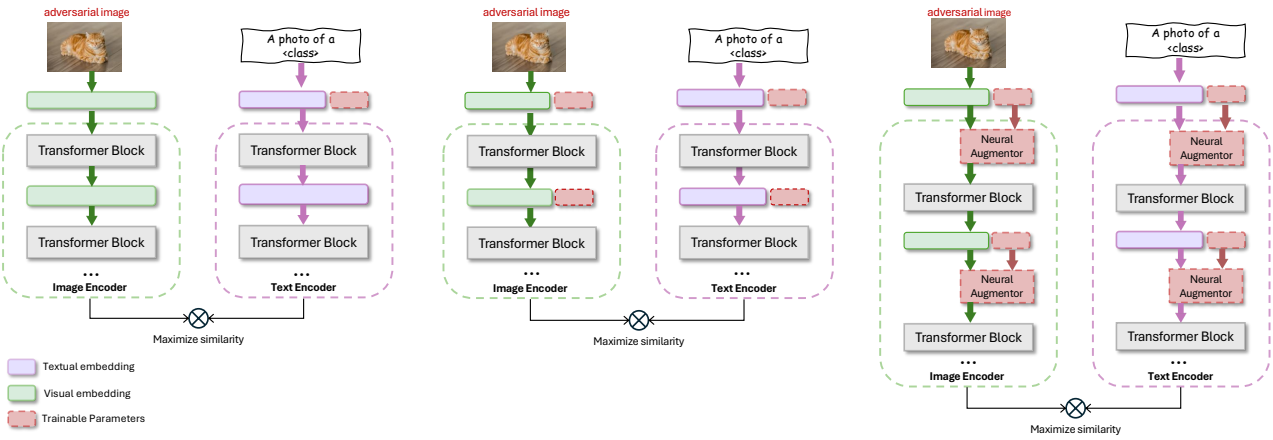
In our previous work [3], we introduced Adversarial Prompt Tuning (AdvPT), a novel approach that enhances the adversarial robustness of VLMs by aligning text embeddings with adversarial image embeddings through learnable text prompts. This method represented a paradigm shift from traditional adversarial training [4] approaches by focusing on prompt-level modifications rather than model parameter retraining, offering significant computational advantages while maintaining effectiveness.

While AdvPT demonstrated the potential of prompt tuning for adversarial defense, it exhibited three critical limitations: (1) its restriction to text modality prompts, (2) its reliance on single-layer prompting, and (3) its emphasis on loss function redesign rather than architectural innovations. Following our initial work, several approaches [5], [6] have attempted to address some of these issues, particularly by extending prompting to multiple modalities and layers. However, these incremental improvements fail to address the core architectural challenge of prompt tuning in adversarial settings.

The fundamental limitation across all existing approaches—including both our original AdvPT and subsequent works—is the direct transplantation of prompt tuning techniques from the generalization domain (*e.g.,* CoOp [7] and MaPLe [8]) to adversarial defense without rethinking the underlying architecture. This approach overlooks a critical insight: prompt tuning, originally developed for clean data environments, faces fundamentally different challenges in adversarial scenarios where feature spaces are deliberately corrupted. Adversarial perturbations introduce systematic distortions that cannot be effectively countered through loss modifications alone, regardless of whether the prompting occurs in single or multiple modalities, or at one or multiple layers.

To comprehensively address all three limitations, we propose the Neural Augmentor framework for Multi-modal Adversarial Prompt Tuning (**NAP-Tuning**). Our approach not only extends prompting to multiple modalities and layers but, most critically, fundamentally reconceptualizes prompt architecture for adversarial defense by integrating feature purification mechanisms. The Neural Augmentor framework incorporates token refiners—lightweight neural networks that learn to reconstruct clean feature representations from adversarially perturbed inputs through residual connections. This architecture enables modality-specific and layer-specific feature correction, yielding substantially enhanced adversarial robustness. Fig. 1 illustrates the evolution from text-only prompting to our comprehensive NAP-Tuning, demonstrating how our method facilitates clean feature reconstruction from adversarially corrupted inputs across both modalities and multiple network layers.

Through extensive experiments under rigorous evaluation protocols, we demonstrate that our approach significantly outperforms our initial AdvPT and other state-of-the-art methods, achieving superior robustness while maintaining competitive clean accuracy. Our analysis revealed that NAP-



(a) Original AdvPT approach (b) Existing Multimodal AdvPT (c) Our proposed NAP-Tuning

Fig. 1: Comparison of adversarial prompt tuning approaches: (a) The original AdvPT that uses only text prompts, (b) Existing multimodal approaches that incorporate prompts in both visual and text pathways, and (c) Our proposed NAP-Tuning framework that extends multimodal prompting with feature purification via token refiners, enabling the reconstruction of clean feature representations from adversarially perturbed inputs.

Tuning effectively learns to correct adversarial distortions in feature space, providing insights into the working mechanisms of successful adversarial defenses.

As an extension of our conference paper [3], this work makes several significant contributions:

- **Multi-modal defense framework:** We extend the initial text-only AdvPT to a comprehensive multi-modal framework that incorporates learnable prompts in both text and visual pathways, creating a more effective defense mechanism that addresses vulnerabilities across modalities.
- **Multi-layer prompt architecture:** We expand from single-layer prompting to multi-layer prompt architectures, allowing for more nuanced and layer-specific defenses that can address adversarial perturbations manifesting differently at various depths of the network.
- **Neural Augmentor design:** Most significantly, we propose a novel architecture-level redesign through our Neural Augmentor approach, which implements feature purification via token refiners. This represents a paradigm shift from previous loss-focused prompt tuning methods to architecture-focused prompt tuning defense, addressing the fundamental limitations of existing approaches.
- **Comprehensive evaluation:** We provide extensive experimental validation across multiple datasets and attack types under stricter evaluation settings. Our approach demonstrates significant performance improvements over the previous conference version (AdvPT), achieving average gains of 42.6% under PGD-100 attacks and 47.5% under the more challenging AutoAttack benchmark, establishing new standards for adversarial robustness in VLMs.

II. Related Work

A. Vision-Language Models

VLMs have established themselves as powerful frameworks for bridging visual and textual modalities. These models can be categorized into two principal architectures. The first category comprises generative models built upon Large Language Models (LLMs), such as LLaVA [9] and Qwen-VL [10], which extend text-only LLMs with vision encoders to enable multimodal understanding and generation.

The second category—which is the focus of adversarial prompt tuning techniques—utilizes contrastive learning to establish joint embeddings of images and text. Models such as CLIP [1] and ALIGN [2] project both modalities into a unified semantic space where related content exhibits higher similarity. These models demonstrate exceptional capabilities across diverse tasks through their dual-encoder architecture. However, their susceptibility to adversarial attacks, particularly in the visual domain, poses significant security concerns as their deployment expands into critical applications.

B. Prompt Learning

Prompt learning emerged in natural language processing as an efficient parameter-efficient adaptation technique for pre-trained language models [11], [12]. This paradigm focuses on optimizing input transformations rather than modifying model parameters.

For VLMs, text-based prompt learning methods such as CoOp [7] and CoCoOp [13] demonstrated that replacing hand-crafted templates with learnable context vectors could significantly improve model performance. Our previous work, AdvPT [3], applied this concept to adversarial defense, showing that learnable text prompts could enhance robustness against adversarial attacks without modifying the underlying model. Multi-modal prompt learning approaches including MaPLe [8] and PromptKD [14] subsequently extended this paradigm by incorporating prompts in both textual and visual pathways. These architectural innovations have provided foundational techniques for multi-modal adversarial prompt tuning, though their primary focus on generalization rather than robustness presents limitations when applied to adversarial defense.

C. Adversarial Prompt Tuning

Conventional adversarial defense methods such as adversarial training [4], TRADES [15], and MART [16] have proven effective but remain computationally prohibitive for large-scale VLMs. These approaches typically require training the whole model parameters from scratch, making them impractical for deployment in VLMs. An innovative alternative approach was introduced by Mao et al. [17] through their TeCoA framework, which intelligently leverages pre-trained VLM weights as initialization and selectively finetunes CLIP's vision encoder.

Our previous work, AdvPT [3], introduced a more efficient and more effective paradigm by applying prompt learning to adversarial defense in VLMs. By aligning text embeddings with adversarial image embeddings through learnable prompts, AdvPT demonstrated that robust representations could be achieved with minimal computational overhead. Recent extensions to this approach include FAP [5], which leverages limited data to learn adversarially correlated text supervision while enhancing multi-modal feature consistency and uni-modal feature differentiation between natural and adversarial examples.

Recent advances in test-time optimization have yielded particularly impressive results in defending VLMs against adversarial attacks. Sheng et al. [18] developed R-TPT, an elegant test-time prompt tuning method that enhances the robustness of vision-language models by minimizing pointwise entropy and introducing a reliability-based weighted ensembling strategy. Their approach demonstrates the power of adaptive defenses that operate at inference time. TAPT [19] introduces an unsupervised approach that optimizes bimodal prompts at test time to enhance CLIP's robustness while maintaining clean performance. In parallel, Xing et al. [20] introduced TTC, an innovative training-free defense mechanism that ingeniously leverages CLIP's pre-trained vision encoder to counteract adversarial images during inference. Their approach offers complementary protection that can be effectively combined with existing robust models, demonstrating the value of ensemble strategies in comprehensive defense frameworks.

While these methods have achieved incremental improvements through loss function innovations, they overlook a critical consideration: the fundamental mismatch between prompt architectures designed for generalization and the unique requirements of adversarial robustness. Addressing adversarial vulnerabilities requires mechanisms specifically engineered to counteract feature-level distortions—a challenge that far exceeds general-purpose representation enhancement. This insight motivates our current work, which demonstrates that architectural innovations in prompt design can substantially improve adversarial robustness beyond what is possible through loss function modifications alone. By reimagining prompt architecture specifically for adversarial defense, we address the feature distortion mechanisms that underlie successful attacks on VLMs.

III. Preliminary: Adversarial Prompt Tuning

To provide context for our extended approach, we first revisit the key concepts of Adversarial Prompt Tuning (AdvPT) as introduced in our previous work [3].

A. Revisiting CLIP

We first provide a brief overview of VLMs, with a particular emphasis on the CLIP [1] architecture. Although our methods are specifically designed for CLIP, they are potentially extendable to a broader range of VLMs that are based on the contrastive learning paradigm. CLIP consists of an image encoder f_I and a text encoder f_T, which project images and text into a shared embedding space. The model is trained to maximize the similarity between matched image-text pairs while minimizing the similarity between unmatched pairs.

For an image x and a text description t, CLIP computes embeddings $f_I(x)$ and $f_T(t)$, and the similarity is calculated as:

$$s(x,t) = \frac{f_I(x) \cdot f_T(t)}{||f_I(x)|| \cdot ||f_T(t)||} \tag{1}$$

For classification tasks, CLIP computes the similarity between an image and a set of text templates describing each class (e.g., "a photo of a [CLASS]"), and selects the class with the highest similarity.

B. Adversarial Attacks on VLMs

Adversarial attacks on VLMs can be categorized into two types based on the attacker's access to model components. In our previous work [3], we considered a restricted threat model where the attacker only has access to the image encoder. Under this assumption, adversarial examples are crafted by maximizing the KL divergence between clean and perturbed image features:

$$\begin{aligned} x_{adv} &= \arg\max_{x'} D_{KL}(f_I(x), f_I(x')) \\ &\text{s.t.} \quad ||x' - x||_p \leq \epsilon, \end{aligned} \tag{2}$$

where D_{KL} is the KL divergence measure. In this work, we consider a more challenging threat model that assumes the attacker has access to both image and text encoders, including textual prompt. This stronger adversary can generate more effective attacks by maximizing the cross-entropy loss between image-text pairs (even when the text is represented as learnable vectors):

$$\begin{aligned} x_{adv} &= \arg\max_{x'} \mathcal{L}_{CE}(f_I(x'), f_T(t_y)) \\ &\text{s.t.} \quad ||x' - x||_p \leq \epsilon, \end{aligned} \tag{3}$$

where t_y is the text template for class y, and $\mathcal{L}_{CE}$ is the cross-entropy loss measuring image-text similarity.

C. Basic Adversarial Prompt Tuning

AdvPT addresses the vulnerability of VLMs to adversarial attacks by learning to align text embeddings with adversarial image embeddings through prompt tuning. The key insight is that this alignment can enhance robustness without requiring modifications to the model architecture. In AdvPT, the standard text template "a photo of a [CLASS]" is replaced with a template containing learnable context vectors:

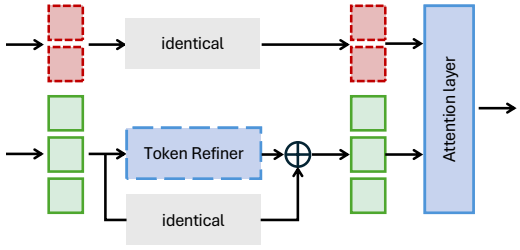


Fig. 2: Overview of our proposed Neural Augmentor module for multi-modal AdvPT.

$$(\mathcal{V}_t, y) = [\mathcal{V}_t, [\text{CLASS}]], \tag{4}$$

where $\mathcal{V}_t$ are learnable vectors that are optimized to align with adversarial image embeddings, and y represents class label. The training objective for AdvPT is to maximize the similarity between the clean text embedding (with learned prompt) and the adversarial image embedding:

$$\min_{\mathcal{V}_t} \mathcal{L}(x_{adv}, y) = \max_{\mathcal{V}_t} s(x_{adv}, (\mathcal{V}_t, y)), \tag{5}$$

where x_{adv} is the adversarial version of image x, and $(\mathcal{V}_t, y)$ is the text template for the true class y.

IV. Neural Augmentor for Multi-Modal Adversarial Prompt Tuning

Adversarial attacks fundamentally operate by introducing carefully crafted perturbations to input data that cause neural networks to make incorrect predictions. In the context of VLMs, these perturbations manipulate feature representations in ways that disrupt the crucial alignment between visual and textual information. Building upon our preliminary work on AdvPT, we introduce a comprehensive redesign of prompt-based adversarial defenses through our NAP-Tuning framework. This section details our approach, its architectural innovations, and theoretical foundations.

A. Framework Overview

Fig. 1 (c) introduces the overall framework, while Fig. 2 illustrates the architecture of our Neural Augmentor module. At its core, our approach aims to defend VLMs against adversarial attacks by addressing their fundamental vulnerability: the distortion of feature representations. Unlike previous prompt defenses that focus on loss function modifications or robust classification boundaries, our method directly targets the feature corruption mechanism through specialized architectural components.

The framework consists of three primary components that work in concert. First, a multi-modal prompting system introduces learnable prompt vectors that operate simultaneously in both textual and visual pathways of the model. Second, a multi-layer prompt architecture implements hierarchical prompting structures at different depths within the transformer architecture to address adversarial effects at various levels of feature abstraction. Third, Neural Augmentor modules serve as specialized neural networks that perform feature purification through token refiners.

B. Structural Innovations

1) Multi-Modal Prompting: A key limitation of our original AdvPT approach was its exclusive focus on the text modality, which left the visual pathway vulnerable to direct attacks. Our enhanced framework addresses this limitation by implementing learnable prompts in both textual and visual pathways, creating a coordinated defense system.

For the text pathway, we build upon the original AdvPT framework with enhanced prompt vectors $(\mathcal{V}_t, y)$. For the visual pathway, we introduce a parallel set of prompt vectors that operate on the visual inputs $(\mathcal{V}_i, x_{adv})$, where $\mathcal{V}_i$ are learnable visual prompt vectors. These visual prompts serve a complementary function to the text prompts, helping to guide perturbed visual representations back toward their clean manifold. Equation (5) can therefore be reformulated as:

$$\min_{\mathcal{V}_i, \mathcal{V}_t} \mathcal{L}(x_{adv}, y) = \max_{\mathcal{V}_i, \mathcal{V}_t} s((\mathcal{V}_i, x_{adv}), (\mathcal{V}_t, y)). \tag{6}$$

The dual-modal prompt system enables a more comprehensive defense, as it can address attacks that target either modality or the cross-modal alignment. This is particularly important in VLMs, where adversarial perturbations can disrupt the crucial cross-modal matching that underlies the model's performance.

2) Multi-Layer Prompt Architecture: Adversarial perturbations manifest differently across network depths, affecting both low-level perceptual features and high-level semantic representations. Our multi-layer prompt architecture addresses this by placing learnable prompts at multiple depths within the transformer network:

$$h^{(l)} = \text{Layer}^{(l)}([\mathcal{V}^{(l)}, h^{(l-1)}]), \tag{7}$$

where h^l represents the output of layer l, $\mathcal{V}^{(l)}$ are layer-specific learnable prompt vectors, and $\text{Layer}^{(l)}$ is the transformer layer function. This hierarchical prompting structure allows for targeted interventions at different levels of feature abstraction. Let $\mathbf{V}_i = \{\mathcal{V}_i^j\}_{j=1}^l$ and $\mathbf{V}_t = \{\mathcal{V}_t^j\}_{j=1}^l$, then Equation (6) can be reformulated as:

$$\min_{\mathbf{V}_i, \mathbf{V}_t} \mathcal{L}(x_{adv}, y) = \max_{\mathbf{V}_i, \mathbf{V}_t} s((\mathbf{V}_i, x_{adv}), (\mathbf{V}_t, y)). \tag{8}$$

The multi-layer design offers several advantages over the single-layer approach of our original AdvPT. It enables depth-specific defense, where different layers can address distinct manifestations of adversarial perturbations, from low-level texture disruptions to high-level semantic shifts. It also facilitates progressive feature refinement, where corrections applied at earlier layers can be further refined by later layers, creating a cascading purification effect. More importantly, the increased parameter capacity allows the model to learn more complex defense strategies without modifying the underlying VLM weights.

This hierarchical structure represents a significant advancement over our preliminary work, which was limited to modifying only the textual prompts. By extending the defense mechanism throughout the network, we create a more robust barrier against adversarial perturbations that might otherwise bypass single-layer defenses.

C. Neural Augmentor Design

The most significant innovation in our framework is the Neural Augmentor—a specialized neural module designed explicitly for feature purification. Unlike conventional prompt tuning that focuses on optimizing context vectors, our Neural Augmentor actively transforms feature representations to counteract adversarial distortions.

1) Feature Distortion Theory: To formalize our approach, we first characterize the effect of adversarial perturbations in feature space. Let $\mathcal{X}$ denote the input space and $\mathcal{F}$ the feature space. For a clean image $x \in \mathcal{X}$ and its adversarial counterpart $x_{adv} = x + \delta$, the feature distortion can be expressed as:

$$\Delta f_I = f_I(x_{adv}) - f_I(x), \tag{9}$$

where $f_I(x)$ represents the feature representation of input x. Traditional adversarial defenses aim to make classification boundaries robust to these distortions. In contrast, our Neural Augmentor directly targets the distortion itself, attempting to recover:

$$\hat{f}_I(x) \approx f_I(x_{adv}) + \Phi(f_I(x_{adv})), \tag{10}$$

where Φ is a learned correction function that approximates $-\Delta f$, effectively "purifying" the adversarial features (including both intermediate and final feature representations) back toward their clean counterparts.

2) TokenRefiner Architecture: The core component of the Neural Augmentor is the TokenRefiner R—a lightweight neural network that processes individual token representations to identify and correct adversarial distortions. For each token representation $z \in \mathbb{R}^d$, the TokenRefiner computes a corrective term $\tilde{z} = R(z)$. The TokenRefiner function is implemented as a two-layer network with a residual connection. It ensures stable gradient flow during training, facilitating effective optimization. Moreover, when no correction is needed (e.g., for clean inputs), the network can learn to output values near zero, effectively preserving the original features through an identity fallback mechanism. The R operates on both text and visual tokens, with modality-specific parameters that allow it to learn distinct correction patterns for each pathway.

We apply the TokenRefiner R to correct potentially perturbed feature representations before combining them with learnable vectors for the attention mechanism. Therefore, Equation (7) can be reformulated as:

$$h^{(l)} = \text{Layer}^{(l)}\big(\text{Attention}(\mathcal{V}^{(l)}, \tilde{h}^{(l-1)})\big), \tag{11}$$

where $\tilde{h}^{(l-1)} = R(h^{(l-1)})$. Through this attention mechanism, we expect the learnable vectors $\mathcal{V}^{(l)}$ and the augmented feature representations $\tilde{h}^{(l-1)}$ to cooperatively optimize and enhance robustness against adversarial attacks. Consequently, the final learning objective can be expressed as an extension of Equation (8):

$$\min_{\substack{\mathbf{V}_i, \mathbf{V}_t, \\ \theta_i, \theta_t}} \mathcal{L}(x_{adv}, y) = \max_{\substack{\mathbf{V}_i, \mathbf{V}_t, \\ \theta_i, \theta_t}} s((\mathbf{V}_i, x_{adv}; R_i), (\mathbf{V}_t, y; R_t)), \tag{12}$$

where θ_i and θ_t represent the learnable parameters of the TokenRefiner R in the visual and text branches, respectively.

The Neural Augmentor implements a feature purification mechanism that operates across three primary dimensions. At the token level, each representation is individually refined, allowing for localized corrections that address token-specific perturbations. At the layer level, TokenRefiner deployed at different network depths learn to address distinct types of perturbations, from low-level feature distortions to high-level semantic shifts. At the modality level, separate TokenRefiner for text and visual pathways allow for specialized correction patterns that address the unique vulnerabilities of each modality.

This multi-dimensional purification approach enables a comprehensive defense against diverse adversarial attacks, addressing perturbations at their source—the feature representations themselves. Rather than treating adversarial robustness as a classification boundary problem, our approach directly targets the mechanism by which adversarial examples cause misclassifications: the distortion of feature representations. By restoring these representations to their clean counterparts, we enable the model to maintain its performance even in the presence of adversarial perturbations.

D. Training Methodology

The efficacy of our NAP-Tuning framework relies on a theoretically grounded training methodology that addresses the fundamental challenge in adversarial learning: optimizing the trade-off between standard accuracy and adversarial robustness while maintaining feature-space integrity. Unlike conventional adversarial training approaches that focus primarily on decision boundary robustification, our method directly targets feature-level purification—a strategy particularly well-aligned with the core operating principle of VLMs, which fundamentally depends on cross-modal feature alignment.

1) Adversarial Example Generation: During training, we generate adversarial examples using the Projected Gradient Descent (PGD) method with the following objective:

$$x_{adv} = \arg\max_{x'} \mathcal{L}_{adv}(f_\theta(x'), y) \quad \text{s.t.} \quad \|x' - x\|_\infty \leq \epsilon, \tag{13}$$

where f_θ represents our model and $\mathcal{L}_{adv}$ denotes the cross-entropy loss. We employ a standard multi-step PGD implementation to generate strong adversarial examples, ensuring that our defense is trained against sophisticated perturbations.

Crucially, the feature-level corrections learned by our model exhibit transferability across different attack types due to the commonality in how various attacks distort the underlying feature manifold. This transferability represents a significant advantage of our feature purification approach over methods that merely harden decision boundaries against specific attack patterns.

2) Principled Loss Formulation: In contrast to previous defense methods that employ complex, multi-term loss functions with heuristically determined components, we derive a principled, minimal training objective from the theoretical foundations of robust learning:

$$\mathcal{L}(\theta) = \mathcal{L}_{clean}(\theta) + \alpha(\tau) \cdot \mathcal{L}_{adv}(\theta), \tag{14}$$

where $\alpha(\tau)$ is a theoretically motivated dynamic balancing coefficient that evolves with training epoch τ. The clean loss $\mathcal{L}_{clean}$ and adversarial loss $\mathcal{L}_{adv}$ are defined as:

$$\mathcal{L}_{clean}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\mathcal{L}(f_\theta(x), y)], \tag{15}$$

$$\mathcal{L}_{adv}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\mathcal{L}(f_\theta(x_{adv}), y)], \tag{16}$$

where $\mathcal{D}$ represents the data distribution, and expectations are taken over the training dataset.

The dynamic balancing coefficient $\alpha(\tau)$ follows a sigmoid schedule that systematically transitions from emphasizing clean performance to prioritizing adversarial robustness:

$$\alpha(\tau) = \alpha_0 \cdot \frac{1}{1 + \exp\left(-10\left(\frac{\tau}{T_{max}} - 0.5\right)\right)}, \tag{17}$$

where α_0 is the maximum weight assigned to the adversarial loss, and T_{max} denotes the total number of training epochs. This schedule implements a curriculum learning strategy that allows the model to first establish representational capacity on clean data before gradually adapting to adversarial inputs.

The theoretical justification for this scheduling approach stems from optimization landscape analysis: the loss surface for adversarial examples typically contains sharper curvature and more local minima than that of clean examples. By initially focusing on clean examples, we guide optimization toward regions of the parameter space with favorable generalization properties before refining these parameters to accommodate adversarial inputs. This procedure effectively navigates the complex optimization landscape of robust learning while mitigating the well-documented trade-off between standard accuracy and adversarial robustness.

V. Experiments

This section systematically evaluates the effectiveness of our proposed NAP-Tuning. We first introduce the experimental setup, then validate the key components of our method through a series of carefully designed experiments, and comprehensively compare with existing approaches.

A. Experimental Setup

1) Datasets and Models: To comprehensively evaluate the effectiveness of our approach, we conduct tests on 11 widely used image classification datasets, including ImageNet [21], Caltech101 [22], DTD [23], EuroSAT [24], FGVC Aircraft [25], Food101 [26], Oxford Flowers [27], Oxford Pets [28], Stanford Cars [29], SUN397 [30], and UCF101 [31]. These datasets encompass a diverse range of visual tasks from fine-grained recognition to scene classification. We follow the training and testing splits defined in [7]. For the ImageNet test set, consistent with prior adversarial attack studies [32]–[34], we randomly sample 1,000 images, ensuring one image per class.

Our experiments are conducted on the CLIP model, where the default configuration employs the publicly available ViT-B/16 architecture unless otherwise specified. We also include results for the ViT-B/32 model as a supplementary evaluation. In accordance with previous work, we use hand-crafted prompts as textual inputs (e.g., "`a photo of a <class>, a type of pet`" for Oxford Pets).

2) Attack Methods: To evaluate adversarial robustness, we implement both white-box and black-box adversarial attacks with more stringent settings compared to our conference version. For white-box attacks, we use learnable textual vectors to construct cross-entropy loss and employ stronger PGD-100 [4] and AutoAttack [35]. For black-box attacks, we utilize more advanced methods such as CWA [36] and AGS [37].

3) Defense Baselines: We compare our method against several strong baselines, including variants of AdvPT and the recent FAP [5]:

- **AdvPT**: the original Adversarial Prompt Tuning method, which performs prompt tuning exclusively on the text side. It adapts the textual input prompts to improve robustness without modifying the visual inputs.
- **AdvPT-V**: an extension of AdvPT that applies prompt tuning on the vision side. Specifically, it introduces additional learnable prompts into the visual encoder, enabling adaptation of the visual input space to adversarial perturbations.
- **AdvPT-VLJ** (vision-language joint prompt tuning): this variant simultaneously introduces learnable prompts to both the vision and text modalities. Moreover, it establishes a mapping between the vision and language prompts, allowing joint adversarial tuning of both modalities. This structure follows a design similar to MaPLe [8].
- **AdvPT-VLI** (vision-language independent prompt tuning): a modification of AdvPT-VLJ that removes the mapping between vision and language prompts. In this setup, the learnable prompts for the visual and textual inputs are optimized independently, without explicit cross-modal interactions.
- **FAP** [5]: a recent method proposed for few-shot adversarial robustness, which carefully designs the objective function.

4) Implementation Details: Our training framework spans 90 epochs with a batch size of 512. We adopt the AdamW optimizer with an initial learning rate of 1×10^{-4}, which is modulated through a cosine annealing schedule. For adversarial example generation during training (as defined in Equation (12)), we implement PGD-5 with a default perturbation magnitude $\epsilon = 1/255$ for both training and testing phases, though we analyze the effects of different ϵ values in subsequent sections. For the TokenRefiner R, we implement a default 2-layer neural network architecture, with the impact of network depth examined in our ablation studies. Regarding the multi-layer prompt architecture, our default configuration utilizes all 12 transformer layers, with layer-specific effects analyzed in detailed experiments. The hyperparameter α_0 in Equation (17)

TABLE I: Evaluation results on ViT-B16 under clean and adversarial settings. Performance is reported across multiple datasets under white-box (PGD and AutoAttack) and black-box attacks (CWA and AGS), where † presents our method. Best results are highlighted in **bold**.

		ImageNet	Caltech	DTD	Eurosat	Aircraft	Food	Flowers	Pets	Cars	SUN	UCF	Avg
Clean	Vanilla	66.7	93.3	44.1	48.3	24.7	**85.8**	70.7	89.1	65.6	62.6	67.5	65.3
	FAP	59.7	92.2	60.8	67.8	25.4	71.3	84.5	87.1	61.1	64.0	75.3	70.6
	AdvPT	**67.5**	**94.1**	**69.6**	69.0	28.7	85.0	87.9	**91.7**	70.8	**70.7**	**77.4**	**73.8**
	AdvPT-V	60.7	89.8	36.9	28.1	16.1	63.8	54.4	84.7	51.7	58.4	54.1	57.0
	AdvPT-VLI	63.8	85.7	37.2	18.2	12.0	62.3	53.6	78.7	46.9	52.8	50.5	55.4
	AdvPT-VLJ	65.9	88.0	37.8	20.0	10.3	69.6	59.7	82.1	53.5	56.6	55.8	60.0
	NAP-Tuning†	57.6	92.4	59.7	**83.0**	**56.3**	60.6	**96.1**	82.4	**84.6**	62.2	73.4	71.9
CWA	Vanilla	50.1	91.5	40.5	18.3	21.9	**76.1**	66.7	85.9	60.9	59.8	61.8	61.1
	FAP	49.6	91.8	56.8	23.2	24.3	66.6	82.2	85.4	58.6	61.2	70.6	65.6
	AdvPT	52.1	**93.1**	**61.8**	46.9	30.8	74.9	88.2	**89.8**	**67.8**	**70.5**	**74.3**	**73.0**
	AdvPT-V	51.2	88.4	36.0	23.5	15.4	56.5	53.3	81.9	48.3	55.8	51.6	51.3
	AdvPT-VLI	**54.4**	84.7	35.4	17.7	11.4	55.5	52.5	76.9	45.2	50.1	48.5	50.8
	AdvPT-VLJ	53.0	86.5	35.2	16.4	8.5	52.6	50.7	77.7	42.5	51.0	48.6	49.1
	NAP-Tuning†	48.7	92.2	58.2	**78.9**	**55.3**	59.2	**95.7**	81.7	83.9	60.4	72.3	**73.9**
AGS	Vanilla	53.5	86.8	30.9	11.0	18.1	56.9	56.9	78.0	54.1	53.7	49.7	54.3
	FAP	51.1	88.2	50.2	12.3	22.2	**58.9**	77.4	80.1	53.4	55.0	61.3	58.5
	AdvPT	54.6	88.5	51.5	26.1	26.1	55.9	76.0	**80.4**	62.6	**62.8**	64.1	61.5
	AdvPT-V	53.0	85.8	33.2	15.6	13.6	47.6	47.7	78.2	44.3	51.3	44.3	50.8
	AdvPT-VLI	**56.3**	81.1	33.6	14.7	10.6	46.2	48.5	74.3	41.5	46.2	41.7	47.5
	AdvPT-VLJ	54.9	83.0	31.6	12.1	7.5	44.0	45.3	73.9	38.6	46.2	41.4	47.3
	NAP-Tuning†	50.5	**90.4**	**54.1**	**72.1**	**52.5**	53.7	**93.5**	77.2	**81.0**	56.3	**68.1**	**67.2**
PGD	Vanilla	2.7	28.3	2.0	0.1	0.0	14.4	3.2	8.8	1.4	1.4	4.0	6.7
	FAP	24.0	62.4	26.7	0.0	5.2	21.8	51.7	33.7	19.5	25.4	10.9	23.3
	AdvPT	1.5	27.0	6.6	0.2	0.6	0.9	4.3	2.8	0.7	1.6	2.1	5.2
	AdvPT-V	19.6	61.5	18.6	8.1	3.8	12.4	25.0	38.0	9.8	17.3	17.0	20.4
	AdvPT-VLI	21.8	59.1	18.6	10.0	2.9	12.9	21.7	38.2	10.0	15.7	14.2	20.2
	AdvPT-VLJ	20.0	58.7	16.4	10.2	2.3	11.4	20.5	32.8	8.6	14.9	12.7	18.9
	NAP-Tuning†	**29.6**	**80.8**	**38.9**	**48.7**	**34.4**	**28.8**	**86.6**	**52.1**	**64.8**	**35.2**	**49.5**	**50.7**
AutoAttack	Vanilla	0.0	0.0	0.1	0.1	0.1	0.0	0.0	0.0	0.1	0.0	0.1	0.0
	FAP	1.6	1.1	4.7	2.3	2.7	1.4	1.1	1.9	1.4	2.6	1.8	2.2
	AdvPT	0.1	0.0	0.2	0.2	0.4	0.1	0.1	0.1	0.1	0.1	0.0	0.1
	AdvPT-V	14.9	55.3	15.0	2.2	2.2	8.6	18.8	32.1	5.9	12.5	13.0	17.9
	AdvPT-VLI	17.0	54.3	14.3	9.2	1.6	9.1	16.2	33.6	4.8	11.4	11.3	17.0
	AdvPT-VLJ	10.3	43.9	10.3	3.7	1.2	4.4	9.7	19.8	1.9	6.7	6.0	11.1
	NAP-Tuning†	**28.3**	**80.6**	**38.4**	**44.4**	**34.1**	**28.3**	**86.2**	**50.7**	**64.2**	**33.6**	**48.7**	**51.4**

is set to 5.0. All experiments are conducted on NVIDIA A800 80GB GPUs.

B. Main Results

The main evaluation results are summarized in Tables I and II. Overall, NAP-Tuning achieves highly competitive clean accuracy compared to existing baselines. While its clean performance is slightly lower than that of AdvPT, this is expected, as AdvPT explicitly prioritizes clean accuracy in the robustness-accuracy trade-off, often at the cost of adversarial robustness.

In terms of adversarial robustness, NAP-Tuning demonstrates clear and consistent advantages. Under black-box attacks, particularly CWA and AGS, our method outperforms all baselines across both ViT-B16 and ViT-B32 backbones, highlighting its superior generalization to unseen adversarial threats. More importantly, under white-box attacks such as PGD and AutoAttack, NAP-Tuning exhibits significant improvements. **AutoAttack** is widely regarded as the most rigorous and comprehensive white-box evaluation benchmark, combining several strong attacks to provide a thorough assessment of model robustness. Under this challenging setup, NAP-Tuning achieves substantial gains over all competitors: on ViT-B16, it attains **51.4%** on average under AutoAttack, outperforming the strongest baseline (AdvPT-V) by **33.5%**; on ViT-B32, it reaches **44.0%**, exceeding the best competitor (AdvPT-VLJ) by **33.0%**. These results clearly demonstrate the robustness of NAP-Tuning against powerful, diverse adversarial threats.

C. Multi-layer Prompt Architecture

To investigate how the depth of our multi-layer prompt architecture affects model robustness, we conduct an ablation study by varying the number of layers where prompts are

TABLE II: Evaluation results on ViT-B32 under clean and adversarial settings. Performance is reported across multiple datasets under white-box (PGD and AutoAttack) and black-box attacks (CWA and AGS), where † presents our method. Best results are highlighted in **bold**.

		ImageNet	Caltech	DTD	Eurosat	Aircraft	Food101	Flowers	Pets	Cars	SUN	UCF	Avg
Clean	Vanilla	62.0	91.4	44.2	45.4	19.3	**80.4**	66.6	87.4	60.2	62.1	63.5	62.0
	FAP	54.0	91.0	57.1	47.1	22.4	65.5	82.5	81.6	50.0	62.6	63.8	61.6
	AdvPT	**63.5**	**93.5**	**64.1**	79.0	27.5	79.0	91.3	**88.5**	70.5	**71.9**	**77.8**	**73.3**
	AdvPT-V	58.8	90.2	40.8	24.6	17.1	70.4	59.1	83.6	50.3	59.5	57.1	55.6
	AdvPT-VLI	61.4	89.2	33.3	24.9	10.0	68.6	54.3	83.9	47.1	54.3	56.9	53.1
	AdvPT-VLJ	61.4	87.0	31.1	19.2	9.3	65.7	49.5	83.6	46.0	53.9	55.2	51.1
	NAP-Tuning†	48.9	89.8	55.1	**80.4**	**48.2**	60.2	**94.2**	78.0	**77.2**	59.2	69.3	69.1
CWA	Vanilla	48.6	90.5	40.6	33.4	16.1	**74.6**	61.8	85.2	55.7	59.5	60.1	60.1
	FAP	50.8	90.4	55.4	40.1	21.7	63.9	81.2	80.6	48.9	61.5	62.1	59.7
	AdvPT	**51.2**	**92.9**	**60.0**	54.9	26.7	73.7	88.2	**87.5**	65.5	**69.3**	**74.3**	67.7
	AdvPT-V	48.0	89.2	36.5	10.8	14.3	59.0	52.5	78.0	43.3	54.3	50.5	48.8
	AdvPT-VLI	50.7	87.5	29.4	23.5	8.4	57.9	49.2	80.2	38.3	48.7	48.3	47.5
	AdvPT-VLJ	**51.2**	83.9	29.1	21.7	7.4	56.7	46.0	80.2	35.7	49.3	48.6	46.3
	NAP-Tuning†	46.1	89.5	54.0	**77.6**	**48.2**	57.4	**94.3**	77.6	**77.0**	58.2	68.6	**68.0**
AGS	Vanilla	46.1	85.0	29.8	11.8	13.4	50.9	49.1	73.6	47.5	49.6	46.9	45.8
	FAP	**52.0**	**89.9**	**54.4**	31.6	21.7	**63.4**	81.0	**80.1**	48.8	**61.1**	61.5	58.7
	AdvPT	48.6	88.0	48.5	29.9	23.9	50.5	71.9	74.9	57.3	57.5	60.1	55.6
	AdvPT-V	47.7	85.9	33.6	3.6	12.3	49.3	46.7	73.6	37.9	48.9	44.3	44.0
	AdvPT-VLI	50.6	83.8	26.4	14.7	6.7	49.3	43.0	76.1	33.9	43.7	41.8	42.7
	AdvPT-VLJ	51.2	80.0	26.7	14.6	7.2	48.0	39.5	74.5	32.0	44.5	43.1	41.9
	NAP-Tuning†	45.2	87.4	48.4	**70.6**	**45.9**	52.0	**91.7**	72.3	**73.1**	54.0	**64.9**	**64.1**
PGD	Vanilla	2.0	29.1	4.9	0.2	0.0	7.4	1.7	4.2	0.4	1.5	3.0	6.7
	FAP	23.4	67.1	32.8	0.4	7.0	26.2	57.3	40.0	17.6	29.8	31.5	23.3
	AdvPT	2.8	31.7	10.0	0.4	0.8	1.8	8.7	3.2	2.0	2.9	4.5	5.2
	AdvPT-V	19.0	64.6	19.8	0.3	2.9	15.0	23.0	36.3	9.6	18.9	18.3	20.4
	AdvPT-VLI	20.7	62.1	16.3	5.7	1.5	15.9	19.3	37.6	8.4	16.2	17.7	20.2
	AdvPT-VLJ	20.9	59.9	15.2	10.1	2.0	15.3	18.8	37.5	9.6	16.9	18.2	18.9
	NAP-Tuning†	**26.5**	**77.9**	**34.2**	**49.1**	**25.4**	**28.4**	**82.4**	**41.3**	**52.4**	**31.5**	**45.0**	**44.9**
AutoAttack	Vanilla	0.0	0.4	0.3	0.1	0.1	0.0	0.0	0.0	0.0	0.0	0.1	0.1
	FAP	2.0	1.3	5.0	1.6	3.5	2.0	1.9	1.5	1.0	3.2	2.3	2.2
	AdvPT	0.0	0.6	0.5	0.1	0.2	0.0	0.0	0.1	0.0	0.1	0.0	0.2
	AdvPT-V	8.3	44.8	12.4	0.0	0.5	5.4	8.8	16.3	2.8	7.4	7.2	10.0
	AdvPT-VLI	9.3	44.2	10.5	0.2	0.4	5.6	7.6	16.4	2.1	6.2	6.8	10.1
	AdvPT-VLJ	10.1	44.7	10.3	1.2	0.6	6.1	7.9	18.9	2.3	7.2	8.3	11.0
	NAP-Tuning†	**20.9**	**77.7**	**34.0**	**48.8**	**25.2**	**28.0**	**82.3**	**40.9**	**50.7**	**31.1**	**44.6**	**44.0**

inserted, from 1 to 12. Fig. 3 presents the performance on eleven diverse datasets under both clean and adversarial conditions.

The results reveal several key insights. First, as shown in the average performance (Fig. 3l), both clean and robust accuracy generally improve as the number of layers increases, confirming the effectiveness of our hierarchical prompting strategy. Interestingly, we observe dataset-specific patterns related to task complexity. For more challenging datasets such as ImageNet (Fig. 3f), SUN397 (Fig. 3j), and Food101 (Fig. 3e), performance follows an inverted U-shape, peaking at intermediate layer depths before declining. This pattern indicates mild overfitting when increasing parameter capacity without corresponding increases in training examples (shot=16). The highlighted regions in these plots mark the optimal layer ranges for these datasets.

These findings reveal the potential for further performance enhancements through larger training sets and dataset-specific layer optimization, suggesting that improved robustness can be achieved through straightforward data augmentation. Nevertheless, our method demonstrates strong performance even with the default configuration (layer=12) as shown in Table I and Table II, consistently outperforming baseline approaches across diverse datasets.

D. Adversarial Regularization Parameter

We analyze the impact of adversarial regularization weight (α) on both standard (clean) and adversarial (robust) accuracy. Fig. 4 presents results across α values from 0.01 to 100, with panel (a) showing both metrics on a single y-axis and panel (b) using dual y-axes to highlight relative trends.

The results show that clean accuracy remains stable (98.5%-99.1%) across all α values, while robust accuracy increases substantially from 79.9% at α=0.01 to 93.9% at α=2.0, with diminishing gains thereafter. For $\alpha \geq 5.0$, robust accuracy

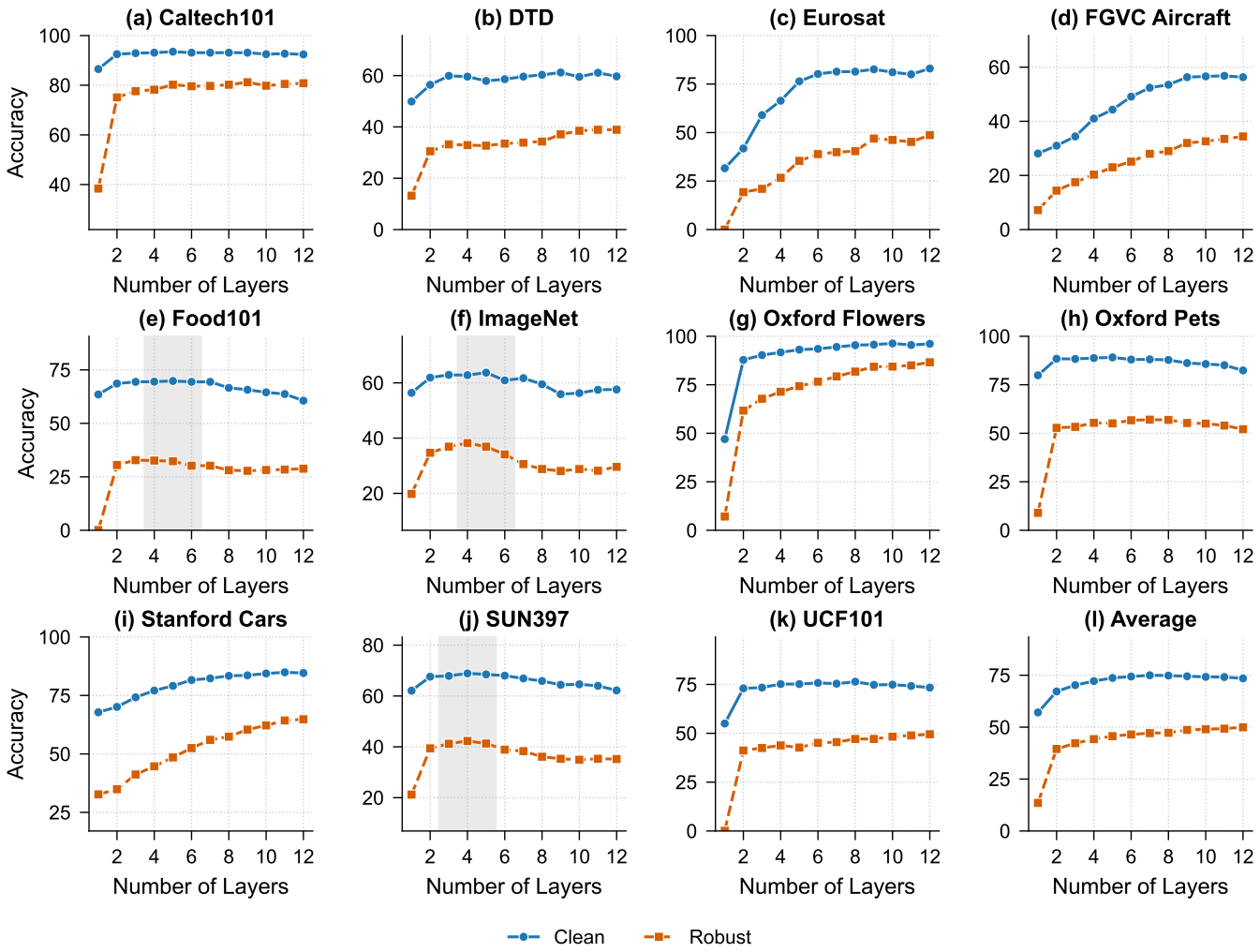


Fig. 3: Clean and robust accuracy across datasets when varying the number of prompt layers (1-12). Complex datasets (ImageNet, Food101, SUN397) show optimal performance at intermediate depth, while other datasets benefit from deeper architectures.

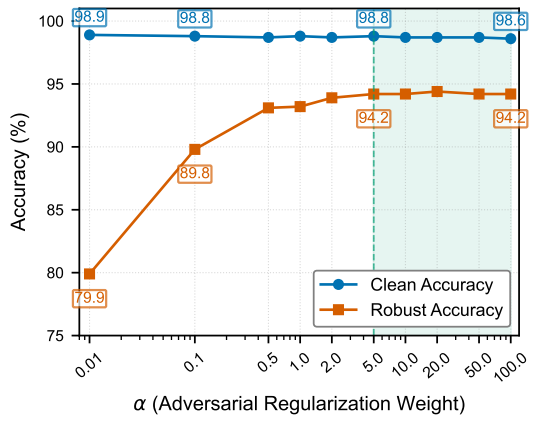
consistently exceeds 94%. Rather than specifically optimizing for the peak performance at α=20.0 (94.4% robust accuracy), we adopt α=5.0 as our representative configuration, which achieves 98.8% clean accuracy and 94.2% robust accuracy. These findings demonstrate that our approach effectively balances clean accuracy and adversarial robustness across a broad range of α values, exhibiting strong performance without requiring fine-tuned parameter selection.

E. TokenRefiner Architecture

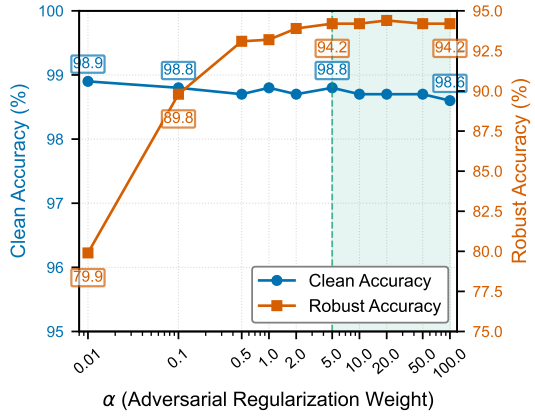
We analyze the impact of TokenRefiner network depth on model performance using Oxford Flowers (simpler task) and ImageNet (more challenging task). Fig. 5 presents clean and robust accuracy across different TokenRefiner layer configurations. Results demonstrate that TokenRefiner is essential for model convergence—its absence (layer = 0) prevents the model from converging under 5-step PGD training, resulting in poor performance (4.6% clean accuracy on ImageNet, 25.8% on Oxford Flowers). Performance improves substantially as layers increase from 0 to 2, with ImageNet clean accuracy rising from 4.6% to 54.4% and robust accuracy from 1.9% to 28.0%. Similarly, Oxford Flowers shows improvements from 25.8% to 96.1% (clean) and from 15.1% to 86.6% (robust). Performance stabilizes beyond 2 layers, leading us to select a 2-layer architecture for our final implementation. Notably, unlike previous approaches such as FAP [5] and APD [6], which lack specialized prompt structure designs and are therefore limited to weaker adversarial search strategies, our TokenRefiner enables convergence under stronger adversarial perturbations. The consistent pattern across datasets of varying complexity suggests our approach generalizes well without requiring task-specific architectural adjustments.

F. Perturbation Magnitude

We analyze the impact of perturbation budgets during both training and testing phases. Table III shows results on the Flowers dataset with varying ϵ values. The results reveal a clear trade-off between clean accuracy and adversarial robustness. As training perturbation budget increases from $\epsilon = 1$ to $\epsilon = 4$, clean accuracy decreases from 96.1% to 87.5%,

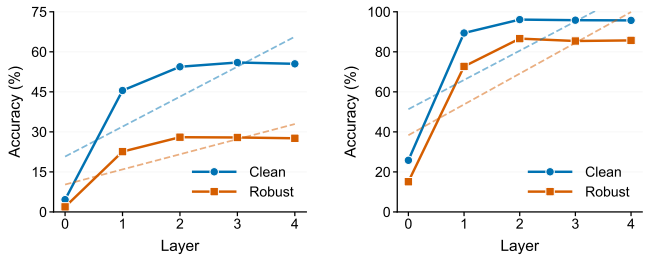


(a) Effect of Adversarial Regularization Weight (α)



(b) Trade-off between Clean and Robust Accuracy across α Values

Fig. 4: Effect of adversarial regularization weight (α) on clean and robust accuracy. Clean accuracy remains stable (98.5%-99.1%) while robust accuracy improves significantly with increasing α, stabilizing above 94% for $\alpha \geq 5.0$.



(a) ImageNet (b) Flowers

Fig. 5: TokenRefiner performance across different network depths. Performance improves dramatically from 0 to 2 layers before stabilizing, with consistent patterns across datasets of varying complexity.

while robustness against stronger attacks improves substantially. Models trained with larger ϵ values perform significantly better when tested under matching or higher perturbation budgets—a model trained with $\epsilon = 4$ achieves 55.9% accuracy when tested at the same level, versus only 17.1% for a model trained with $\epsilon = 1$. Models experience rapid performance degradation

TABLE III: Accuracy (%) under ℓ_∞-bounded adversarial perturbations. Models are trained and evaluated with varying perturbation magnitudes (measured in $\epsilon/255$). Bold values indicate the highest robustness at each test-time perturbation level.

Training ϵ	Clean Accuracy	Robust Accuracy @ Test-time ϵ			
		1	2	4	8
1	**96.1**	**86.5**	67.0	17.1	0.1
2	94.2	87.2	**75.3**	38.1	1.3
4	87.5	79.2	75.7	**55.9**	**11.8**

when tested with perturbations larger than their training budget. This observation suggests that, similar to traditional adversarial training, adversarial prompt tuning exhibits a clear trade-off between clean accuracy and adversarial robustness.

G. Few-shot Learning

We examine the impact of shot count on model performance. Fig. 6 illustrates the relationship between number of shots (1-16) and model accuracy across diverse datasets. Clean accuracy shows consistent improvement, with average accuracy increasing from 40.6% (single shot) to 73.2% (16 shots)—a 1.8× improvement. More remarkably, robust accuracy exhibits steeper relative gains, rising from 14.1% to 49.8%—a dramatic 3.5× improvement.

This differential impact reveals a key insight: while few-shot learning is known to enhance clean accuracy, its benefits for robust performance are substantially more pronounced. The results indicate that robustness requires a richer representation of class concepts that becomes increasingly available with additional examples. These findings align with our observations in Section V-C, confirming that increasing shot count offers tremendous potential for performance enhancement in our method, providing a straightforward path to significantly improved robustness without architectural modifications.

H. Context Vector Count

We examine how the number of learnable context vectors ($\mathcal{V}_t$) affects model performance. Fig. 7 shows results for models with 1-16 context vectors on ImageNet and Oxford Flowers datasets. For ImageNet, clean accuracy remains relatively stable (55.9%-58.1%) across different vector counts, while robust accuracy shows modest improvements with increased vectors, peaking at 29.6% with 14-16 vectors. On Oxford Flowers, both metrics maintain high performance across all configurations, with clean accuracy ranging from 95.5% to 96.5% and robust accuracy between 85.2% and 86.7%. These findings suggest that the number of context vectors $\mathcal{V}_t$ has a relatively minor impact on overall performance compared to the influence of layer count in Multi-Layer Prompts observed in Section V-C. This indicates that input-level representational capacity is not the primary performance bottleneck, particularly for less complex datasets where even few context vectors prove sufficient.

VI. Conclusion and Discussion

This paper presents the Neural Augmentor framework for Multi-modal Adversarial Prompt Tuning (NAP-Tuning), a

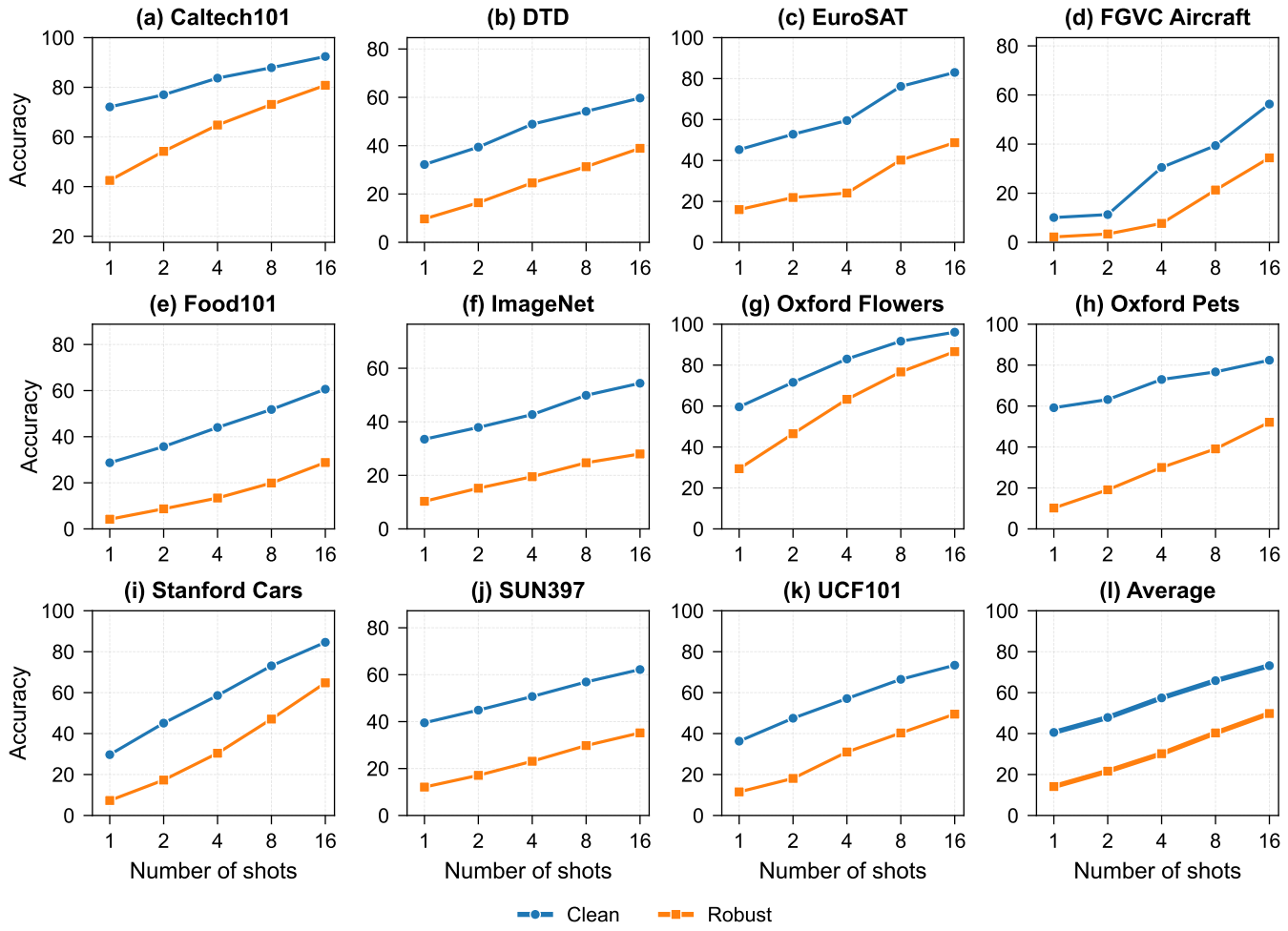


Fig. 6: Impact of shot count on clean and robust accuracy across eleven datasets. Each subplot (a-k) shows performance on individual datasets, while subplot (l) presents the average across all datasets.

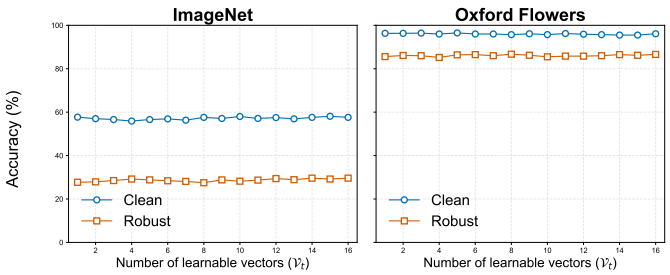


Fig. 7: Effect of context vector count (1-16) on ImageNet and Oxford Flowers. Performance varies minimally across different vector counts, indicating context vectors are not a primary bottleneck for model capability.

novel architecture-focused approach to enhancing adversarial robustness in Vision-Language Models. Our work addresses significant limitations in existing adversarial prompt tuning methods by fundamentally reconceptualizing prompt architecture through feature purification mechanisms.

A. *Conclusion*

We have extended adversarial prompt tuning from text-only to multi-modal contexts and from single-layer to multi-layer architectures while introducing token refiners that enable modality-specific and layer-specific feature correction. This architectural innovation represents a paradigm shift from previous loss-focused approaches, emphasizing the critical role of feature purification in addressing adversarial distortions. Through extensive experimentation, we have demonstrated that NAP-Tuning substantially outperforms existing methods, achieving state-of-the-art robustness against strong attacks while maintaining competitive clean accuracy.

B. *Broader Implications*

Our findings have several significant implications: 1) highlight the insufficiency of merely transplanting prompt tuning techniques from generalization domains to adversarial defense without architectural reconsideration; 2) underscore the importance of addressing adversarial perturbations at the feature level rather than solely through loss modification.

C. Discussion

Future work could explore adaptive token refiners that dynamically adjust their behavior based on detected perturbation characteristics, integration of uncertainty quantification mechanisms to improve refinement reliability, and extension to other multi-modal tasks beyond classification. Additionally, investigating the theoretical connections between feature purification and information bottleneck principles could yield deeper insights into adversarial robustness in representation learning.

In conclusion, our work establishes that addressing adversarial robustness in VLMs requires fundamental architectural innovations rather than merely adapting existing prompt tuning techniques. The Neural Augmentor framework provides a principled approach to this challenge, offering both improved performance and conceptual insights that advance our understanding of robust multi-modal learning.

REFERENCES

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[2] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig, "Scaling up visual and vision-language representation learning with noisy text supervision," in *International conference on machine learning*. PMLR, 2021, pp. 4904–4916.

[3] J. Zhang, X. Ma, X. Wang, L. Qiu, J. Wang, Y.-G. Jiang, and J. Sang, "Adversarial prompt tuning for vision-language models," in *European Conference on Computer Vision*. Springer, 2024, pp. 56–72.

[4] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2018.

[5] Y. Zhou, X. Xia, Z. Lin, B. Han, and T. Liu, "Few-shot adversarial prompt learning on vision-language models," *Advances in Neural Information Processing Systems*, vol. 37, pp. 3122–3156, 2024.

[6] L. Luo, X. Wang, B. Zi, S. Zhao, and X. Ma, "Adversarial prompt distillation for vision-language models," *arXiv preprint arXiv:2411.15244*, 2024.

[7] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Learning to prompt for vision-language models," *International Journal of Computer Vision*, vol. 130, no. 9, pp. 2337–2348, 2022.

[8] M. U. Khattak, H. Rasheed, M. Maaz, S. Khan, and F. S. Khan, "Maple: Multi-modal prompt learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19 113–19 122.

[9] H. Liu, C. Li, Y. Li, and Y. J. Lee, "Improved baselines with visual instruction tuning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 26 296–26 306.

[10] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei *et al.*, "Qwen2. 5 technical report," *arXiv preprint arXiv:2412.15115*, 2024.

[11] X. Liu, K. Ji, Y. Fu, W. L. Tam, Z. Du, Z. Yang, and J. Tang, "P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks," *arXiv preprint arXiv:2110.07602*, 2021.

[12] X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," *arXiv preprint arXiv:2101.00190*, 2021.

[13] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Conditional prompt learning for vision-language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 816–16 825.

[14] Z. Li, X. Li, X. Fu, X. Zhang, W. Wang, S. Chen, and J. Yang, "Promptkd: Unsupervised prompt distillation for vision-language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 26 617–26 626.

[15] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan, "Theoretically principled trade-off between robustness and accuracy," in *International conference on machine learning*. PMLR, 2019, pp. 7472–7482.

[16] Y. Wang, D. Zou, J. Yi, J. Bailey, X. Ma, and Q. Gu, "Improving adversarial robustness requires revisiting misclassified examples," in *International conference on learning representations*, 2019.

[17] C. Mao, S. Geng, J. Yang, X. Wang, and C. Vondrick, "Understanding zero-shot adversarial robustness for large-scale models," in *The Eleventh International Conference on Learning Representations*, 2023.

[18] L. Sheng, J. Liang, Z. Wang, and R. He, "R-tpt: Improving adversarial robustness of vision-language models through test-time prompt tuning," *arXiv preprint arXiv:2504.11195*, 2025.

[19] X. Wang, K. Chen, J. Zhang, J. Chen, and X. Ma, "Tapt: Test-time adversarial prompt tuning for robust inference in vision-language models," *arXiv preprint arXiv:2411.13136*, 2024.

[20] S. Xing, Z. Zhao, and N. Sebe, "Clip is strong enough to fight back: Test-time counterattacks towards zero-shot adversarial robustness of clip," *arXiv preprint arXiv:2503.03613*, 2025.

[21] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[22] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer Vision and Pattern Recognition Workshop*, 2004.

[23] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 3606–3613.

[24] P. Helber, B. Bischke, A. Dengel, and D. 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*, vol. 12, no. 7, pp. 2217–2226, 2019.

[25] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," *arXiv preprint arXiv:1306.5151*, 2013.

[26] L. Bossard, M. Guillaumin, and L. V. Gool, "Food-101–mining discriminative components with random forests," in *European conference on computer vision*. Springer, 2014, pp. 446–461.

[27] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*. IEEE, 2008, pp. 722–729.

[28] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. Jawahar, "Cats and dogs," in *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012, pp. 3498–3505.

[29] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3d object representations for fine-grained categorization," in *Proceedings of the IEEE international conference on computer vision workshops*, 2013, pp. 554–561.

[30] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2010, pp. 3485–3492.

[31] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[32] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9185–9193.

[33] X. Wang and K. He, "Enhancing the transferability of adversarial attacks through variance tuning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1924–1933.

[34] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. L. Yuille, "Improving transferability of adversarial examples with input diversity," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2730–2739.

[35] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in *International conference on machine learning*. PMLR, 2020, pp. 2206–2216.

[36] H. Chen, Y. Zhang, Y. Dong, X. Yang, H. Su, and J. Zhu, "Rethinking model ensemble in transfer-based adversarial attacks," in *The Twelfth International Conference on Learning Representations*, 2024.

[37] R. Wang, Y. Guo, and Y. Wang, "Ags: Affordable and generalizable substitute training for transferable adversarial attack," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 6, 2024, pp. 5553–5562.