SAMPLE-WISE TARGETED ADVERSARIAL ATTACKS ON TEST-TIME ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Test-time adaptation (TTA) effectively counters distribution shifts but exposes models to adversarial manipulation via the unlabeled test stream. Existing class-wise targeted attacks remain impractical for stealthy exploitation in this setting: since TTA operates on batches, forcing a subset of samples toward a target label unintentionally pulls similar benign samples along, resulting in a conspicuously high frequency of the target label that is easy to detect. To capture a more realistic threat, we introduce a sample-wise targeted attack. Unlike prior approaches, the attacker aims to misclassify only inputs carrying an attacker-chosen trigger, while preserving a benign-like output label distribution to evade detection. To achieve this, we propose a meta-learning-based attack with a novel priority-aware gradient alignment strategy that explicitly prioritizes attack success. The strategy formulates the gradient update as an ellipsoidal trust-region problem, mitigating the misalignment between attack success and stealth, while providing theoretical guarantees for effective optimization of the attack objective in the presence of gradient misalignment. Extensive experiments on CIFAR-10-C, CIFAR-100-C, and ImageNet-C across TTA protocols demonstrate that our method achieves high targeted success rates while maintaining prediction behavior close to benign adaptation under multiple label-free stealth metrics, making it difficult to detect in unlabeled TTA deployment scenarios. Furthermore, we demonstrate that our attack shows strong robustness against existing defenses.

1 INTRODUCTION

Deep models deployed in the wild inevitably face distribution shift between training and deployment data due to changes in sensors, environments, or usage patterns, which can severely degrade performance even for models that are robust on in-distribution benchmarks Hendrycks & Dietterich (2019a). Training-time remedies such as domain adaptation Li et al. (2024) and robust training Qian et al. (2022) mitigate some shifts but cannot anticipate all future conditions and often require source data and labels that are unavailable at deployment. Test-time adaptation (TTA) addresses this gap by updating model parameters online using unlabeled test inputs Liang et al. (2025): by minimizing entropy Wang et al. (2020); Lee et al. (2022), performing self-training Su et al. (2022); Sinha et al. (2023), or enforcing consistency on the incoming test stream Wang et al. (2022); Brahma & Rai (2023). Without the training set, TTA can effectively recover performance under distribution shift.

However, because TTA implicitly trusts the unlabeled test stream to drive online updates, it introduces a new attack surface Wu et al. (2023). By manipulating a subset of the test samples used for a single update, an adversary can influence predictions on other inputs processed in the same batch by a shared TTA model Su et al. (2024). Existing literature largely focuses on two threat models. The first is *untargeted* attack, in which the adversary drives the model into broadly poor performance, offering limited concrete benefit to a rational attacker. The second is *class-wise targeted* attack Wu et al. (2023); Cong et al. (2024); Su et al. (2024); Rifat et al. (2025). The latter, while more strategic, remains suboptimal: by indiscriminately biasing predictions for all samples resembling the victim class, it causes widespread, easily detectable anomalies (e.g., constant prediction of the target class). Since TTA has no access to ground truth labels during deployment, direct accuracy monitoring is infeasible. Thus, we consider *distributional stealth*, i.e., maintaining label-free prediction statistics close to a benign baseline, as a practical proxy for stealth, motivated by label-shift detection via prediction distributions Lipton et al. (2018) and evidence that marginal distribution shifts are effective

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

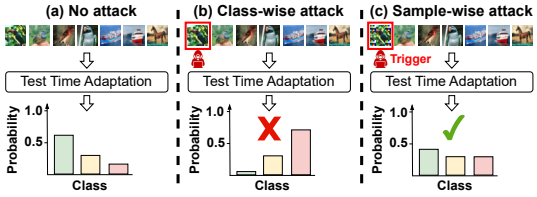


Figure 1: Illustration of distributional stealth. Unlike class-wise attacks **(b)** that create detectable anomalies by indiscriminately affecting *all* samples in a class, the proposed sample-wise attack **(c)** uses a specific trigger to misclassify *only* the intended victim, preserving a natural, stealthy prediction distribution to evade detection.

Table 1: Taxonomy of existing adversarial threat models under TTA. Unlike prior works that only enable class-wise targeted attacks, our threat model allows an adversary to selectively induce misclassification on specific inputs, possibly from different classes (sample-wise).

Threat model	Model Knowledge	Targeted Attack	Selectivity	Access to user data
DIA Wu et al. (2023)	White-box	✓	Global/ class-wise	✓
TePA Cong et al. (2024)	Grey-box	✗	Global	✗
FCA Rifat et al. (2025)	Grey-box	✓	Global/ class-wise	✗
RTTDP Su et al. (2024)	Grey-box	✓	Global/ class-wise	✗
Ours (this work)	Grey-box	✓	**Sample-wise**	✗

drift signals Rabanser et al. (2019). As illustrated in Figure 1, *class-wise targeted* attacks are unstealthy as they induce an unnatural concentration of predictions on the target, which could trigger anomaly alarms, especially when the victim class appears frequently. In contrast, realistic adversaries often prioritize selective manipulation to avoid detection. Therefore, it is important to investigate a finer-grained threat model that explicitly captures the selective nature of attacks under TTA, to guide the development of more robust TTA defenses.

We study *sample-wise targeted* attacks on TTA, in which only inputs carrying an attacker-chosen trigger (possibly from different classes) should be redirected to a target label, while non-triggered inputs from any user preserve a benign-like prediction distribution to evade detection. Instantiating this threat model presents two key challenges. First, enforcing sample-wise selectivity under realistic TTA constraints is difficult, as the attacker has no access to benign or victim user data and must rely solely on attacker-controlled samples Su et al. (2024). Second, existing class-wise attacks Wu et al. (2023); Su et al. (2024); Rifat et al. (2025) do not consider trigger information during optimization; consequently, naively applying them to the sample-wise setting yields poor performance.

To overcome these limitations, we propose a meta-learning-based attack that simulates the adaptation process to best utilize the attacker's limited data. This framework optimizes the adaptation process with two objectives: ***(i)*** a primary attack objective that maximizes the success rate on trigger-carrying inputs, and ***(ii)*** a secondary distributional stealth objective that prevents the conspicuous output anomalies. We introduce a novel loss term for ***(ii)***, tailored to the TTA setting. However, we identify a critical optimization challenge: the gradients of these two objectives are strongly antagonistic. Standard multi-task formulations Sener & Koltun (2018); Yu et al. (2020); Liu et al. (2021) fail here because they treat tasks symmetrically, seeking Pareto compromises that balance competing objectives. In an adversarial context, however, such symmetry is detrimental: it often sacrifices the attack effectiveness for marginal gains in stealth. Since a failed attack has zero utility to the adversary regardless of its stealthiness, the objectives are inherently *asymmetric* in our setting, where attack success is primary, while stealth functions as a constraint. To enforce this, we propose a novel *priority-aware gradient alignment* formulated via an ellipsoidal trust-region. Our update direction is constructed to stay close to the attack gradient while penalizing movement toward misaligned stealth-gradient directions, yielding updates that are attack-driven yet distributionally consistent. We provide theoretical guarantees showing that this aligned direction ensures descent on the attack objective for the unprojected update even under severe gradient misalignment, whereas symmetric multi-objective baselines lack such guarantees and may fail to make progress. Finally, we empirically demonstrate that our method achieves a high sample-wise targeted attack success rate of around 90% while maintaining label-free prediction statistics close to the no-attack baseline. Contributions of this paper are summarized as follows:

- We formalize a sample-wise targeted threat model for TTA that enforces strict selectivity: attacking only trigger-carrying inputs while maintaining a benign-like prediction behavior for non-triggered queries.
- We propose a meta-learning-based attack with a novel priority-aware gradient alignment that mitigates the gradient misalignment between attack objectives and distributional stealth, with a descent guarantee for the unprojected update and a sufficient descent condition under projection.

- Extensive experiments demonstrate that our method achieves high targeted attack success rates while maintaining benign-like behavior under multiple label-free stealth metrics.

2 Related Works

We discuss the related work of this study below, with additional discussion provided in Appendix A.

Test-time adaptation and attacks. TTA handles distribution shifts by updating model parameters on unlabeled test streams, typically via entropy minimization Wang et al. (2020); Zhang et al. (2024) or self-training Niu et al. (2023); Rusak et al. (2021). While effective, this mechanism implicitly trusts the incoming data, exposing a critical attack surface. Prior adversaries largely focus on *untargeted* Cong et al. (2024) or *class-wise targeted* attacks Wu et al. (2023); Su et al. (2024); Rifat et al. (2025). Both render the attack unstealthy. In contrast, we propose a sample-wise targeted threat model: misclassifying only trigger-carrying inputs while maintaining a benign-like prediction behavior. The key differences are summarized in Table 1.

Connection to backdoor attacks. Our use of input triggers parallels traditional backdoor attacks Gu et al. (2017); Li et al. (2022). However, unlike standard backdoor attacks that require poisoning training data or model parameters during training, our attack operates strictly at test time. We exploit the TTA process to induce backdoor-like behavior in clean, pre-trained models on the fly, without any prior access to either the training phase or the training data.

Gradient conflict resolution. Balancing the attack and stealth objectives resembles Multi-Task Learning (MTL), where conflicting or misaligned gradients often destabilize optimization Zhang & Yang (2021). However, unlike standard MTL solvers (e.g., Yu et al. (2020); Liu et al. (2021)) that seek symmetric trade-offs or Pareto optimality, our setting is priority-asymmetric: maximizing attack success is the primary goal, while stealth is a secondary objective. Consequently, we propose a priority-aware alignment strategy that penalizes movement toward increasingly misaligned stealth directions, rather than balancing both objectives equally.

3 Threat Model

We consider a shared TTA service that adapts on batches containing inputs from multiple sources, including an attacker-controlled subset, as shown in Figure 2. These inputs jointly influence adaptation and the predictions returned for the batch. The same shared-deployment scenario is also considered in prior TTA attack work Wu et al. (2023); Cong et al. (2024); Su et al. (2024); Rifat et al. (2025). The core difference in our setting lies in the adversary goal, i.e., class-wise vs. sample-wise targeting. We next formalize the system model, adversary goals, and adversary capabilities.

3.1 System Model

Let $\mathcal{X} \subset \mathbb{R}^D$ denote the input space and $\mathcal{Y} = \{1, \ldots, C\}$ the label space. We consider a multi-class classification service that deploys a TTA model $f_{\theta_t} : \mathcal{X} \to \mathbb{R}^C$, where θ_t represents the model parameter at adaptation step t. θ_0 denotes the original pre-trained parameters prior to any adaptation. During deployment, each incoming test batches $X^{(t)}$ comprises of three disjoint subsets: benign inputs $X_B^{(t)} := \{x_{B,i}^{(t)}\}_{i=1}^{N_B}$, victim inputs $X_V^{(t)} := \{x_{V,i}^{(t)}\}_{i=1}^{N_V}$ (some of which may carry the attacker's trigger), and attacker-controlled inputs $X_A^{(t)} := \{x_{A,i}^{(t)}\}_{i=1}^{N_A}$. These subsets identify the roles of inputs in the attack, rather than requiring distinct accounts or devices. All inputs lie in the same input space $\mathcal{X}$ and label space $\mathcal{Y}$, with $X^{(t)} = X_B^{(t)} \cup X_V^{(t)} \cup X_A^{(t)}$. For every batch $X^{(t)}$, the service internally updates the model parameters via a TTA mechanism $\mathcal{A}$, i.e., $\theta_{t+1} = \mathcal{A}(\theta_t, X^{(t)})$, and subsequently outputs predictions $f_{\theta_{t+1}}(X^{(t)})$, without access to ground truth labels Su et al. (2024); Rifat et al. (2025).

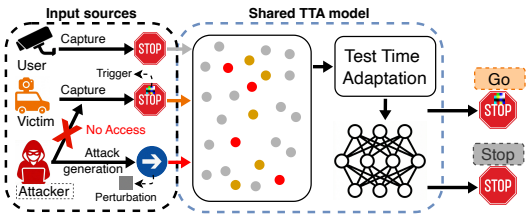


Figure 2: Attack setting under shared TTA. The attack is selective, where it forces trigger-carrying victim inputs (red stream) to a target label while maintaining benign-like prediction behavior on non-triggered inputs (gray stream). Trigger injection occurs upstream or in the physical environment, and is naturally captured by victim inputs. The attacker has no access to victim data.

3.2 Adversary Goals

The objective of the *sample-wise targeted* attack on TTA is twofold. First, it seeks to induce **targeted misclassification on a selected subset**, ensuring that a predefined portion of victim inputs $\hat{X}_v^{(t)} \subseteq X_V^{(t)}$ is mapped to a specific target label $y_{\text{tgt}} \in \mathcal{Y}$ after adaptation. Second, since direct accuracy monitoring is infeasible due to the absence of ground truth labels during deployment of TTA, the attack prioritizes **stealth via label-free prediction consistency**, requiring that the adapted model maintains benign-like prediction behavior on all remaining non-targeted inputs, i.e., $X_B^{(t)} \cup \left(X_V^{(t)} \setminus \hat{X}_v^{(t)}\right)$, thereby rendering the attack difficult to detect.

Trigger-defined victim subset. Rather than selecting victims by their semantic class, we assume the attacker defines $\hat{X}_v^{(t)}$ via the presence of a *trigger*, adopting the formalism from backdoor attacks Gu et al. (2017); Chen et al. (2017). Let $T : \mathcal{X} \to \mathcal{X}$ denote a trigger operator (e.g., a localized patch or a sinusoidal pattern), and define the triggered victim set $\hat{X}_v^{(t)} := \{\, T(x) \mid x \in X_v^{(t)} \,\}$, where $X_v^{(t)}$ is the selected untriggered set. Importantly, trigger injection occurs *outside* the shared TTA pipeline: it may be physical (e.g., a patch on an object captured by sensors) or upstream/digital (e.g., a marker in attacker-provided content processed by a victim application). In either case, the attacker controls the trigger pattern without observing or accessing the resulting victim inputs (Figure 2). The attack succeeds if, after the TTA update, all triggered victim inputs are forced to the same target label, i.e., $\forall x \in \hat{X}_v^{(t)} : f_{\theta_{t+1}}(x) = y_{\text{tgt}}$, while prediction behavior on non-triggered inputs remains close to the benign baseline.

Key distinction from class-wise targeted attacks. Prior class-wise targeted attacks couple the misbehavior to a specific source class (e.g., all "stop" sign → "go" sign), which inevitably distorts the global output into a degenerate probability spike. In contrast, our victim set is trigger-defined and thus independent of the underlying semantic class. This selectivity gives the attacker greater flexibility (e.g., affecting samples from multiple classes simultaneously) while allowing the adapted model to maintain benign-like prediction behavior on non-triggered inputs. In the absence of labels, this makes the attack harder to distinguish from natural adaptation.

3.3 Adversary Capabilities

The attacker can submit its own inputs to the shared TTA service but cannot observe or modify other users' submitted inputs. The attack operates when attacker-controlled and victim inputs are processed in the same adaptation batch. We evaluate varying attacker participation in Appendix I.1. We assume a *grey-box* setting, following realistic attack scenarios in Su et al. (2024); Rifat et al. (2025): the attacker has access to the architecture and parameters of the initial deployed model f_{θ_0} (e.g., a publicly released checkpoint). The attacker may run f_{θ_0} locally to craft adversarial examples set $\hat{X}_A^{(t)} := X_A^{(t)} + \delta_{N_A}^{(t)}$, where $\delta_{N_A}^{(t)}$ is a set of perturbations, each defined in $\mathcal{X}$, and constrained within the ℓ_∞ budget ϵ, i.e., $||\delta||_\infty \leq \epsilon$. However, the attacker does not know the deployed TTA algorithm $\mathcal{A}$, its hyperparameters, the sequence of adapted parameters $\{\theta_t\}_{t\geq 1}$, or any specific inputs from other users $X_B^{(t)} \cup X_V^{(t)}$.

4 Attack Design

An overview of our attack generation pipeline is shown in Figure 3. We simulate deployment-time TTA by repeatedly sampling batch from attacker's available data to emulate victim inputs (by applying trigger $T(\cdot)$), support inputs (by adding learnable perturbation δ), and benign inputs. The attack is optimized in a meta-learning manner by jointly optimizing the attack objective and a newly introduced stealth objective (Section 4.1). Crucially, to alleviate the strong disagreement between objectives, we avoid symmetric combination and instead introduce a novel priority-aware gradient alignment that explicitly preserves the attack direction while maintaining stealth, with theoretical descent analysis (Section 4.2). The resulting aligned gradient is used to update the attack parameters for the next meta-optimization step. After optimization, the learned perturbations are applied to the attacker's inputs and used to deploy the attack.

4.1 Attack Generation via Meta-Learning

Under the threat model in Section 3, we now detail how the attacker constructs concrete attack inputs under the grey-box setting. Since the attacker has access only to a local surrogate model f_{θ_0} and

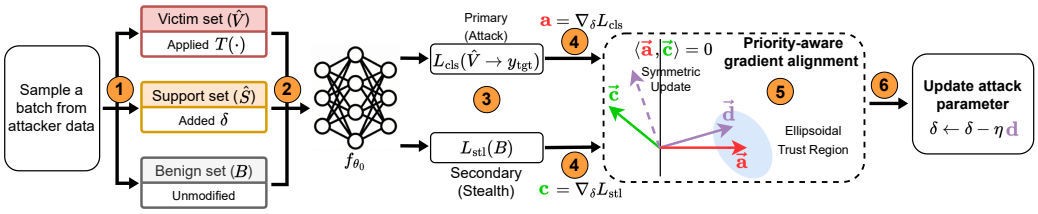


Figure 3: Overview of the attack generation using the proposed meta-learning workflow, where misalignment between attack and stealth gradients is resolved via priority-aware alignment.

attacker-controlled data, the perturbations must generalize beyond attacker data. To this end, we introduce a meta-learning-based attack generation framework to learn the perturbations. Specifically, we parameterize the attack by a set of perturbations δ_{N_A}, omitting the superscript t for notational simplicity without loss of generality. We then construct multiple *tasks*, as described below. The same δ_{N_A} is shared across many such tasks and will be optimized to work robustly across them.

Task construction. Each optimization iteration samples a batch from attacker data X_A by randomly partitioning X_A into three disjoint roles: ***(i)*** a victim set $\hat{V}$, to which the trigger $T(\cdot)$ is applied and which should be classified as the target label y_{tgt}; ***(ii)*** a support set $\hat{S}$, which contains N_S $(< N_A)$ samples carrying the corresponding learnable perturbation; and ***(iii)*** a benign set B, which remains unmodified and is used to preserve stealth behavior. To improve generalization, the mixture ratios of the three sets are randomly sampled for each task from a uniform distribution. Across many randomly sampled tasks, all attacker samples are selected as support samples at different iterations, ensuring that the perturbations are optimized for the entire X_A.

Objectives. For each task, we define an attack loss that pushes triggered victim samples in $\hat{V}$ toward y_{tgt}. This is implemented via the standard cross-entropy loss $L_{\text{cls}} = \text{CE}\big(f_{\theta_0}([\hat{V}, \hat{S}, B])\big|_{\hat{V}}, y_{\text{tgt}}\big)$, where $\big|_{\hat{V}}$ restricts the batch outputs to the victim part only. This objective works via cross-sample coupling within a batch (see Appendix B). In addition, the attacker must preserve distributional stealth on non-victim inputs. Since TTA typically operates in a batch-wise mode, modifications in $\hat{V}$ and $\hat{S}$ inevitably alter the shared batch statistics used for B. To address this problem, we encourage prediction consistency between clean and perturbed versions of the same samples. Specifically, we propose a new distributional stealth loss that penalizes deviations in the model's output on benign inputs B caused by the presence of $\hat{V}$ and $\hat{S}$ in the same batch. Formally,

$$L_{\text{stl}} = \text{KL}\big(p_{\theta_0}([V, S, B])\big|_B \,||\, p_{\theta_0}([\hat{V}, \hat{S}, B])\big|_B\big),$$

where $p_{\theta_0}(\cdot) = \text{softmax}\,(f_{\theta_0}(\cdot))$, KL is the Kullback-Leibler divergence, V denotes the victim inputs without the trigger $T(\cdot)$, S denotes the support inputs without the perturbations, and $\big|_B$ restricts the batch outputs to the benign part only. L_{stl} ensures that the benign predictions remain stable despite the shift in batch statistics caused by the attack.

Meta-learning update. Within one iteration, gradients from a predefined number of batches are aggregated to obtain batch-level estimates for the attack and stealth objectives. Based on these gradients, we construct an aligned gradient update direction d^k (Section 4.2), and update the perturbations via an ℓ_∞-constrained projected gradient step:

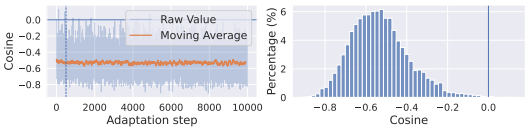
$$\delta_{N_A}^{k+1} = \Pi_{\mathcal{B}_\infty(\epsilon)}\big(\delta_{N_A}^k - \eta d^k\big), \tag{1}$$

where k is the current step of the meta learning, $\eta > 0$ is the step size, $\mathcal{B}_\infty(\epsilon)$ is the ℓ_∞ ball of radius ϵ, and $\Pi_{\mathcal{B}_\infty(\epsilon)}$ denotes element-wise projection onto $\mathcal{B}_\infty(\epsilon)$. Repeating this procedure over multiple iterations yields perturbations that have been optimized across many randomly constructed tasks, i.e., in a meta-learning fashion. At deployment, the attacker applies the learned δ_{N_A} to its own queries X_A to create $\hat{X}_A := X_A + \delta_{N_A}$ before sending them to the TTA service.

4.2 Attack Success and Stealth Misalignment

In this section, we describe how the aligned direction d^k is constructed. For brevity, we omit the superscript k. Empirically, we observe that the gradient of L_{cls} and the gradient of L_{stl} with respect to the perturbation are strongly antagonistic. As shown in Figure 4, the cosine similarity between the two gradients is negative in nearly all iterations (99.9%), placing our optimization in the conflicting-gradients regime Yu et al. (2020).

Because a failed attack has zero utility regardless of its stealthiness, standard symmetric multi-task learning Sener & Koltun (2018); Liu et al. (2021), which treats all objectives equally, is unsuitable. Instead, we propose a priority-asymmetric objective where attack success is primary, and stealth is a secondary objective. Our update rule aligns closely with the attack gradient while penalizing movement toward stealth gradient that deviates from it (Section 4.2.1), which theoretically guarantees descent for the unprojected update even under severe gradient disagreement (Section 4.2.2).



(a) Cosine across steps. (b) Frequency of cosine.

Figure 4: (a) Cosine similarity between gradients of L_{cls} and L_{stl} across attack optimization steps, where the vertical dashed line marks the 500 steps used in the experiments. (b) Frequency distribution of cosine similarity values. Nearly all steps (99.9%) exhibit negative cosine similarity, indicating persistent gradient misalignment.

4.2.1 Priority-aware Gradient Alignment

Let $a := \nabla_\delta L_{\text{cls}}(\delta_{N_A})$, $c := \nabla_\delta L_{\text{stl}}(\delta_{N_A})$ denote the gradient of L_{cls} and L_{stl}, respectively. To encode our attack-first preference, we formulate the update as a trust-region problem centered at the attack gradient a under an ellipsoidal norm. Concretely, we seek an update direction d that ***(i)*** stays close to a, ensuring the update remains attack-driven, and ***(ii)*** incorporates c while penalizing movement toward directions that increasingly deviate from a.

To encode ***(i)***, we additionally require that d stays in a small neighbourhood of a. We measure distance using a Mahalanobis norm $\|v\|_M := \sqrt{v^\top M v}$ ($M \succ 0$), and enforce a trust region of the form $\|d - a\|_M \leq \rho$. This defines an ellipsoid centred at a with radius $\rho > 0$ in the M-metric, so any admissible d remains close to a.

To encode ***(ii)***, we choose the metric M so that movement toward stealth directions that deviate from a is penalized more strongly. Specifically, we set $M = I + \lambda u u^\top$, where u denotes the deviation from a toward c, and $\lambda \geq 0$ increases with the disagreement between a and c. Intuitively, this choice leaves all directions orthogonal to u unchanged, but makes movement along the deviation direction u (i.e., deviating from a) more expensive: along u, the admissible Euclidean radius shrinks from ρ to $\rho/\sqrt{1+\lambda}$. A precise construction of u and λ is given in Appendix C.

We consider convex combinations of the attack and stealth gradients, i.e., $g_w = wa + (1-w)c$ for $w \in [0, 1]$, which interpolates between purely attack-driven ($w = 1$) and stealth-driven ($w = 0$) directions. For an update $-\eta d$ (Eq. (1)), the first-order approximation $L(\delta - \eta d) \approx L(\delta) - \eta \nabla L^\top d$ shows that a larger inner product $\nabla L^\top d$ corresponds to a greater predicted loss reduction. However, a fixed weighting can mask a decline in one objective through improvement in the other. We therefore choose d to maximize the worst-case first-order improvement over all g_w, while prioritizing the attack objective through the trust region centered at a. Setting the trust-region radius $\rho := \gamma \|a\|_2$ ($\gamma > 0$), gives the primal problem:

$$\max_{d \in \mathbb{R}^d} \min_{w \in [0,1]} g_w^\top d \quad \text{s.t.} \quad \|d - a\|_M \leq \rho. \tag{2}$$

We solve Eq. (2) using Lagrangian Boyd & Vandenberghe (2004), detailed in Appendix D. For a fixed w, the optimal update is:

$$d^\star(w) = a + \rho \frac{M^{-1} g_w}{\|g_w\|_{M^{-1}}}, \tag{3}$$

where $\|g_w\|_{M^{-1}} := \sqrt{g_w^\top M^{-1} g_w}$. We then have the following one-dimensional convex program:

$$\min_{w \in [0,1]} a^\top g_w + \rho \|g_w\|_{M^{-1}}. \tag{4}$$

We solve Eq. (4) over $w \in [0, 1]$ using a scalar convex optimizer (e.g., Brent's method), obtain $w^\star$, and finally set $d^\star = d^\star(w^\star)$ via Eq. (3).

4.2.2 Theoretical Understanding

In this section, we characterize the effect of one update step on the perturbation δ_{N_A} when using the ellipsoidal direction $d^\star$ from Eq. (3). We assume L_{cls} is L-smooth in δ_{N_A}. The following theorem summarizes the behavior of one step along $d^\star$, with the proof deferred to Appendix E.

Theorem 4.1 (Alignment-adaptive descent). *Let L_{cls} be L-smooth, and let $d^\star$ be the ellipsoidal update direction with radius $\rho = \gamma\|a\|_2$ for $\gamma \in (0,1)$. Then there exists a factor $\xi \in \left[\frac{1}{\sqrt{1+\lambda}}, 1\right]$ that depends on how strongly $d^\star$ points into the deviation direction u such that, for the step size $\eta := \frac{1-\gamma\xi}{L(1+\gamma\xi)^2}$, the update $\delta^+_{N_A} = \delta_{N_A} - \eta\, d^\star$ satisfies*

$$L_{\mathrm{cls}}(\delta^+_{N_A}) \le L_{\mathrm{cls}}(\delta_{N_A}) - \frac{(1-\gamma\xi)^2}{2L(1+\gamma\xi)^2}\,\|a\|_2^2. \tag{5}$$

Theorem 4.1 has two key implications. ***(i)*** In the isotropic case, $\lambda = 0$ and $\xi = 1$, recovering the standard Euclidean trust-region update ($M = I$). ***(ii)*** When c deviates from a, $\lambda > 0$ can yield $\xi < 1$, resulting in a strictly stronger descent bound. Thus, the ellipsoidal geometry adapts to gradient disagreement by discouraging movement toward misaligned stealth directions, while reducing to the isotropic update when no disagreement is present.

Theorem 4.1 analyzes the unprojected update, whereas our implementation projects the perturbation onto the feasible ℓ_∞ ball (Eq. 1). In Appendix G, we derive a sufficient condition for descent under projection and empirically show that the projected update fails to decrease the attack objective in only a small fraction of iterations, with negligible impact on attack performance.

5 Experiment

In this section, we first describe the experimental setup used to evaluate our proposed attack in Section 5.1. We then present the main results in Section 5.2, followed by an ablation study in Section 5.3. Finally, in Section 5.4, we examine existing defenses' performance against our proposed attack. More experiments are presented in Appendix I.

5.1 Experimental Setting

Datasets and backbones. We evaluate the attack on standard benchmarks for TTA, namely CIFAR10-C, CIFAR100-C, and ImageNet-C. These datasets are constructed by applying a diverse set of synthetic corruptions to clean images Hendrycks & Dietterich (2019b), thereby simulating realistic distribution shifts. Following common practice Su et al. (2024); Rifat et al. (2025), for CIFAR10-C and CIFAR100-C, we adopt ResNet-32 as the backbone and evaluate performance across all 15 corruption types and all five severity levels; for ImageNet-C, we use ResNet-50 and report results at corruption severity level 3. We provide cross-dataset backbone results in Appendix I.6.

Baselines and TTA methods. We compare our attack against two class-wise targeted TTA attacks, FCA Rifat et al. (2025) and RTTDP Su et al. (2024), under the threat model presented in Section 3. Following prior work Wu et al. (2023); Cong et al. (2024); Su et al. (2024); Rifat et al. (2025), we evaluate attacks under five BN-affine TTA methods: TENT Wang et al. (2020), RPL Rusak et al. (2021), EATA Niu et al. (2022), SAR Niu et al. (2023), and DeYO Lee et al. (2024). More details on each TTA method are presented in Appendix H.

Evaluation protocol. We randomly select 5% from distinct classes as triggered victims. We evaluate patch Gu et al. (2017) and SIG Barni et al. (2019) triggers (Figure 8). Results with an imperceptible trigger are reported in Appendix I.2. We measure attack effectiveness by attack success rate (ASR) and stealth by entropy distribution drift (EDD) and augmentation disagreement inconsistency (ADI), where lower values indicate greater stealth. EDD captures shifts in the predictive-entropy distribution of non-targeted inputs, while ADI captures changes in their augmentation consistency relative to benign adaptation. We set $\epsilon = 16/255$ Su et al. (2024). Further implementation details and sensitivity analyses are provided in Appendices H and I.5, respectively.

5.2 Main Results

Table 2 compares our attack with FCA Rifat et al. (2025) and RTTDP Su et al. (2024), both adapted to our threat model. Our method consistently achieves the highest ASR across all datasets, triggers, and TTA methods, frequently exceeding 80% on CIFAR and 95% on ImageNet-C. Adapting RTTDP and FCA to our meta-learning framework substantially improves their performance over their original formulations (values in parentheses), confirming that explicitly optimizing for the sample-wise threat model is important. Nevertheless, both baselines remain considerably weaker than our method,

Table 2: ASR, Mean EDD, and Mean ADI of attacks under various TTA methods. The best results are highlighted in **bold**. ⋆ denotes methods adapted to our threat model (see Appendix H for details), and values in parentheses report the original results without adaptation.

Dataset	Attack	SIG Trigger					Patch Trigger					Mean EDD	Mean ADI
		TENT	RPL	EATA	SAR	DeYO	TENT	RPL	EATA	SAR	DeYO		
CIFAR-10-C (ResNet32)	No attack	2.85	3.07	3.10	3.04	2.85	2.77	3.16	3.02	3.64	2.97	-	-
	RTTDP⋆	25.39 (7.56)	29.65 (8.17)	22.80 (7.53)	37.70 (8.30)	26.10 (7.51)	32.72 (6.55)	42.54 (7.57)	35.98 (6.57)	60.58 (8.95)	35.47 (6.76)	**0.16**	**18.08**
	FCA⋆	54.80 (6.67)	59.62 (7.27)	51.99 (6.76)	63.26 (7.29)	54.09 (6.52)	50.77 (5.43)	58.36 (6.40)	50.15 (5.68)	58.97 (7.45)	51.74 (5.74)	0.60	28.06
	Ours	**89.59**	**93.17**	**90.21**	**95.90**	**91.15**	**79.28**	**86.78**	**80.34**	**93.33**	**82.48**	0.23	21.14
CIFAR-100-C (ResNet32)	No attack	0.07	0.09	0.11	0.09	0.09	0.87	0.91	0.96	0.91	0.82	-	-
	RTTDP⋆	30.17 (0.15)	39.31 (0.12)	37.35 (0.14)	41.92 (0.15)	31.11 (0.13)	64.80 (0.40)	76.16 (0.40)	75.30 (0.39)	77.84 (0.39)	69.44 (0.35)	**0.10**	**14.39**
	FCA⋆	45.81 (0.15)	49.13 (0.13)	51.22 (0.14)	54.49 (0.15)	43.65 (0.14)	73.03 (0.40)	76.88 (0.40)	80.61 (0.40)	80.84 (0.40)	70.95 (0.36)	0.62	19.09
	Ours	**66.93**	**76.17**	**76.42**	**81.49**	**70.66**	**88.04**	**91.82**	**91.83**	**93.47**	**88.97**	0.16	15.36
ImageNet-C (ResNet50)	No attack	0.00	0.00	0.09	0.00	0.09	0.11	0.11	0.05	0.11	0.05	-	-
	RTTDP⋆	49.50 (0.00)	54.08 (0.00)	37.14 (0.00)	53.54 (0.00)	41.67 (0.00)	32.67 (0.00)	31.63 (0.00)	20.95 (0.00)	32.32 (0.00)	21.30 (0.00)	**0.15**	**17.95**
	FCA⋆	92.08 (0.00)	94.90 (0.00)	86.67 (0.00)	95.96 (0.00)	87.04 (0.00)	84.16 (0.00)	85.71 (0.00)	73.33 (0.00)	81.82 (0.00)	71.96 (0.00)	0.60	31.07
	Ours	**96.79**	**96.99**	**95.22**	**96.98**	**95.49**	**92.29**	**92.49**	**88.48**	**92.28**	**89.21**	0.24	22.29

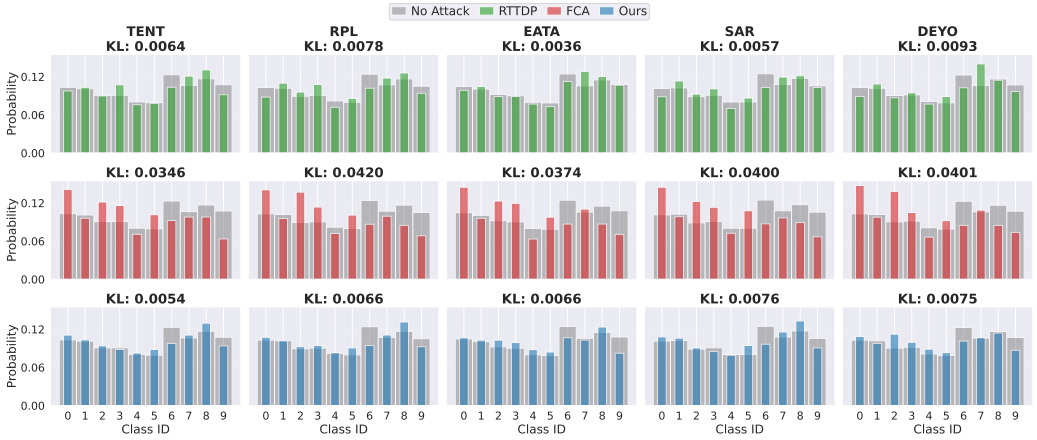


Figure 5: Comparison of label distribution shifts induced by RTTDP, FCA, and ours on CIFAR-10-C.

indicating that the proposed attack objective and priority-aware gradient alignment are critical for achieving high attack success under this setting.

Regarding stealthiness, across all three datasets, our method consistently achieves lower mean EDD and mean ADI than FCA while simultaneously obtaining substantially higher ASR. RTTDP achieves the lowest EDD and ADI, but this comes at the cost of much weaker attack effectiveness. These results indicate that our method achieves a favorable balance between attack success and label-free stealth, rather than improving ASR simply by inducing large changes in model behavior. Figure 5 provides complementary evidence using the output-label distribution and KL divergence between the attacked and benign distributions. Class-wise attacks such as FCA produce noticeable probability spikes and larger distribution shifts (KL > 0.03), whereas our trigger-defined attack remains close to the no-attack distribution (KL $= 0.006$), further demonstrating that high attack success can be achieved without creating an obvious concentration in the predicted target class.

5.3 Ablation Study

Table 3 analyzes the contribution of each component in our framework. Optimizing L_{cls} alone already yields a strong ASR of 87.43%, but produces clear probability spikes and substantially larger label-distribution shifts than our method (Appendix I.4). Naively combining the attack and stealth objectives ($L_{\text{cls}} + L_{\text{stl}}$) achieves the lowest EDD and ADI, but reduces ASR to only 37.65%. This sharp trade-off is consistent with the severe gradient misalignment observed in Figure 4, and shows that simply emphasizing stealth can render the attack ineffective. Standard conflict-resolution methods do not adequately resolve this asymmetry. PCGrad Yu et al. (2020) and CAGrad Liu et al. (2021) achieve ASRs of only 38.55% and 48.66%, respectively, as their symmetric treatment of the two objectives continues to sacrifice the primary attack objective. The Euclidean trust region (isotropic approach) better preserves attack effectiveness, reaching 79.88% ASR, but still falls substantially below the attack-only baseline. In contrast, our priority-aware ellipsoidal alignment achieves the highest ASR of 91.03%, exceeding even L_{cls} alone and outperforming the isotropic trust region by more than 11 percentage points. While its EDD and ADI remain comparable to the attack-only setting, the label-distribution analysis in Appendix I.4 shows substantially improved distributional

Table 3: ASR, EDD, and ADI of attacks averaged across various TTA methods, SIG trigger, CIFAR-10-C dataset.

Method	ASR	EDD	ADI
L_{cls}	87.43	0.245	20.12
$L_{\text{cls}} + L_{\text{stl}}$	37.65	**0.078**	**5.81**
W/ PCGrad	38.55	0.153	10.17
W/ CAGrad	48.66	0.165	10.96
W/ Euclidean trust region	79.88	0.218	16.64
Ours	**91.03**	0.225	20.19

Table 4: ASR of our attack under CoTTA defense Wang et al. (2022).

Trigger	No attack	Ours	Ours w/ Aug
SIG	3.14	13.60	**41.44**
Patch	2.48	12.95	**41.59**

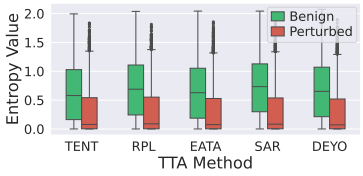


Figure 6: Entropy of benign and perturbed samples.

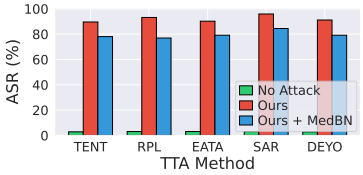


Figure 7: ASR of our attack under MedBN defense Park et al. (2024).

behavior. Overall, the ablation confirms the motivation of our formulation: rather than seeking a symmetric compromise that substantially weakens the attack, our method preserves the primary attack objective while incorporating stealth as a secondary objective.

5.4 Existing Defenses Performance

Sample filtering. A potential defense against our attack is sample filtering, which discards high-entropy samples to prevent noisy data from degrading TTA performance Niu et al. (2022; 2023); Gong et al. (2023). However, as illustrated in Figure 6, our attack generates samples with entropy values even lower than those of benign samples, as the attack optimization minimizes uncertainty to ensure target success. Thus, entropy-based filtering fails to remove our adversarial inputs. This ineffectiveness is empirically confirmed by the high ASR achieved against EATA Niu et al. (2022) and SAR Niu et al. (2023) in Table 2.

TTA-specific defense. Figure 7 reports performance against MedBN defense Park et al. (2024), which replaces standard BN with Median BN for robust batch statistics. Our attack still achieves high ASR ($\approx 80\%$). MedBN primarily hinders optimization through gradient masking: the derivative of the median is zero for non-median samples, obstructing gradient flow. However, under our grey-box threat model, the attacker crafts perturbations using the original pre-deployment model equipped with standard BN, which provides a smooth gradient landscape for optimization. Since shifts in batch statistics induced under BN can transfer to the median-based statistics, the resulting perturbations remain effective against MedBN at test time.

Exponential moving average (EMA). Another potential defense involves using an EMA of the model weights during adaptation, rather than instantaneous updates Wang et al. (2022). This strategy initially mitigates our attack, reducing the ASR to about 13% (Table 4). However, by simply applying data augmentations (e.g., blur, crop, flip) to the adversarial samples, we can recover the ASR to roughly 41%. This occurs because augmentation forces the generation of robust and consistent gradients that accumulate in the EMA buffer, rather than brittle noise that gets averaged out. Notably, this recovery is achieved without explicitly optimizing the attack against the EMA, suggesting that simple heuristics are sufficient to compromise this defense.

6 Conclusion

In this paper, we formalize a sample-wise targeted threat model for TTA, where only trigger-carrying inputs are misclassified while non-triggered inputs maintain benign-like prediction behavior. We propose a meta-learning framework enhanced by a novel priority-aware gradient alignment, which mitigates the misalignment between attack success and stealth, supported by theoretical descent analysis of the attack objective. Our approach significantly outperforms existing baselines, achieving high targeted attack success rates while remaining consistent with benign adaptation under multiple label-free stealth metrics. Additionally, we investigate the attack effectiveness against existing defenses, highlighting security risks in deployment-time adaptation.

AI USE STATEMENT

We used generative AI tools primarily to assist with manuscript writing, including grammar correction, language polishing, improving clarity, and refining the presentation of existing technical content. We also used generative AI to assist with generating and debugging well-defined sections of experimental code for which the intended functionality, inputs, and outputs were specified by the authors. All AI-assisted text and code were reviewed, edited, and verified by the authors before inclusion in the final work.

ETHICS STATEMENT

This work investigates security vulnerabilities in Test-Time Adaptation (TTA) systems and therefore has potential dual-use implications. The proposed attack is developed solely for research purposes and is evaluated in controlled experimental settings using standard benchmark datasets and models; we do not target real-world deployed systems or use sensitive user data. While the methodology could potentially be misused to compromise adaptive systems, exposing such vulnerabilities is important for understanding the security risks of deployment-time learning and for motivating more robust defenses. We therefore present the attack to support responsible security research and the development of safer TTA mechanisms. To the best of our knowledge, this work does not involve human subjects, private data, or other ethical concerns beyond the security and dual-use considerations described above.

REPRODUCIBILITY STATEMENT

We provide detailed experimental settings and implementation details in the main paper and appendices, including the attack configuration, hyperparameters, TTA settings, baseline adaptations, trigger specifications, and evaluation protocols. The official implementation code will be made publicly available upon acceptance.

REFERENCES

Mauro Barni, Kassem Kallas, and Benedetta Tondi. A new backdoor attack in cnns by training set corruption without label poisoning. In *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 101–105. IEEE, 2019.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Dhanajit Brahma and Piyush Rai. A probabilistic framework for lifelong test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3582–3591, 2023.

Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*, pp. 794–803. PMLR, 2018.

Tianshuo Cong, Xinlei He, Yun Shen, and Yang Zhang. Test-time poisoning attacks against test-time adaptation models. In *2024 IEEE Symposium on Security and Privacy (SP)*, pp. 1306–1324. IEEE, 2024.

Bao Gia Doan, Ehsan Abbasnejad, and Damith C Ranasinghe. Februus: Input purification defense against trojan attacks on deep neural network systems. In *Proceedings of the 36th Annual Computer Security Applications Conference*, pp. 897–912, 2020.

Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1625–1634, 2018.

Taesik Gong, Yewon Kim, Taeckyung Lee, Sorn Chottananurak, and Sung-Ju Lee. Sotta: Robust test-time adaptation on noisy data streams. *Advances in Neural Information Processing Systems*, 36:14070–14093, 2023.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.

Jisu Han, Jaemin Na, and Wonjun Hwang. Ranked entropy minimization for continual test-time adaptation. *arXiv preprint arXiv:2505.16441*, 2025.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019a.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019b.

Zixuan Hu, Yichun Hu, Xiaotong Li, Shixiang Tang, and Ling-Yu Duan. Beyond entropy: Region confidence proxy for wild test-time adaptation. *arXiv preprint arXiv:2505.20704*, 2025.

Minguk Jang, Sae-Young Chung, and Hye Won Chung. Test-time adaptation via self-training with nearest neighbor information. *arXiv preprint arXiv:2207.10792*, 2022.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7482–7491, 2018.

Jae-Hong Lee and Joon-Hyuk Chang. Stationary latent weight inference for unreliable observations from online test-time adaptation. In *Forty-first International Conference on Machine Learning*, 2024.

Jonghyun Lee, Dahuin Jung, Saehyung Lee, Junsung Park, Juhyeon Shin, Uiwon Hwang, and Sungroh Yoon. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. *arXiv preprint arXiv:2403.07366*, 2024.

Jungsoo Lee, Debasmit Das, Jaegul Choo, and Sungha Choi. Towards open-set test-time adaptation utilizing the wisdom of crowds in entropy minimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16380–16389, 2023.

Yoonho Lee, Annie S Chen, Fahim Tajwar, Ananya Kumar, Huaxiu Yao, Percy Liang, and Chelsea Finn. Surgical fine-tuning improves adaptation to distribution shifts. *arXiv preprint arXiv:2210.11466*, 2022.

Jingjing Li, Zhiqi Yu, Zhekai Du, Lei Zhu, and Heng Tao Shen. A comprehensive survey on source-free domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8): 5743–5762, 2024.

Yiming Li, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. Backdoor learning: A survey. *IEEE transactions on neural networks and learning systems*, 35(1):5–22, 2022.

Yudong Li, Shigeng Zhang, Weiping Wang, and Hong Song. Backdoor attacks to deep learning models and countermeasures: A survey. *IEEE Open Journal of the Computer Society*, 2023.

Yuezun Li, Yiming Li, Baoyuan Wu, Longkang Li, Ran He, and Siwei Lyu. Invisible backdoor attack with sample-specific triggers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 16463–16472, 2021.

Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, 133(1):31–64, 2025.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In *International conference on machine learning*, pp. 3122–3130. PMLR, 2018.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890, 2021.

Fan Lyu, Kaile Du, Yuyang Li, Hanyu Zhao, Zhang Zhang, Guangcan Liu, and Liang Wang. Variational continual test-time adaptation. *arXiv preprint arXiv:2402.08182*, 2024.

Jing Ma. Improved self-training for test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23701–23710, 2024.

Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.

Anh Nguyen and Anh Tran. Wanet–imperceptible warping-based backdoor attack. *arXiv preprint arXiv:2102.10369*, 2021.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International conference on machine learning*, pp. 16888–16905. PMLR, 2022.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. *arXiv preprint arXiv:2302.12400*, 2023.

Hyejin Park, Jeongyeon Hwang, Sunung Mun, Sangdon Park, and Jungseul Ok. Medbn: Robust test-time adaptation against malicious test samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5997–6007, 2024.

Zhuang Qian, Kaizhu Huang, Qiu-Feng Wang, and Xu-Yao Zhang. A survey of robust adversarial training in pattern recognition: Fundamental, theory, and methodologies. *Pattern Recognition*, 131: 108889, 2022.

Stephan Rabanser, Stephan Günnemann, and Zachary Lipton. Failing loudly: An empirical study of methods for detecting dataset shift. *Advances in Neural Information Processing Systems*, 32, 2019.

Shahriar Rifat, Jonathan Ashdown, Michael J De Lucia, Ananthram Swami, and Francesco Restuccia. On the adversarial vulnerability of label-free test-time adaptation. In *The Thirteenth International Conference on Learning Representations*, 2025.

Evgenia Rusak, Steffen Schneider, George Pachitariu, Luisa Eck, Peter Gehler, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. If your data distribution shifts, use self-learning. *arXiv preprint arXiv:2104.12928*, 2021.

Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 11957–11965, 2020.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.

Samarth Sinha, Peter Gehler, Francesco Locatello, and Bernt Schiele. Test: Test-time self-training under distribution shift. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2759–2769, 2023.

Maurice Sion. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171–176, 1958.

Yongyi Su, Xun Xu, and Kui Jia. Revisiting realistic test-time training: Sequential inference and adaptation by anchored clustering. *Advances in Neural Information Processing Systems*, 35: 17543–17555, 2022.

Yongyi Su, Yushu Li, Nanqing Liu, Kui Jia, Xulei Yang, Chuan-Sheng Foo, and Xun Xu. On the adversarial risk of test time adaptation: An investigation into realistic test-time data poisoning. *arXiv preprint arXiv:2410.04682*, 2024.

Jiaxu Tian and Fan Lyu. Parameter-selective continual test-time adaptation. In *Proceedings of the Asian Conference on Computer Vision*, pp. 1384–1400, 2024.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7201–7211, 2022.

Emily Wenger, Josephine Passananti, Arjun Nitin Bhagoji, Yuanshun Yao, Haitao Zheng, and Ben Y Zhao. Backdoor attacks against deep learning systems in the physical world. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6206–6215, 2021.

Tong Wu, Feiran Jia, Xiangyu Qi, Jiachen T Wang, Vikash Sehwag, Saeed Mahloujifar, and Prateek Mittal. Uncovering adversarial risks of test-time adaptation. *arXiv preprint arXiv:2301.12576*, 2023.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33: 5824–5836, 2020.

Qingyang Zhang, Yatao Bian, Xinke Kong, Peilin Zhao, and Changqing Zhang. Come: Test-time adaptation by conservatively minimizing entropy. *arXiv preprint arXiv:2410.10894*, 2024.

Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE transactions on knowledge and data engineering*, 34(12):5586–5609, 2021.

A Extended Related Works

Test-time Adaptation. TTA updates model parameters at inference time using unlabeled test data to handle distribution shift Liang et al. (2025). Unlike training-time domain adaptation Li et al. (2024), it assumes no access to source data or labels at deployment. TTA methods are commonly grouped by the adaptation signal: (i) *entropy/confidence-based* updates that minimize prediction entropy Wang et al. (2020), improved via sample filtering Lee et al. (2023); Lee & Chang (2024); Lee et al. (2024), conservative entropy minimization Zhang et al. (2024); Han et al. (2025), and confidence regularization Hu et al. (2025); and (ii) *self-training/consistency-based* updates using sharpness-aware adaptation Niu et al. (2023), robustness-regularized self-training Rusak et al. (2021); Jang et al. (2022); Ma (2024), or EMA teacher-student guidance Wang et al. (2022); Lyu et al. (2024); Tian & Lyu (2024). Since TTA relies on online updates from the unlabeled test stream, implicitly trusting the test stream, model behavior becomes dependent on the adaptation trajectory, thereby exposing an attack surface. In this work, we study adversaries that exploit this by targeting the adaptation process.

Attacks on Test-time Adaptation. Since TTA updates model parameters from unlabeled test inputs, adversaries can manipulate a subset of the test samples to steer the adaptation trajectory. Existing attacks Wu et al. (2023); Cong et al. (2024); Su et al. (2024); Rifat et al. (2025) mainly demonstrate untargeted degradation of post-adaptation performance. In practice, however, attackers often benefit more from targeted manipulation. Wu et al. (2023) demonstrates targeted attacks in a strong setting (white-box access and access to victim data), and subsequent works relax these assumptions, including grey-box settings Rifat et al. (2025) and realistic test-time data poisoning without requiring victim-data access Su et al. (2024). However, these attacks are *class-wise targeted*, and thus become unstealthy as they unnaturally concentrate predictions on the target class (Figure 1). In contrast, we study a *sample-wise targeted* threat model: only inputs carrying the attacker's trigger should be redirected to the target label, while non-triggered inputs should maintain a label distribution consistent with the no-attack baseline (Table 1).

Connection to Backdoor Attacks. Our threat model adopts the concept of trigger-activated misbehavior from traditional backdoor attacks Gu et al. (2017); Saha et al. (2020); Li et al. (2022), utilizing triggers that can be realized digitally (e.g., pixel patterns Gu et al. (2017), steganography Li et al. (2021)) or physically (e.g., patches or accessories in the scene Wenger et al. (2021); Eykholt et al. (2018)). However, standard backdoors require compromising the training data (poisoning) or the model weights (trojaning) prior to deployment, which are often infeasible. In contrast, our attack operates strictly at test-time, exploiting the online adaptation mechanism of TTA. This allows an adversary to induce backdoor-like behavior in a clean, pre-trained model on the fly, without ever having accessed the training phase.

Multi-Task Learning and Conflicting Gradients Multi-task learning (MTL) Zhang & Yang (2021) optimizes shared parameters for multiple objectives, but often suffers from gradient interference Yu et al. (2020), where misaligned task gradients lead to unstable training or poor trade-offs. Prior work mitigates this via loss reweighting Kendall et al. (2018); Chen et al. (2018) or multi-objective/gradient-balancing updates Sener & Koltun (2018); Yu et al. (2020); Liu et al. (2021). Our problem is closely related: the attack and stealth objectives may produce gradients that are partially misaligned, or even directly conflicting when their inner product is negative. However, unlike standard MTL formulations that balance tasks symmetrically or approximate Pareto-optimal trade-offs, our setting is priority-asymmetric: maximizing attack success is the primary goal, while maintaining benign-like behavior is a secondary objective. This motivates a priority-aware alignment strategy that keeps the update close to the attack gradient and increasingly penalizes movement toward stealth-gradient directions as their angular disagreement with the attack direction grows, rather than treating both objectives equally as in conventional MTL.

B Causal Link Through Shared BatchNorm Statistics

Fundamentally, our attack exploits the cross-sample dependency in standard BatchNorm-affine TTA methods, consistent with observations in prior works Wu et al. (2023); Cong et al. (2024); Su et al.

(2024); Rifat et al. (2025). During a mixed-batch forward pass, the normalized representation of any victim sample is dynamically coupled to the support set through the shared batch mean and variance.

Concretely, let h_i^c denote the pre-BN activation of sample i in channel c, and let the batch of m samples be formed by the concatenation $[\hat{V}, \hat{S}, B]$, representing the victim, support, and clean sets, respectively. For a BN layer,

$$\mu_c = \frac{1}{m}\sum_{k=1}^{m} h_k^c, \qquad (\sigma_c^2) = \frac{1}{m}\sum_{k=1}^{m}(h_k^c - \mu_c)^2,$$

and the normalized activation of a victim sample $i \in \hat{V}$ is $z_i^c = (h_i^c - \mu_c)/\sqrt{\sigma_c^2 + \epsilon}$. Although the attack loss L_{cls} is evaluated only on the triggered victim samples, its gradient with respect to a support perturbation δ_j on $j \in \hat{S}$ is non-zero because δ_j changes the shared BN statistics:

$$\frac{\partial L_{\text{cls}}}{\partial \delta_j} = \sum_{i \in \hat{V}, c} \frac{\partial L_{\text{cls}}}{\partial z_i^c} \left(\frac{\partial z_i^c}{\partial \mu_c}\frac{\partial \mu_c}{\partial h_j^c} + \frac{\partial z_i^c}{\partial \sigma_c^2}\frac{\partial \sigma_c^2}{\partial h_j^c} \right) \frac{\partial h_j^c}{\partial \delta_j}.$$

Through this cross-sample gradient flow, the support perturbations are learned to steer the shared BN normalization in a direction that makes the triggered victim samples more likely to be classified as the target label.

C Misalignment Metric Construction

We instantiate M to penalize movement toward directions that deviate from the primary attack gradient a. For $\|a\|_2 > 0$ and $\|c\|_2 > 0$, let

$$\hat{a} := \frac{a}{\|a\|_2}, \qquad \hat{c} := \frac{c}{\|c\|_2}, \qquad s := \hat{a}^\top \hat{c},$$

If either gradient has zero norm, we fall back to the attack-only direction. Here, s is the cosine similarity between the attack gradient and the stealth gradient. A larger angular disagreement between a and c indicates that the stealth gradient is less aligned with the primary attack objective.

We define the deviation axis as the direction from the attack gradient toward the stealth gradient:

$$u := \begin{cases} (\hat{c} - \hat{a})/\|\hat{c} - \hat{a}\|_2, & \text{if } \|\hat{c} - \hat{a}\|_2 > \varepsilon, \\ 0, & \text{otherwise,} \end{cases}$$

for a small numerical threshold $\varepsilon > 0$. Next, we set the anisotropy strength $\lambda \geq 0$ as a monotone function of the angular disagreement between a and c:

$$\lambda := \kappa(1 - s),$$

where $\kappa > 0$ is a scalar hyperparameter. Thus, λ is small when c is aligned with a, increases as c moves away from a, and is largest when c points in the opposite direction of a.

The resulting metric is

$$M = I + \lambda\, uu^\top. \tag{6}$$

Since $\lambda \geq 0$, this matrix is positive definite and satisfies $M \succeq I$.

To see its effect, consider a displacement purely along the deviation axis, i.e., $d - a = tu$,

$$\|d - a\|_M^2 = \|tu\|_M^2 = (1 + \lambda)\,\|tu\|_2^2 = (1 + \lambda)\,t^2. \qquad (\text{since } \|u\|_2 = 1)$$

Therefore,

$$\|d - a\|_M = \|tu\|_M \;=\; \sqrt{1 + \lambda}\,|t|.$$

Under the trust-region constraint $\|d - a\|_M \leq \rho$, this implies

$$|t| \leq \frac{\rho}{\sqrt{1 + \lambda}}.$$

By contrast, a Euclidean trust region $\|d - a\|_2 \leq \rho$ would allow $|t| \leq \rho$ along the same direction. Hence, using M shrinks the admissible movement along the deviation axis by a factor of $1/\sqrt{1 + \lambda}$.

This gives a selective penalty: movement in the direction that deviates from the attack gradient toward the stealth gradient is suppressed, while movement orthogonal to u receives no additional penalty beyond the Euclidean term.

D SOLVING ELLIPSOIDAL TRUST-REGION PROBLEM

Setting the trust-region radius $\rho := \gamma\|a\|_2$ ($\gamma > 0$), we have the primal problem in Eq. (2):

$$\max_{d\in\mathbb{R}^d} \min_{w\in[0,1]} g_w^\top d \quad \text{s.t.} \quad \|d-a\|_M \le \rho.$$

The objective is linear in d and g_w is affine in w, while the feasible set in d is convex. Hence, the inner-max outer-min problem satisfies the conditions of Sion's minimax theorem Sion (1958), and we may swap the order of $\max$ and $\min$:

$$\min_{w\in[0,1]} \max_{d:\|d-a\|_M\le\rho} g_w^\top d. \tag{7}$$

For a fixed w, the inner problem in Eq. (7) is a problem of maximizing a linear functional $g_w^\top d$ over an ellipsoid. Its solution can be obtained in closed form via the Lagrangian Boyd & Vandenberghe (2004). Consider:

$$\mathcal{L}(d,\mu) = g_w^\top d - \mu\big((d-a)^\top M(d-a) - \rho^2\big), \qquad \mu \ge 0.$$

Stationarity with respect to d gives:

$$\nabla_d \mathcal{L} = g_w - 2\mu M(d-a) = 0 \quad \Rightarrow \quad d = a + \frac{1}{2\mu} M^{-1} g_w.$$

Since the objective is linear and the feasible set is bounded, the optimum lies on the boundary of the ellipsoid, so the constraint is active at $d^\star$, and we have

$$\|d^\star - a\|_M^2 = \rho^2 \quad \Longleftrightarrow \quad \left\| \frac{1}{2\mu^\star} M^{-1} g_w \right\|_M^2 = \rho^2.$$

Using $\|v\|_M^2 = v^\top M v$ and the definition of the dual norm $\|g_w\|_{M^{-1}} := \sqrt{g_w^\top M^{-1} g_w}$, this yields

$$\frac{1}{4(\mu^\star)^2}\, g_w^\top M^{-1} g_w = \rho^2 \quad \Rightarrow \quad \mu^\star = \frac{1}{2\rho}\, \|g_w\|_{M^{-1}}.$$

Substituting $\mu^\star$ back into the expression for d gives the optimal update for fixed w:

$$d^\star(w) = a + \rho\, \frac{M^{-1} g_w}{\|g_w\|_{M^{-1}}}.$$

Note that if $\|g_{w^\star}\|_{M^{-1}} \le \varepsilon$, we fall back to the attack-only direction $d^\star = a$. This avoids the degenerate case where the normalized trust-region direction is undefined.

E PROOF OF THEOREM 4.1

Unit M-norm. For any $w \in [0,1]$, the solution of the inner trust-region problem is

$$d^\star(w) = a + \rho p_w, \qquad p_w := \frac{M^{-1} g_w}{\|g_w\|_{M^{-1}}}.$$

Let $w^\star$ be a minimizer of Eq. (4), and define

$$p^\star := p_{w^\star}, \qquad d^\star := d^\star(w^\star) = a + \rho p^\star.$$

$p^\star$ has unit M-norm:

$$\|p^\star\|_M^2 = (p^\star)^\top M p^\star = \frac{g_{w^\star}^\top M^{-1} M M^{-1} g_{w^\star}}{g_{w^\star}^\top M^{-1} g_{w^\star}} = 1.$$

Bounds on ξ. Define $\xi := \|p^\star\|_2$. Since $M \succeq I$, we have

$$\xi = \|p^\star\|_2 \le \|p^\star\|_M = 1.$$

Since $M = I + \lambda uu^\top$ with $\|u\|_2 = 1$, for any vector p we can write

$$p_u := (u^\top p)u, \qquad p_{\text{rest}} := p - p_u, \qquad u^\top p_{\text{rest}} = 0.$$

Then

$$\|p\|_M^2 = p^\top M p = \|p_{\text{rest}}\|_2^2 + (1+\lambda)\|p_u\|_2^2.$$

Applying this to $p = p^\star$ and using $\|p^\star\|_M = 1$, we get

$$\begin{aligned}
\|p^\star_{\text{rest}}\|_2^2 + (1+\lambda)\|p^\star_u\|_2^2 &\le (1+\lambda)\left(\|p^\star_{\text{rest}}\|_2^2 + \|p^\star_u\|_2^2\right), \\
\iff \|p^\star\|_M^2 = 1 &\le (1+\lambda)\|p^\star\|_2^2, \\
\iff \frac{1}{\sqrt{1+\lambda}} &\le \|p^\star\|_2, \\
\iff \frac{1}{\sqrt{1+\lambda}} &\le \xi \le 1.
\end{aligned}$$

Descent bound. Using $d^\star = a + \rho p^\star$ and $\rho = \gamma\|a\|_2$, we obtain

$$\begin{aligned}
a^\top d^\star &= a^\top a + \rho a^\top p^\star = \|a\|_2^2 + \rho a^\top p^\star \\
&\ge \|a\|_2^2 - \rho\|a\|_2\|p^\star\|_2 \qquad \text{(Cauchy-Schwarz and } -\|x\|\|y\| \le x^\top y) \\
&= \|a\|_2^2 - \rho\|a\|_2\xi \qquad (\|p^\star\|_2 = \xi) \\
&= (1-\gamma\xi)\|a\|_2^2. \qquad (8)
\end{aligned}$$

Thus, when $a \neq 0$, $\xi \le 1$, and $\gamma < 1$, $-d^\star$ is a descent direction for L_{cls}.

Step norm bound. We also have the following norm bound

$$\|d^\star\|_2 = \|a + \rho p^\star\|_2 \le \|a\|_2 + \rho\|p^\star\|_2 = (1+\gamma\xi)\|a\|_2. \qquad (9)$$

Applying smoothness. Since L_{cls} is L-smooth, for any $\eta > 0$, the descent lemma Nesterov (2013) gives

$$L_{\text{cls}}(\delta_{N_A} - \eta d^\star) \le L_{\text{cls}}(\delta_{N_A}) - \eta a^\top d^\star + \frac{L\eta^2}{2}\|d^\star\|_2^2. \qquad (10)$$

Substituting the bounds (8) and (9) into (10) gives

$$L_{\text{cls}}(\delta_{N_A} - \eta d^\star) \le L_{\text{cls}}(\delta_{N_A}) - \eta(1-\gamma\xi)\|a\|_2^2 + \frac{L\eta^2}{2}(1+\gamma\xi)^2\|a\|_2^2.$$

Choose

$$\eta = \frac{1-\gamma\xi}{L(1+\gamma\xi)^2}.$$

Then

$$\begin{aligned}
L_{\text{cls}}(\delta_{N_A} - \eta d^\star) &\le L_{\text{cls}}(\delta_{N_A}) \;-\; \eta(1-\gamma\xi)\,\|a\|_2^2 \;+\; \frac{\eta(1-\gamma\xi)}{2}\,\|a\|_2^2 \\
&= L_{\text{cls}}(\delta_{N_A}) \;-\; \frac{\eta(1-\gamma\xi)}{2}\,\|a\|_2^2 \\
&= L_{\text{cls}}(\delta_{N_A}) \;-\; \frac{(1-\gamma\xi)^2}{2L\,(1+\gamma\xi)^2}\,\|a\|_2^2.
\end{aligned}$$

This completes the proof. □

F ATTACK IMPLEMENTATION

Algorithm 1 summarizes the meta-learning procedure used to generate the attack. Starting from an initial zero perturbations δ_{N_A}, we run K outer iterations. At each iteration, we sample T batches from the attacker's data X_A, where each batch includes three components as described in Section 4.1. For each batch, use the base model f_{θ_0} to evaluate the attack loss L^{τ}_{cls} and the distributional stealth loss L^{τ}_{stl} under the current perturbation. The corresponding gradients with respect to δ_{N_A} are accumulated and averaged across batches to obtain batch-level attack and stealth gradients a and c. These aggregated gradients are then fed into the ELLIPSOIDALDIRECTION, which solves the ellipsoidal trust-region problem from Eqs. (2)–(4) and returns an aligned direction d^k that prioritizes the attack objective while penalizing movement toward misaligned stealth directions. It is important to note that, although we write $M = I + \lambda uu^\top$, we do not instantiate the full $F \times F$ matrix, where F is the flattened perturbation dimension. Since M is a rank-one perturbation of the identity, we compute $M^{-1}g_w = g_w - \frac{\lambda}{1+\lambda}u(u^\top g_w)$, so each trust-region evaluation only costs $O(F)$ time and memory. Finally, we update δ_{N_A} via an ℓ_∞-projected gradient step along $-d^k$, enforcing the perturbation budget $\|\delta_{N_A}\|_\infty \le \epsilon$. After K iterations, the learned perturbations are added to all attacker inputs to produce the perturbed attacker data $\hat{X}_A = X_A + \delta_{N_A}$ used to steer the TTA process at deployment.

Algorithm 1 Pseudo-code of the attack generation

Input: Model f_{θ_0}; attacker data X_A split into batches; target label y_{tgt}; perturbation budget ϵ; step size η; trust-radius parameter γ; misalignment scaling κ; number of outer iterations K; number of batches per iteration T.
Output: Perturbed data $\hat{X}_A := X_A + \delta_{N_A}$.
Initialize $\delta_{N_A} \leftarrow 0$
for $k = 0, 1, \ldots, K-1$ **do**:
  Sample batches $\{\tau_1, \ldots, \tau_T\}$ from X_A
  Initialize aggregated gradients $a \leftarrow 0$, $c \leftarrow 0$
  for each batch τ **do**:
    Use f_{θ_0} to compute attack loss $L^{\tau}_{\text{cls}}(\delta_{N_A})$ and stealth loss $L^{\tau}_{\text{stl}}(\delta_{N_A})$
    $a \leftarrow a + \nabla_\delta L^{\tau}_{\text{cls}}(\delta_{N_A}), \quad c \leftarrow c + \nabla_\delta L^{\tau}_{\text{stl}}(\delta_{N_A})$
  $a \leftarrow a/T, \quad c \leftarrow c/T$
  $d^k \leftarrow$ ELLIPSOIDALDIRECTION(a, c, γ, κ) # *Eqs.* (2)–(4)
  $\delta_{N_A} \leftarrow \Pi_{\mathcal{B}_\infty(\epsilon)}\big(\delta_{N_A} - \eta\, d^k\big)$ # *Projection and clipping*
return $\hat{X}_A := X_A + \delta_{N_A}$
Subroutine ELLIPSOIDALDIRECTION(a, c, γ, κ):
  Construct (u, λ) from (a, c) (Appendix C)
  $\rho \leftarrow \gamma\|a\|_2$
  $g_w \leftarrow wa + (1-w)c$
  Solve $\min_{w\in[0,1]} a^\top g_w + \rho\|g_w\|_{M^{-1}}$
  return $d^\star(w^\star)$

G DESCENT ANALYSIS UNDER ℓ_∞ PROJECTION

Sufficient decent condition. Theorem 4.1 analyzes the unprojected update $\delta^+ = \delta - \eta d^\star$. In the actual attack, however, the perturbation is constrained to an ℓ_∞ ball, and the update is therefore $\delta^+ = \Pi_{\mathcal{B}_\infty(\epsilon)}(\delta - \eta d^\star)$. We next establish a sufficient condition under which this projected update still decreases the attack objective.

Define the projected gradient mapping $G_\eta := \frac{\delta - \delta^+}{\eta}$. Since the current iterate δ is feasible, the optimality condition of Euclidean projection gives

$$(d^\star)^\top G_\eta \ge \|G_\eta\|_2^2. \tag{11}$$

Table 5: Effect of enforcing descent after the ℓ_∞ projection. ASR is reported in %, and rejected updates denote the fraction of projected updates that do not decrease L_{cls}.

	TENT	**RPL**	**EATA**	**SAR**	**DeYO**
Original ASR	89.59	93.17	90.21	95.90	91.15
Filtered ASR	89.28	92.86	90.00	95.72	90.77
Rejected updates (%)	3.20	3.19	3.28	3.18	3.41

Let $a := \nabla_\delta L_{\text{cls}}(\delta)$. By L-smoothness of L_{cls},

$$\begin{aligned} L_{\text{cls}}(\delta^+) &\le L_{\text{cls}}(\delta) + a^\top(\delta^+ - \delta) + \frac{L}{2}\|\delta^+ - \delta\|_2^2 \\ &= L_{\text{cls}}(\delta) - \eta a^\top G_\eta + \frac{L\eta^2}{2}\|G_\eta\|_2^2. \end{aligned}$$

Writing $a = d^\star - (d^\star - a)$ and using Eq. (11), we obtain

$$\begin{aligned} a^\top G_\eta &= (d^\star)^\top G_\eta - (d^\star - a)^\top G_\eta \\ &\ge \|G_\eta\|_2^2 - \|d^\star - a\|_2 \|G_\eta\|_2. \end{aligned}$$

Substituting this inequality into the smoothness bound yields

$$L_{\text{cls}}(\delta^+) \le L_{\text{cls}}(\delta) - \eta \left[\left(1 - \frac{L\eta}{2}\right) \|G_\eta\|_2 - \|d^\star - a\|_2 \right] \|G_\eta\|_2. \tag{12}$$

For our ellipsoidal direction $\|d^\star - a\|_2 = \gamma\xi\|a\|_2$, where $\xi \in [1/\sqrt{1+\lambda}, 1]$ as defined in Theorem 4.1. Therefore, the projected update strictly decreases L_{cls} whenever

$$\left(1 - \frac{L\eta}{2}\right) \|G_\eta\|_2 > \gamma\xi\|a\|_2. \tag{13}$$

Thus, projection changes the quantitative guarantee of Theorem 4.1, but a sufficient descent condition can still be established for the update used in practice. When the projection is inactive, $G_\eta = d^\star$ and the update reduces to the unprojected step analyzed in Theorem 4.1.

Empirical effect of projection. We further examine whether projection frequently prevents descent in practice. For this diagnostic experiment, we accept a projected update only if

$$L_{\text{cls}}(\delta^+) < L_{\text{cls}}(\delta), \tag{14}$$

and otherwise discard the update. Table 5 compares this descent-filtered variant with the original attack. Only 3.18%-3.41% of projected updates are rejected, while the ASR decreases by at most 0.38 percentage points. This indicates that the projected update rarely fails to decrease L_{cls} in practice and has negligible impact on the overall attack effectiveness. Together with Eq. (13), these results clarify the relationship between the theoretical analysis and the projected update used in our implementation.

H Additional Details on Experimental Setting

TTA method. Following prior work Wu et al. (2023); Cong et al. (2024); Su et al. (2024); Rifat et al. (2025), we evaluate attacks under five BN-affine TTA methods: TENT Wang et al. (2020) adapts batch-normalization parameters by minimizing entropy, RPL Rusak et al. (2021) uses robust pseudo-labels, EATA Niu et al. (2022) updates the model only on confident and stable samples, SAR Niu et al. (2023) performs sharpness-aware minimization, DeYO Lee et al. (2024) regularizes adaptation through a distributional entropy.

Implementation detail. We use the test batch size of 200. For attack generation, we set the number of outer iterations to $K = 500$ and use $T = 1$ batch per iteration. To improve robustness, at each iteration we randomly sample the data composition ratios from uniform distributions: the victim



(a) Reference. (b) Patch. (c) SIG.

Figure 8: Trigger patterns on victim input.

data ratio is sampled between 5% and 20%, the support data ratio between 40% and 60%, and the remainder consists of benign data. The step size for perturbation updates is set to $\eta = 1.0$. For the priority-aware gradient alignment, we set the trust-radius parameter to $\gamma = 0.5$ and use the misalignment scaling κ of 10, 10, and 0.01 for CIFAR10-C, CIFAR100-C, and ImageNet-C, respectively. Our experiments are conducted on a server equipped with a 64-core AMD Ryzen Threadripper 3990X CPU @ 2.9GHz, 128 GB of RAM, and two NVIDIA RTX 4500 Ada GPUs.

Adaptation of baseline. Naively applying prior class-wise attacks Su et al. (2024); Rifat et al. (2025) to our sample-wise setting yields suboptimal performance, as their original optimization does not account for trigger-based selectivity. To ensure a fair and rigorous comparison, we adapt these methods by integrating their original loss functions into our proposed meta-learning framework (Section 4.1). This allows the baselines to be optimized over simulated adaptation tasks, enabling them to learn perturbations that generalize to trigger-carrying inputs effectively. We retain the exact loss formulations from their official implementations (RTTDP: `https://github.com/Gorilla-Lab-SCUT/RTTDP`, FCA: `https://github.com/Restuccia-Group/tta-adv`) to isolate the impact of our proposed alignment strategy versus their respective objectives.

Trigger specification. We evaluate two common triggers in the experiment. For patch trigger Gu et al. (2017), we use the trigger pattern at four corners, with the size of the trigger being 15% of the image size. For SIG trigger Barni et al. (2019), we superimpose a sinusoidal signal onto the entire image, parameterized by a frequency of 10 and an amplitude of $16/255$. The visualization is shown in Figure 8. We also include an additional experiment using the imperceptible WaNet Nguyen & Tran (2021) trigger in Appendix I.2.

Evaluation protocol. We follow Su et al. (2024) and set the ratio of attacker-controlled samples to the combined victim and benign samples to $1:1$. Experiments with dynamically varying participation are provided in Appendix I.1. We use two label-free metrics to quantify stealth on non-targeted inputs. EDD is the Wasserstein distance between their predictive-entropy distributions under benign and attacked adaptation. ADI measures the change in prediction disagreement between an input and its augmented version relative to benign adaptation.

I Additional Experiments

I.1 Attack Effectiveness under Random Attacker Participation

We further evaluate a more dynamic setting in which the attacker-controlled ratio varies across batches and is sampled from $[0.1, 0.5]$ using either a uniform distribution or a Beta distribution skewed toward lower ratio (i.e., 0.1), as shown in Tables 6 and 7, respectively. Under randomly varying attacker participation, our method remains substantially more effective than both baselines. With uniformly sampled attacker-controlled ratios, our attack achieves an average ASR of approximately 60%, compared with only about 9% for RTTDP and 14% for FCA. Even when the ratio is Beta-distributed and skewed toward the low-participation regime around 0.1, our method maintains an average ASR of approximately 38%, whereas both baselines remain below 10%. This improvement does not come with a disproportionate increase in the label-free stealth metrics. Under uniform sampling, our entropy drift is comparable to FCA, while our augmentation inconsistency is consistently lower. Under the Beta-skewed setting, our method exhibits substantially lower entropy drift than FCA and comparable augmentation inconsistency, despite achieving roughly 4-$5\times$ higher ASR. RTTDP yields lower values on both stealth metrics, but this is accompanied by substantially weaker attack effectiveness. Overall,

Table 6: Attack effectiveness and stealthiness under uniformly sampled attacker-controlled ratios. Results are reported as mean ± standard deviation over three runs.

Metric	Method	TENT	RPL	EATA	SAR	DeYO
ASR (%)	RTTDP	6.48±0.33	12.89±1.61	5.78±0.22	13.10±0.26	6.85±1.17
	FCA	14.86±0.26	14.02±0.95	13.41±0.66	13.10±0.32	14.33±1.31
	Ours	**59.26±0.31**	**63.14±2.08**	**56.40±0.61**	**67.09±1.59**	**59.75±0.93**
EDD	RTTDP	0.0737±0.0005	0.0645±0.0076	0.0765±0.0049	0.0708±0.0025	0.0755±0.0004
	FCA	0.1803±0.0005	0.1983±0.0028	0.1774±0.0063	0.2227±0.0011	0.1849±0.0008
	Ours	0.1783±0.0024	0.1984±0.0039	0.1822±0.0051	0.2323±0.0009	0.1872±0.0019
ADI (%)	RTTDP	6.07±0.16	4.89±0.10	5.64±0.36	3.94±0.23	5.56±0.05
	FCA	9.69±0.26	9.47±0.03	8.65±0.42	9.14±0.66	9.31±0.35
	Ours	9.05±0.29	8.04±0.01	8.40±0.05	8.28±0.43	8.18±0.89

Table 7: Attack effectiveness and stealthiness when the attacker-controlled ratio is sampled from a Beta distribution skewed toward 0.1. Results are reported as mean ± standard deviation over three runs.

Metric	Method	TENT	RPL	EATA	SAR	DeYO
ASR (%)	RTTDP	9.88±0.31	7.55±0.31	8.84±0.59	6.69±1.94	8.28±1.35
	FCA	8.64±0.62	9.43±1.47	7.90±0.30	6.71±0.30	8.12±0.74
	Ours	**35.19±0.42**	**42.14±0.63**	**31.10±0.31**	**44.73±0.14**	**37.66±0.98**
EDD	RTTDP	0.0962±0.0076	0.0969±0.0014	0.1012±0.0010	0.0910±0.0073	0.1029±0.0023
	FCA	0.2902±0.0043	0.3156±0.0004	0.2875±0.0011	0.3449±0.0088	0.2953±0.0046
	Ours	0.2389±0.0020	0.2450±0.0051	0.2446±0.0047	0.2580±0.0035	0.2462±0.0004
ADI (%)	RTTDP	10.72±0.20	8.93±0.89	9.86±0.84	8.11±0.04	9.61±0.70
	FCA	15.25±0.33	15.62±0.24	14.81±0.67	15.32±0.14	14.99±0.11
	Ours	15.31±0.10	14.55±0.21	15.11±0.97	15.80±0.05	14.99±0.05

these results demonstrate that our attack remains effective even when attacker participation varies across batches.

I.2 Effectiveness with Imperceptible Triggers

The triggers used in our main evaluation (Section 5) are representative choices rather than requirements of our attack. Our framework is compatible with a broad range of stealthy triggers developed in the data-poisoning literature Li et al. (2023). To verify that the attack does not rely on visually salient triggers, we further evaluate it using WaNet Nguyen & Tran (2021), a widely used imperceptible warping-based trigger.

As shown in Table 8, our attack remains highly effective across all TTA methods, achieving ASRs between 80.80% and 99.68%. This demonstrates that the attack is not limited to the visible triggers considered in the main evaluation and remains effective with an imperceptible trigger.

More broadly, malicious perturbations may also resemble natural corruptions or distribution shifts, which are precisely the conditions under which TTA is commonly deployed. Consequently, manual inspection is neither reliable nor scalable, particularly for imperceptible triggers and high-volume online adaptation streams. This further motivates the need for automated defenses against such attacks.

I.3 Additional Defenses

I.3.1 MedBN TTA Defense Park et al. (2024)

Table 9 reports the full attack performance of our method against the MedBN TTA defense Park et al. (2024).

Table 8: ASR (%) of our attack using the imperceptible WaNet Nguyen & Tran (2021) trigger.

TENT	RPL	EATA	SAR	DeYO
80.80	94.65	83.64	99.68	89.38

Table 9: ASR (%) of our attack against the MedBN TTA defense Park et al. (2024) across different TTA methods on the CIFAR-10-C dataset, using SIG and patch triggers.

Attack	SIG Trigger					Patch Trigger				
	TENT	RPL	EATA	SAR	DeYO	TENT	RPL	EATA	SAR	DeYO
No attack	2.85	3.07	3.10	3.04	2.85	2.77	3.16	3.02	3.64	2.97
Ours	89.59	93.17	90.21	95.90	91.15	79.28	86.78	80.34	93.33	82.48
Ours + Park et al. (2024)	78.03	76.92	79.15	84.44	79.14	71.85	81.29	73.82	83.06	73.93

I.3.2 Trigger Purification Defense Doan et al. (2020)

We evaluate our attack against a trigger purification defense designed to mitigate backdoor attacks Doan et al. (2020). The results are presented in Table 10. Although the defense reduces the ASR to some extent, our attack remains effective.

I.4 Label Distribution Shift

To complement the ablation study in Section 5.3, Figures 9 and 10 visualize the label distribution shifts induced by attacks optimized using only the attack loss L_{cls}. It exhibits substantially higher KL divergence with pronounced probability spikes compared to our approach.

I.5 Sensitivity Analysis

I.5.1 Dependency on Attacker Data Ratio

We evaluate the effectiveness of the attack on various attacker data ratios. The results are shown in Figure 11a. Increasing the attacker data ratio consistently improves ASR across all TTA methods on CIFAR-10-C. At 50%, the attack achieves the highest ASR, ranging from 89.59% to 95.90%. As the ratio decreases to 20%, ASR declines. This behavior is expected, as increasing the proportion of attacker-controlled data within a batch makes it easier for the attacker to influence the batch statistics. A similar trend has been observed in prior TTA attacks Wu et al. (2023); Su et al. (2024); Rifat et al. (2025).

However, it is important to note that even at a 20% attacker-controlled ratio, our attack remains substantially more effective than existing baselines, as shown in Figure 12. On average, our attack achieves an ASR approximately $3.6\times$ higher than FCA and $12.5\times$ higher than RTTDP.

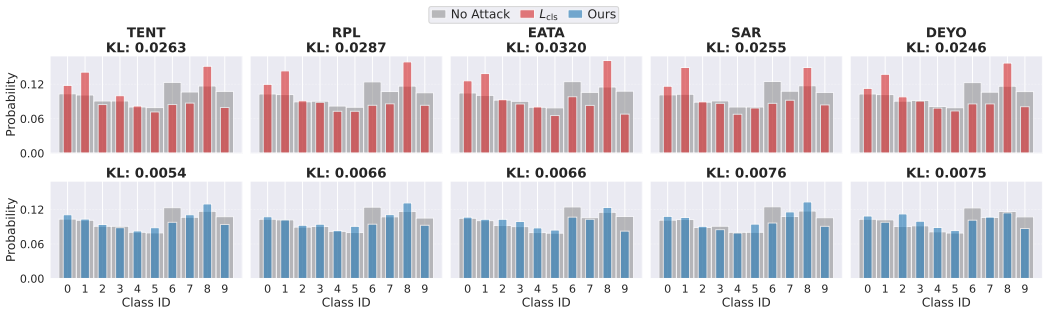


Figure 9: Comparison of label distribution shifts induced by L_{cls} and our method on CIFAR-10-C.

Table 10: ASR (%) of our attack against the trigger purification defense Doan et al. (2020) across different TTA methods on the CIFAR-10-C dataset, using patch triggers.

Attack	Patch Trigger				
	TENT	RPL	EATA	SAR	DeYO
No attack	2.77	3.16	3.02	3.64	2.97
Ours	79.28	86.78	80.34	93.33	82.48
Ours + Doan et al. (2020)	51.54	58.18	50.76	66.45	54.26

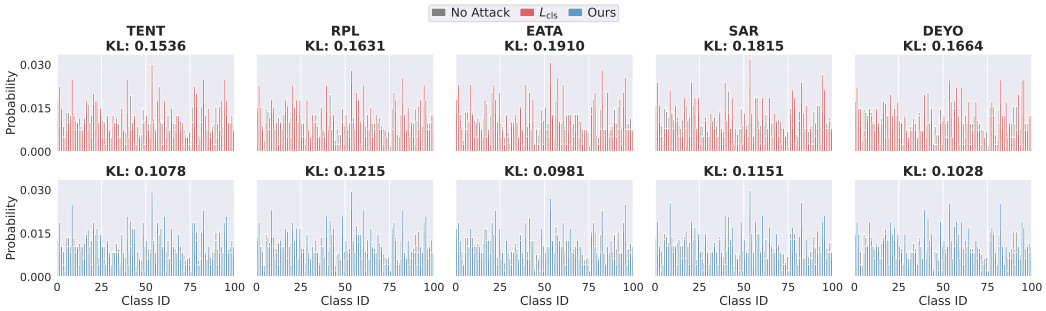


Figure 10: Comparison of label distribution shifts induced by L_{cls} and our method on CIFAR-100-C.

I.5.2 Dependency on Victim Ratio

We vary the victim ratio while keeping the batch size fixed, and report the ASR in Figure 11b. From the figure, increasing the victim ratio while keeping the batch size fixed generally leads to higher ASR. Intuitively, a larger victim ratio means more trigger-carrying victim samples in each batch and fewer unaffected benign inputs, since both the batch size and the number of attack-controlled samples remain fixed. As a result, the adaptation update is more consistently driven toward the malicious objective, making the attack easier to induce.

I.5.3 Dependency on Attack Budget

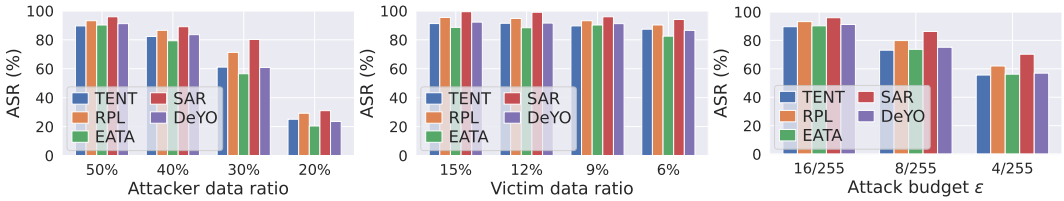
We further evaluate the attack's effectiveness on different attack budgets, with results reported in Figure 11c. Specifically, we consider $\epsilon = 16/255, \epsilon = 8/255, \epsilon = 4/255$. Increasing the attack budget ϵ consistently improves ASR across all TTA methods. With the largest budget of $16/255$, the attack achieves the highest ASR. As the budget decreases to $8/255$ and $4/255$, ASR correspondingly drops across all methods, indicating that stronger perturbation budgets lead to more effective attacks.

I.6 Cross-Dataset Backbone

Table 11: ASR (%) of our attack using ResNet-32 backbone across TTA methods and datasets.

Dataset	SIG					Patch				
	TENT	RPL	EATA	SAR	DeYO	TENT	RPL	EATA	SAR	DeYO
CIFAR-10-C	89.59	93.17	90.21	95.90	91.15	79.28	86.78	80.34	93.33	82.48
CIFAR-100-C	66.93	76.17	76.42	81.49	70.66	88.04	91.82	91.83	93.47	88.97
ImageNet-C	99.07	99.28	99.05	99.23	98.15	91.59	91.76	91.51	91.57	90.74

Table 11 and Table 12 show the attack effectiveness using ResNet-32 and ResNet-50 as a backbone across three datasets, respectively. The attack remains highly effective under both ResNet-32 and ResNet-50 backbones, achieving consistently high ASR across all settings. For the SIG trigger, ImageNet-C exhibits near-perfect ASR for both architectures, while CIFAR-10-C and CIFAR-100-C show comparatively lower but still substantial attack success. For the patch trigger, the attack generally achieves higher ASR on CIFAR-100-C and ImageNet-C than on CIFAR-10-C. Comparing



(a) Attacker data ratio. (b) Victim data ratio. (c) Attack budget.

Figure 11: Sensitivity analysis of the attack under different settings.

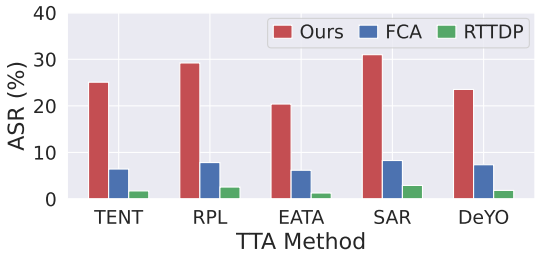


Figure 12: ASR (%) across TTA methods at a 20% attacker-controlled data ratio.

Table 12: ASR (%) of our attack using ResNet-50 backbone across TTA methods and datasets.

Dataset	SIG					Patch				
	TENT	RPL	EATA	SAR	DeYO	TENT	RPL	EATA	SAR	DeYO
CIFAR-10-C	72.66	63.18	47.31	76.09	40.63	61.45	52.21	37.23	72.65	45.48
CIFAR-100-C	61.22	54.17	57.41	66.00	68.09	84.00	84.78	81.63	92.31	87.23
ImageNet-C	96.79	96.99	95.22	96.98	95.49	92.29	92.49	88.48	92.28	89.21

architectures, ResNet-32 typically yields higher ASR than ResNet-50, suggesting that the attack transfers more effectively to the smaller backbone.

I.7 Attack Effectiveness across Various Severities

Table 13: ASR (%) of our attack under various TTA methods and severities, with SIG and patch trigger.

Dataset	Severity	SIG Trigger					Patch Trigger				
		TENT	RPL	EATA	SAR	DeYO	TENT	RPL	EATA	SAR	DeYO
CIFAR-10-C (ResNet32)	Severity 1	87.33	90.46	88.27	94.29	88.70	72.12	80.62	73.60	89.73	75.39
	Severity 2	89.10	92.24	89.27	95.30	90.17	76.73	84.45	76.95	91.64	79.79
	Severity 3	89.72	93.78	89.66	95.89	91.18	79.40	87.54	80.35	93.23	83.03
	Severity 4	90.54	94.47	91.53	96.85	92.26	82.21	89.27	83.28	95.41	85.24
	Severity 5	91.28	94.87	92.33	97.15	93.46	85.95	92.05	87.53	96.67	88.95
CIFAR-100-C (ResNet32)	Severity 1	55.35	64.62	63.58	70.96	58.45	82.32	87.74	87.83	89.97	83.51
	Severity 2	61.98	72.37	72.81	79.39	66.47	85.86	90.88	90.28	92.93	87.11
	Severity 3	67.69	76.48	77.57	81.95	70.59	88.62	91.76	92.03	93.64	89.30
	Severity 4	72.98	81.75	82.73	86.54	77.54	90.84	93.90	94.34	95.23	91.83
	Severity 5	76.65	85.65	85.41	88.62	80.26	92.59	94.79	94.69	95.55	93.10
ImageNet-C (ResNet50)	Severity 3	96.79	96.99	95.22	96.98	95.49	92.29	92.49	88.48	92.28	89.21

Table 13 summarizes attack effectiveness under varying corruption severities. For ImageNet-C, experiments are conducted only at severity level 3, as specified in Section 5.1. Overall, ASR consistently increases as the corruption severity becomes stronger, indicating that heavier distribution shifts make TTA methods more vulnerable to the attack. On CIFAR-10-C and CIFAR-100-C, the

attack achieves progressively higher ASR from Severity 1 to Severity 5 for all methods and trigger types. The SIG trigger generally performs better on CIFAR-10-C, while the patch trigger achieves particularly strong ASR on CIFAR-100-C. On ImageNet-C, the attack maintains very high ASR across all TTA methods for both trigger types.

I.8 Attack Effectiveness across Various Corruption Types

Table 14: ASR (%) of our attack under various TTA methods and corruption types, with SIG and patch trigger.

Dataset	Corruption Type	SIG Trigger					Patch Trigger				
		TENT	RPL	EATA	SAR	DeYO	TENT	RPL	EATA	SAR	DeYO
CIFAR-10-C (ResNet32)	Brightness	75.84	81.33	77.22	85.26	77.66	60.55	69.07	62.34	79.74	63.93
	Contrast	97.15	98.06	97.31	98.70	96.80	85.22	89.58	84.74	90.84	84.47
	Defocus Blur	91.45	92.71	91.87	95.27	91.88	80.61	85.18	80.71	90.71	82.98
	Elastic Transform	92.55	95.26	92.50	96.89	93.46	84.38	90.46	84.03	94.57	86.52
	Fog	95.47	97.35	95.44	97.86	94.98	80.76	86.99	81.70	90.58	82.70
	Frost	86.55	90.79	89.56	94.41	89.63	69.50	78.34	72.53	91.54	73.58
	Gaussian Noise	89.29	94.26	88.87	97.33	91.12	77.25	89.59	78.07	96.58	81.87
	Glass Blur	94.42	97.72	94.92	98.99	96.33	87.83	94.66	88.90	97.35	91.57
	Impulse Noise	86.80	93.81	87.94	96.85	90.50	75.07	87.30	77.36	96.50	82.05
	Jpeg Compression	88.88	92.96	90.14	96.70	91.19	89.51	95.16	90.26	98.76	91.91
	Motion Blur	94.27	96.50	93.50	97.52	95.08	85.99	90.66	86.06	94.35	87.33
	Pixelate	86.76	91.99	87.58	95.18	89.11	83.85	89.43	83.85	95.41	86.13
	Shot Noise	86.41	91.16	86.81	96.40	87.83	73.31	85.26	74.51	95.43	77.97
	Snow	86.37	89.19	87.59	94.09	89.12	69.17	79.24	73.01	93.04	75.57
	Zoom Blur	91.70	94.38	91.94	97.01	92.62	86.23	90.86	87.05	94.61	88.62
CIFAR-100-C (ResNet32)	Brightness	45.36	51.22	51.64	57.34	47.73	67.84	73.98	74.41	78.40	68.49
	Contrast	76.44	86.51	84.04	89.89	81.17	91.95	93.43	92.92	93.99	91.80
	Defocus Blur	61.06	67.84	66.18	73.84	63.81	84.30	88.04	87.47	90.27	86.01
	Elastic Transform	66.89	75.12	75.16	81.03	69.40	88.20	92.69	91.32	93.55	88.77
	Fog	74.41	81.98	84.72	88.26	78.94	91.19	93.14	94.27	95.49	91.91
	Frost	63.25	73.17	75.92	80.12	69.21	81.66	87.47	87.65	88.87	84.09
	Gaussian Noise	73.93	85.51	87.07	89.87	80.00	92.33	97.25	97.48	97.70	93.22
	Glass Blur	72.42	81.85	83.12	86.72	76.22	95.20	97.90	98.28	98.69	96.08
	Impulse Noise	74.95	84.69	84.16	88.21	78.50	89.78	94.73	94.26	96.22	91.75
	Jpeg Compression	71.52	82.03	83.37	86.73	74.53	93.50	96.46	97.51	98.08	94.24
	Motion Blur	69.09	76.30	76.59	80.75	71.40	90.71	93.02	92.34	94.18	90.90
	Pixelate	56.28	65.05	64.92	73.47	59.02	86.60	90.29	90.87	92.61	86.19
	Shot Noise	69.10	83.00	81.95	86.03	73.51	91.51	96.32	96.26	97.62	93.11
	Snow	61.49	69.27	69.80	76.59	64.61	84.71	89.05	89.26	91.60	86.27
	Zoom Blur	67.78	79.07	77.64	83.55	71.89	91.21	93.48	93.22	94.71	91.70
ImageNet-C (ResNet50)	Brightness	84.02	84.62	82.84	84.02	84.52	72.19	72.19	66.86	71.60	66.07
	Contrast	100.00	100.00	99.35	100.00	99.36	94.59	94.59	92.26	94.59	92.95
	Defocus Blur	100.00	100.00	100.00	100.00	100.00	100.00	100.00	97.94	100.00	97.89
	Elastic Transform	92.36	92.36	88.61	92.99	88.20	85.99	86.62	77.85	85.99	78.40
	Fog	97.89	98.59	97.95	98.59	97.95	85.21	85.21	84.93	85.21	84.35
	Frost	100.00	100.00	98.94	100.00	98.92	100.00	100.00	96.81	100.00	96.81
	Gaussian Noise	100.00	100.00	98.10	100.00	98.15	99.01	98.98	97.14	98.99	97.22
	Glass Blur	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	Impulse Noise	98.15	98.15	98.25	98.15	98.25	98.15	98.13	93.86	98.15	96.49
	Jpeg Compression	93.38	93.38	86.99	94.12	89.73	83.09	83.09	77.40	83.09	79.45
	Motion Blur	95.04	95.87	93.39	95.87	93.50	95.04	95.04	88.43	95.04	91.06
	Pixelate	92.76	92.76	90.85	92.76	91.50	80.92	81.58	74.34	79.61	75.32
	Shot Noise	100.00	100.00	99.12	100.00	100.00	100.00	100.00	97.37	100.00	98.25
	Snow	99.06	99.05	94.78	99.06	93.91	95.28	96.19	87.83	96.23	89.47
	Zoom Blur	99.15	100.00	99.16	99.15	98.37	94.92	95.73	94.17	95.73	94.35

In Table 14, we report the attack effectiveness across the 15 corruption types defined in Hendrycks & Dietterich (2019b). Stronger distribution-shift corruptions such as contrast, glass blur, Gaussian noise, shot noise, and JPEG compression generally lead to higher ASR, while brightness is consistently among the least vulnerable corruption types. On CIFAR-10-C, most corruption types yield high ASR, especially contrast, glass blur, fog, motion blur, and zoom blur. On CIFAR-100-C, the attack remains effective but shows larger variation across corruption types, with brightness and pixelate producing relatively lower ASR. For ImageNet-C, the ASR is near-saturated for many corruptions, including defocus blur, frost, Gaussian noise, glass blur, and shot noise, indicating that the attack remains highly effective under diverse corruption-induced distribution shifts.

J LIMITATION AND FUTURE WORK

We acknowledge two limitations of our current study and outline corresponding future directions where applicable.

Scope of Stealth Evaluation. We evaluate stealth using two label-free monitoring signals, EDD and ADI, which capture entropy-distribution drift and augmentation inconsistency relative to benign adaptation. While these metrics cover complementary aspects of prediction behavior, they do not encompass all possible runtime monitors, such as temporal drift detectors or feature-level anomaly detection. Evaluating against a broader range of label-free monitoring strategies is a direction for future work.

Performance under Very Low Attacker Participation. While our attack outperforms existing baselines under dynamically varying attacker participation (Appendix I.1) and under lower ratios (Appendix I.5.1), its effectiveness decreases as the attacker-controlled fraction becomes very small. Similar declines are also reported by prior TTA attacks Wu et al. (2023); Su et al. (2024); Rifat et al. (2025), reflecting the reduced influence of fewer attacker-controlled samples on shared batch statistics and adaptation updates. Improving effectiveness under highly constrained injection budgets is a direction for future TTA attacks.