Energy Waveify and Redistribution for Test-Time Adaptation: A Control System Perspective

Supplementary Material

1. Relate Works

1.1. Test-Time Adaptation

Test-time adaptation (TTA) has recently emerged as an effective strategy to improve model robustness under distribution shifts, where the test data differ from the training distribution. Unlike traditional domain adaptation methods that require access to target domain samples during training, TTA focuses on adapting the model online at inference time, using only the unlabeled test data. Early works such as TENT [29] and its variants [23, 24] proposed optimizing batch normalization statistics or entropy-based objectives to adjust the model parameters dynamically. Subsequent research extended these ideas by introducing self-supervised auxiliary tasks [35], consistency regularization [32], and memory-based mechanisms [3, 13] to stabilize adaptation and prevent catastrophic forgetting. Other approaches leverage lightweight parameter updates, such as feature normalization, affine parameter refinement, or the use of adapters, to balance efficiency and performance [25]. Recently, several studies have explored test-time adaptation in more challenging settings, including continual domain shifts [30], streaming data [10], and large-scale vision-language models [16, 37], further broadening its applicability. Overall, TTA provides a promising direction for building models that can autonomously adapt to unseen environments without retraining.

1.2. Test-Time Energy Adaptation

Recently, incorporating energy function into TTA has emerged as a promising trend for improving model robustness under distribution shifts [33, 36]. Unlike traditional approaches that rely on entropy minimization or normalization statistic updates, energy-based methods define an explicit energy function to measure the compatibility between inputs and outputs. This formulation provides a more expressive and theoretically grounded objective, allowing the model to capture uncertainty and adapt its predictions beyond the limitations of entropy-based confidence measures. Typically, energy-based TTA methods perform adaptation by minimizing the energy associated with each test sample. During inference, the model iteratively refines its predictions through stochastic sampling (often using Monte Carlo or Langevin dynamics) to explore low-energy regions in the prediction space. This process encourages the model to converge toward more consistent and confident outputs while maintaining stability under unseen domains. However, despite their effectiveness, these sampling-based energy adaptation methods are computationally demanding. The iterative optimization over each test instance requires multiple forward and backward passes, leading to significant inference latency. Consequently, while energy-based TTA provides a principled and powerful framework for handling distribution shifts, its high computational cost poses a major challenge for real-time or large-scale deployment scenarios.

1.3. Control System-Inspired AI Methods

Control theory has recently emerged as an important source of inspiration for developing more stable, robust, and interpretable AI systems [4, 21, 28]. Classical control principles, such as feedback regulation, stability analysis, and Lyapunov-based optimization, have been increasingly integrated into modern learning frameworks to address the instability and unpredictability of purely data-driven models. For example, control-inspired reinforcement learning methods incorporate feedback control structures or model predictive control to enhance policy stability and safety under dynamic and uncertain environments [2, 26]. In supervised and unsupervised learning, control-theoretic ideas have been used to design optimization algorithms with guaranteed convergence or robustness properties, such as Lyapunov-stable training dynamics and adaptive gain tuning [8]. Moreover, differentiable control architectures and neural ordinary differential equations extend the connection between continuous-time control systems and deep networks, offering new perspectives on dynamic representation learning [9]. Overall, these control-inspired AI methods bridge theoretical rigor from control systems with the flexibility of data-driven learning, enabling more reliable and physically grounded intelligent systems.

2. Algorithm Protocol

Algorithms 1 and 2 present the pseudocode of our proposed APT framework. Algorithm 1 outlines the initialization phase, where the energy of training samples from the pre-trained model f_θ is transformed into a wave representation. Subsequently, Algorithm 2 details the core TTA process. This process adapts the pre-trained model to the target domain by minimizing the probability of high-energy states, rather than minimizing the energy values directly. Crucially, the two algorithms differ in the target of automatic differentiation: Algorithm 1 requires gradient computation for the energy, whereas Algorithm 2 requires gradient computation for the raw test input samples.

Algorithm 1 Energy waveify before test time

Input: Training distribution input $\mathbf{x}_s$ (Batch size B), pre-trained model f_θ, potential V_0, threshold a, boundary $\{\tilde{x}_l, \tilde{x}_r\}$, hyperparameters $\{\alpha, \delta\}$
Output: Initial wave mapping $\Psi_{\hat{\theta}}$ with parameters $\hat{\theta}$
Initialization Phase
$\mathbf{X} \leftarrow f_\theta(\mathbf{x}_s)$; $\mathbf{X}$.requires_grad(True)
$\mathbf{\Psi} \leftarrow \Psi_{\hat{\theta}}(\mathbf{X})$ ▷ Energy waveify
Wave Equation Residual ▷ Eq.(6)
$\mathbf{\Psi}^{(1)} \leftarrow \text{AutoGrad}(\mathbf{\Psi}, \mathbf{X})$, $\mathbf{\Psi}^{(2)} \leftarrow \text{AutoGrad}(\mathbf{\Psi}^{(1)}, \mathbf{X})$
$\mathcal{M}_L \leftarrow \mathbb{I}[\mathbf{X} < a]$, $\mathcal{M}_R \leftarrow \mathbb{I}[\mathbf{X} \geq a]$ ▷ Left/right region mask
$\mathcal{R}_L \leftarrow -\mathbf{\Psi}^{(2)} - \mathbf{\Psi}$, $\mathcal{R}_R \leftarrow -\mathbf{\Psi}^{(2)} + V_0\mathbf{\Psi} - \mathbf{\Psi}$
$\mathcal{L}_{\text{def}} \leftarrow \frac{1}{B}\sum_{i=1}^{B}\left[\mathcal{M}_L^{(i)} \cdot |\mathcal{R}_L^{(i)}|^2 + \mathcal{M}_R^{(i)} \cdot |\mathcal{R}_R^{(i)}|^2\right]$
Continuity Condition ▷ Eq.(7)
$\mathbf{X}_L^{(a)} \leftarrow \mathbf{1}_B \otimes (a - \delta)$, $\mathbf{X}_R^{(a)} \leftarrow \mathbf{1}_B \otimes (a + \delta)$
$\mathbf{\Psi}_L^{(a)} \leftarrow \Psi_{\hat{\theta}}(\mathbf{X}_L^{(a)})$, $\mathbf{\Psi}_R^{(a)} \leftarrow \Psi_{\hat{\theta}}(\mathbf{X}_R^{(a)})$
$\mathcal{L}_{\text{value}} \leftarrow \frac{1}{B}\sum_{i=1}^{B}|\mathbf{\Psi}_L^{(a,i)} - \mathbf{\Psi}_R^{(a,i)}|^2$ ▷ Value continuity
$\mathbf{\Psi}_L^{(1)} \leftarrow \text{AutoGrad}(\mathbf{\Psi}_L^{(a)}, \mathbf{X}_L^{(a)})$
$\mathbf{\Psi}_R^{(1)} \leftarrow \text{Autograd}(\mathbf{\Psi}_R^{(a)}, \mathbf{X}_L^{(a)})$
$\mathcal{L}_{\text{grad}} \leftarrow \frac{1}{B}\sum_{i=1}^{B}|\mathbf{\Psi}_L^{(1,i)} - \mathbf{\Psi}_R^{(1,i)}|^2$ ▷ Gradient continuity
Boundary Conditions ▷ Eq.(8)
$\mathbf{X}_\partial \leftarrow [\mathbf{1}_{B/2} \otimes \tilde{x}_l; \mathbf{1}_{B/2} \otimes \tilde{x}_r]$
$\mathbf{\Psi}_\partial \leftarrow \Psi_{\hat{\theta}}(\mathbf{X}_\partial)$
$\mathcal{L}_{\text{bound}} \leftarrow \frac{1}{B}\sum_{i=1}^{B}\left[|\text{Re}(\mathbf{\Psi}_\partial^{(i)})|^2 + |\text{Im}(\mathbf{\Psi}_\partial^{(i)})|^2\right]$
Total Loss & Optimization
$\mathcal{L}_{\text{wave}} \leftarrow \mathcal{L}_{\text{def}} + \alpha \cdot (\mathcal{L}_{\text{value}} + \mathcal{L}_{\text{grad}} + \mathcal{L}_{\text{bound}})$ ▷ Eq.(9)
$\mathbf{g}_{\hat{\theta}} \leftarrow \text{AutoGrad}(\mathcal{L}_{\text{wave}}, \hat{\theta})$
$\hat{\theta} \leftarrow \text{Optimizer.Step}(\hat{\theta}, \mathbf{g}_{\hat{\theta}})$
return $\hat{\theta}$

Algorithm 2 Redistribute energy at test time

Input: Test batch $\mathbf{x}_t$ (size B'), pre-trained model f_θ, potential V_0 and MLP $\Psi_{\hat{\theta}}$, threshold a, boundaries $\{\tilde{x}_l, \tilde{x}_r\}$, hyperparams $\{\alpha, \beta, \delta\}$
Output: Adapted model parameters θ
Initialization
$\mathbf{X} \leftarrow f_\theta(\mathbf{x}_t)$; $\mathbf{x}_t$.requires_grad(True) ▷ Different with Alg. 1
$\mathbf{\Psi} \leftarrow \Psi_{\hat{\theta}}(\mathbf{X})$ ▷ Energy waveify
$\mathbf{s} \leftarrow \sum(\mathbf{\Psi}^2)$ ▷ Energy probability
Penalize High-Energy ▷ Eq. (12)
$\mathcal{M}_{\text{InD}} \leftarrow \mathbb{I}[\mathbf{X} < a]$
$\mathbf{s}_{\text{OOD}} \leftarrow \mathbf{s}[\neg\mathcal{M}_{\text{InD}}]$
$\mathcal{L}_{\text{penalize}} \leftarrow \begin{cases} \text{Mean}(\mathbf{s}_{\text{OOD}}) & \text{if } \mathbf{s}_{\text{OOD}} \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$
Wave Equation Residual ▷ Eq. (6)
$\mathbf{\Psi}^{(1)} \leftarrow \text{AutoGrad}(\mathbf{\Psi}, \mathbf{X})$, $\mathbf{\Psi}^{(2)} \leftarrow \text{AutoGrad}(\mathbf{\Psi}^{(1)}, \mathbf{X})$
$\mathcal{M}_R \leftarrow \mathbb{I}[\mathbf{X} \geq a]$
$\mathcal{R}_L \leftarrow -\mathbf{\Psi}^{(2)} - \mathbf{\Psi}$, $\mathcal{R}_R \leftarrow -\mathbf{\Psi}^{(2)} + V_0\mathbf{\Psi} - \mathbf{\Psi}$
$\mathcal{L}_{\text{def}} \leftarrow \frac{1}{B'}\sum_{i=1}^{B'}\left[\mathcal{M}_{\text{InD}}^{(i)}|\mathcal{R}_L^{(i)}|^2 + \mathcal{M}_R^{(i)}|\mathcal{R}_R^{(i)}|^2\right]$
Continuity Condition ▷ Eq. (7)
$\mathbf{X}_L^{(a)} \leftarrow \mathbf{1}_{B'} \otimes (a - \delta)$, $\mathbf{X}_R^{(a)} \leftarrow \mathbf{1}_{B'} \otimes (a + \delta)$
$\mathbf{\Psi}_L^{(a)} \leftarrow \Psi_{\hat{\theta}}(\mathbf{X}_L^{(a)})$, $\mathbf{\Psi}_R^{(a)} \leftarrow \Psi_{\hat{\theta}}(\mathbf{X}_R^{(a)})$
$\mathcal{L}_{\text{value}} \leftarrow \frac{1}{B'}\sum_{i=1}^{B'}|\mathbf{\Psi}_L^{(a,i)} - \mathbf{\Psi}_R^{(a,i)}|^2$ ▷ Value continuity
$\mathbf{\Psi}_L^{(1)} \leftarrow \text{AutoGrad}(\mathbf{\Psi}_L^{(a)}, \mathbf{X}_L^{(a)})$
$\mathbf{\Psi}_R^{(1)} \leftarrow \text{AutoGrad}(\mathbf{\Psi}_R^{(a)}, \mathbf{X}_R^{(a)})$
$\mathcal{L}_{\text{grad}} \leftarrow \frac{1}{B'}\sum_{i=1}^{B'}|\mathbf{\Psi}_L^{(1,i)} - \mathbf{\Psi}_R^{(1,i)}|^2$ ▷ Gradient continuity
Boundary Condition ▷ Eq. (8)
$\mathbf{X}_\partial \leftarrow [\mathbf{1}_{B'/2} \otimes \tilde{x}_l; \mathbf{1}_{B'/2} \otimes \tilde{x}_r]$
$\mathbf{\Psi}_\partial \leftarrow \Psi_{\hat{\theta}}(\mathbf{X}_\partial)$
$\mathcal{L}_{\text{bound}} \leftarrow \frac{1}{B'}\sum_{i=1}^{B'}\left[|\text{Re}(\mathbf{\Psi}_\partial^{(i)})|^2 + |\text{Im}(\mathbf{\Psi}_\partial^{(i)})|^2\right]$
Total Loss & Optimization ▷ Eq. (13)
$\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{penalize}} + \beta\left(\mathcal{L}_{\text{def}} + \alpha(\mathcal{L}_{\text{value}} + \mathcal{L}_{\text{grad}} + \mathcal{L}_{\text{bound}})\right)$
$\mathbf{g}_\theta \leftarrow \text{AutoGrad}(\mathcal{L}_{\text{total}}, \theta)$
$\theta \leftarrow \text{Optimizer.Step}(\theta, \mathbf{g}_\theta)$
return θ

3. Formal Proof of Theorem 1

Before proving Theorem 1, we first present two auxiliary results that will be used in the derivation.

Lemma 1 (Gaussian Integral Formula) *For complex a with $\text{Re}(a) > 0$ and any complex b,*

$$\int_{-\infty}^{\infty} e^{-ax^2+bx}\,\mathrm{d}x = e^{b^2/(4a)}\sqrt{\frac{\pi}{a}}.$$

Brief Proof. Completing the square yields

$$-ax^2 + bx = -a\left(x - \frac{b}{2a}\right)^2 + \frac{b^2}{4a}.$$

Shifting the integration variable $t = x - \frac{b}{2a}$ gives

$$\int_{-\infty}^{\infty} e^{-ax^2+bx}\,\mathrm{d}x = e^{b^2/(4a)}\int_{-\infty}^{\infty} e^{-at^2}\,\mathrm{d}t.$$

The last integral equals $\sqrt{\pi/a}$ by analytic continuation of the real case.
Intuition: Lemma 1 provides a closed-form evaluation for a shifted Gaussian integral, which will be used to express each infinitesimal kernel in an exponential form.

Lemma 2 (Trotter Product Formula) *Let $\hat{A}$ and $\hat{B}$ be infinitesimal generators of strongly continuous contraction semigroups $e^{t\hat{A}}$ and $e^{t\hat{B}}$ on a Hilbert space $\mathcal{H}$. Assume that the closure of $\hat{A} + \hat{B}$ is also a generator. Then, for all $t \geq 0$ and $x \in \mathcal{H}$,*

$$\lim_{n\to\infty}\left(e^{\frac{t}{n}\hat{A}}e^{\frac{t}{n}\hat{B}}\right)^n x = e^{t(\hat{A}+\hat{B})}x,$$

in the strong operator topology.

Brief Proof. Define $F(t) = e^{t\hat{A}}e^{t\hat{B}}$. Since $F(t)$ is a family of contractions strongly continuous in t, and for all $x \in D(\hat{A}) \cap D(\hat{B})$,

$$\lim_{t\to 0}\frac{F(t)x - x}{t} = (\hat{A} + \hat{B})x,$$

Chernoff's theorem [5] implies

$$\lim_{n\to\infty} F\left(\frac{t}{n}\right)^n x = e^{t(\hat{A}+\hat{B})}x.$$

By density and uniform boundedness, this extends to all $x \in \mathcal{H}$.
Intuition: Lemma 2 ensures that the sequential composition of two infinitesimal evolutions $e^{\frac{t}{n}\hat{A}}$ and $e^{\frac{t}{n}\hat{B}}$ approximates the combined evolution $e^{t(\hat{A}+\hat{B})}$ in the limit $n \to \infty$.

Proof of Theorem 1. Let $K(\hat{\mathbf{x}}, t; \hat{\mathbf{x}}_0, t_0)$ be an integral kernel on a suitable function space such that

$$\psi(\hat{\mathbf{x}}, t) = \int K(\hat{\mathbf{x}}, t; \hat{\mathbf{x}}_0, t_0)\, \psi(\hat{\mathbf{x}}_0, t_0)\, \mathrm{d}\hat{\mathbf{x}}_0.$$

(1) Discretization of the evolution: Partition the interval $[t_0, t]$ into N subintervals of length $\epsilon = (t - t_0)/N$. By iterative composition,

$$K(\hat{\mathbf{x}}, t; \hat{\mathbf{x}}_0, t_0) = \int \prod_{k=1}^{N-1} \mathrm{d}\hat{\mathbf{x}}_k\, K(\hat{\mathbf{x}}, t; \hat{\mathbf{x}}_{N-1}, t_{N-1}) \cdots K(\hat{\mathbf{x}}_1, t_1; \hat{\mathbf{x}}_0, t_0).$$

(2) Local approximation of each infinitesimal kernel: For sufficiently small ϵ, by Lemma 1 one obtains

$$K(\hat{\mathbf{x}}_{k+1}, t+\epsilon; \hat{\mathbf{x}}_k, t) = \frac{1}{\sqrt{2\pi i\epsilon}} \exp\left(i \left[\frac{(\hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_k)^2}{2\epsilon} - V(\hat{\mathbf{x}}_k)\epsilon \right] \right).$$

Define the discrete functional

$$S_k = \left[\frac{1}{2} \left(\frac{\hat{\mathbf{x}}_{k+1} - \hat{\mathbf{x}}_k}{\epsilon} \right)^2 - V(\hat{\mathbf{x}}_k) \right] \epsilon,$$

so that $K(\hat{\mathbf{x}}_{k+1}, t + \epsilon; \hat{\mathbf{x}}_k, t) \propto e^{iS_k}$.

(3) Composition over all subintervals: By Lemma 2, multiplying these kernels and taking the limit gives

$$K(\hat{\mathbf{x}}, t; \hat{\mathbf{x}}_0, t_0) = \lim_{N\to\infty} \int \prod_{k=1}^{N-1} \mathrm{d}\hat{\mathbf{x}}_k \exp\left(i \sum_{k=0}^{N-1} S_k \right).$$

(4) Continuous limit: As $N \to \infty$, the discrete sum converges to an integral over continuous trajectories:

$$K(\hat{\mathbf{x}}, t; \hat{\mathbf{x}}_0, t_0) = \int \mathcal{D}[\hat{\mathbf{x}}(t)]\, e^{iS[\hat{\mathbf{x}}(t)]},$$

where

$$S[\hat{\mathbf{x}}(t)] = \int_{t_0}^{t} \left[\frac{1}{2} \dot{\hat{x}}^{\,2} - V(\hat{\mathbf{x}}) \right] \mathrm{d}t.$$

Substitution into the evolution of ψ, therefore,

$$\psi(\hat{\mathbf{x}}, t) = \int \mathcal{P}[\hat{\mathbf{x}}(t)]\, \psi(\hat{\mathbf{x}}_0, t_0)\, e^{iS[\hat{\mathbf{x}}(t)]}\, \mathrm{d}\hat{\mathbf{x}}_0,$$

where $\mathcal{P}[\hat{\mathbf{x}}(t)]$ denotes the induced measure over all admissible trajectories from $\hat{\mathbf{x}}_0$ to $\hat{\mathbf{x}}$. The integral representation above coincides with the expression in Eq.(4), thus proving Theorem 1.

4. Formal Proof of Theorem 2

Before presenting the proof of Theorem 2, we introduce two auxiliary lemmas that will be used repeatedly.

Lemma 3 (Divergence Theorem) *Let $\mathbf{F}$ be a continuously differentiable vector field defined on a compact region V with piecewise smooth boundary $S = \partial V$ and outward unit normal $\hat{\mathbf{n}}$. Then the following equality holds:*

$$\oint_S \mathbf{F} \cdot \mathrm{d}\mathbf{S} = \int_V (\nabla \cdot \mathbf{F})\, \mathrm{d}V, \qquad (\blacklozenge)$$

where $\mathrm{d}\mathbf{S} = \hat{\mathbf{n}}\, \mathrm{d}S$ is the oriented surface element.

Brief Proof. For a small rectangular box V with edges $\{\Delta x, \Delta y, \Delta z, \cdots\}$, the flux through opposite faces in each direction gives contributions $\{\partial_x F_x\, \Delta V,\ \partial_y F_y\, \Delta V,\ \partial_z F_z\, \Delta V, \cdots\}$, whose sum equals $(\nabla \cdot \mathbf{F})\, \Delta V$. Summing over all boxes in a partition of V and taking the limit as the maximum box size tends to zero yields the Eq.(♦). This is the standard proof of the divergence theorem.

Intuition. If a vector field represents a flow (such as probability current), then the net outflow across the boundary measures the cumulative rate of sources or sinks inside the region. In the context of probability conservation, this theorem ensures that any local change in probability density is exactly balanced by the net probability flux through the boundary, thus connecting the continuity equation to the global conservation law.

Lemma 4 (Vector Identity) *For any twice continuously differentiable vector field $\mathbf{F}$, the following vector identity holds:*

$$\nabla \times (\nabla \times \mathbf{F}) = \nabla(\nabla \cdot \mathbf{F}) - \nabla^2 \mathbf{F}, \qquad (\blacktriangledown)$$

where $\nabla\times$ denotes the curl, $\nabla\cdot$ the divergence, and ∇^2 the vector Laplacian acting componentwise.

Brief Proof. In index notation using the Levi-Civita symbol ϵ_{ijk} and Einstein summation convention, the i-th component of the left-hand side is

$$[\nabla \times (\nabla \times \mathbf{F})]_i = \epsilon_{ijk} \partial_j (\epsilon_{klm} \partial_l F_m).$$

Using the identity $\epsilon_{ijk}\epsilon_{klm} = \delta_{il}\delta_{jm} - \delta_{im}\delta_{jl}$, we obtain

$$[\nabla\times(\nabla\times\mathbf{F})]_i = \partial_i(\partial_j F_j) - \partial_j\partial_j F_i = \partial_i(\nabla\cdot\mathbf{F}) - (\nabla^2 F_i).$$

Since this holds for each $i = 1, 2, 3$, the vector form Eq.(▼) follows.

Intuition. The operation $\nabla \times (\nabla \times \mathbf{F})$ measures how the curl of a field itself varies spatially, while the right-hand side decomposes this variation into two intuitive parts: the gradient of the field's divergence, representing local expansion or compression, and the negative Laplacian of the field,

representing diffusion or spatial smoothing of its components. This identity underlies many conservation and wave equations.

Proof of Theorem 2. Assume that $\psi(\hat{\mathbf{x}}, t) \in C^2(\mathbb{R}^3 \times [0, \infty))$ and that both ψ and $\nabla\psi$ vanish sufficiently fast at spatial infinity so that all surface integrals below converge. Let the potential $V(\hat{\mathbf{x}}) \in \mathbb{R}$ be real-valued.

Taking the complex conjugate of Eq.(3) yields

$$-i\,\frac{\partial \psi^*}{\partial t} = -\nabla^2\psi^* + V(\hat{\mathbf{x}})\psi^*.$$

Multiplying the original equation by ψ^* and its conjugate by ψ, then subtracting the two eliminates the real potential term:

$$i\left(\psi^*\frac{\partial \psi}{\partial t} + \psi\frac{\partial \psi^*}{\partial t}\right) = -\left(\psi^*\nabla^2\psi - \psi\nabla^2\psi^*\right). \qquad (\spadesuit)$$

Define the probability density $\rho := |\psi|^2 = \psi^*\psi$ and the probability current density

$$\mathbf{j} := \frac{1}{i}\left(\psi^*\nabla\psi - \psi\nabla\psi^*\right).$$

Using the vector identity in Lemma 4, the right-hand side of Eq. (♠) can be written as a divergence:

$$\psi^*\nabla^2\psi - \psi\nabla^2\psi^* = \nabla\cdot(\psi^*\nabla\psi - \psi\nabla\psi^*).$$

Substituting this into Eq. (♠) gives the *continuity equation*:

$$\frac{\partial \rho}{\partial t} + \nabla\cdot\mathbf{j} = 0. \qquad (\clubsuit)$$

Eq.(♣) expresses local conservation of probability: any local decrease in ρ is balanced by outward flow of $\mathbf{j}$.

Boundary and continuity conditions. At a potential discontinuity $\hat{\mathbf{x}} = a$, the continuity of ψ and its derivative,

$$\psi_{\text{left}}(a,t) = \psi_{\text{right}}(a,t), \qquad \left.\frac{\partial \psi_{\text{left}}}{\partial \hat{\mathbf{x}}}\right|_a = \left.\frac{\partial \psi_{\text{right}}}{\partial \hat{\mathbf{x}}}\right|_a,$$

ensures that the probability current is continuous across the interface:

$$\mathbf{j}_{\text{left}}(a,t) = \mathbf{j}_{\text{right}}(a,t).$$

Thus, no probability sources or sinks arise at a. For bound states, the boundary condition $\lim_{\hat{\mathbf{x}}\to\text{boundary}}\psi = \lim_{\hat{\mathbf{x}}\to\text{boundary}}\nabla\psi = 0$ implies

$$\lim_{\hat{\mathbf{x}}\to\text{boundary}} \mathbf{j} = 0.$$

Global conservation. Integrating Eq.(♣) over all space and applying the divergence theorem (Lemma 3) gives

$$\frac{d}{dt}\int_{\mathbb{R}^3}\rho\,d^3r = -\int_{\mathbb{R}^3}\nabla\cdot\mathbf{j}\,d^3r = -\oint_{\partial\mathbb{R}^3}\mathbf{j}\cdot d\mathbf{S}.$$

Because $\mathbf{j} \to 0$ at the spatial boundary, the surface integral vanishes, leading to

$$\frac{d}{dt}\int_{\mathbb{R}^3}|\psi|^2\,d^3r = 0.$$

Hence, the total probability is conserved:

$$\int_{\mathbb{R}^3}|\psi(\hat{\mathbf{x}},t)|^2\,d^3r = 1, \qquad \frac{d}{dt}\int_{\mathbb{R}^3}|\psi(\hat{\mathbf{x}},t)|^2\,d^3r = 0.$$

Therefore, the probability current conservation law stated in Theorem 2 holds.

5. Backbones and Baselines.

For image corruption, we employ WideResNet-28-10 [34] with BatchNorm [14] and ResNet-50 [11] with GroupNorm [31], consistent with TENT [29] and SAR [24]. For domain generalization, we use ResNet-18 following TEA [33]. We evaluate APT alongside the unadapted source model and eleven SoTA TTA methods spanning three categories: normalization-based (BN [27], DUA [22]), pseudo-labeling-based (PL [18], SHOT [20]), entropy-based (TENT [29], ETA [23], EATA [23], SAR [24], CRKD [15], DISTA [1]), and energy-based (TEA [33]). Notably, CRKD and DISTA leverage training data, with CRKD employing a proxy and DISTA directly accessing it during testing, which poses privacy risks.

6. Implementation Details.

Model weight consistency is maintained using pre-trained WideResNet-28-10 (BatchNorm) weights from RobustBench [6] on CIFAR-10. When unavailable, models are trained following the protocol in [34]. The MLP comprises four layers, with hidden layer dimension is 128 for CIFAR-10 and CIFAR-100 and 256 for TinyImageNet-200-C and PACS. The MLP is trained for 1000 epochs. Hyperparameters α and β are assigned 0.01 and 1.0, respectively. We normalize each loss term by dividing it by its detached value to balance their contributions in the total loss. Standard settings mirror those of TENT [29] and TEA [33], while the remaining APT-specific values (omitted from the main text) are detailed in Table.4. The batch size follows the TEA configuration. All our experiments are performed with 3× NVIDIA Tesla P100 and 4× NVIDIA GeForce RTX 3090 Ti GPUs.

7. Detailed of Datasets

We conducted experiments on two tasks across four datasets. As shown in Table.5 and Table.6. The image corruption task utilized clean CIFAR-10, CIFAR-100 [17], and TinyImageNet-200 [7] datasets, alongside their corrupted counterparts, CIFAR-10-C, CIFAR-100-C, and TinyImageNet-200-C [12]. The domain generalization task employed the PACS dataset [19].

Table 4. Summary of Hyperparameters.

	WRN-28-10			ResNet-50			ResNet-18
	CIFAR-10(C)	CIFAR-100(C)	TinyImageNet-200(C)	CIFAR-10(C)	CIFAR-100(C)	TinyImageNet-200(C)	PACS
Left Boundary	-20	-30	-25	-20	-30	-30	-25
Right Boundary	0	0	25	0	0	10	25
LR	0.001	0.002	0.00001	0.00002	0.0001	0.0001	0.001
#Hidden Dim.	128	128	256	128	128	256	128
Batch Size	200	200	1000	200	200	1000	full

Table 5. Clean and corruption datasets overview.

Dataset	**#Train**	**#Test**	**#Corr.**	**#Severity**	**#Class.**
CIFAR-10	50,000	10,000	1	1	10
CIFAR-100	50,000	10,000	1	1	100
Tiny-ImageNet	100,000	10,000	1	1	200
CIFAR-10-C	-	950,000	15	5	10
CIFAR-100-C	-	950,000	15	5	100
Tiny-ImageNet-C	-	750,000	15	5	200

Table 6. PACS datasets overview.

Domain	**#Sample**	**#Class**	**Size**
Photo	1,670	7	3x227x227
Art	2,048	7	3x227x227
Cartoon	2,344	7	3x227x227
Sketch	3,929	7	3x227x227

7.1. Clean Datasets

CIFAR-10, CIFAR-100, and TinyImageNet-200, are used for pre-trained model training. CIFAR-10 contains $60,000$ color images ($3\times32\times32$ pixels) across 10 classes, with 6,000 images per class ($5,000$ training, $1,000$ test). CIFAR-100 includes $60,000$ color images ($3\times32\times32$ pixels) across 100 classes, with 600 images per class (500 training, 100 test). TinyImageNet-200 comprises $110,000$ color images ($3\times64\times64$ pixels) across 200 classes, with 500 training and 50 test images per class.

7.2. Corrupted Datasets

Their corrupted counterparts, CIFAR-10-C, CIFAR-100-C, and TinyImageNet-200-C, are designed to assess model robustness against image corruptions. CIFAR-10-C and CIFAR-100-C each contain $10,000$ test images ($3\times32\times32$ pixels) from their respective clean datasets ($1,000$ per class for CIFAR-10-C, 100 per class for CIFAR-100-C), while TinyImageNet-200-C includes $10,000$ test images ($3\times64\times64$ pixels) across 200 classes (50 per class). Each dataset applies 19 corruption types (*e.g.*, noise, blur, weather effects) at five severity levels, producing 95 corrupted variants per image. The corruptions encompass 15 primary types: Gaussian noise, shot noise, impulse noise, defocus blur, glass blur, motion blur, zoom blur, snow, frost, fog, brightness, contrast, elastic transformation, pixelation, and JPEG compression. These corruptions simulate diverse distributional shifts, such as environmental variations and imaging artifacts, that models may face in real-world applications, thereby testing their generalization and robustness under challenging conditions.

7.3. PACS Dataset

The PACS dataset, designed for domain generalization, comprises $9,991$ images across four distinct domains: Photo, Art Painting, Cartoon, and Sketch. It includes seven object categories (dog, elephant, giraffe, guitar, horse, house, person) with varying image counts per category and domain. Each image is of size $3\times227\times227$ pixels, suitable for evaluating model performance across diverse visual styles and abstractions. The dataset challenges models to generalize across domains with significant stylistic differences, mimicking real-world scenarios where data distributions vary.

8. Additional Experiments

8.1. The Expansion Results for Left-Right Ratio Analysis

This section extends the left-right ratio analysis presented in Experiments section of the main text by examining the relationship between the increase in the left-right ratio and the enhancement in generalizability across 15 distinct types of corruption. Detailed results are provided in Figure 9 and Figure 10, which analyze each corruption type across five severity levels. Specifically, for each corruption type, we investigate the correlation between the magnitude of the left-right ratio increase and the performance metrics, evaluated both pre- and post-adaptation, across increasing severity levels.

Across all corruption types, we observe that performance increases almost linearly with the left–right ratio, thereby establishing a strong coupling between the model's generalization capability and the energy distribution metric. This observation validates the main-text assertion that test data exhibiting lower energy are inherently easier for the classifier. Conversely, for specific corruption types at minimal severity, such as the Defocus corruption at level 1 (Figure 9), our method yields only a marginal improvement in the

Table 7. Results of hyperparameter sensitivity analysis.

Parameter Group	Hyperparameter	Corr Severity 5		Corr Severity 1–5	
		Acc(↑)	mCE(↓)	Acc(↑)	mCE(↓)
Boundary Params	$\tilde{x}_l = -25.0$	86.12	33.50	90.30	45.00
	$\tilde{x}_l = -20.0$	86.93	31.82	91.12	43.48
	$\tilde{x}_l = -15.0$	82.20	45.80	87.05	53.50
	$\tilde{x}_r = -5.0$	83.55	43.50	87.80	52.00
	$\tilde{x}_r = 0.0$	86.93	31.82	91.12	43.48
	$\tilde{x}_r = 5.0$	86.36	32.50	90.70	44.50
Hidden Dimension	$H_{dim} = 64$	86.40	32.40	90.75	44.40
	$H_{dim} = 128$	86.93	31.82	91.12	43.48
	$H_{dim} = 256$	87.02	31.70	91.25	43.30
Loss Weights	$\alpha = 0.001$	80.63	50.32	85.90	59.32
	$\alpha = 0.01$	86.93	31.82	91.12	43.48
	$\alpha = 0.1$	86.70	33.03	90.58	45.02
	$\beta = 0.01$	84.89	40.07	88.85	48.54
	$\beta = 0.1$	86.93	31.82	91.12	43.48
	$\beta = 1.0$	86.44	35.02	89.87	46.56
Infinitesimal Increment	$\delta = 0.0001$	86.93	31.82	91.12	43.48
	$\delta = 0.01$	84.11	42.01	88.35	50.57

left-right ratio. We hypothesize that this limited enhancement stems from the close proximity of these distributions to the source domain, causing the uniform hyperparameters employed during our adaptation to be suboptimal for such minor shifts.

8.2. Detailed Results for Image Corruption

This section complements the main adaptation results in Table 1 of the main text by providing per-corruption performance under the highest severity level, as detailed in Table.8. Experiments on CIFAR-10-C, CIFAR-100-C, and TinyImageNet-200-C show that APT outperforms baselines on most corruption types. We observe that our advantage narrows on less severe corruption types, such as Elastic Transformation, Pixelation, and JPEG Compression. This could stem from the close similarity between training and test distributions, enabling methods like DISTA [1] and CRKD [15] to approximate the correct distribution via direct or indirect access to training data, or allowing pseudo-labeling techniques, such as SHOT [20], to produce accurate labels.

8.3. Hyperparameter Sensitivity Analysis

As reported in Table.7, we perform a comprehensive sensitivity analysis on the key hyperparameters of our model. First, regarding the boundary parameters, we observe a distinct asymmetric sensitivity: moderately relaxing the boundary range (*i.e.*, decreasing $\tilde{x}_l$ or increasing $\tilde{x}_r$) has a negligible impact on performance. Conversely, tightening the boundaries (*i.e.*, increasing $\tilde{x}_l$ or decreasing $\tilde{x}_r$) leads to significant performance degradation. We attribute this to the fact that overly narrow boundaries impede the effective process of energy waveify. Consequently, this leads to the failure of Eq.(11), indicating that the prerequisite for probability current conservation is not met, which ultimately invalidates the theoretical guarantees of Theorem 2.

Second, regarding model capacity, the hidden dimension exhibits a clear trend of diminishing marginal returns. While increasing the dimension from 64 to 128 yields considerable gains, a further increase to 256 does not result in substantial performance improvements. This suggests that selecting an appropriate dimension is crucial for balancing performance with computational overhead.

Finally, concerning the loss weights and infinitesimal increment, results indicate that α and β play critical roles. A reduction in α causes a sharp deterioration in performance, suggesting that these constraints (*i.e.*, continuity and boundary conditions) are essential for stable adaptation. Similarly, the sensitivity analysis of β reveals that merely suppressing high-energy states is insufficient; substantial weight (*i.e.*, a larger β) must be assigned to the probability current conservation constraint to ensure model performance. Furthermore, δ serves as a minute increment at the energy threshold a for enforcing continuity constraints. Our experiments show that an excessively large δ (*e.g.*, 0.01) causes the local continuity assumption to fail, thereby impairing model effectiveness.

Table 8. We comprehensively compare APT and baselines on CIFAR-10-C, CIFAR-100-C, and TinyImageNet-200-C. All models adopt the WRN-28-10 architecture with BatchNorm. For CIFAR-10-C, model weights are obtained from RobustBench to ensure fair comparison. Evaluations report Accuracy (%) per corruption, along with Average Accuracy (Acc %), and Mean Corruption Error (mCE %). Results from original papers are marked with $*$, and our reproductions with †. Best results are highlighted with **boldface**.

	Method	Noise			Blur				Weather			Digital					Avg	
		Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG	Acc(↑)	mCE(↓)
CIFAR-10(-C)	Source	27.68	34.25	27.07	53.01	45.67	65.24	57.99	74.87	58.68	73.98	90.70	53.37	73.39	41.56	69.71	56.47	100.00
	BN	71.93	73.88	63.74	87.19	64.72	85.83	87.89	82.73	82.61	84.75	91.61	87.35	76.25	80.33	72.70	79.56	52.65
	DUA*	72.60	75.40	64.70	86.90	65.10	85.40	88.40	83.20	82.50	86.90	92.40	85.90	77.30	80.70	73.80	80.10	50.78
	PL	14.46	17.77	14.48	44.27	31.29	66.70	60.49	76.46	55.01	79.35	91.05	43.49	75.36	31.61	69.63	51.42	106.98
	SHOT	62.91	66.15	46.41	82.85	61.64	82.39	83.16	79.03	78.24	82.35	89.68	82.20	76.08	75.34	73.19	74.77	63.19
	TENT	75.20	76.52	67.03	88.00	68.22	86.27	89.24	84.06	83.81	86.30	92.16	87.91	78.01	82.77	75.77	81.41	48.13
	ETA	72.21	73.88	63.72	87.21	64.70	85.84	87.89	82.73	82.58	84.77	91.60	87.32	76.22	80.37	72.69	79.58	52.64
	EATA	72.25	73.88	63.74	87.19	64.72	85.83	87.89	82.73	82.61	84.75	91.61	87.35	76.25	80.33	72.70	79.59	52.62
	SAR	71.95	74.14	64.11	87.39	65.20	86.00	88.06	83.08	82.66	85.07	91.90	87.20	76.69	80.41	72.79	79.77	51.94
	CRKD*	76.80	81.60	75.00	86.90	73.50	86.10	89.40	85.60	86.60	87.90	91.70	90.20	80.60	86.10	**82.30**	84.02	42.52
	DISTA†	76.82	79.77	71.90	87.27	72.23	85.36	88.95	83.18	84.46	84.53	90.14	88.56	78.13	85.09	79.36	82.38	47.89
	TEA*	78.33	79.87	70.94	88.89	71.31	87.87	89.77	85.56	85.29	87.61	92.37	88.98	79.32	84.90	78.99	83.33	43.69
	APT	**80.76**	**83.12**	**73.35**	**93.03**	**73.94**	**91.81**	**94.36**	**89.45**	**89.16**	**91.85**	**97.06**	**93.08**	**83.48**	**87.39**	82.13	**86.93**	**31.82**
CIFAR-100(-C)	Source	9.87	11.58	4.15	35.57	20.56	46.92	44.20	51.13	39.07	45.07	71.42	27.43	51.01	30.19	42.82	35.39	100.00
	BN	46.53	48.62	37.15	70.94	47.36	69.04	71.25	63.00	62.96	66.08	75.89	71.31	58.79	64.56	47.46	60.06	63.54
	PL	29.54	34.09	14.99	64.06	40.81	65.36	67.74	60.42	59.47	62.92	75.98	57.29	58.72	59.41	50.34	53.40	72.12
	SHOT	37.50	39.68	21.27	67.90	45.52	67.42	69.86	61.42	60.75	65.01	75.12	63.96	58.96	62.54	51.09	56.53	68.01
	TENT	53.44	54.12	45.53	71.69	51.17	71.54	71.63	64.88	65.30	68.41	75.14	73.59	59.25	66.81	53.93	63.09	59.42
	ETA	48.64	50.95	38.05	69.66	47.52	67.76	70.24	62.51	61.73	66.03	73.40	71.15	56.93	64.40	48.34	59.82	64.52
	EATA	48.76	51.60	39.47	69.29	47.49	68.13	70.68	62.94	62.53	65.14	74.48	71.61	57.53	64.24	49.67	60.24	63.75
	SAR	51.34	54.08	44.62	72.24	50.10	71.06	72.43	64.96	65.35	68.40	76.23	73.95	60.04	67.26	52.29	62.95	59.37
	CRKD†	50.63	53.06	46.04	71.54	52.32	70.73	72.68	65.17	65.07	68.64	74.8	74.79	60.92	67.5	54.18	63.20	59.17
	DISTA†	55.79	57.98	47.51	73.64	53.84	72.17	73.03	66.07	66.72	68.61	75.77	73.76	60.22	69.06	56.49	64.71	56.99
	TEA*	54.29	56.55	48.59	72.96	53.78	72.63	74.20	67.78	67.14	69.98	76.74	**75.71**	62.18	68.65	55.32	65.10	56.18
	TEA	**59.72**	**59.81**	**51.00**	**75.84**	**55.92**	**74.73**	**76.68**	**69.93**	**69.62**	**71.50**	**80.12**	70.39	**65.44**	**72.30**	**60.20**	**67.55**	**51.81**
Tiny-ImageNet(-C)	Source	12.37	16.07	5.55	6.63	5.67	20.71	19.38	26.79	31.88	14.18	29.75	1.95	32.03	49.00	46.19	21.21	100.00
	BN	24.00	25.82	18.22	28.43	18.46	35.20	33.41	27.59	30.26	25.58	32.58	6.04	34.12	40.81	35.72	27.74	93.42
	PL	20.75	25.99	11.41	17.28	14.42	36.07	33.66	30.19	35.01	22.44	34.87	2.03	**41.31**	51.76	46.75	28.26	91.22
	SHOT	22.25	27.10	14.54	18.74	15.17	36.78	34.63	30.95	**35.68**	23.52	36.01	1.91	41.01	**51.99**	**46.85**	29.14	90.16
	TENT	23.80	25.29	18.68	27.05	17.54	33.21	31.36	26.21	28.56	24.76	30.08	5.68	31.37	37.77	33.33	26.31	95.52
	ETA	24.44	25.76	18.00	28.05	18.25	34.59	32.85	27.11	29.87	25.42	31.86	5.93	32.84	39.55	34.82	27.28	94.12
	EATA	24.48	25.55	17.95	27.68	18.41	34.83	32.84	27.03	29.67	25.39	31.94	5.87	32.71	39.87	35.09	27.28	94.09
	SAR	25.10	26.33	18.87	28.80	18.44	35.59	33.87	28.31	30.62	26.32	33.31	6.30	34.24	41.16	35.97	28.21	92.82
	CRKD†	23.65	26.60	18.70	32.26	20.47	39.13	**41.76**	30.67	31.73	26.44	34.08	4.61	39.36	43.43	39.13	30.13	90.10
	DISTA†	24.29	28.61	21.86	31.93	22.21	42.22	39.69	29.94	33.06	29.34	35.58	6.45	40.07	45.4	42.51	31.54	88.18
	TEA*	25.50	29.35	20.58	32.76	21.14	41.72	39.06	31.43	34.19	29.02	36.83	6.89	38.94	46.38	41.31	31.67	87.99
	APT	**29.49**	**30.36**	**24.17**	**35.71**	**24.10**	**42.89**	39.35	**32.60**	35.15	**30.28**	**37.18**	**7.82**	39.97	47.86	42.46	**33.29**	**85.98**

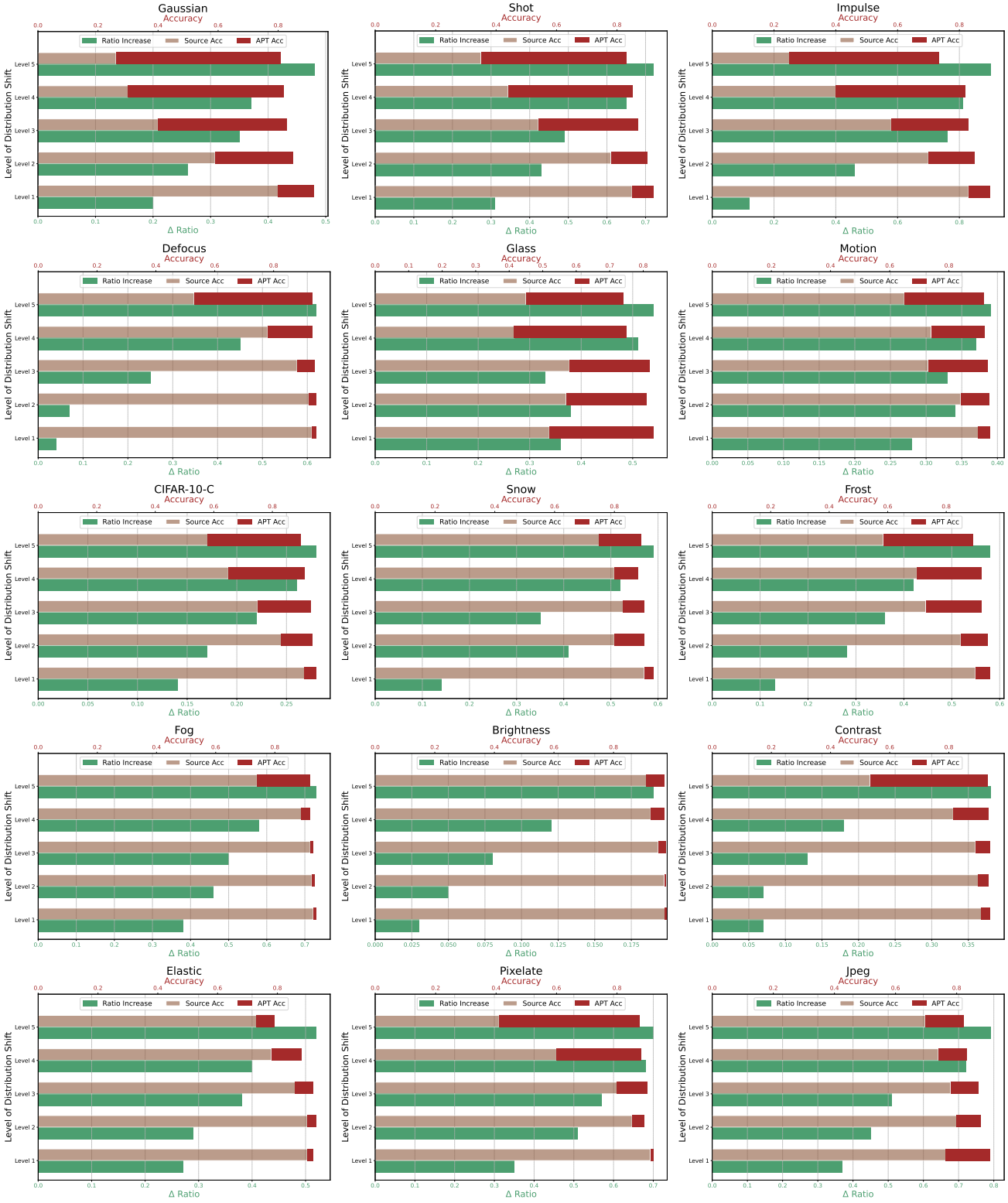

Figure 9. The correlation between the APT ratio increase and accuracy gains on CIFAR-10-C across various distribution types and shift severities. In each sub-plot, corruption severity is placed on the y-axis, the APT ratio increase on the lower x-axis, and accuracy on the upper x-axis. Within the accuracy scale, the solid red bar marks APT, whereas the translucent bar indicates the baseline.

Figure 10. The correlation between the APT ratio increase and accuracy gains on CIFAR-100-C across various distribution types and shift severities. In each sub-plot, corruption severity is placed on the y-axis, the APT ratio increase on the lower x-axis, and accuracy on the upper x-axis. Within the accuracy scale, the solid red bar marks APT, whereas the translucent bar indicates the baseline.

References

[1] Motasem Alfarra, Alvaro Correia, Bernard Ghanem, and Christos Louizos. Test-time adaptation with source based auxiliary tasks. *Transactions on Machine Learning Research*, 2025. 4, 6

[2] L Brunke et al. Safe learning in robotics: from learning-based control to safe reinforcement learning (2021). *arXiv preprint arXiv:2108.06266*. 1

[3] Ziyang Chen, Yongsheng Pan, Yiwen Ye, Mengkang Lu, and Yong Xia. Each test image deserves a specific prompt: Continual test-time adaptation for 2d medical image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11184–11193, 2024. 1

[4] Richard Cheng, Gábor Orosz, Richard M Murray, and Joel W Burdick. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. In *Proceedings of the AAAI conference on artificial intelligence*, pages 3387–3395, 2019. 1

[5] Paul R Chernoff. *Product formulas, nonlinear semigroups, and addition of unbounded operators*. American Mathematical Soc., 1974. 2

[6] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020. 4

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 4

[8] Mahyar Fazlyab, Alejandro Ribeiro, Manfred Morari, and Victor M Preciado. Analysis of optimization algorithms via integral quadratic constraints: Nonstrongly convex problems. *SIAM Journal on Optimization*, 28(3):2654–2689, 2018. 1

[9] Chris Finlay, Jörn-Henrik Jacobsen, Levon Nurbekyan, and Adam Oberman. How to train your neural ode: the world of jacobian and kinetic regularization. In *International conference on machine learning*, pages 3154–3164. PMLR, 2020. 1

[10] 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. 1

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[12] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 4

[13] Junyuan Hong, Lingjuan Lyu, Jiayu Zhou, and Michael Spranger. Mecta: Memory-economic continual test-time model adaptation. In *2023 International Conference on Learning Representations*, 2023. 1

[14] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015. 4

[15] Juwon Kang, Nayeong Kim, Donghyeon Kwon, Jungseul Ok, and Suha Kwak. Leveraging proxy of training data for test-time adaptation. 2023. 4, 6

[16] Adilbek Karmanov, Dayan Guan, Shijian Lu, Abdulmotaleb El Saddik, and Eric Xing. Efficient test-time adaptation of vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14162–14171, 2024. 1

[17] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 4

[18] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, page 896. Atlanta, 2013. 4

[19] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017. 4

[20] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International conference on machine learning*, pages 6028–6039. PMLR, 2020. 4, 6

[21] Guan-Horng Liu and Evangelos A Theodorou. Deep learning theory review: An optimal control and dynamical systems perspective. *arXiv preprint arXiv:1908.10920*, 2019. 1

[22] M Jehanzeb Mirza, Jakub Micorek, Horst Possegger, and Horst Bischof. The norm must go on: Dynamic unsupervised domain adaptation by normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14765–14775, 2022. 4

[23] 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*, pages 16888–16905. PMLR, 2022. 1, 4

[24] 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. 1, 4

[25] Sunghyun Park, Seunghan Yang, Jaegul Choo, and Sungrack Yun. Label shift adapter for test-time adaptation under covariate and label shifts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16421–16431, 2023. 1

[26] Hongbin Ren, Yunong Li, Yang Wang, Chih-Keng Chen, Lin Yang, and Yuzhuang Zhao. Learning-based model predictive control for safe path planning and control. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 239(9):3991–4004, 2025. 1

[27] Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *Advances in neural information processing systems*, 33:11539–11551, 2020. 4

[28] Soumyabrata Talukder and Ratnesh Kumar. Robust stability of neural-network-controlled nonlinear systems with parametric variability. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 53(8):4820–4832, 2023. 1

[29] 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. 1, 4

[30] 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*, pages 7201–7211, 2022. 1

[31] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018. 4

[32] Xiao Yang, Jiyao Wang, Yuxuan Fan, Can Liu, Houcheng Su, Weichen Guo, Zitong Yu, Dengbo He, and Kaishun Wu. Not only consistency: Enhance test-time adaptation with spatio-temporal inconsistency for remote physiological measurement. *arXiv preprint arXiv:2507.07908*, 2025. 1

[33] Yige Yuan, Bingbing Xu, Liang Hou, Fei Sun, Huawei Shen, and Xueqi Cheng. Tea: Test-time energy adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23901–23911, 2024. 1, 4

[34] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016. 4

[35] Luca Zancato, Alessandro Achille, Tian Yu Liu, Matthew Trager, Pramuditha Perera, and Stefano Soatto. Train/test-time adaptation with retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15911–15921, 2023. 1

[36] Xiaoran Zhang, Byung-Woo Hong, Hyoungseob Park, Daniel H Pak, Anne-Marie Rickmann, Lawrence H Staib, James S Duncan, and Alex Wong. Progressive test time energy adaptation for medical image segmentation. *arXiv preprint arXiv:2503.16616*, 2025. 1

[37] Lihua Zhou, Mao Ye, Shuaifeng Li, Nianxin Li, Xiatian Zhu, Lei Deng, Hongbin Liu, and Zhen Lei. Bayesian test-time adaptation for vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29999–30009, 2025. 1