MAGE: MARGIN-ALIGNED GRADIENT ENERGY FOR TARGET-SUPERVISED ROBUST ADAPTATION OF VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language models are vulnerable to adversarial perturbations, and fixed confidence-based energy objectives need not guide an attacked image toward its correct class. We introduce **MAGE** (**Margin-Aligned Gradient Energy**), a target-supervised, parameter-efficient robust adaptation method that learns a scalar energy head on frozen vision-language features. For each target dataset, MAGE independently trains the head using its training split, class names, and labels, shaping the negative input-energy gradient to restore the ground-truth margin while preserving clean predictions. At deployment, the adapted head performs a few bounded test-time image updates without test-label access or persistent parameter updates. Under adaptive PGD-20 that differentiates through the complete deployment pipeline, MAGE improves over fixed ET3 in all evaluated backbone-dataset combinations and achieves 37.0% average robust accuracy in the nine-dataset comparison, while also improving average clean accuracy by 7.6 points. These results show that target-supervised energy adaptation is most useful when its test-time gradient is aligned with semantic recovery rather than confidence alone.

1 INTRODUCTION

Pretrained vision-language models map images and text into a shared representation space, enabling open-vocabulary recognition, retrieval, and multimodal reasoning without task-specific classifier training. CLIP (Radford et al., 2021) and its descendants are now widely used as generic visual encoders. Their semantic transferability, however, does not imply robustness to input perturbations. A small L_∞ perturbation can substantially alter zero-shot predictions (Goodfellow et al., 2015), and this visual vulnerability can propagate to downstream image captioning, visual question answering, and embodied decision-making systems.

Recent contrastive, self-supervised, instruction-tuned, and grounded VLMs broaden transferable visual representations and multimodal capabilities (Li et al., 2024a; Zhai et al., 2023; Oquab et al., 2023; Li et al., 2023; Dai et al., 2023; Liu et al., 2023; Bai et al., 2023; Zhu et al., 2023; Ye et al., 2023; Liu et al., 2024; Girdhar et al., 2023; Peng et al., 2024; Kirillov et al., 2023).

Adversarial training and robust fine-tuning remain the most direct approaches to this problem (Mao et al., 2023; Schlarmann et al., 2024; Madry et al., 2018). Although effective, they require access to training data, repeated adversarial example generation, and optimization of a large visual backbone. They are also costly to revisit when the threat model, data distribution, or downstream vocabulary changes. We instead consider target-supervised parameter-efficient robust adaptation: retain the frozen backbone and adapt only a small module for each target dataset. Test-time input transformation is then the deployment mechanism through which that adapted module acts on the current image. This separation introduces a central design question: what adaptation objective produces a test-time correction that is semantically meaningful rather than merely confidence-increasing?

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.

ET3 (Mirza et al., 2026) provides a particularly simple answer. It interprets the negative LogSumExp of classification logits as an energy and updates the input by descending this energy within a constrained neighborhood. ET3 requires no auxiliary generative model, no backbone fine-tuning, and only a few image-space optimization steps. Under local linearity and ground-truth gradient dominance, its update can improve the relative score of the correct class. Yet these geometric conditions are not directly enforced by the fixed energy. If an adversarial input is already assigned high confidence to an incorrect class, its negative LogSumExp energy can still be low, and further minimization may reinforce the wrong prediction.

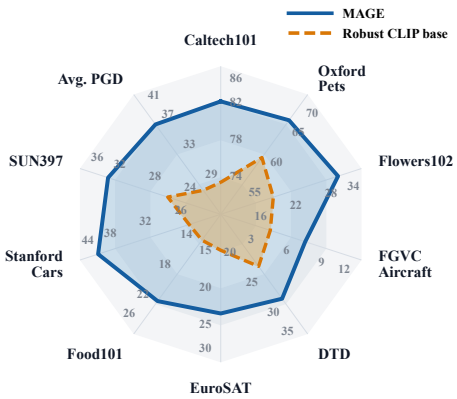

Figure 1: Adaptive PGD accuracy of MAGE and the Robust CLIP base across nine datasets.

As shown in Figure 1, under adaptive PGD against each method's complete inference pipeline, MAGE improves robust accuracy over the Robust CLIP base in all nine plotted target-dataset settings, raising the average PGD-20 accuracy from 27.5% to 37.0%; complete values are reported in Table 3.

This observation motivates a different view of test-time energy. A useful defense does not merely require a scalar that separates clean and adversarial samples. It requires an energy whose **negative input gradient is restorative**. For an adversarial example x_{adv} with label y, descending the energy should increase the margin between the ground-truth logit and its strongest competitor. We therefore ask:

> Can a lightweight energy function be learned on top of a frozen VLM such that its test-time gradient reliably points toward semantic recovery while preserving clean predictions?

We answer this question with MAGE. MAGE attaches a small multilayer perceptron to frozen image features and trains it with three complementary signals. An energy-ranking loss assigns higher energy to adversarial than clean inputs. A differentiable restoration loss unrolls one energy-descent step and optimizes the classification loss and ground-truth margin after that step. A clean-stability loss constrains the same update to preserve clean predictions and visual features. Once trained, the energy head and frozen backbone define a constrained test-time image transformation with no label access and no persistent parameter update.

Our contributions are threefold:

1. We identify that fixed confidence energy does not guarantee a restorative descent direction, and formulate margin-aligned gradient energy to address this limitation.

2. MAGE trains only a small scalar head on frozen VLM features using energy ranking, differentiable recovery, margin improvement, and clean stability; deployment uses bounded label-free image updates.

3. Component analyses and multi-backbone experiments show consistent gains over ET3 under adaptive PGD, while comparisons retain the differing adaptation budgets of competing methods. Code is available at `https://anonymous.4open.science/r/MAGE-CEE5`.

2 Related Work

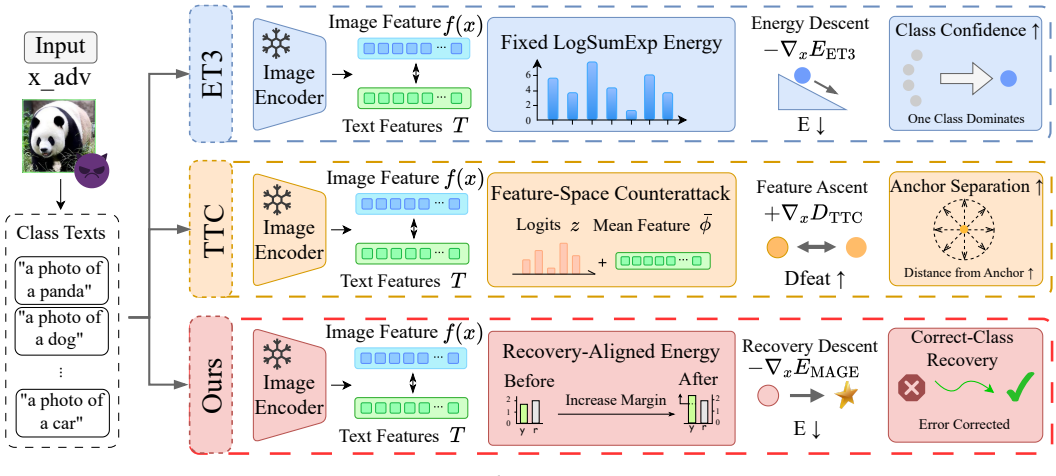

Figure 2: Test-time input-transformation framework of TTC, ET3, and MAGE. All three share a frozen CLIP encoder and bounded projected updates but differ in objective—feature counterattack, fixed LogSumExp energy, and learned recovery energy—as the rightmost panels illustrate.

Adversarial training remains the dominant robustness paradigm, but it repeatedly generates attacks and updates many backbone parameters (Madry et al., 2018). TeCoA and FARE preserve vision-language alignment through text-guided contrastive supervision and unsupervised adversarial fine-tuning, respectively (Mao et al., 2023; Schlarmann et al., 2024); pretrained-model guidance further limits semantic drift (Wang et al., 2024). Standard attack and robustness analyses provide complementary foundations (Croce & Hein, 2020; Croce et al., 2022; 2021; Athalye et al., 2018; Cohen et al., 2019). These methods supply MAGE's robust backbones, while their retraining cost makes lightweight adaptation a useful complement.

Parameter-efficient adaptation changes a small subspace while keeping the encoders frozen. CoOp/CoCoOp learn fixed or input-conditioned context, while Visual Prompt Tuning and MaPLe adapt visual or multimodal prompts (Zhou et al., 2022a;b; Jia et al., 2022; Khattak et al., 2023). One Prompt Word, AdvPT (adversarial prompt tuning), and Few-Shot Adversarial Prompt Learning extend this idea to adversarial settings (Li et al., 2024b; Zhang et al., 2024; Zhou et al., 2024), but prompt updates affect both attack gradients and decisions. MAGE instead adapts a scalar energy head and acts through image transformation.

Test-time adaptation moves optimization to deployment: TPT and its variants update prompts, while MTA-style methods aggregate augmented views (Shu et al., 2022; Yoon et al., 2024; Sheng et al., 2025; Zanella & Ben Ayed, 2024). Related test-time training and entropy-minimization methods provide complementary adaptation paradigms (Sun et al., 2020; Wang et al., 2021). A recent 2025 zero-shot, training-free approach instead avoids learned prompts or auxiliary modules and performs direct inference-time correction (Tong et al., 2025); unlike MAGE, it does not use target-dataset labels to shape a recovery objective. These methods keep the backbone frozen but allocate different online computation budgets, so Table 3 reports both clean and PGD accuracy.

In parallel with parameter and prompt adaptation, test-time input defenses act directly on the current image. Anti-Adversaries and transformation ensembling are representative approaches that modify the evidence received by a fixed classifier rather than redefine the downstream classifier (Alfarra et al., 2022; Perez et al., 2021). Other methods counter-optimize the attack process or use interpretability and spectral signals to construct corrective directions (Wu et al., 2021; Kulkarni & Weng, 2024; Singh et al., 2024). Diffusion and score-based purification instead use a stronger generative prior to project inputs toward the natural-image distribution, but require an auxiliary model and long sampling chains (Nie et al., 2022; Yoon et al., 2021); adaptive analyses such as DiffBreak empha-

size evaluating the complete purification pipeline (Kassis et al., 2025). Our focus is the lightweight regime that uses neither a generative model nor a long sampling trajectory.

For VLMs, Test-Time Counterattacks (TTC) (Xing et al., 2025) showed that the input to a frozen CLIP model can be corrected at inference without a clean reference image: starting from the attacked image, the update maximizes the distance between its features and the attacked input's own feature anchor, leaving the "falsely stable" attacked representation behind. ET3 (Mirza et al., 2026) turns this idea into a unified energy objective: it builds a negative LogSumExp energy from classification logits and performs bounded projected descent around the observed input, without a clean reference image or additional training. However, fixed LogSumExp measures aggregate confidence, so an input dominated by an incorrect class can still have low energy; neither TTC nor ET3 directly enforces semantic restorability of the descent direction. Recent 2026 work continues this input-side line: multi-view guided adaptive counterattacks (MAC) strengthen the correction signal at test time (Kim & Um, 2026), and contrastive spectral rectification (CSR) corrects the input in the spectral domain (Nie et al., 2026). Both remain label-free test-time transformations that do not learn a target-supervised recovery objective, so the same limitation persists.

This gap is consistent with energy-based modeling, where a scalar defines input-space geometry through its gradients and level sets (Grathwohl et al., 2020; Liu et al., 2020; Du & Mordatch, 2019; LeCun et al., 2006); robust-classifier analyses further link confidence to the energy landscape (Mirza et al., 2024). A detection-oriented energy does not constrain how the classification margin changes after a gradient step. MAGE addresses this distinction while retaining the bounded image-update interface of the **TTC** $\rightarrow$ **ET3** progression, but learns a recovery-aligned local potential instead of a fixed one: because the head is trained with ground-truth margin supervision, the descent direction itself becomes restorative, and it adds only a small scalar head on the frozen backbone while remaining label-free at deployment. MAGE therefore complements prompt-side adaptation rather than replacing it.

3 Method

3.1 Problem Formulation and Adversarial Example Construction

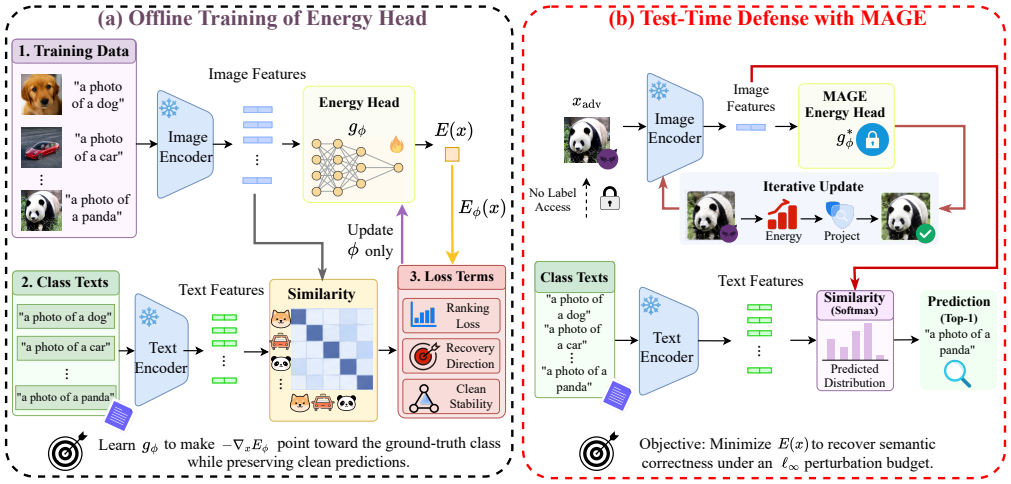

Figure 3: Target-supervised training and test-time transformation pipeline of MAGE. The left panel trains the energy head on clean-adversarial pairs via ranking, recovery, and clean stability; the right panel performs label-free energy descent at inference.

Let $h_\theta(\cdot)$ denote a frozen image encoder, and let $\{t_k\}_{k=1}^K$ be text prototypes obtained by encoding the names of K downstream classes. For an image x, zero-shot classification logits are

$$s_k(x) = \gamma \cdot \frac{h_\theta(x)}{\|h_\theta(x)\|_2}^\top \frac{t_k}{\|t_k\|_2}, \tag{1}$$

where γ is the CLIP logit scale. Following standard first-order threat models and attack practice (Madry et al., 2018; Croce & Hein, 2020), an attacker constructs $x_{adv} = x + \delta$ under $\|\delta\|_\infty \le \epsilon_a$. Given only the observed input, the defense applies a small transformation $\mathcal{D}(\cdot)$ with the goal of increasing the relative advantage of $s_y(\mathcal{D}(x_{adv}))$, without updating θ or accessing the test label.

The original logits remain the decision interface; the energy head only defines a local restoration field. Attacks use ϵ_a around the clean image, transformations use ϵ_d around the observed input, and labels are used only offline. The ground-truth margin $M_y(x) = s_y(x) - \max_{j \neq y} s_j(x)$ supervises the direction during training; deployment follows only the label-free energy gradient.

3.2 From Fixed ET3 Energy to Margin-Aligned Energy

ET3 uses the fixed temperature-scaled negative LogSumExp energy $E_0(x) = -\tau \log \sum_k \exp(s_k(x)/\tau)$. Its negative gradient is confidence-weighted by the current class logits and can therefore align with a dominant incorrect class rather than improve the ground-truth margin. MAGE learns this direction explicitly.

We introduce a scalar energy head g_ϕ on frozen normalized image features:

$$u(x) = \frac{h_\theta(x)}{\|h_\theta(x)\|_2}, \qquad E_\phi(x) = g_\phi(u(x)). \tag{2}$$

The head is a three-layer MLP ($d \to 128 \to 32 \to 1$) with SiLU and LayerNorm; it has 70,081 / 102,849 parameters for ViT-B/32 / ViT-L/14. It is not a classifier: predictions remain frozen-CLIP logits, while its scalar output affects only the input gradient. At test time, E_{def} is E_0, E_ϕ, or $E_0 + \lambda_a E_\phi$ for ET3, MAGE, or the diagnostic combination.

3.3 Restoration-Oriented Training Objective

MAGE is trained with four losses that play distinct roles: energy separation identifies suspicious inputs, restoration cross-entropy supplies class-level recovery supervision, a margin objective directly rewards local progress, and clean stability limits destructive motion around correctly recognized inputs.

Energy ranking. We first require the adversarial example to have higher energy than its paired clean image:

$$\mathcal{L}_{rank} = \max\left(0, m_E + E_\phi(x) - E_\phi(x_{adv})\right). \tag{3}$$

Ranking separates clean and adversarial energies but does not determine a restorative gradient, so training also unrolls one restoration step.

Differentiable one-step restoration. Starting from x_{adv}, we take one energy-descent step with detached mean-absolute gradient scaling:

$$\tilde{x}_{adv} = \Pi_{\mathcal{B}_{\epsilon_d}^{(\infty)}(x_{adv})} \left[x_{adv} - \alpha \frac{\nabla_x E_\phi(x_{adv})}{\mathrm{stopgrad}\left(\frac{1}{n}\sum_j |[\nabla_x E_\phi(x_{adv})]_j|\right) + \xi} \right]. \tag{4}$$

This Mean-Absolute scaling is used by the differentiable one-step training surrogate and by MAGE deployment. We supervise both absolute recovery and local boundary progress:

$$\begin{aligned} \mathcal{L}_{restore} &= \mathrm{CE}(s(\tilde{x}_{adv}), y), \\ \mathcal{L}_{margin} &= \max\left(0, m_R - [M_y(\tilde{x}_{adv}) - M_y(x_{adv})]\right). \end{aligned} \tag{5}$$

Cross-entropy encourages correct recovery, whereas the margin term rewards progress from the attacked input near a boundary. Their gradients pass through $\nabla_x E_\phi$ to train a restorative field.

Clean-input stability. Applying an unconditional image update can alter inputs that are already correctly classified. We therefore apply the same one-step transformation to a clean image, obtaining $\tilde{x}$, and constrain both its predictive distribution and image feature:

$$\begin{aligned} \mathcal{L}_{clean} &= D_{KL}(p(x)\|p(\tilde{x})) + \lambda_f[1-\cos(h_\theta(x), h_\theta(\tilde{x}))], \\ \mathcal{L} &= \lambda_1\mathcal{L}_{rank} + \lambda_2\mathcal{L}_{restore} + \lambda_3\mathcal{L}_{margin} + \lambda_4\mathcal{L}_{clean}. \end{aligned} \tag{6}$$

Thus, ranking separates states, recovery and margin shape the direction, and stability limits clean-image drift; all CLIP components remain frozen.

3.4 Test-Time Transformation and Deployment

For a test input z, MAGE applies T projected energy-descent steps with Mean-Absolute image-gradient normalization:

$$z^{(t)} = \Pi_{\mathcal{B}_{\epsilon_d}^{(\infty)}(z^{(0)})}\left[z^{(t-1)} - \alpha \frac{\nabla_z E_{def}(z^{(t-1)})}{\frac{1}{n}\sum_j |[\nabla_z E_{def}(z^{(t-1)})]_j| + \xi}\right]. \tag{7}$$

The final prediction is $\arg\max_k s_k(z^{(T)})$. Unless stated otherwise, $T = 5$, $\epsilon_d = 10/255$, $\alpha = 2.5/255$, and $\tau = 1.0$; the update requires no label or persistent parameter change.

For defense-aware evaluation, let $\mathcal{D}_m$ denote the complete test-time mapping of method m; for MAGE, $\mathcal{D}_{\text{MAGE}}$ contains all $T = 5$ energy-update steps above. The adaptive white-box attacker directly maximizes the post-defense classification loss,

$$\delta_m^* = \arg\max_{\|\delta\|_\infty \le \epsilon_a} \text{CE}(s(\mathcal{D}_m(x+\delta)), y)\,. \tag{8}$$

Attack gradients pass through all deployed input updates, while $\mathcal{D}_m$ remains label-free.

Algorithm 1: MAGE offline training and test-time defense

Input : Labeled target set $\mathcal{D}$, frozen VLM, test image $z^{(0)}$
Output: Prediction $\hat{y}$

Initialize scalar head g_ϕ;
foreach $(x, y) \in \mathcal{D}$ **do**
  $x_{adv} \leftarrow \text{PGD}(x, y)$; compute $\mathcal{L}_{rank}$ and one-step $\tilde{x}_{adv}$;
  $\mathcal{L} \leftarrow \lambda_1\mathcal{L}_{rank} + \lambda_2\mathcal{L}_{restore} + \lambda_3\mathcal{L}_{margin} + \lambda_4\mathcal{L}_{clean}$; update ϕ;
for $t \leftarrow 1$ **to** T **do**
  $z^{(t)} \leftarrow \Pi_{\mathcal{B}_{\epsilon_d}^{(\infty)}(z^{(0)})}\big(z^{(t-1)} - \alpha\ \text{MeanAbsNorm}(\nabla_z E_{def}(z^{(t-1)}))\big)$;
return $\hat{y} \leftarrow \arg\max_k s_k(z^{(T)})$;

Projection bounds the displacement and clipping keeps pixels valid; adaptive evaluation differentiates through all T deployed steps.

4 Experiments

4.1 Experimental Setup

We use OpenAI CLIP ViT-B/32 for CIFAR-10 mechanism analysis and FARE4/TeCoA4 at ViT-B/32 and ViT-L/14 for multi-backbone evaluation. Caltech101, CIFAR-100, DTD, and ImageNet form the matched backbone study, while Table 3 extends the comparison to nine target datasets by adding Oxford Pets, Flowers102, FGVC Aircraft, EuroSAT, Food101, Stanford Cars, and SUN397. Each backbone–dataset–seed is a separate target adaptation task in which MAGE trains one scalar energy head from scratch on the corresponding training split and evaluates it only on the matched test

split; the frozen image/text encoders, class prototypes, and logit scale retain the original zero-shot decision rule, and labels are used only offline.

Training uses five-step PGD pairs and differentiable one-step recovery, while deployment applies five projected image updates with Mean-Absolute normalization, $\epsilon_d = 10/255$, $\alpha = 2.5/255$, and $\tau = 1.0$. Evaluation uses randomly initialized adaptive PGD-CE ($\epsilon_a = 2/255$, 20 steps, step size $0.5/255$) through each method's complete deployment pipeline, following recommendations for adaptive test-time-defense evaluation (Croce et al., 2022; 2021; Athalye et al., 2018); full split, preprocessing, and implementation details are given in Appendix A.

4.2 Main Results: Energy Ablations and Multi-Dataset, Multi-Backbone Evaluation

Restoration training is necessary. Table 1 dissects the MAGE training objective on CIFAR-10 with standard OpenAI CLIP ViT-B/32, where PGD accuracy is zero without defense; the fixed ET3 energy recovers 12.1% robust accuracy while retaining 74.8% clean accuracy after defense, a 10.4-point clean drop. A learned energy trained only for clean-adversarial ranking reaches 37.3% robust accuracy, confirming that frozen visual features contain information useful for separating attacked inputs, but it incurs a 33.5-point clean drop, demonstrating that detection-oriented energy does not yield a safe gradient field.

Cross-entropy alone reaches 54.1% robust accuracy, 16.8 points above ranking-only, whereas the margin term alone reaches 40.8%, a 3.5-point gain over ranking-only that establishes an independent robust contribution of the margin objective; combining both reaches 55.1%, 1.0 point above CE-only, while clean-stability regularization yields 55.5% and reduces the clean drop to 21.6 points, which does not eliminate degradation on standard CLIP but materially constrains the destructive component of the learned update.

CE and margin supervision are complementary rather than redundant: CE anchors the transformed sample to its labeled class, whereas the margin term judges whether the update increases separation from the strongest competitor; clean stability then recovers 11.9 points of clean accuracy relative to ranking-only, so the full objective trades a modest amount of peak clean accuracy for a substantially safer update field whenever the same transformation is applied to adversarial and already-correct inputs.

Table 1: Component analysis of MAGE objectives on CIFAR-10 under adaptive PGD through the complete defense. The restoration objective is decomposed into cross-entropy (CE) and ground-truth margin terms. Gains are computed from unrounded means.

Variant	Test-time energy	Training objective	Clean	Clean after defense	Clean drop	PGD	Gain over no defense
No defense	none	none	85.2	85.2	0.0	0.0	-
Original ET3	E_0	none	85.2	74.8	10.4	12.1	+12.1
Ranking only	E_ϕ	ranking	85.2	51.7	33.5	37.3	+37.3
Ranking + $\mathcal{L}_{\text{restore}}$	E_ϕ	ranking + CE	85.2	55.3	29.9	54.1	+54.1
Ranking + $\mathcal{L}_{\text{margin}}$	E_ϕ	ranking + margin	85.2	52.5	32.7	40.8	+40.8
Ranking + $\mathcal{L}_{\text{restore}} + \mathcal{L}_{\text{margin}}$	E_ϕ	ranking + CE + margin	85.2	57.7	27.5	55.1	+55.1
MAGE Full	E_ϕ	**ranking + CE + margin + stability**	85.2	**63.6**	**21.6**	**55.5**	**+55.5**
ET3 + MAGE	$E_0 + \lambda E_\phi$	stable restoration	85.2	61.4	23.7	48.9	+48.9

The combined energy reaches 48.9% robust accuracy, below the learned-only result of 55.5%. It retains 61.4% clean accuracy after defense, corresponding to a 23.7-point clean drop, and is therefore less stable on clean inputs than both Original ET3 and learned-only MAGE. E_0 and E_ϕ need not share a numerical scale, gradient norm, curvature, or local direction. Their uncalibrated sum can therefore dilute the restorative gradient learned by MAGE, so compatible energy values need not imply compatible recovery directions. We consequently use learned-only MAGE as the primary method and treat direct energy addition as a diagnostic variant.

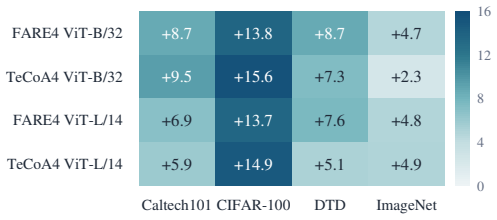

Figure 4: MAGE gains over the robust base across matched backbone–dataset settings.

Table 2 groups clean and robust results by model. Figure 4 summarizes the corresponding MAGE gains over the robust base. For each backbone, we first report Base and MAGE clean accuracy, followed by PGD accuracy for Base, Original ET3, MAGE, and ET3+MAGE in the same dataset order. `Avg.` is the arithmetic mean over the four datasets, and `Improv.` is the absolute gain over the Base row under the same clean or robust condition.

Table 2: Clean and adaptive robust accuracy of the base model, Original ET3, and MAGE across multiple datasets and FARE4/TeCoA backbones at ViT-B/32 and ViT-L/14 scales. PGD differentiates through each method's complete inference pipeline. Each MAGE result uses an energy head independently trained on the corresponding target dataset. `Improv.` denotes the absolute gain over the corresponding base result; averages and gains are computed from unrounded values.

Model	Method	Caltech101	CIFAR-100	DTD	ImageNet	Avg.	Improv.
FARE4 ViT-B/32	Base (Clean)	86.2	49.7	36.9	54.4	56.8	-
	+ MAGE (Clean)	89.0	57.2	42.5	53.3	60.5	+3.7
	Base (Robust)	74.6	29.9	24.7	33.4	40.7	-
	+ ET3 (Robust)	76.7	32.0	24.6	35.6	42.2	+1.6
	+ MAGE (Robust)	**83.3**	**43.7**	**33.4**	**38.1**	**49.6**	**+9.0**
	+ ET3 + MAGE (Robust)	77.6	35.1	26.7	36.0	43.9	+3.2
TeCoA4 ViT-B/32	Base (Clean)	83.6	46.0	29.9	58.9	54.6	-
	+ MAGE (Clean)	87.7	59.7	36.4	59.2	60.8	+6.2
	Base (Robust)	72.7	31.7	21.8	45.5	42.9	-
	+ ET3 (Robust)	74.6	31.8	22.0	45.4	43.5	+0.5
	+ MAGE (Robust)	**82.2**	**47.3**	**29.1**	**47.8**	**51.6**	**+8.7**
	+ ET3 + MAGE (Robust)	79.4	40.3	25.8	46.8	48.1	+5.2
FARE4 ViT-L/14	Base (Clean)	88.4	57.4	41.3	61.8	62.2	-
	+ MAGE (Clean)	90.1	64.3	46.8	62.1	65.8	+3.6
	Base (Robust)	78.6	42.2	30.7	48.6	50.0	-
	+ ET3 (Robust)	80.3	44.8	32.4	50.2	51.9	+1.9
	+ MAGE (Robust)	**85.5**	**55.9**	**38.3**	**53.4**	**58.3**	**+8.2**
	+ ET3 + MAGE (Robust)	83.1	49.8	35.2	52.1	55.0	+5.0
TeCoA4 ViT-L/14	Base (Clean)	83.4	51.9	34.2	61.3	57.7	-
	+ MAGE (Clean)	86.6	63.9	39.4	62.8	63.2	+5.5
	Base (Robust)	76.4	36.3	26.3	50.9	47.5	-
	+ ET3 (Robust)	77.8	39.2	27.9	52.3	49.3	+1.8
	+ MAGE (Robust)	**82.3**	**51.2**	**31.4**	**55.8**	**55.2**	**+7.7**
	+ ET3 + MAGE (Robust)	80.4	44.0	28.6	54.1	51.8	+4.3

MAGE outperforms Original ET3 in all 16 matched backbone–dataset settings, with average robust gains of 7.7–9.0 points over the robust base versus 0.5–1.9 for ET3. The combined energy remains below learned-only MAGE throughout, indicating that direct energy addition can dilute the learned restorative direction.

Clean and robust performance do not exhibit a simple trade-off: MAGE improves clean accuracy in 15 of 16 settings while providing consistent robust gains across backbones and datasets. These matched results establish target-supervised energy-head adaptation as a complement to robust backbones, rather than a replacement for robust training or evidence of a transferable cross-dataset head.

The gains persist across scales: average robust accuracy rises from 40.7% to 49.6% (FARE4 ViT-B/32) and 50.0% to 58.3% (ViT-L/14), while TeCoA4 rises from 42.9% to 51.6% and 47.5% to 55.2%, respectively. MAGE also improves defended clean accuracy in 15 of 16 settings (3.6–6.2 points on average), confirming that target-supervised energy adaptation complements robust pretraining.

4.3 Comparison with Lightweight Methods and Comprehensive Analysis

Table 3 compares target-supervised adaptation, image transformation, multi-view aggregation, and adversarial prompt learning across nine datasets. Each cell reports clean and PGD accuracy under the corresponding deployment pipeline. Since methods retain different offline supervision and online computation, this is not an identical-cost comparison.

Table 3: Comparison across lightweight robust adaptation and test-time transformation methods on nine datasets. Each entry reports Clean / PGD accuracy, where PGD is adaptive to the complete deployment pipeline of the corresponding method; Avg. PGD is the mean robust accuracy across datasets. Each MAGE result aggregates three heads independently trained on the corresponding target dataset; no head is reused across datasets. Methods retain their respective offline supervision and online-compute protocols, so this is not an identical-cost comparison. Clean and PGD are ranked independently in each dataset: **bold**, underlined, and *italicized* values denote the best, second-best, and third-best results, respectively. Avg. PGD is ranked by mean robust accuracy.

Method	Caltech101	Pets	Flowers	Aircraft	DTD	EuroSAT	Food101	Stanford Cars	SUN397	Avg. PGD
Robust CLIP base	83.6 / 72.7	*72.8* / 57.4	25.6 / 17.0	5.3 / 3.3	29.9 / 21.8	17.8 / 13.8	23.5 / 12.7	38.5 / 24.2	38.2 / 24.7	27.5
TTC	71.7 / 50.8	60.5 / 26.1	36.7 / 16.4	5.1 / 1.7	24.9 / 14.4	16.6 / 11.7	**29.9** / 13.0	20.4 / 16.6	35.4 / *26.2*	19.7
Original ET3	83.7 / 74.6	72.1 / *59.0*	23.2 / 17.5	3.8 / 2.3	27.8 / 22.0	15.8 / 12.8	20.8 / 14.2	33.2 / 23.7	33.1 / 25.4	27.9
MAGE	87.7 / **82.2**	77.4 / **65.7**	39.8 / 30.0	10.5 / 7.2	36.4 / **29.1**	*27.7* / 23.4	29.3 / 21.6	**52.7** / **40.9**	41.8 / **32.8**	**37.0**
MTA	84.2 / 75.1	72.5 / 56.6	25.5 / 17.1	5.0 / 3.4	29.6 / 21.2	16.4 / 13.3	23.7 / 12.8	*38.7* / 24.6	*38.5* / 24.9	27.7
R-TPT	**91.2** / 81.6	**84.0** / 62.9	63.5 / **36.4**	**18.4** / **11.1**	42.8 / 29.0	*32.1* / 14.7	24.6 / **23.5**	35.6 / 32.3	36.8 / 28.2	35.5
APT	*86.8* / *76.2*	70.3 / 42.0	**71.9** / *32.9*	*16.0* / 5.6	**44.2** / 25.9	**58.8** / *23.2*	*27.4* / *17.3*	29.9 / *26.4*	**45.9** / 11.1	*29.0*
FAP	86.4 / 69.1	41.1 / 33.2	*57.2* / 34.4	16.8 / *6.2*	*42.3* / *26.6*	45.7 / **32.4**	22.8 / 8.4	*42.5* / 13.7	32.8 / 25.7	27.7

MAGE attains the highest average PGD accuracy (37.0%) and improves average clean accuracy over the robust base by 7.6 points. It is strongest on Caltech101, Oxford Pets, DTD, Stanford Cars, and SUN397; R-TPT leads on Flowers, Aircraft, and Food101, while FAP leads on EuroSAT. The distribution of winners indicates that energy adaptation and prompt learning exploit different priors.

R-TPT obtains higher average clean accuracy through per-image prompt optimization over augmented views, which entails a larger online adaptation budget than MAGE's fixed five-step image update. Table 3 should therefore be read as a comparison across distinct adaptation protocols, whereas Table 2 supplies the matched Base/ET3/MAGE evidence. Together, the ablation and both comparisons show that margin-aligned energy adaptation yields robust gains across target domains while remaining complementary to other mechanisms.

5 Limitations and Future Directions

MAGE learns a local recovery field from global image features and fixed class prototypes, so it does not yet address language-conditioned tasks such as captioning or visual question answering. It also trains a separate head for each backbone–dataset–seed configuration; transfer to unseen datasets, vocabularies, or visual domains remains untested. Finally, inference uses a fixed update budget for every sample, without sample-adaptive stopping or fallback, and extending MAGE to joint image–text energy, transferable heads, and adaptive update policies is therefore an important direction.

6 Conclusion

We presented MAGE, a target-supervised parameter-efficient robust adaptation method for vision-language models, deployed through test-time input transformation. Rather than merely separating clean and adversarial inputs by energy value, MAGE uses differentiable restoration and ground-truth margin supervision to shape the semantic direction of the negative energy gradient, while clean-stability constraints limit destructive updates; CIFAR-10 component analysis shows that energy ranking alone is insufficient, with restoration and stability raising adaptive PGD accuracy to 55.5%. Under adaptive PGD, MAGE outperforms fixed ET3 across all 16 matched backbone–dataset settings and achieves the highest average robust accuracy of 37.0% over nine datasets and differing adaptation protocols, while improving average clean accuracy by 7.6 points over the robust backbone.

From TTC's feature counterattack and ET3's fixed discriminative energy to MAGE's learned recovery potential, the ceiling of test-time input transformation is set not by update count or computation but by whether the update direction is aligned with task semantics; MAGE accordingly turns energy from a static confidence score into a trainable, analyzable recovery interface, offering a new modeling approach for lightweight robust inference with a frozen backbone.

AI USE STATEMENT

In this work, we used generative AI tools only as limited assistance for debugging code, drafting selected portions of implementation code, improving the clarity and fluency of the manuscript, and optimizing formatting and presentation. Specifically, AI tools were used to help identify and fix programming errors, suggest partial code snippets that were subsequently reviewed and integrated by the authors, polish wording at the sentence and paragraph level, and refine LaTeX/Markdown-style formatting.

We did not use generative AI tools for research ideation, problem formulation, method design, theoretical derivation, experimental design, data generation or modification, result interpretation, drawing scientific conclusions, or selecting/citing references. All AI-assisted code was manually checked and tested by the authors, and all AI-assisted text and formatting changes were reviewed, revised, and approved by the authors. The authors take full responsibility for the final content of this paper, including all claims, results, code, figures, tables, references, and any artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies the adversarial robustness of vision-language models using only publicly available benchmark datasets under their original licenses; no human subjects, personal, or private data are involved, and no new dataset is released. Although adversarial methods are inherently dual-use, our contribution is defensive, and the threat models, perturbation budgets, and evaluation protocols follow standard practice in the literature. The authors declare no conflicts of interest.

REFERENCES

Motasem Alfarra, Juan C. Perez, Ali Thabet, Adel Bibi, Philip H. S. Torr, and Bernard Ghanem. Combating adversaries with anti-adversaries. In *AAAI Conference on Artificial Intelligence*, 2022.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International Conference on Machine Learning*, 2018.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.

Jeremy M. Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, 2019.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning*, 2020.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: A standardized adversarial robustness benchmark. In *NeurIPS Datasets and Benchmarks Track*, 2021.

Francesco Croce, Sven Gowal, Thomas Brunner, Evan Shelhamer, Matthias Hein, and A. Taylan Cemgil. Evaluating the adversarial robustness of adaptive test-time defenses. In *International Conference on Machine Learning*, 2022.

Yubo Cui, Xianchao Guan, Zijun Xiong, and Zheng Zhang. AGFT: Alignment-guided fine-tuning for zero-shot adversarial robustness of vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22836–22846, 2026.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems*, 2023.

Junhao Dong, Yifei Zhang, Hao Zhu, Yew-Soon Ong, and Piotr Koniusz. Hierarchically robust zero-shot vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 37642–37652, 2026.

Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. In *Advances in Neural Information Processing Systems*, 2019.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

Will Grathwohl, Kuan-Chieh Wang, Jorn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. In *International Conference on Learning Representations*, 2020.

Jia-Wei Hai, Yijun Wang, and Xiu-Shen Wei. Towards fine-grained robustness: Attention-guided test-time prompt tuning for vision-language models. In *International Conference on Machine Learning*, 2026.

Jaehyuk Jang, Minseok Seo, Seungju Cho, Kangwook Ko, and Changick Kim. T-VSS: Test-time visual subspace steering for adversarial robustness of vision-language models. *arXiv preprint arXiv:2606.23132*, 2026.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, 2022.

Andre Kassis, Urs Hengartner, and Yaoliang Yu. Diffbreak: Is diffusion-based purification robust? In *Advances in Neural Information Processing Systems*, 2025.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. MaPLe: Multi-modal prompt learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Sunoh Kim and Daeho Um. When CLIP sees more, it fights back harder: Multi-view guided adaptive counterattacks for test-time adversarial robustness. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15689–15699, 2026.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *IEEE/CVF International Conference on Computer Vision*, 2023.

Akshay Kulkarni and Tsui-Wei Weng. Interpretability-guided test-time adversarial defense. In *European Conference on Computer Vision*, 2024.

Yann LeCun, Sumit Chopra, Raia Hadsell, Marc'Aurelio Ranzato, and Fu-Jie Huang. A tutorial on energy-based learning. In *Predicting Structured Data*. MIT Press, 2006.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*, 2023.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024b.

Chaohu Liu, Tianyi Gui, Yu Liu, and Linli Xu. AdPO: Enhancing the adversarial robustness of large vision-language models with preference optimization. In *International Conference on Learning Representations*, 2026a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, 2023.

Jiaxiang Liu, Jiawei Du, Xiao Liu, Shangyang Li, Songchen Ma, Changshuo Wang, Prayag Tiwari, and Mingkun Xu. Self-calibrated consistency can fight back for adversarial robustness in vision-language models. In *International Conference on Machine Learning*, 2026b.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, 2024.

Weitang Liu, Xiaoyun Wang, John D. Owens, and Yixuan Li. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems*, 2020.

Xiao Liu, Jiaxiang Liu, Boci Peng, Boren Hu, Yusong Wang, Xiwen Chen, Prayag Tiwari, Liming Zhang, and Mingkun Xu. Closed-loop bidirectional prompting for adversarial robustness of vision language models. *arXiv preprint arXiv:2605.25922*, 2026c.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Chengzhi Mao, Scott Geng, Junfeng Yang, Xin Wang, and Carl Vondrick. Understanding zero-shot adversarial robustness for large-scale models. In *International Conference on Learning Representations*, 2023.

Mujtaba Hussain Mirza, Maria Rosaria Briglia, Senad Beadini, and Iacopo Masi. Shedding more light on robust classifiers under the lens of energy-based models. In *European Conference on Computer Vision*, 2024.

Mujtaba Hussain Mirza, Antonio D'Orazio, Odelia Melamed, and Iacopo Masi. A provable energy-guided test-time defense boosting adversarial robustness of large vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8598–8609, 2026.

Sen Nie, Jie Zhang, Zhuo Wang, Shiguang Shan, and Xilin Chen. Contrastive spectral rectification: Test-time defense towards zero-shot adversarial robustness of CLIP. In *International Conference on Machine Learning*, 2026.

Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. In *International Conference on Machine Learning*, 2022.

Maxime Oquab, Timothee Darcet, Theo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. In *International Conference on Learning Representations*, 2024.

Juan C. Perez, Motasem Alfarra, Guillaume Jeanneret, Laura Rueda, Ali Thabet, Bernard Ghanem, and Pablo Arbelaez. Enhancing adversarial robustness via test-time transformation ensembling. In *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763, 2021.

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 *International Conference on Machine Learning*, 2024.

Lijun Sheng, Jian Liang, Zilei Wang, and Ran He. R-TPT: Improving adversarial robustness of vision-language models through test-time prompt tuning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 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 *Advances in Neural Information Processing Systems*, 2022.

Anurag Singh, Mahalakshmi Sabanayagam, Krikamol Muandet, and Debarghya Ghoshdastidar. Robust feature inference: A test-time defense strategy using spectral projections. *Transactions on Machine Learning Research*, 2024.

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

Baoshun Tong, Hanjiang Lai, Yan Pan, and Jian Yin. On the zero-shot adversarial robustness of vision-language models: A truly zero-shot and training-free approach. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. doi: 10.1109/CVPR52734.2025.01855.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. TENT: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.

Sibo Wang, Jie Zhang, Zheng Yuan, and Shiguang Shan. Pre-trained model guided fine-tuning for zero-shot adversarial robustness. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Boxi Wu, Heng Pan, Li Shen, Jindong Gu, Shuai Zhao, Zhifeng Li, Deng Cai, Xiaofei He, and Wei Liu. Attacking adversarial attacks as a defense. *arXiv preprint arXiv:2106.04938*, 2021.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Songlong Xing, Weijie Wang, Zhengyu Zhao, Jindong Gu, Philip H. S. Torr, and Nicu Sebe. Fine-tune like you pretrain: Boosting zero-shot adversarial robustness in vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Findings*, pp. 737–747, 2026.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, Chenliang Li, Yuanhong Xu, Hehong Chen, Junfeng Tian, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mPLUG-Owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023.

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

Jongmin Yoon, Sung Ju Hwang, and Juho Lee. Adversarial purification with score-based generative models. In *International Conference on Machine Learning*, 2021.

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

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *IEEE/CVF International Conference on Computer Vision*, 2023.

Jiaming Zhang, Xingjun Ma, Xin Wang, Lingyu Qiu, Jiaqi Wang, Yu-Gang Jiang, and Jitao Sang. Adversarial prompt tuning for vision-language models. In *European Conference on Computer Vision*, 2024.

Yuanhe Zhang, Xueting Wang, Yanbin Ren, Haoran Gao, Xinhan Zheng, Zhenhong Zhou, Fanyu Meng, Li Sun, and Sen Su. Structure-guided visual perturbation neutralization for LVLMs. *arXiv preprint arXiv:2605.27927*, 2026.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 2022a.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022b.

Yiwei Zhou, Xiaobo Xia, Zhiwei Lin, Bo Han, and Tongliang Liu. Few-shot adversarial prompt learning on vision-language models. In *Advances in Neural Information Processing Systems*, 2024. doi: 10.52202/079017-0102.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023.

Xingyu Zhu, Huanshen Wu, Shuo Wang, Beier Zhu, Jiannan Ge, Jiaheng Zhang, and Long Chen. Robustifying vision-language models via test-time prompt adaptation. In *International Conference on Machine Learning*, 2026.

A Datasets and Experimental Protocol

This appendix specifies the datasets, splits, preprocessing, text prototypes, training and evaluation protocol, and reporting conventions underlying the main tables. Two principles govern the design: every hyperparameter is fixed before final test evaluation, and each backbone–dataset–seed is treated as a separate adaptation task with one head trained from scratch on the corresponding training split. The subsections below document the datasets and their roles, preprocessing and class-text handling, the full reproducibility specification, and the metric and aggregation rules.

A.1 Datasets and Splits

Table A.1 summarizes the datasets and their roles. MAGE trains an independent energy head within each target dataset; no head is transferred across datasets.

Table A.1: Datasets, class counts, and experimental roles.

Dataset	Classes	Task	Split and role
CIFAR-10	10	General objects	Official train/test split; component ablations
Caltech101	100	General objects	70/30 seed-fixed split after removing background class; Tables 2–3
CIFAR-100	100	Low-resolution objects	Official train/test split; multi-backbone evaluation
DTD	47	Texture recognition	Official train/test split; Tables 2–3
ImageNet	1000	Natural images	Official train/validation split; Table 2
Oxford Pets	37	Fine-grained pets	Standard Zhou train+val/test split; Table 3
Flowers102	102	Fine-grained flowers	Standard Zhou train+val/test split; Table 3
FGVC Aircraft	100	Fine-grained aircraft	Official variant trainval/test split; Table 3
EuroSAT	10	Remote sensing scenes	Standard Zhou train+val/test split; Table 3
Food101	101	Fine-grained food categories	Official train/test split; Table 3
Stanford Cars	196	Fine-grained car models	Official train/test split; Table 3
SUN397	397	Scene recognition	Official train/test split; Table 3

A.2 Preprocessing and Text Prototypes

All images use the preprocessing associated with the loaded CLIP checkpoint, including resize, center crop, tensor conversion, and RGB normalization. Class names are converted to natural-language forms and encoded with fixed prompt templates. Text prototypes and image features are L_2 normalized; the text encoder, image encoder, prototypes, and logit scale remain frozen. The energy head consumes normalized image features, while classification is still performed by the original image–text similarity logits.

Table A.2: Preprocessing and class-text protocol.

Stage	Processing	Protocol
Image input	Checkpoint-specific resize, center crop, tensor conversion, and RGB normalization	Training and evaluation use the same image interface without additional augmentation.
Class names	Dataset-provided names converted to natural language	Remove the Caltech101 background class; align ImageNet WNIDs with the class-name file.
Text prototypes	Fixed templates encoded and L_2 normalized, then averaged	Text prototypes, tokenizer, and logit scale remain frozen.
Image features	Visual encoder output followed by L_2 normalization	The MAGE head consumes these features; classification still uses image–text logits.
Data splits	Train/test split fixed before DataLoader construction	The same dataset–seed split is used for training, evaluation, and aggregation.

A.3 Training and Evaluation Protocol

MAGE uses a $d \to 128 \to 32 \to 1$ MLP energy head trained independently for every backbone–dataset–seed combination. AdamW uses learning rate 10^{-3} and weight decay 10^{-4}. The training adversary uses five PGD steps with radius $2/255$ and step size $1/255$. The one-step differentiable training surrogate uses detached Mean-Absolute gradient scaling. At deployment, MAGE uses Mean-Absolute normalization. All defenses use five projected updates, defense radius $10/255$, $\alpha = 2.5/255$, and temperature 1.0. The evaluation attack is randomly initialized adaptive PGD-CE

with radius $2/255$, step size $0.5/255$, and 20 steps, differentiating through the complete method-specific inference pipeline. The complete reproducibility specification, including scope-dependent training budgets and sample caps, is given in Table A.3. No test split is used for hyperparameter selection: defaults are fixed before final test evaluation, and the sensitivity scans are diagnostic analyses rather than test-set tuning. The three additional target datasets (Food101, Stanford Cars, and SUN397) use a clean-stability weight of 0.5 instead of 1.0; all other settings retain the values below.

Table A.3: Complete reproducibility specification. The reported test sets are never used for hyperparameter selection.

Scope	Parameter	Value
All MAGE runs	Energy head	$d \to 128 \to 32 \to 1$
All MAGE runs	Optimizer	AdamW
All MAGE runs	Learning rate / weight decay	10^{-3} / 10^{-4}
All MAGE runs	Training epochs	10 (CIFAR-10 ablation); 5 (multi-dataset runs)
CIFAR-10 ablation	Training / evaluation batch size	32 / 32
Multi-dataset runs	Training / evaluation batch size	8 / 8
CIFAR-10	Training / test sample cap	all / all
CIFAR-100	Training / test sample cap	all / all
DTD	Training / test sample cap	all / all
Caltech101	Training / test sample cap	all / all
ImageNet	Training / test sample cap	all / all
Oxford Pets, Flowers102, FGVC Aircraft, EuroSAT, Food101, Stanford Cars, SUN397	Training / test sample cap	all / all
Training attack	PGD radius / step size / steps	$2/255$ / $1/255$ / 5
Loss weights	$(\lambda_1, \lambda_2, \lambda_3, \lambda_4)$	$(1.0, 1.0, 0.5, 0.5)$
Loss margins	(m_E, m_R)	$(0.5, 0.1)$
Clean stability	λ_f	1.0 (0.5 for the additional datasets)
Training proxy	Gradient normalization	Mean-Absolute
Test-time defense	Radius / coefficient / steps	$10/255$ / $2.5/255$ / 5
Test-time defense	Temperature τ	1.0
ET3+MAGE	Combined-energy coefficient λ_a	1.0
Evaluation attack	PGD radius / step size / steps	$2/255$ / $0.5/255$ / 20
Hyperparameter selection	Test-set usage	None

A.4 Metrics, Aggregation, and Reporting

For each seed, all metrics are computed on the same test split before the required seed aggregation. Main tables report means without standard deviations; per-seed records and sensitivity scans are retained in Appendix B.

Table A.4: Metric definitions and aggregation rules.

Metric	Definition	Reported in
Clean	Natural-input accuracy before test-time defense	Tables 1–3 and Appendix B
Clean after defense	Natural-input accuracy after ET3, MAGE, or another method	Tables 1–3
Clean drop	Clean minus Clean after defense; negative values indicate improvement	Table 1 and Appendix B
PGD	Accuracy after the complete method-specific defense-aware PGD pipeline	Tables 1–3 and Appendix B
Gain	Difference from undefended PGD accuracy for the same backbone and dataset	Tables 1–2
Avg. PGD	Unweighted mean PGD accuracy across target datasets	Table 3

B Additional Results

B.1 Hyperparameter Sensitivity

We conduct one-factor sensitivity analysis on TeCoA4 ViT-B/32 and CIFAR-10 using the stable-restoration variant and seed 0. Unless otherwise stated, all settings follow the defaults above. Clean is the undefended natural accuracy, and MAGE Clean is the accuracy after learned-energy recovery.

Table B.1: Hyperparameter sensitivity on CIFAR-10 (%).

Parameter	Value	Clean	MAGE Clean	Drop	PGD Base	PGD ET3	PGD MAGE	PGD ET3+MAGE
α	$0.5/255$	64.9	74.9	-10.0	47.0	50.7	59.9	57.6
α	$1.0/255$	64.9	80.3	-15.4	47.0	52.4	67.6	63.1
α	$1.5/255$	64.9	82.6	-17.6	47.0	53.1	71.3	65.7
α	$2.5/255$	64.9	84.0	-19.1	47.0	53.8	73.7	68.3
α	$4.0/255$	64.9	83.8	-18.9	47.0	54.2	73.7	68.1
ϵ_d	$5/255$	64.9	82.3	-17.4	47.0	52.3	69.3	64.7
ϵ_d	$10/255$	64.9	84.0	-19.1	47.0	53.8	73.7	68.3
ϵ_d	$15/255$	64.9	82.8	-17.9	47.0	54.4	73.4	67.3
ϵ_d	$20/255$	64.9	84.0	-19.1	47.0	54.7	74.6	67.6
T	1	64.9	75.0	-10.1	47.0	50.4	59.7	56.9
T	3	64.9	82.3	-17.3	47.0	53.1	70.9	65.4
T	5	64.9	84.0	-19.1	47.0	53.8	73.7	68.3
T	7	64.9	84.4	-19.4	47.0	54.1	74.8	69.4
λ	0	64.9	84.0	-19.1	47.0	53.8	73.7	53.8
λ	0.25	64.9	84.0	-19.1	47.0	53.8	73.7	59.2
λ	0.5	64.9	84.0	-19.1	47.0	53.8	73.7	63.0
λ	1.0	64.9	84.0	-19.1	47.0	53.8	73.7	68.3
λ	2.0	64.9	84.0	-19.1	47.0	53.8	73.7	72.4
Clean-stability	0	64.9	88.1	-23.2	47.0	53.8	78.4	69.7
Clean-stability	0.25	64.9	88.5	-23.6	47.0	53.8	78.9	70.0
Clean-stability	0.5	64.9	84.0	-19.1	47.0	53.8	73.7	68.3
Clean-stability	1.0	64.9	77.4	-12.4	47.0	53.8	66.9	64.0

The main gains are obtained by increasing α from $0.5/255$ to $2.5/255$ and by using three to five recovery steps. The larger radius $20/255$ gives only a marginal learned-only improvement over $10/255$ while allowing larger input displacement. The direct energy combination improves with λ in this scan but remains below learned-only MAGE. Clean-stability is backbone-dependent: the TeCoA4 scan favors 0–0.25, whereas the main OpenAI CLIP ablation supports a moderate value. Thus, this weight should be selected per backbone rather than treated as a universal optimum.

B.2 Normalization and Training-Proxy Ablation

This section studies how the training-time input-gradient scaling and the deployment-time defense normalization affect MAGE. Unlike the component-level ablation in the main paper, this study uses three independent runs to isolate the training proxy and deployment normalization variables, while keeping the training budget, attack setup, and defense budget fixed within each configuration. The goal is to compare the recovery geometry induced by different normalization choices; Table 1 in the main text still analyzes the role of ranking, restoration, and clean stability.

We compare three configurations: detached Mean-Absolute scaling during training with L_2 normalization at deployment (MeanAbs/L2); L_2 normalization at both training and deployment (L2/L2); and Mean-Absolute normalization at both training and deployment (MeanAbs/MeanAbs).

Let the input energy gradient be

$$g(x) = \nabla_x E_\phi(x). \tag{B.1}$$

For the Mean-Absolute training proxy, the gradient scaling statistic is defined as

$$q_{\mathrm{MA}}(g) = \mathrm{stopgrad}\left(\frac{1}{n}\sum_{j=1}^{n}|g_j|\right), \tag{B.2}$$

and the one-step differentiable recovery uses

$$\frac{g}{q_{\mathrm{MA}}(g)+\xi}. \tag{B.3}$$

For the L_2 training proxy, we instead use the detached L_2 norm as the scaling statistic. In both cases, the normalization statistic is stop-gradiented, so it only sets the scale of the recovery step and does not receive outer-loop gradients.

We evaluate on CIFAR-10 with OpenAI CLIP ViT-B/32 and the stable_restoration objective. Robustness is measured by adaptive PGD-CE through each method's complete test-time defense pipeline.

The attack uses an L_∞ radius of $2/255$, step size $0.5/255$, and 20 steps. The defense uses 5 input updates, projection radius $10/255$, coefficient $\alpha = 2.5/255$, and temperature $\tau = 1.0$. At every PGD step, the attacker replays the full defense mapping and maximizes the post-defense cross-entropy.

For each row, ET3, MAGE, and ET3+MAGE use the deployment normalization specified by that row, so the energy forms are compared under the same input optimizer. The shared clean reference is 85.2%, and the undefended Base accuracy under adaptive PGD-20 is 0.0%; because these references do not change across normalization settings, they are not repeated in Table B.2.

Table B.2: Training-proxy and deployment-normalization ablation on CIFAR-10 (%).

Part 1: clean accuracy and prediction transitions

Train / Deploy	Clean	MAGE Clean	Drop	Changed	CW	WC
MeanAbs / L_2	85.2	65.3 ± 4.8	19.9 ± 4.8	33.6 ± 5.5	24.0 ± 4.8	4.1 ± 0.1
L_2 / L_2	85.2	46.9 ± 4.7	38.3 ± 4.7	51.2 ± 4.9	41.3 ± 4.6	3.0 ± 0.1
MeanAbs / MeanAbs	85.2	65.5 ± 2.4	19.7 ± 2.4	33.4 ± 2.7	23.9 ± 2.5	4.2 ± 0.1

Part 2a: adaptive robustness

Train / Deploy	Base	ET3	MAGE	ET3+MAGE
MeanAbs / L_2	0.0	12.9 ± 0.1	37.7 ± 1.6	36.2 ± 0.5
L_2 / L_2	0.0	12.9 ± 0.1	28.6 ± 1.4	30.2 ± 1.3
MeanAbs / MeanAbs	0.0	46.6 ± 0.3	55.1 ± 1.0	53.0 ± 1.6

Part 2b: post-defense margin statistics

Train / Deploy	Margin	Δ Margin
MeanAbs / L_2	-0.89 ± 0.06	15.51 ± 0.05
L_2 / L_2	-1.12 ± 0.18	15.28 ± 0.19
MeanAbs / MeanAbs	$+0.57 \pm 0.07$	16.97 ± 0.07

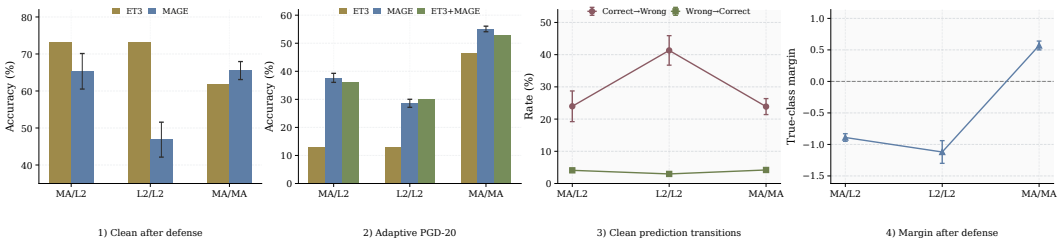

Figure B.1: Normalization and training-proxy ablation on CIFAR-10. MA/L2 denotes Mean-Absolute training scaling with L_2 deployment, L2/L2 uses L_2 scaling in both stages, and MA/MA uses Mean-Absolute scaling in both stages. The panels summarize defended clean accuracy, adaptive PGD-20 accuracy, clean prediction transitions, and post-defense true-class margin. Error bars reflect the three independent normalization-ablation runs where available.

The table reports the percentage of samples whose prediction changed across defense, along with the CleanWrong and WrongClean transition rates. Prediction Changed also includes transitions from one wrong class to another, so it is not the sum of the latter two quantities.

At deployment, when L_2 normalization is used throughout, the training proxy makes a large difference. MeanAbs/L2 achieves 37.7% adaptive PGD MAGE accuracy, 9.1 points above L2/L2, while also improving defended clean accuracy from 46.9% to 65.3%, a gain of 18.4 points.

This matches the prediction-stability behavior on clean inputs. Under L2/L2, 41.3% of samples undergo a CleanWrong transition, whereas MeanAbs/L2 reduces that rate to 24.0%. Prediction

Changed also drops from 51.2% to 33.6%. Under fixed deployment optimizer, attack budget, and defense budget, the detached Mean-Absolute training proxy therefore learns a local energy geometry that better supports semantic recovery while preserving clean predictions.

The same pattern appears in the feature shifts. Across the three runs, the average clean feature shift is about 0.155 for MeanAbs/L2 and about 0.232 for L2/L2. The smaller shift is consistent with the lower CleanWrong rate and the higher defended clean accuracy, indicating that MeanAbs training proxy both improves recovery on adversarial inputs and better limits damage to already-correct semantics.

With a fixed MeanAbs-trained head, changing the deployment normalization also changes the recovery trajectory. Switching deployment normalization from L_2 to MeanAbs raises MAGE adaptive PGD accuracy from 37.7% to 55.1%, a gain of 17.4 points, while moving the average true-class margin from -0.89 to $+0.57$. More importantly, MAGE clean accuracy remains almost unchanged, from 65.3% to 65.5%.

The same deployment normalization change also affects ET3. Its adaptive PGD accuracy rises from 12.9% to 46.6%, showing that test-time gradient normalization alone can substantially reshape the actual update trajectory. ET3 clean accuracy is 73.18% under L_2 deployment and about 61.86% under MeanAbs deployment, whereas MAGE remains much more stable across the two settings. This suggests a different local geometric interaction between learned recovery energy and the input optimizer than what is induced by the fixed LogSumExp energy.

Note that Mean-Absolute and L_2 normalization produce different effective coordinate updates on the same high-dimensional gradient. With fixed $\alpha = 2.5/255$ and an L_∞ projection radius, deployment normalization controls both the per-step displacement distribution and the projected trajectory. The main experiments therefore retain each method's stated deployment rule—Original ET3 with L_2 normalization and MAGE with Mean-Absolute normalization—while matching the backbone, step count, projection radius, coefficient, and attack budget. The main tables should consequently be read as method-level comparisons rather than as a pure energy-function isolation. Table B.2 provides the corresponding same-normalizer controls: under L_2 deployment, MAGE reaches 37.7% versus 12.9% for ET3, a 24.8-point advantage; under Mean-Absolute deployment, MAGE reaches 55.1% versus 46.6% for ET3, an 8.5-point advantage.

Overall, B.2 gives two complementary conclusions. First, under the same L_2 deployment normalization, detached Mean-Absolute training scaling significantly improves adaptive robustness and reduces prediction drift on clean inputs relative to L_2 training scaling. Second, with a MeanAbs-trained head fixed, deployment normalization materially changes the local recovery trajectory, robust accuracy, and final margin. Thus both the learned energy field and the deployment optimizer contribute to the final method behavior, and the default ET3–MAGE gap should not be attributed to the energy function alone. Both conclusions also hold across the three runs of each configuration: in the adaptive-robustness panel the largest standard deviation is 1.6 points, whereas the MeanAbs/L2 advantage over L2/L2 under L_2 deployment is 9.1 points and switching deployment normalization moves MAGE by 17.4 points, so neither effect depends on which run enters the mean. The per-seed records for the remaining methods are listed in the following subsection.

B.3 Per-Seed and Per-Dataset Results

The following tables retain the raw seed-level records used to check the main-table aggregates. Table B.3 decomposes the CIFAR-10 component ablation into three seeds, and its mean rows reproduce the values in Table 1; the per-seed ordering of variants stays stable even where individual seeds vary, confirming that the restoration-training conclusion is not an artifact of a single run. Table B.4 reports the TTC/MTA and APT/FAP per-seed logs behind Table 3 for all nine target datasets, including the three additional ones (Food101, Stanford Cars, and SUN397) appended at the end of each block. Tables B.5–B.7 add the R-TPT records and the TeCoA4 and FARE4 ViT-B/32 records, with the clean, defended, and Original ET3 references of each run. Every configuration is run with three seeds (0–2), and each mean row is their unweighted average, so the reported aggregates can be checked against individual runs; the ViT-L/14 settings appear only as means in the main text. Together these tables serve as an audit trail for the reported aggregates and verify that no headline number is driven by a single anomalous run.

Table B.3: CIFAR-10 component and restoration-loss ablation by seed (%).

Variant	Seed	Clean	Clean Def.	Drop	Base	ET3	MAGE
Ranking only	0	85.1	49.8	35.3	0.0	12.6	37.2
	1	85.6	55.3	30.3	0.0	12.9	40.8
	2	85.0	50.0	35.0	0.0	10.8	34.0
	Mean	85.2	51.7	33.5	0.0	12.1	37.3
Ranking + CE	0	85.1	51.8	33.3	0.0	12.6	52.8
	1	85.6	58.4	27.2	0.0	12.9	55.3
	2	85.0	55.8	29.2	0.0	10.8	54.2
	Mean	85.2	55.3	29.9	0.0	12.1	54.1
Ranking + margin	0	85.1	49.2	35.9	0.0	12.6	36.9
	1	85.6	56.3	29.3	0.0	12.9	43.7
	2	85.0	52.1	32.9	0.0	10.8	41.7
	Mean	85.2	52.5	32.7	0.0	12.1	40.8
Ranking + CE + margin	0	85.1	57.9	27.2	0.0	12.6	55.1
	1	85.6	57.8	27.8	0.0	12.9	54.8
	2	85.0	57.4	27.6	0.0	10.8	55.3
	Mean	85.2	57.7	27.5	0.0	12.1	55.1
MAGE Full	0	85.1	61.3	23.8	0.0	12.6	53.4
	1	85.6	65.7	19.9	0.0	12.9	55.3
	2	85.0	63.9	21.1	0.0	10.8	57.7
	Mean	85.2	63.6	21.6	0.0	12.1	55.5

Table B.4: TTC/MTA and APT/FAP per-seed results (%). Each cell is Clean / PGD. Each method lists its nine datasets in consecutive rows, and the two method groups are placed side by side to reduce vertical space.

		TTC / MTA					APT / FAP		
Method	Dataset	Seed 0	Seed 1	Seed 2	Method	Dataset	Seed 0	Seed 1	Seed 2
TTC	Caltech101	71.7/50.8	71.8/50.7	71.7/51.1	APT	Caltech101	86.9/76.1	86.4/74.9	87.1/77.6
TTC	Oxford Pets	59.9/26.2	60.7/26.4	61.0/25.7	APT	Oxford Pets	69.5/41.5	70.5/42.3	70.8/42.2
TTC	Oxford Flowers	36.8/16.3	36.7/16.5	36.8/16.3	APT	Oxford Flowers	71.8/32.3	69.8/32.0	74.0/34.1
TTC	FGVC Aircraft	5.2/1.6	5.0/1.8	5.2/1.8	APT	FGVC Aircraft	15.8/5.5	15.0/5.6	17.1/5.6
TTC	DTD	24.8/14.3	25.0/14.2	25.0/14.6	APT	DTD	44.2/28.3	44.0/25.6	44.3/23.8
TTC	EuroSAT	16.6/11.7	16.6/11.6	16.6/11.7	APT	EuroSAT	58.0/25.7	60.4/22.5	58.1/21.4
TTC	Food101	27.5/10.8	30.1/13.2	32.1/15.0	APT	Food101	25.1/15.1	27.5/17.5	29.6/19.3
TTC	Stanford Cars	18.2/14.4	20.5/16.8	22.5/18.6	APT	Stanford Cars	27.7/24.1	30.0/26.5	32.0/28.6
TTC	SUN397	33.2/24.0	35.5/26.3	37.5/28.3	APT	SUN397	43.5/9.1	46.0/11.3	48.2/12.9
MTA	Caltech101	83.7/74.1	84.6/75.9	84.3/75.3	FAP	Caltech101	86.4/70.0	85.5/68.8	87.3/68.4
MTA	Oxford Pets	72.5/56.6	72.5/56.6	72.5/56.6	FAP	Oxford Pets	42.6/32.7	47.0/34.9	33.7/31.9
MTA	Oxford Flowers	25.5/17.1	25.5/17.1	25.5/17.1	FAP	Oxford Flowers	61.3/37.4	57.0/34.1	53.2/31.6
MTA	FGVC Aircraft	5.0/3.4	5.0/3.4	5.0/3.4	FAP	FGVC Aircraft	17.5/6.3	15.3/5.8	17.6/6.6
MTA	DTD	29.6/21.2	29.6/21.2	29.6/21.2	FAP	DTD	42.4/26.8	45.9/27.7	38.7/25.5
MTA	EuroSAT	16.4/13.3	16.4/13.3	16.4/13.3	FAP	EuroSAT	61.8/41.2	20.6/18.1	54.8/37.8
MTA	Food101	21.5/10.6	23.9/13.0	25.7/14.8	FAP	Food101	20.5/6.5	22.9/8.5	25.0/10.2
MTA	Stanford Cars	36.3/22.2	38.8/24.7	41.0/26.9	FAP	Stanford Cars	40.2/11.5	42.6/13.8	44.7/15.8
MTA	SUN397	36.2/22.5	38.6/25.0	40.7/27.2	FAP	SUN397	30.5/23.4	32.9/25.8	35.0/27.9

Table B.5: R-TPT per-seed logs (%).

Dataset	Clean 0	Clean 1	Clean 2	PGD 0	PGD 1	PGD 2
Caltech101	91.3	91.4	91.0	82.2	81.2	81.5
Oxford Pets	84.0	84.1	84.0	63.3	62.4	62.9
Oxford Flowers	63.6	63.4	63.6	36.2	36.4	36.5
FGVC Aircraft	18.5	18.5	18.2	11.3	10.7	11.3
DTD	43.3	42.9	42.1	29.0	29.3	28.7
EuroSAT	32.1	32.2	32.1	14.7	15.0	14.4
Food101	22.4	24.7	26.7	21.2	23.7	25.6
Stanford Cars	33.3	35.7	37.8	30.0	32.5	34.4
SUN397	34.5	36.9	39.0	25.9	28.3	30.4

Table B.6: TeCoA4 ViT-B/32 per-seed results (%).

Dataset	Seed	Clean	MAGE Clean	PGD	ET3	MAGE
Caltech101	0	82.7	86.9	71.9	74.5	81.8
	1	84.1	87.6	72.7	74.0	82.1
	2	83.9	88.5	73.4	75.3	82.8
EuroSAT	0	17.8	29.4	13.8	12.8	25.1
	1	17.8	27.3	13.8	12.8	22.8
	2	17.8	26.6	13.8	12.8	22.3
DTD	0	30.2	36.8	21.9	22.4	29.2
	1	31.2	37.3	22.1	22.6	29.4
	2	28.4	35.0	21.4	20.9	28.6
FGVC Aircraft	0	5.3	10.4	3.3	2.3	7.4
	1	5.3	10.4	3.3	2.3	7.0
	2	5.3	10.7	3.3	2.3	7.3
Oxford Flowers	0	25.6	39.6	17.0	17.5	29.6
	1	25.6	39.2	17.0	17.5	29.0
	2	25.6	40.7	17.0	17.5	31.4
Oxford Pets	0	72.8	77.8	57.4	59.0	65.7
	1	72.8	77.0	57.4	59.0	65.7
	2	72.8	77.2	57.4	59.0	65.6
Food101	0	21.2	27.1	10.8	12.1	19.5
	1	23.6	29.4	12.9	14.4	21.9
	2	25.7	31.4	14.4	16.1	23.4
Stanford Cars	0	36.5	50.4	21.9	21.4	38.5
	1	38.4	52.8	24.3	23.8	41.1
	2	40.6	54.9	26.4	25.9	43.1
SUN397	0	35.8	39.4	22.4	23.3	30.5
	1	38.1	41.9	24.8	25.5	32.9
	2	40.7	44.1	26.9	27.4	35.0

Table B.7: Additional FARE4 and TeCoA4 ViT-B/32 per-seed results (%).

Backbone	Dataset	Seed	Clean	MAGE Clean	PGD	ET3	MAGE	ET3+MAGE
FARE4	Caltech101	0	85.0	87.3	73.9	76.2	81.9	77.1
		1	85.9	89.9	73.9	75.8	83.9	76.8
		2	87.8	89.8	75.9	78.2	84.1	79.0
FARE4	CIFAR-100	0	47.5	52.9	29.1	31.7	39.4	34.5
		1	49.9	58.8	31.6	33.3	46.4	36.2
		2	51.8	59.9	28.9	31.1	45.2	34.5
FARE4	DTD	0	37.6	42.9	24.8	24.6	34.4	26.9
		1	36.8	42.6	24.8	25.4	33.1	27.2
		2	36.4	42.1	24.5	23.7	32.7	26.1
TeCoA4	CIFAR-100	0	45.0	58.5	31.6	30.8	46.0	41.0
		1	47.5	61.4	32.6	32.5	48.6	40.4
		2	45.6	59.1	30.8	32.2	47.3	39.4
FARE4	ImageNet	0	52.8	51.8	31.0	33.2	36.5	33.5
		1	53.9	52.5	34.2	35.9	37.4	36.4
		2	56.6	55.7	35.0	37.7	40.5	38.0
TeCoA4	ImageNet	0	60.2	59.3	46.3	45.4	48.4	46.3
		1	58.4	57.8	44.9	45.5	46.0	46.2
		2	58.1	60.6	45.2	45.4	48.9	47.8

B.4 PGD Iteration Sensitivity

This section checks whether the number of adaptive white-box attack iterations materially changes the robustness estimate for MAGE. Unlike attacks that only generate adversarial examples against the frozen base classifier, the PGD attack here replays each method's full test-time defense at every attack step and directly maximizes the post-defense classification loss.

For a method m, let the complete test-time mapping be D_m, and define the attack objective as

$$\delta_m^* = \arg \max_{\|\delta\|_\infty \le \epsilon_a} \mathrm{CE}(s(D_m(x+\delta)), y)\,. \tag{B.4}$$

For MAGE, Original ET3, and ET3+MAGE, D_m includes all five test-time input updates. The attack graph therefore covers energy evaluation, input gradients, projection, and the full multi-step recovery process, so the attacker optimizes directly against the final deployed prediction.

To decouple attack-step sweeps from the normalization ablation, we keep one fixed defense configuration throughout this sensitivity study and run three independent trials. PGD uses an L_∞ radius of $2/255$ and a fixed step size of $0.5/255$, with 10, 20, 50, and 100 attack iterations. The defense always uses 5 input updates, an L_∞ projection radius of $10/255$, coefficient $\alpha = 2.5/255$, and temperature $\tau = 1.0$. This sweep is an independent attack-convergence diagnostic and is not used to replace the default MAGE numbers reported in the main tables.

Table B.8 reports robustness under different adaptive PGD iteration counts. Each result is optimized against the full test-time inference pipeline of the corresponding method, and values are reported as means over three independent runs.

Table B.8: Sensitivity to standard PGD iteration count on CIFAR-10 (%).

Steps	Clean	MAGE Clean	Drop	Base	ET3	MAGE	ET3+MAGE	Margin
10	85.2	70.1	15.1	0.03	14.4	37.2	30.0	−1.07
20	85.2	70.1	15.1	0.00	12.6	36.2	28.2	−1.17
50	85.2	70.1	15.1	0.00	13.6	36.8	29.4	−1.09
100	85.2	70.1	15.1	0.00	15.2	37.1	30.9	−1.01

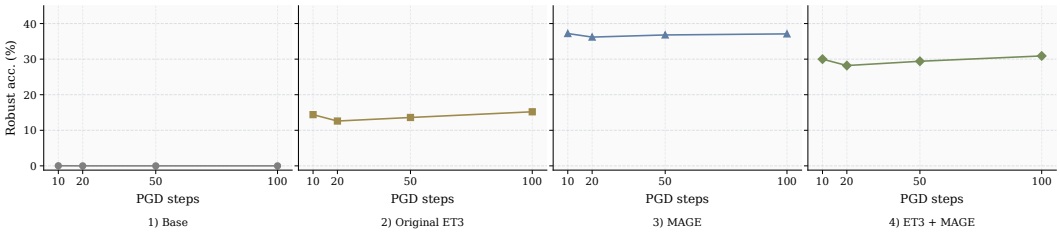

Figure B.2: Adaptive PGD iteration sensitivity on CIFAR-10 for the fixed diagnostic configuration of Table B.8. Across 10–100 attack steps, MAGE remains in a narrow 36.2%–37.2% robust-accuracy band, while Base remains near zero and Original ET3 stays substantially lower. These values belong to this independent sensitivity sweep and are not substitutes for the default MAGE results in the main tables.

As the number of attack iterations increases from 10 to 100, MAGE remains in a narrow 36.2%–37.2% robust-accuracy band, with a maximum spread of only 1.0 point. In particular, the 20-, 50-, and 100-step results are 36.2%, 36.8%, and 37.1%, all within the variation range of the three runs.

Within this fixed sensitivity configuration, the 20-step result already lies in the same narrow band as the 50- and 100-step results. Extending the attack budget therefore does not produce a systematic reduction in MAGE robustness in this sweep. Base is already near 0% at 20 steps and beyond, while MAGE remains well above Original ET3 at every attack length. Figure B.2 shows the same convergence pattern visually.

Different attack lengths follow different PGD optimization trajectories, so the final accuracy on a highly non-convex defense objective need not be monotone in the number of steps. More importantly, all four runs use the same attack radius, step size, and defense-aware objective, changing only the attack-iteration budget. The narrow variation therefore indicates that MAGE's evaluation is stable over the 10–100 step range.

The table reports only the final post-defense Margin statistic; no Δ Margin column is available for this sweep. We therefore restrict the interpretation to the quantities explicitly reported in Table B.8 rather than inferring an unreported margin-gain trend.

Thus, B.4 shows that, within this fixed defense configuration, increasing the adaptive PGD iteration count from 10 to 100 does not erase MAGE's advantage over Original ET3. It supports attack-iteration stability for this diagnostic setting without conflating the sweep with the default MAGE accuracy configuration used in the main results.

B.5 Robustness under Different Attacks

We isolate the effect of the attack objective by re-evaluating the robust base and MAGE on five datasets—the three additional ones (Food101, Stanford Cars, SUN397) plus CIFAR-100 and DTD from the main comparison—under four attacks: PGD-CE, APGD-CE, APGD-DLR, and CW-L_∞. All attacks use the same frozen TeCoA4 ViT-B/32 backbone, cover the full target split, and perturb only the frozen base model, after which MAGE applies its standard deployment transformation (radius $10/255$, step size $2.5/255$, five Mean-Absolute updates), which is left untuned across attacks. The energy heads are those of the main experiments, one per dataset and seed, trained with the stable-restoration objective for five epochs and reused here without retraining. Table B.9 lists the four attackers and Table B.10 reports the results.

Table B.9: Attack configurations used in the four-attack sweep. All attacks are L_∞ with $\epsilon_a = 2/255$ and perturb only the frozen robust base. The APGD variants use a single restart and `best_loss=True`, i.e., the maximum-loss perturbation per sample without clean fallback.

Attack	Steps	Loss	Step size	Attacked logits
PGD-CE	20	cross-entropy	fixed $0.5/255$	CLIP-scaled ($\times 100$)
APGD-CE	100	cross-entropy	per-sample adaptive	unscaled cosine
APGD-DLR	100	DLR	per-sample adaptive	unscaled cosine
CW-L_∞	100	CW margin $\max_{j\neq y} z_j - z_y$	$2\epsilon_a/\text{steps}$	CLIP-scaled ($\times 100$)

Table B.10: Robust accuracy (%) under four attacks on five datasets. Each cell reports the robust base / MAGE accuracy, averaged over three seeds; the first row reports clean accuracy without any attack, and the attack configurations are listed in Table B.9. Attacks are generated on the frozen base model and MAGE is applied afterwards without attack-specific tuning.

	(Robust CLIP base / MAGE)					
Attack	Food101	Cars	SUN397	CIFAR-100	DTD	Mean
Clean	23.5 / 29.3	38.5 / 52.7	38.2 / 41.8	46.0 / 59.7	29.9 / 36.4	35.2 / 44.0
PGD-CE	12.0 / 20.7	26.0 / 41.7	25.6 / 32.1	30.2 / 50.2	22.9 / 30.8	23.3 / 35.1
APGD-CE	13.6 / 22.2	30.4 / 45.4	27.7 / 34.0	33.6 / 51.9	24.8 / 32.0	26.0 / 37.1
APGD-DLR	11.3 / 23.9	24.3 / 46.1	23.9 / 33.7	29.7 / 55.6	22.4 / 33.3	22.3 / 38.5
CW-L_∞	11.2 / 19.9	24.2 / 40.0	23.9 / 31.0	29.0 / 50.2	22.4 / 31.0	22.2 / 34.4

MAGE's advantage over the robust base persists under every attack and on every dataset: the mean gain is 11.8 points under PGD-CE, 11.1 under APGD-CE, 16.2 under APGD-DLR, and 12.3 under CW-L_∞, and the largest per-dataset gains are on CIFAR-100 (+18.3 to +25.9) and Stanford Cars (+15.0 to +21.8). Absolute robust accuracy also depends on the attacker: the robust base is weakest under CW-L_∞ (22.2% on average) and strongest under APGD-CE (26.0%), whereas MAGE is

strongest under APGD-DLR (38.5%) and weakest under CW-L_∞ (34.4%). The comparison therefore does not hinge on a single attack objective, which complements the adaptive evaluation used in the main text.

C Comparison Methods and Fairness

C.1 Implementation and Adaptation Mechanisms

Base, Original ET3, MAGE, and ET3+MAGE form the controlled energy comparison. TTC, MTA, R-TPT, APT, and FAP retain their public implementation, checkpoint, prompt structure, view strategy, and few-shot protocol. MAGE trains only the energy head; it does not update the CLIP backbone, text encoder, class prototypes, or test-time prompt.

Table C.1: Adaptation mechanisms and implementation sources.

Method	Adapted object	Deployment operation	Supervision
Base	None	Zero-shot classification	None
Original ET3	None	Five fixed-energy input updates	None
MAGE	Energy head	Five learned-energy input updates	Target labels offline
TTC	None	Feature-side counterattack	None
MTA	None	Eight-view aggregation	None
R-TPT	Text context	Per-image prompt update	No offline labels
APT	Prompt parameters	Fixed learned prompt	Few-shot labels
FAP	Multimodal prompt	Fixed learned prompt	Few-shot labels

C.2 Per-Method Protocol Details

Base directly performs zero-shot classification. Original ET3 descends the fixed temperature-scaled LogSumExp energy, while MAGE replaces it with the learned scalar energy and keeps the same image-update interface. TTC uses its published feature-side counterattack, MTA aggregates eight augmented views, R-TPT updates text context independently for each image, and APT/FAP use their published few-shot prompt-learning procedures before evaluation.

Table C.2: Differences in offline supervision and online computation.

Method	Offline supervision	Persistent state	Main online budget
Base	None	None	One zero-shot forward
Original ET3	None	None	Five input-gradient updates
MAGE	Target split and labels	Energy head	Five input-gradient updates
TTC	None	None	Published counterattack
MTA	None	None	Eight-view aggregation
R-TPT	No offline labels	Per-image text prompt	Multi-view prompt updates
APT	16-shot labels	Learned prompt	Fixed prompt forward
FAP	Published few-shot data	Learned multimodal prompt	Fixed prompt forward

The purpose of this comparison is to expose the source of supervision and the granularity of online optimization rather than to collapse all methods into a single FLOPs number. In particular, the per-image prompt optimization of R-TPT and the multi-view computation of MTA are retained when interpreting their clean and PGD results.

C.3 Detailed Method Explanations

- **Base (Robust CLIP base).** An adversarially fine-tuned robust CLIP backbone (the TeCoA4 and FARE4 checkpoints used in Tables 2–3), evaluated without any test-time defense. Classification remains zero-shot through the raw image–text similarity logits, with no online correction, no learnable head, and no test-time label access. It is the undefended reference against which every robustness gain and clean-accuracy change is measured.

- **TTC (Test-Time Counterattacks, CVPR 2025).** A training-free test-time input defense that counterattacks the attacked image by maximizing the feature distance between the updated input and the attacked input's own feature anchor (the "falsely stable" representation), without a clean reference image or learned parameters (Xing et al., 2025). It corrects the input at inference but defines no objective over the classification margin, so its correction direction is not supervised for semantic recovery.
- **Tong et al. (zero-shot, training-free defense, CVPR 2025).** A truly zero-shot VLM defense that performs direct inference-time correction without learned prompts, auxiliary modules, or target-dataset labels (Tong et al., 2025). It marks the training-free extreme of the landscape, whereas MAGE uses target labels offline to shape a recovery objective.
- **ET3 (Energy-guided test-time defense, CVPR 2026).** Turns test-time defense into bounded projected descent on a fixed negative LogSumExp energy built from classification logits, with no additional training (Mirza et al., 2026). It is provable and label-free, but the LogSumExp energy measures aggregate confidence, so a low-energy input can still be dominated by a wrong class and the descent direction is not guaranteed to restore the ground-truth class.
- **MAGE (ours).** Learns a small scalar energy head on frozen image features with ranking, differentiable restoration, and ground-truth margin supervision, then deploys it as bounded test-time image updates. Unlike fixed-energy and prompt methods, the learned gradient is explicitly supervised to move adversarial inputs toward their labeled class while preserving clean predictions; only the tiny head is trained offline with target labels.
- **MTA (Multi-view test-time aggregation).** A training-free method that aggregates predictions over eight augmented views at deployment to stabilize the zero-shot decision (Zanella & Ben Ayed, 2024). It reduces attack variance by ensembling but neither moves the input along a recovery direction nor learns parameters.
- **R-TPT (Test-time prompt tuning, CVPR 2025).** Optimizes the text context independently for each image through multi-view test-time prompt updates, using no offline labels (Sheng et al., 2025). It attains high clean accuracy but spends a larger per-image online budget, and because it changes the decision interface it affects attack gradients and predictions differently from an input transformation.
- **APT (adversarial prompt tuning; the One Prompt Word method of Li et al., 2024b).** Learns a fixed adversarial prompt from 16-shot labels before evaluation and keeps it frozen at test time. It modifies the decision interface rather than the input, and its 16-shot supervision is far weaker than MAGE's full target-split training signal.
- **FAP (Few-shot multimodal prompt).** Learns a visual-and-text multimodal prompt from published few-shot data (Zhou et al., 2024). Like APT, it adapts the prompt instead of the input, so it shapes no input-space recovery field and changes both the attack surface and the decision interface.
- **TeCoA.** A robust backbone produced by text-guided contrastive adversarial fine-tuning that preserves vision-language alignment during adversarial training (Mao et al., 2023). It supplies the TeCoA4 checkpoints used in Tables 2–3, and MAGE adapts on top of these frozen weights.
- **FARE.** A robust backbone produced by unsupervised adversarial fine-tuning (Schlarmann et al., 2024). It supplies the FARE4 checkpoints used in Table 2; MAGE treats it as a frozen base and adds only the learned energy head.

We also reference, but do not evaluate, the surrounding prompt-learning foundations (CoOp/CoCoOp, MaPLe), diffusion and score-based purification, and adaptive-attack evaluations such as DiffBreak and Chain of Attack; they define the landscape rather than the controlled comparison.

Concurrent 2026 work targets the same robustness goal through other mechanisms; the entries below are cited for context and are not part of the controlled comparison.

- **AGFT (alignment-guided fine-tuning).** Aligns adversarial fine-tuning with the pretrained cross-modal alignment so that zero-shot transfer is retained (Cui et al., 2026).
- **Hierarchically Robust.** Exploits hierarchical semantic structure during robust fine-tuning of vision-language models (Dong et al., 2026).
- **A-TPT (attention-guided test-time prompt tuning).** Adds attention-guided enhancement to adversarial test-time prompt tuning (Hai et al., 2026).

- **T-VSS (test-time visual subspace steering).** Performs lightweight test-time adaptation directly in the visual feature subspace (Jang et al., 2026).
- **AdPO (adversarial preference optimization).** Improves LVLM robustness through preference optimization during training (C. Liu et al., 2026).
- **SCC (self-calibrated consistency).** Restores robustness at test time through semantic and spatial self-consistency (J. Liu et al., 2026).
- **CLBP (closed-loop bidirectional prompting).** Recovers attacked cross-modal semantics through closed-loop bidirectional prompting (X. Liu et al., 2026).
- **AdvFLYP (finetune like you pretrain).** Mimics the CLIP pretraining objective during adversarial fine-tuning (Xing et al., 2026).
- **SIGN (structure-guided neutralization).** Neutralizes input-side visual perturbations for LVLMs in a lightweight, plug-and-play way (Zhang et al., 2026).
- **RITA (robust test-time prompt adaptation).** Adapts test-time prompts through distribution-level alignment, optimal transport, and dynamic caching (Zhu et al., 2026).

Table C.3: Essential distinction between the 2025 and 2026 related-work methods most relevant to MAGE.

Year	Work	Paradigm	Essential distinction
2025	TTC	Test-time feature counterattack	Training-free; maximizes feature distance from the attacked anchor, learns no gradient objective.
2025	R-TPT	Test-time prompt tuning	Optimizes per-image text context; no offline labels.
2025	Tong et al. (zero-shot, training-free defense)	Zero-shot, training-free	Direct inference-time correction without target labels or learned modules.
2026	ET3	Energy-guided test-time defense	Fixed LogSumExp descent; provable but no semantic recovery supervision.
2026	MAGE (ours)	Target-supervised energy adaptation	Learned head supervised by restoration and ground-truth margin.

C.4 Fairness and Compute Budget

For Tables 1–2, Base, Original ET3, MAGE, and ET3+MAGE use the same backbone weights, class names, templates, preprocessing, split, attack budget, five-step update budget, and projection radius for every matched backbone–dataset–seed combination. Their only controlled difference is the energy function and, for MAGE, the learned head. For Table 3, all methods use the same target datasets, test sets, attack budget, metric definition, and seed aggregation, while each method retains the online operation and checkpoint required by its original protocol. R-TPT's EuroSAT clean result is available from three matched runs and is included as 32.1%.

Table C.4: Fairness audit for the controlled and cross-method comparisons.

Protocol item	Controlled energy comparison	Cross-method comparison
Backbone and weights	Identical within each matched combination	Public or prescribed weights retained
Preprocessing and classes	Fully identical	Same target task; prompt mechanisms retained
Offline supervision	MAGE target split; ET3/Base none	Each method follows its published supervision
Persistent updates	MAGE head only	Prompt methods retain prompt parameters
Online budget	Five input updates	MTA uses eight views; R-TPT updates per image
Attack and aggregation	Same PGD and seed protocol	Same budget; each method is attacked through its own pipeline

D Complexity and Efficiency

D.1 Parameter Count and Computational Complexity

MAGE freezes the image encoder, text encoder, class-text prototypes, and zero-shot classifier, and trains only the scalar energy head g_ϕ on normalized visual features. The trainable parameter count is therefore fully determined by the energy head and does not grow with the number of test-time classes.

For OpenAI CLIP ViT-B/32, the normalized visual feature dimension is $d = 512$, and the energy head uses a

$$512 \to 128 \to 32 \to 1 \tag{D.1}$$

three-layer MLP. The first linear layer contains

$$512 \times 128 + 128 = 65,664 \tag{D.2}$$

parameters, the second layer contains

$$128 \times 32 + 32 = 4,128 \tag{D.3}$$

parameters, and the output layer contains

$$32 \times 1 + 1 = 33 \tag{D.4}$$

parameters. The 128-dimensional LayerNorm contributes another 256 trainable affine parameters, giving

$$65,664 + 4,128 + 33 + 256 = 70,081. \tag{D.5}$$

Using the recorded model size of 151,277,313 parameters as the denominator, the trainable head accounts for only

$$\frac{70,081}{151,277,313} \times 100\% \approx 0.0463\%. \tag{D.6}$$

Table D.1: MAGE parameter count on OpenAI CLIP ViT-B/32.

Backbone	Feature dim.	Energy-head parameters	Ratio
OpenAI CLIP ViT-B/32	512	70,081	0.0463%

For ViT-L/14, used in the main experiments, the visual feature dimension increases to $d = 768$, and the same $768 \to 128 \to 32 \to 1$ structure contains only 102,849 trainable parameters. MAGE therefore remains highly parameter-efficient even as the visual backbone grows.

Computationally, the head forward pass is much cheaper than the visual encoder. The main deployment overhead comes from input optimization: each step computes the energy through the frozen visual encoder and differentiates the scalar energy with respect to the input image. Deployment cost therefore scales primarily linearly with the number of test-time defense steps T, rather than with the size of the energy head.

This gives MAGE and Original ET3 nearly identical deployment structure and the same update count. They differ in the energy definition and in their default image-gradient normalizer, while the learned MAGE head is tiny relative to the frozen visual encoder. The matched implementation benchmark below fixes the deployment normalizer to isolate runtime and memory overhead.

D.2 Measured Deployment Efficiency

We further measure the deployment overhead of Base, Original ET3, and MAGE under the same runtime environment. The measurement uses CIFAR-10, OpenAI CLIP ViT-B/32, and batch size 64. For this timing-only matched implementation benchmark, Original ET3 and MAGE both use five test-time input updates with defense radius $10/255$, coefficient $\alpha = 2.5/255$, and image-gradient L_2 normalization at deployment. This benchmark is intended to compare overhead under the same optimizer implementation; it is not the default MAGE accuracy configuration in the main experiments, where MAGE uses Mean-Absolute normalization.

Before timing, we run two warm-up batches and then measure ten batches, totaling 640 images. We record total wall-clock time, average latency per image, and peak memory. All methods are measured with the same runtime and batch configuration.

Table D.2: Measured deployment efficiency on CIFAR-10.

Method	Images	Total time (s)	ms/image	Peak MB
Base	640	0.398	0.622	820.8
Original ET3	640	5.877	9.183	1984.8
MAGE	640	5.875	9.180	1984.8

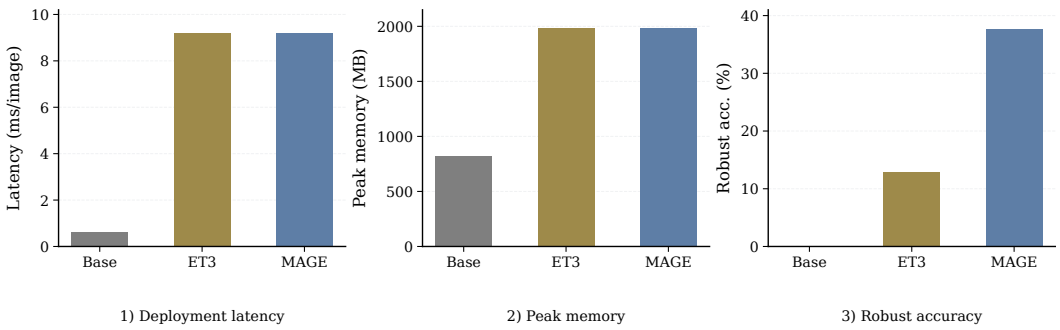

Figure D.1: Matched deployment-efficiency benchmark on CIFAR-10. Latency and peak memory are measured with five updates and L_2 deployment normalization for both Original ET3 and MAGE. The robust-accuracy panel reports the corresponding matched-L_2 diagnostic values (Base 0.0%, ET3 12.9%, MAGE 37.7%) from Table B.2; these are not the default Mean-Absolute MAGE results in Table 1.

Five input updates increase average latency from 0.622 ms/image for Base to about 9.18 ms/image, corresponding to roughly a 14.8$\times$ runtime increase. The additional cost mainly comes from repeated forward passes through the frozen visual encoder and energy gradients with respect to the input image.

In contrast, MAGE and Original ET3 have nearly indistinguishable deployment overhead. Their total processing times for 640 images are 5.875 s and 5.877 s, respectively, an absolute difference of only 0.002 s and a relative difference of about 0.03%. Their average latencies are 9.180 and 9.183 ms/image, and their peak memory usage is identical.

At the matched-L_2 setting visualized in Figure D.1, MAGE retains a substantial robust-accuracy advantage over Original ET3 while their measured latency and peak memory are effectively indistinguishable. The default main-experiment MAGE configuration uses Mean-Absolute deployment normalization and is reported separately. Thus, the learned energy head does not obtain its gains by using more test-time update steps, more peak memory, or a materially larger inference-time budget.

E Energy and Recovery Analysis

This section analyzes MAGE's test-time behavior from three perspectives: clean/adversarial energy separation, multi-step recovery trajectories, and representative recovery patterns. The experiments use CIFAR-10 with OpenAI CLIP ViT-B/32, and MAGE performs five test-time energy-descent updates with its default Mean-Absolute image-gradient normalization.

We use an independent run corresponding to the mechanism analysis and compute paired energy statistics on the full 10,000-image CIFAR-10 test set. Adversarial inputs are generated by an adaptive attack against the complete MAGE inference pipeline. Independently trained energy heads may have different scalar energy scales, so clean/adversarial energy gaps are interpreted only within the same head; recovery directions are compared through true-class margin and final classification outcomes.

Unless otherwise stated, labels, true-class margins, and clean reference features are used only for offline analysis. They are not used by the deployed defense, which still relies only on the observed image, frozen backbone, and trained g_ϕ.

E.1 Clean/Adversarial Learned-Energy Separation

We first examine whether the learned energy E_ϕ consistently separates clean inputs from their paired adaptive adversarial examples.

For N paired clean/adversarial samples, the paired energy ranking accuracy is

$$\mathrm{RankAcc} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\left[E_\phi\left(x_{\mathrm{adv}}^{(i)}\right) > E_\phi\left(x^{(i)}\right)\right]. \tag{E.1}$$

The mean adversarial-minus-clean energy gap is

$$\Delta E = \frac{1}{N}\sum_{i=1}^{N}\left(E_\phi(x_{\mathrm{adv}}^{(i)}) - E_\phi(x^{(i)})\right). \tag{E.2}$$

Across all 10,000 pairs, $E_\phi(x_{\mathrm{adv}}) > E_\phi(x)$, giving 100.0% paired energy ranking accuracy and a mean adversarial-minus-clean energy gap of 21.56. This shows that the ranking constraint transfers to held-out test inputs and induces a stable clean/adversarial energy hierarchy.

Table E.1: Clean/adversarial energy separation statistics.

Statistic	Value
Paired samples	10,000
Paired ranking accuracy	100.0%
Mean energy gap	21.56

This result is consistent with MAGE's training design. The ranking term first places perturbed inputs in a higher-energy region, providing a global potential structure for subsequent energy descent; the restoration and margin terms further shape the local gradient so that test-time descent is aligned with true-class recovery. The left panel of Figure E.1 visualizes the full paired energy-gap distribution: all 10,000 observed gaps are positive, with mean $\Delta E = 21.56$.

Energy values and energy gradients play complementary roles. Paired ranking describes the relative scalar potential of different input states, whereas the actual test-time trajectory is determined by $-\nabla_x E_\phi(x)$. Thus, this subsection verifies stable adversarial-to-clean ordering, and the next subsection analyzes how true-class margin evolves along energy descent.

E.2 Multi-Step Recovery Trajectories

To examine how energy descent changes semantic decisions, we analyze eight representative adaptive adversarial examples at steps 0, 1, 3, and 5, recording both prediction trajectories and true-class margins.

The subset intentionally covers different trajectory types, including successful recovery, wrong-class transfer, recovery followed by reversion, and local oscillation near competing classes. It is used for qualitative mechanism analysis; overall performance is still measured by full-test-set accuracy in the main paper and appendix.

Table E.2: Representative MAGE recovery trajectories.

Case	True label	Prediction trajectory (0135)	Margin Step 0	Margin Step 1	Margin Step 3	Margin Step 5	Outcome
0	0	2→2→0→0	-14.04	-1.91	3.46	2.95	Recovered
1	5	3→3→5→5	-10.49	-1.13	3.89	4.30	Recovered
2	0	6→6→9→9	-19.48	-2.60	-3.72	-1.86	Wrong-class transfer
3	4	7→7→4→7	-22.92	-10.37	2.89	-0.54	Recovered then reverted
4	8	9→9→8→8	-9.85	-3.05	6.72	8.17	Recovered
5	4	1→2→4→4	-15.13	-1.88	2.03	3.73	Recovered
6	5	3→3→7→3	-11.06	-4.84	-3.20	-4.30	Oscillatory
7	8	9→9→9→1	-15.29	-6.63	-1.15	-2.83	Wrong-class transfer

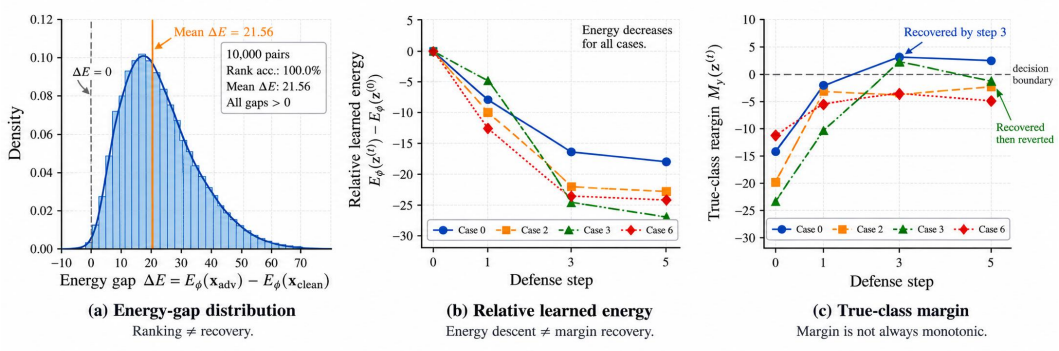

Figure E.1: Energy separation and representative MAGE recovery dynamics on CIFAR-10. Left: the distribution of paired energy gaps $E_\phi(x_{\text{adv}}) - E_\phi(x_{\text{clean}})$ over all 10,000 test pairs. Middle: relative learned energy along the trajectories of Cases 0, 2, 3, and 6 from Table E.2. Right: true-class margins for the same cases, illustrating recovery, wrong-class transfer, recovery followed by reversion, and oscillation. Energy decreases monotonically for all four plotted cases, including failures, showing that lower scalar energy alone is insufficient; semantic recovery depends on how the descent direction is aligned with the local class geometry.

Successful cases usually shrink the negative margin quickly in the first few steps and then cross the decision boundary into the true-class region. For example, Case 0 moves from a Step-0 margin of -14.04 to -1.91 at Step 1 and 3.46 at Step 3, while the prediction recovers from class 2 to the true class 0. Case 1 follows a similar pattern and reaches a final margin of 4.30.

Cases 4 and 5 show more sustained positive recovery. Case 4 improves from -9.85 to 8.17, and Case 5 improves from -15.13 to 3.73. These examples show that, when the local gradient matches the class-boundary geometry, MAGE can substantially increase the true class's advantage over competing classes within only a few updates.

The local trajectories in multi-class decision space are richer, however. Cases 2 and 7 move closer to the true-class boundary but ultimately enter another competing class region, producing wrong-class transfer. Case 6 lies near multiple competing boundaries and changes among wrong classes.

Case 3 highlights a boundary-crossing behavior. Its margin improves from -22.92 to 2.89 at Step 3, and the prediction recovers to the true class 4. After additional updates, however, the Step-5 margin becomes -0.54 and the input re-enters a neighboring class region. This indicates that the same energy field can enable effective early boundary crossing, while the step at which each sample reaches its best semantic margin may differ. The middle panel of Figure E.1 further shows that learned energy can continue decreasing even when the prediction later reverts or remains incorrect, which separates scalar descent from semantic recovery.

E.3 Recovery Patterns and Local Geometry

The previous two subsections reveal MAGE's mechanism at the scalar-energy and multi-step trajectory levels.

First, the 100.0% paired energy ranking accuracy shows that the learned energy establishes a stable potential ordering between clean and adversarial inputs. For an attacked observed input, the higher initial energy provides a common coordinate for test-time descent.

Second, classification recovery is determined by the relationship between the energy gradient and the local class margin. MAGE uses true-label supervision during restoration training to directly optimize the post-descent classification result and margin gain. Its objective is therefore not merely to find lower-energy inputs, but to shape a local vector field that is statistically better aligned with the correct-class direction on the training distribution.

The representative trajectories suggest four recurring behaviors.

Recovered trajectory. The input rapidly reduces its negative margin in the first 1–3 steps and enters the true-class region. Cases 0, 1, 4, and 5 follow this pattern. Some samples continue expanding the margin after entering the correct region, suggesting that the learned potential field provides a continuous recovery direction across both sides of the decision boundary.

Wrong-class transfer. The input substantially improves its position relative to the original wrong class, but enters another competing class region near a multi-class boundary. For instance, Case 7 improves its margin from -15.29 to -1.15, indicating movement toward the true-class boundary, but the final prediction transfers to another wrong class. This behavior reflects competition in multi-class decision geometry rather than a binary relation between the attacked and true classes.

Recovered-then-reverted trajectory. The sample enters the true-class region at an intermediate step and then crosses a neighboring decision boundary under continued energy descent. Case 3, which reaches a positive margin of 2.89 at Step 3, is a representative example. This pattern shows that test-time recovery may have a sample-specific optimal stopping point, and that local margin could inform future dynamic control strategies.

Oscillatory trajectory. Near multiple close class boundaries, a sample may move among several competing classes. Case 6 illustrates this local competition. Even there, the true-class margin improves from -11.06 at Step 0 to -3.20 at Step 3, showing that the recovery vector field first reduces the semantic displacement induced by the attack.

Together, these behaviors show that test-time energy defense has two levels. The first is potential ordering, which describes the energy region occupied by the current input. The second is local recovery geometry, which determines how the true class moves relative to competing classes after following the negative energy gradient.

This distinction explains why MAGE jointly trains energy ranking, differentiable restoration, true-class margin, and clean stability. Ranking provides global state structure, restoration and margin directly shape the local direction, and clean stability constrains the vector field near natural inputs. Together, these objectives make the learned energy not only identify attacked states, but also serve as an effective potential for test-time input correction.

These analyses complement the accuracy results in the main text. MAGE's gains do not come from simply mapping adversarial examples to a lower scalar energy; they come from supervised training that changes the local energy geometry in input space, making the negative energy gradient statistically more aligned with true-class recovery. This is the core reason MAGE achieves stronger adaptive robustness than fixed LogSumExp energy.