Test-Time Adaptation with Binary Feedback

Taeckyung Lee [1] **Sorn Chottananurak** [1] **Junsu Kim** [1] **Jinwoo Shin** [1] **Taesik Gong** [2 *] **Sung-Ju Lee** [1 *]

Abstract

Deep learning models perform poorly when domain shifts exist between training and test data. Test-time adaptation (TTA) is a paradigm to mitigate this issue by adapting pre-trained models using only unlabeled test samples. However, existing TTA methods can fail under severe domain shifts, while recent active TTA approaches requiring full-class labels are impractical due to high labeling costs. To address this issue, we introduce a new setting of TTA with binary feedback. This setting uses a few binary feedback inputs from annotators to indicate whether model predictions are correct, thereby significantly reducing the labeling burden of annotators. Under the setting, we propose BiTTA, a novel dual-path optimization framework that leverages reinforcement learning to balance binary feedback-guided adaptation on uncertain samples with agreement-based self-adaptation on confident predictions. Experiments show BiTTA achieves 13.3%p accuracy improvements over state-of-the-art baselines, demonstrating its effectiveness in handling severe distribution shifts with minimal labeling effort. The source code is available at https://github.com/taeckyung/BiTTA.

1. Introduction

Deep learning has revolutionized various fields, including computer vision (Deng et al., 2009), speech recognition (Gulati et al., 2020), and natural language processing (Brown et al., 2020). However, deep models often suffer from domain shifts, where discrepancies between training and test data distributions lead to significant performance degradation. For example, autonomous driving systems might struggle with new types of vehicles or unexpected weather conditions that differ from the training data (Sakaridis et al., 2018).

Test-time adaptation (TTA) (Wang et al., 2021) dynamically adapts the pre-trained models in real time using only unlabeled test samples. Hence, it is a viable solution to domain shifts. However, without ground-truth labels, existing TTA methods are vulnerable to adaptation failures or suboptimal accuracies in severe distribution shifts (Gong et al., 2022; Niu et al., 2023; Gong et al., 2023b; Lee et al., 2024b; Press et al., 2023). This is largely due to relying on self-supervised metrics such as entropy or confidence (Wang et al., 2021; Gong et al., 2023b) that could be unreliable when adaptation fails (Lee et al., 2024b).

This limitation motivates the potential value of incorporating human feedback into the adaptation loop. In language models, reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022) has emphasized the critical role of human feedback in aligning large language models with human intent. The key challenge in TTA is making human feedback practical and scalable. While full labeling of test samples is prohibitively expensive (Joshi et al., 2010), recent studies on language models have shown that binary feedback (e.g., thumbs up/down) can effectively guide model behaviors (Shankar et al., 2024).

Inspired by this, we introduce the **TTA with binary feedback** setting (Figure 1), which uses binary feedback on the model prediction (*correct* or *incorrect*) to guide adaptation while maintaining efficiency. Our binary-feedback approach requires only minimal label information, significantly reducing labeling costs compared with full-class active TTA (Gui et al., 2024), mitigating the interaction bottlenecks to enable real-world applications.

As a solution, we propose **BiTTA**, a dual-path optimization framework for TTA with binary feedback that incorporates both binary feedback and unlabeled samples. Motivated by the recent reinforcement learning studies that show effectiveness in incorporating human feedback in the optimization process (Ouyang et al., 2022; Fan et al., 2023; Black et al., 2024), BiTTA leverages reinforcement learning to effectively balance two complementary adaptation strategies (Figure 3): *Binary Feedback-guided Adaptation (BFA)* on uncertain samples and *Agreement-Based self-Adaptation (ABA)* on confident samples. Using Monte Carlo dropout (Gal &

*Corresponding authors. [1]KAIST [2]UNIST. Correspondence to: Taesik Gong <taesik.gong@unist.ac.kr>, Sung-Ju Lee <profsj@kaist.ac.kr>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

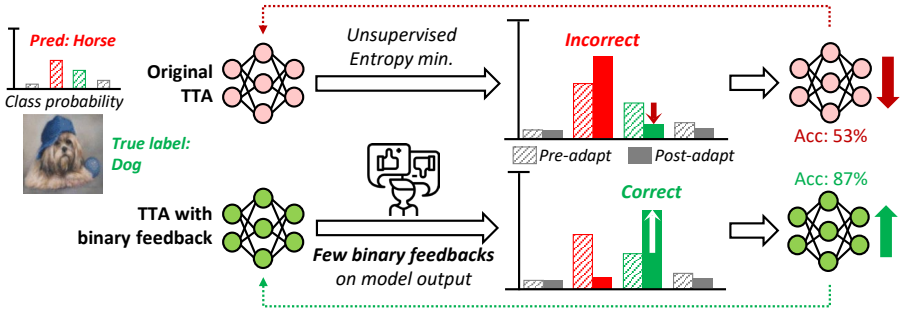


Figure 1: Overview of TTA with binary feedback. Traditional TTA algorithms often fail under severe distribution shifts due to the risk of unlabeled-only adaptation. Our proposed TTA with binary feedback addresses this challenge by offering a few binary feedback (*correct* or *incorrect*) on selected model predictions. TTA with binary feedback significantly improves the adaptation performance with minimal labeling effort, enabling a practical and scalable TTA paradigm for real-world applications.

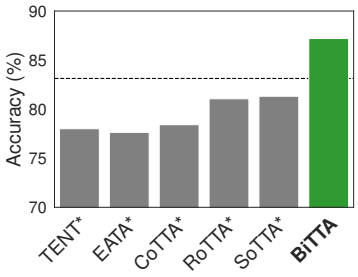


Figure 2: Accuracy (%) of TTA methods with binary feedback on CIFAR10-C. The asterisk indicates a modified algorithm to utilize binary feedback. The dotted line is full-class active TTA (SimATTA).

Ghahramani, 2016) for policy estimation and uncertainty assessment, we select uncertain samples for binary feedback in BFA while utilizing samples with high prediction agreement in ABA. This dual approach enables BiTTA to adapt to new uncertain patterns (via BFA) while maintaining confidence in correct predictions (via ABA), achieving robust performance improvements.

We evaluate BiTTA under the TTA with binary feedback setting with various test-time distribution shift scenarios, including three image corruption datasets (CIFAR10-C, CIFAR100-C, and Tiny-ImageNet-C) and two domain generalization scenarios (domain-wise and mixed data streams). Comparisons with TTA and active TTA methods demonstrate that BiTTA achieves an accuracy improvement of 13.3%p on average. Moreover, BiTTA outperforms active TTA utilizing full-class feedback from the oracle (the ground-truth label, Figure 2) and the state-of-the-art foundational model (GPT-4o, Figure 5), despite using only binary feedback. These results highlight the importance and effectiveness of BiTTA and TTA with binary feedback, enabling robust adaptation with minimal labeling effort.

Our main contributions are as follows:

- We propose a lightweight and scalable setting of TTA with binary feedback (Figure 1), offering *correct* or *incorrect* feedback on selected model predictions.
- We develop BiTTA, a dual-path optimization strategy combining two complementary signals from binary feedback-guided adaptation and agreement-based self-adaptation with a reinforcement learning framework.
- We perform extensive experiments that show BiTTA outperforms TTA and active TTA baselines by 13.3%p. Comparisons with full-class active TTA indicate the effectiveness of BiTTA and TTA with binary feedback.

2. Test-Time Adaptation with Binary Feedback

We propose TTA with binary feedback, a novel TTA setting for adapting pre-trained models during test time using binary feedback from an oracle, which indicates whether a model's predictions are correct (Figure 1). This real-time feedback seamlessly integrates into the adaptation process, enabling continuous refinement of the model's performance.

TTA with binary feedback addresses the critical challenge of adapting pre-trained models to domain shifts with minimal labeling effort. From an information-theory perspective, full-class labeling requires $\log(\texttt{num\_class})$ times more bits than binary feedback to encode the same information (MacKay, 2003). Empirical studies on human annotation also validate the efficiency of binary feedback: full-class labeling on 50-class takes 11.7 seconds on average with a 12.7% error rate, whereas binary comparison requires only 1.6 seconds with 0.8% error rate (Joshi et al., 2010).

Although binary feedback provides only a single bit, the feedback is based on the adapted model's prediction, which is typically better than chance. This makes the feedback more informative and allows it to directly guide model behaviors. Therefore, TTA with binary feedback is an efficient and practical framework for real-world TTA applications.

Notation. Let x denote a test sample selected for binary-feedback labeling, and $y^* = \arg\max_y f_\theta(y|x)$ is the class prediction output of the model parameterized by θ. The binary feedback $B(x, y)$ is defined as:

$$B(x,y) = \begin{cases} 1 & \text{if } y \text{ is correct,} \\ -1 & \text{if } y \text{ is incorrect.} \end{cases} \quad (1)$$

Accordingly, each binary-feedback sample is represented as the tuple $(x, y^*, B(x, y^*))$, consisting of the input instance, the model's predicted label, and the binary feedback.

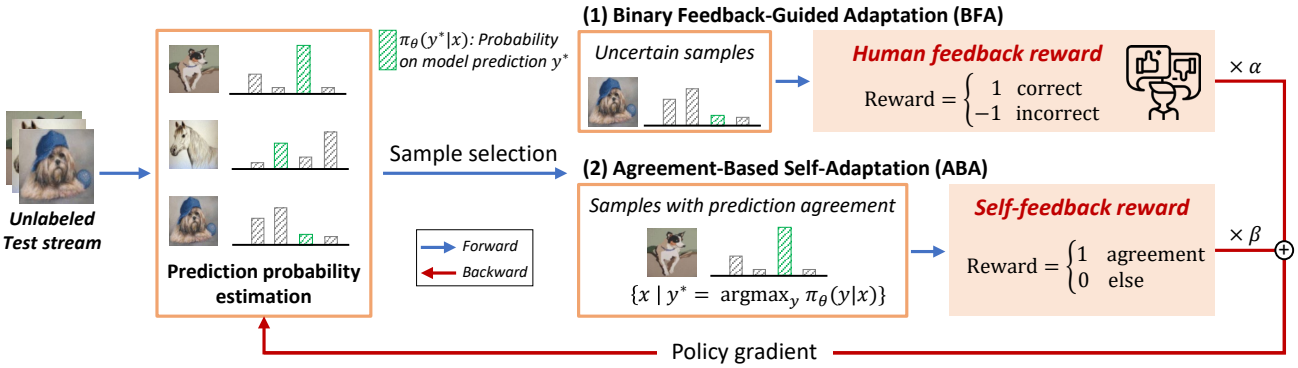


Figure 3: Overview of BiTTA algorithm. BiTTA implements a reinforcement learning-based dual-path optimization that estimates prediction probabilities using MC-dropout. It computes policy gradients from two complementary signals: (1) Binary Feedback-guided Adaptation (BFA) on uncertain samples, using binary rewards of ± 1, and (2) Agreement-Based self-Adaptation (ABA) on confident, unlabeled samples, using reward 1. By jointly optimizing both paths, BiTTA enables robust adaptation under dynamic distribution shift scenarios.

3. BiTTA: Dual-path Optimization to Learn with Binary Feedback

Motivation. Recent advancements in reinforcement learning from human feedback (RLHF) (Ouyang et al., 2022) have demonstrated the effectiveness of incorporating sparse feedback signals in large language model training. Inspired by this, we propose BiTTA, a reinforcement learning (RL) based approach for TTA with binary feedback that effectively adapts to distribution shifts using minimal labeling effort (Figure 3). BiTTA leverages binary feedback as a reinforcement signal, offering several key advantages for TTA. (i) Binary feedback can be seamlessly incorporated as non-differentiable rewards in the RL framework, enabling the model to learn from minimal supervision (Zoph & Le, 2017; Yoon et al., 2020). (ii) The RL framework allows for integrating binary feedback and unlabeled samples into a single objective function optimized through policy gradient methods. By combining sparse binary-feedback samples with unlabeled data, BiTTA provides a robust framework with minimal labeling effort, making TTA more feasible for real-world applications.

Policy gradient modeling. Given a batch of test samples $\mathcal{B} = \{x_1, \ldots, x_n\}$, our goal is to adapt the model parameters θ to improve performance on the test distribution. We formulate the test-time adaptation process as an RL problem by assigning test-time input $x \sim \mathcal{B}$ as a state, the model prediction $y^* = f_\theta(x)$ as an action, and the corresponding prediction probability $\pi_\theta(y|x)$ as a policy, which objective is maximizing the expected reward, defined as:

$$J(\theta) = \mathbb{E}_{x\sim\mathcal{B}, y\sim\pi_\theta(y|x)}[R(x, y)], \tag{2}$$

where $R(x, y)$ represents the reward function defined later. This optimization is performed for each test batch, allowing continuous adaptation to the evolving test distribution.

As binary feedback is a non-differentiable function, we employ the REINFORCE algorithm (Williams, 1992), also known as the "log-derivative trick". This method allows us to estimate the gradient of the expected reward with respect to the model parameters:

$$\nabla_\theta J(\theta) = \mathbb{E}_{x\sim\mathcal{B}, y\sim\pi_\theta(y|x)}[R(x, y)\nabla_\theta \log \pi_\theta(y|x)]. \tag{3}$$

By using this gradient estimator, we can effectively optimize our model parameters using stochastic gradient ascent.

To estimate the policy π_θ, we adopt Monte Carlo (MC) dropout (Gal & Ghahramani, 2016), a practical Bayesian approximation technique that enables robust uncertainty estimation without architectural changes. MC-dropout performs multiple stochastic forward passes with dropout enabled at test time, allowing the model to capture epistemic uncertainty in its predictions.

Formally, we approximate the policy $\pi_\theta(y|x)$ as the mean of softmax outputs across N stochastic forward passes:

$$\pi_\theta(y|x) = \frac{1}{N}\sum_{n=1}^{N} f_\theta^d(y|x), \tag{4}$$

where f_θ^d denotes the model with dropout applied during inference, and N is the number of samples drawn.

This approach is crucial for test-time adaptation under distribution shift. Unlike standard softmax outputs, which are often miscalibrated and overconfident on out-of-distribution (OOD) samples, MC-dropout provides well-calibrated confidence estimates. These improved uncertainty estimates are key to BiTTA's dual-path strategy: selecting the uncertain samples for feedback (BFA) and identifying confidently

predicted ones for self-adaptation (ABA). We empirically validate the calibration benefits of MC-dropout in Section 4.

Dual-path optimization strategy. With the proposed RL framework, BiTTA addresses the challenge of TTA with binary feedback (Figure 3), utilizing (1) *few samples with ground-truth binary feedback* and (2) *many unlabeled samples with potentially noisy predictions* by two complementary strategies:

1. Binary Feedback-guided Adaptation on uncertain samples (BFA, Section 3.1): This strategy focuses on enhancing the model's areas of uncertainty. By selecting samples where the model is least confident and obtaining binary feedback on these, BiTTA efficiently probes the boundaries of the model's current knowledge.

2. Agreement-Based self-Adaptation on confident samples (ABA, Section 3.2): To complement the guided adaptation strategy, BiTTA also leverages the model's existing knowledge through self-adaptation on confidently predicted samples. Without requiring additional feedback, ABA identifies confident samples by the agreement between the model's standard predictions and those obtained via MC-dropout.

The synergy between BFA and ABA enables BiTTA to effectively utilize both labeled and unlabeled samples. BFA drives exploration and adaptation to new patterns in the test distribution through binary feedback on uncertain samples. Concurrently, ABA maintains and refines existing knowledge through self-supervised adaptation on confident predictions. This dual-path optimization allows for effective adaptation across diverse challenging conditions.

3.1. Binary Feedback-Guided Adaptation

In TTA with binary feedback settings where binary feedback is limited and costly, selecting samples to query and using them for effective model adaptation becomes crucial. To address this challenge, we propose Binary Feedback-guided Adaptation on uncertain samples (BFA). This approach refines the model's decision boundaries and improves its understanding of challenging data points through binary feedback guidance, enabling robust and efficient adaptation in test-time distribution shifts.

Sample selection. To guide the adaptation, we focus on the uncertain samples, often the most informative for model improvement (Settles, 2009). We quantify the sample-wise (un)certainty $C(x)$ as the MC-dropout softmax of the original model predicted class:

$$C(x) = \pi_\theta(y^*|x), \tag{5}$$

where $y^* = \arg\max_y f_\theta(y|x)$ is the deterministic class prediction and $\pi_\theta(y|x)$ is MC-dropout softmax confidence. Then, we select the set of k samples to get the binary feedback, noted as $\mathcal{S}_{\texttt{BFA}}$. One straightforward strategy is to select the least confident samples:

$$\mathcal{S}_{\texttt{BFA}} = \texttt{argsort}_x(C(x))[:k]. \tag{6}$$

We further discuss the BFA sample selection strategy and its impact in Appendix B.

Reward function design. For these selected samples, we query the binary feedback $B(x, y)$ (correct/incorrect) and define the reward function $R_{\texttt{BFA}}$ for feedback samples as:

$$R_{\texttt{BFA}}(x, y) = B(x, y) = \begin{cases} 1 & \text{if } y \text{ is correct,} \\ -1 & \text{otherwise.} \end{cases} \tag{7}$$

This binary-feedback reward scheme provides a clear signal for model adaptation, encouraging the prediction probability of correct predictions and penalizing incorrect ones. By selectively applying this reward function to the most uncertain samples, BFA efficiently utilizes the limited labeling budget, maximizing the contribution of each queried label.

3.2. Agreement-Based Self-Adaptation

To complement the binary feedback-guided adaptation on uncertain samples, we propose leveraging the model's confident predictions on the remaining many unlabeled samples. This approach, which we call Agreement-Based self-Adaptation (ABA), aims to reinforce the model's current knowledge without requiring additional oracle feedback.

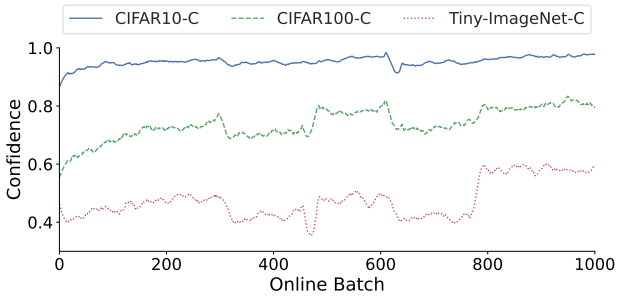
Sample selection. The key idea behind ABA is to identify samples where the model's standard prediction agrees with its MC-dropout prediction. We consider these samples "confident" and use them for self-adaptation. Formally, we define the set of confident samples $\mathcal{S}_{\texttt{ABA}}$ as:

$$\mathcal{S}_{\texttt{ABA}} = \{x \in \mathcal{B} \setminus \mathcal{S}_{\texttt{BFA}} \mid y^* = \arg\max_y \pi_\theta(y|x)\}, \tag{8}$$

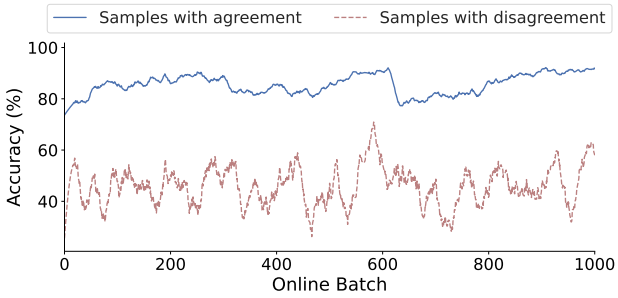
where $\mathcal{B}$ is the entire batch of test samples, $\mathcal{S}_{\texttt{BFA}}$ is the set of samples selected for active feedback, y^* is the original class prediction, and $\pi_\theta(y|x)$ is the MC-dropout softmax.

Unlike existing TTA methods that rely on fixed confidence thresholds (Niu et al., 2022; 2023; Gong et al., 2023b), our approach can dynamically select confident samples based on the agreement between standard and MC-dropout predictions. Figure 4a illustrates the dynamic nature of prediction confidences during distribution shifts—highlighting the need for dynamic sample selection. To demonstrate the effectiveness of ABA further, we compare our agreement-based approach with thresholding strategies (Figure 8 in Appendix B). The results motivate our dynamic agreement-based selection over static confidence thresholding.

Furthermore, our method effectively identifies confident samples for self-adaptation. Figure 4b demonstrates the



(a) Average sample-wise confidence in three image corruption datasets.



(b) Average sample-wise accuracy of samples with agreement ($\mathcal{S}_{\texttt{ABA}}$) and disagreement ($\mathcal{B} \setminus \mathcal{S}_{\texttt{ABA}}$).

Figure 4: Analysis of confidence and accuracy during online adaptation. (a) Average sample-wise confidence over time and dataset, showing dynamic changes that challenge fixed thresholding methods. (b) Average sample-wise accuracy for samples with prediction agreement and disagreement on CIFAR10-C, demonstrating the effectiveness of agreement-based selection for confident samples.

stable accuracies in samples with agreement, while samples with disagreement show unstable and low accuracies. This originates from the prediction agreement of indicating robustness and reliability via the consistency in model outputs across different dropout masks. By leveraging this consistency, ABA can reliably select confident samples for effective self-adaptation.

Reward function design. We now incorporate these samples into our reinforcement learning framework. We introduce a self-feedback reward function $R_{\texttt{ABA}}$ for unlabeled samples. This reward encourages the model to maintain its predictions on confident samples while discarding the adaptation on unreliable ones. Formally, we define $R_{\texttt{ABA}}$ as:

$$R_{\texttt{ABA}}(x, y) = \begin{cases} 1 & \text{if } x \in \mathcal{S}_{\texttt{ABA}}, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

By incorporating this adaptive prediction agreement strategy, ABA enhances the learning process by maintaining the knowledge of confident predictions. While prediction disagreement suggests uncertainty, our analysis shows disagreement samples exhibit unstable accuracy rather than consistent errors (Figure 4b). Therefore, ABA assigns zero rewards to them rather than penalizing (as in BFA), preventing potentially harmful adaptation from noisy signals. This conservative approach is especially valuable in TTA scenarios where distribution shift may be partial or gradual, where most of the model's existing knowledge remains relevant.

3.3. BiTTA Algorithm

Our proposed BiTTA algorithm unifies binary feedback-guided adaptation (BFA, Section 3.1) and agreement-based self-adaptation (ABA, Section 3.2) into a dual-path optimization framework. This design enables robust and efficient test-time adaptation under distribution shifts while maintaining stability across time. Here, we detail the algorithmic formulation and its practical implementation using memory-based sample buffers.

Combined objective. We formulate the overall objective as a weighted sum of expected rewards from BFA and ABA:

$$J(\theta) = \alpha \mathbb{E}_{x \in \mathcal{S}_{\text{BFA}}} \left[R_{\text{BFA}}(x, y) \right] + \beta \mathbb{E}_{x \in \mathcal{S}_{\text{ABA}}} \left[R_{\text{ABA}}(x, y) \right], \tag{10}$$

where α and β balance BFA and ABA. We set $\alpha = \beta = 1$ in all experiments and analyze the impact of these hyperparameters in Figure 10 (Appendix B).

The policy gradient is estimated via REINFORCE (Williams, 1992):

$$\begin{aligned} \nabla_\theta J(\theta) = \alpha \mathbb{E}_{x \in \mathcal{S}_{\text{BFA}}} \left[R_{\text{BFA}}(x, y) \nabla_\theta \log \pi_\theta(y|x) \right] \\ + \beta \mathbb{E}_{x \in \mathcal{S}_{\text{ABA}}} \left[\nabla_\theta \log \pi_\theta(y|x) \right], \end{aligned} \tag{11}$$

where $\pi_\theta(y|x)$ is approximated using MC-dropout.

Practical implementation. We implement the policy gradient in Equation 11 using two FIFO memory pools: $\mathcal{M}_{\text{C}}$ for correct predictions and $\mathcal{M}_{\text{I}}$ for incorrect ones. Each memory stores up to a batch-sized number of samples for computational stability and efficiency. Additionally, we use $\mathcal{S}_{\text{ABA}}$, confident unlabeled samples identified via agreement on the current batch.

For each input x with predicted label y^*, we optimize the MC-dropout softmax probability $\pi_\theta(y^*|x)$ by minimizing or maximizing the cross-entropy depending on its type. The overall loss is defined as:

$$\begin{aligned} \mathcal{L}_{\text{BiTTA}} = \alpha \cdot \frac{1}{|\mathcal{M}_{\text{C}}|} \sum_{x \in \mathcal{M}_{\text{C}}} \left(-\log \pi_\theta(y^*|x) \right) \\ + \alpha \cdot \frac{1}{|\mathcal{M}_{\text{I}}|} \sum_{x \in \mathcal{M}_{\text{I}}} \left(+\log \pi_\theta(y^*|x) \right) \\ + \beta \cdot \frac{1}{|\mathcal{S}_{\text{ABA}}|} \sum_{x \in \mathcal{S}_{\text{ABA}}} \left(-\log \pi_\theta(y^*|x) \right). \end{aligned} \tag{12}$$

Algorithm 1 BiTTA Algorithm

Input: Model f_θ, MC-dropout π_θ, batch $\mathcal{B}$, correct memory $\mathcal{M}_C$, incorrect memory $\mathcal{M}_I$, budget k, epochs E, learning rate η, balancing hyperparameters α, β
`# Binary Feedback Sample Selection`
$Y^* \leftarrow \arg\max f_\theta(*\,|\,\mathcal{B})$ // Deterministic prediction
$I_{\texttt{BFA}} \leftarrow \texttt{argsort}_x(\pi_\theta(Y^*\,|\,\mathcal{B}))[:k]$ // Equation 6
$\mathcal{S}_{\texttt{BFA}} \leftarrow \{(x_i, y_i^*) \mid i \in I_{\texttt{BFA}}\}$ // BFA samples
for each (x, y) in $\mathcal{S}_{\texttt{BFA}}$ **do**
  $B(x, y) \leftarrow$ Binary feedback from oracle
  if $B(x, y) = 1$ **then**
    $\mathcal{M}_C$.update(x, y) // Store correct sample
  else
    $\mathcal{M}_I$.update(x, y) // Store incorrect sample
  end if
end for
`# Test-Time Adaptation`
Update BN stats with $\mathcal{B}$ and freeze them
for $e = 1$ to E **do**
  `# BFA (Section 3.1)`
  $X_C, Y_C \leftarrow \mathcal{M}_C$ // Get correct samples
  $X_I, Y_I \leftarrow \mathcal{M}_I$ // Get incorrect samples
  $\mathcal{L}_{\texttt{BFA}} \leftarrow \ell_{\texttt{CE}}(\pi_\theta(* \mid X_C), Y_C) - \ell_{\texttt{CE}}(\pi_\theta(* \mid X_I), Y_I)$
  `# ABA (Section 3.2)`
  $X_U \leftarrow \mathcal{B} \setminus \mathcal{S}_{\texttt{BFA}}$ // Get unlabeled samples
  $R_{\texttt{ABA}} \leftarrow \mathbb{1}[\arg\max f_\theta(*\,|\,X_U) = \arg\max \pi_\theta(*\,|\,X_U)]$
  $X_{\texttt{ABA}} \leftarrow \{x \in X_U \mid R_{\texttt{ABA}}(x) = 1\}$,
  $Y_{\texttt{ABA}} \leftarrow \arg\max f_\theta(* \mid X_{\texttt{ABA}})$ // Get $\mathcal{S}_{\texttt{ABA}}$
  $\mathcal{L}_{\texttt{ABA}} \leftarrow \ell_{\texttt{CE}}(\pi_\theta(* \mid X_{\texttt{ABA}}), Y_{\texttt{ABA}})$
  `# Final Update (Section 3.3)`
  $\mathcal{L}_{\texttt{BiTTA}} \leftarrow \alpha \cdot \mathcal{L}_{\texttt{BFA}} + \beta \cdot \mathcal{L}_{\texttt{ABA}}$
  $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\texttt{BiTTA}}$
end for

Based on the unified objective $\mathcal{L}_{\texttt{BiTTA}}$, Algorithm 1 outlines the full adaptation procedure. The algorithm proceeds in two phases. In the first phase, we select the top-k uncertain samples (or random samples in large-scale additional studies) from the current batch based on MC-dropout confidence scores over the predicted class. We then query binary feedback from an oracle and update the corresponding FIFO memory: $\mathcal{M}_{\texttt{C}}$ (correct) or $\mathcal{M}_{\texttt{I}}$ (incorrect). This separation ensures balanced and stable adaptation, particularly during early rounds when labeled feedback is scarce or skewed.

In the second phase, we perform E epochs of test-time adaptation. We first freeze the batch normalization (BN) statistics using the current batch. We then compute two types of adaptation losses: (i) BFA over $\mathcal{M}_{\texttt{C}}$ and $\mathcal{M}_{\texttt{I}}$ by minimizing and maximizing cross-entropy, respectively, and (ii) ABA over unlabeled samples with agreement by minimizing cross-entropy. The final objective $\mathcal{L}_{\texttt{BiTTA}}$ combines both components and is optimized via gradient descent. Notably, the policy gradient $\nabla_\theta \log \pi_\theta(y|x)$ is implemented implicitly via backpropagation through cross-entropy on MC-dropout outputs, enabling a simple and effective optimization.

4. Experiments

We present our experimental setup and results across various scenarios. Additional experiments, results, and details are provided in Appendices B, C, and D.

Baselines. We evaluated BiTTA against a comprehensive set of baselines, including source validation (SrcValid) and seven state-of-the-art TTA methods: BN-Stats (Nado et al., 2020), TENT (Wang et al., 2021), EATA (Niu et al., 2022), SAR (Niu et al., 2023), CoTTA (Wang et al., 2022), RoTTA (Yuan et al., 2023), and SoTTA (Gong et al., 2023b). To ensure a fair comparison, we incorporate an equal number of random binary-feedback data into TTA baselines by adding correct-sample loss (cross-entropy) and incorrect-sample loss (complementary label loss Kim et al. (2019)). We also included SimATTA (Gui et al., 2024) as an active TTA baseline, adapting it to use binary-feedback data by incorporating a complementary loss for negative samples. The non-active and full-class active TTA accuracies are reported in Appendix C.

Dataset. To evaluate the robustness of BiTTA across various domain shifts, we used standard image corruption datasets CIFAR10-C, CIFAR100-C, and Tiny-ImageNet-C (Hendrycks & Dietterich, 2019). Additionally, we conducted experiments on the PACS dataset (Li et al., 2017), which is commonly used for domain adaptation tasks. To closely simulate real-world scenarios with evolving distribution shifts, we implemented a continual TTA (Wang et al., 2022) where corruption continuously changes.

Settings and hyperparameters. We configured BiTTA to operate with minimal labeling effort, using only three binary feedback samples within each 64-sample test batch, accounting for less than 5%. We utilize a single value of balancing hyperparameters $\alpha = 2$ and $\beta = 1$ for BiTTA in all experiments. A comprehensive list of settings and hyperparameters is provided in Appendix D.

Overall result. As shown in Table 1, BiTTA consistently outperformed all TTA and active TTA baselines, showcasing its effectiveness in the proposed **TTA with binary feedback** setting. Existing TTA methods, even when adapted to binary feedback, showed suboptimal results, as their fixed filtering strategies (e.g., EATA, SAR, SoTTA) struggle to cope with dynamic uncertainties under continuous distribution shifts.

Table 1: Accuracy (%) comparisons with TTA and active TTA baselines with binary feedback in corruption datasets (severity level 5). Notation * indicates the modified algorithm to utilize binary-feedback samples. B: TTA with binary feedback. Results outperforming all other baselines are highlighted in **bold** fonts. Averaged over three random seeds. Comparison with non-active TTAs and full-class active TTA are in Table 12 in Appendix C.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	25.97	33.19	24.71	56.73	52.02	67.37	64.80	77.97	67.01	74.14	91.51	33.90	76.62	46.38	73.23	57.23
-	BN-Stats	66.96	69.04	60.36	87.78	65.55	86.29	87.38	81.63	80.28	85.39	90.74	86.88	76.72	79.33	71.92	78.42
B	TENT*	75.25	80.71	73.21	85.42	68.89	82.95	85.48	81.96	82.99	83.18	88.88	86.78	75.37	80.77	75.59	80.49
B	EATA*	73.22	74.16	65.72	76.50	62.47	74.62	79.65	76.42	77.31	79.70	85.37	83.66	69.52	77.12	72.16	75.17
B	SAR*	71.57	78.62	73.33	**88.98**	73.30	87.98	89.72	86.00	87.09	87.86	92.46	90.00	82.07	86.69	80.96	83.78
B	CoTTA*	66.97	69.04	60.35	87.77	65.54	86.29	87.38	81.63	80.28	85.40	90.73	86.87	76.74	79.35	71.92	78.42
B	RoTTA*	64.49	69.69	63.89	86.29	69.48	86.78	89.23	85.21	85.39	87.48	92.02	87.09	81.76	85.66	80.18	80.98
B	SoTTA*	71.39	79.27	70.58	85.07	68.39	84.03	87.27	83.47	84.90	85.55	90.81	86.18	78.26	83.41	76.94	81.03
B	SimATTA*	70.21	81.67	71.49	79.59	69.41	82.15	87.28	83.90	86.89	86.49	91.51	83.40	77.94	83.81	82.25	81.09
B	BiTTA	**76.78**	**84.24**	**78.75**	87.51	**77.39**	**88.38**	**91.36**	**89.42**	**90.72**	**90.30**	**94.65**	**92.62**	**86.15**	**92.42**	**87.24**	**87.20**

(a) CIFAR10-C.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	10.63	12.14	7.17	34.86	19.58	44.09	41.94	46.34	34.22	41.08	67.31	18.47	50.36	24.91	44.56	33.18
-	BN-Stats	39.23	40.75	34.10	66.14	42.46	63.57	64.82	53.81	53.49	58.15	68.22	64.48	53.88	56.63	45.17	53.66
B	TENT*	49.70	51.52	39.35	44.63	28.42	27.26	24.74	14.74	10.19	6.44	4.89	3.01	2.99	2.87	2.62	20.89
B	EATA*	10.80	2.75	2.46	1.88	1.68	1.80	1.67	1.56	1.37	1.22	1.30	1.06	1.39	1.42	1.23	2.24
B	SAR*	46.57	55.41	48.54	**66.29**	51.01	**65.27**	**68.49**	60.72	62.35	63.35	71.08	**69.50**	59.82	65.55	56.48	60.70
B	CoTTA*	39.22	40.76	34.10	66.11	42.48	63.57	64.83	53.80	53.48	58.15	68.22	64.48	53.85	56.66	45.17	53.66
B	RoTTA*	36.68	40.73	36.00	62.54	44.58	62.90	66.93	58.09	59.71	60.23	70.33	62.67	59.12	63.55	53.83	55.86
B	SoTTA*	44.39	38.67	23.70	29.92	20.45	26.89	31.05	24.68	26.63	25.25	33.69	16.99	24.57	27.67	24.82	27.96
B	SimATTA*	28.93	41.54	28.94	39.79	34.83	49.11	55.42	46.59	51.06	48.83	60.03	34.67	44.27	45.19	46.22	43.69
B	BiTTA	**50.12**	**58.34**	**52.07**	63.27	**52.70**	63.80	68.16	**62.65**	**65.39**	**63.79**	**71.26**	68.97	**63.93**	**69.45**	**63.38**	**62.49**

(b) CIFAR100-C.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	6.66	8.91	3.68	16.03	10.05	26.62	27.16	29.29	33.40	11.26	30.76	1.96	27.83	40.67	47.96	21.48
-	BN-Stats	32.28	34.27	22.21	32.63	22.02	44.56	46.11	39.31	43.27	31.66	46.01	10.19	43.51	52.14	50.62	36.72
B	TENT*	**36.80**	35.24	17.71	8.21	3.33	3.30	2.91	2.36	2.15	1.99	1.99	1.43	1.88	1.94	1.88	8.21
B	EATA*	35.19	37.98	25.35	**35.82**	24.84	47.04	48.19	42.10	45.53	36.67	48.79	9.16	44.71	53.43	51.60	39.09
B	SAR*	33.90	38.56	27.84	35.49	**26.70**	47.14	48.41	41.54	45.49	37.25	49.53	**14.85**	46.63	52.49	51.41	39.82
B	CoTTA*	32.27	34.26	22.21	32.62	22.04	44.56	46.11	39.30	43.27	31.67	46.01	10.18	43.51	52.18	50.60	36.72
B	RoTTA*	31.54	35.42	23.98	32.80	24.02	45.16	47.01	40.70	44.57	33.07	47.57	14.59	44.65	49.89	49.49	37.63
B	SoTTA*	34.24	33.48	18.21	17.96	10.63	15.41	12.75	9.46	7.35	5.97	7.70	1.64	5.96	6.13	5.70	12.84
B	SimATTA*	16.76	24.53	14.51	17.92	14.72	30.63	34.98	25.21	34.89	17.91	35.13	1.68	33.33	42.04	46.72	26.06
B	BiTTA	34.84	**39.88**	**28.56**	35.37	26.65	**48.41**	**49.57**	**43.62**	**47.90**	**39.53**	**50.95**	12.27	**47.18**	**54.01**	**54.06**	**40.85**

(c) Tiny-ImageNet-C.

SimATTA, an active TTA baseline, also underperformed due to hard-thresholding and clustering with incorrect samples.

To fairly evaluate binary feedback under labeling constraints, we further compared BiTTA against full-class active TTA in two scenarios: (1) *equal labeling cost* — adjusted for information gain (Figure 5), and (2) *equal number of full-class/binary-feedback samples* (Table 12). Scenarios are further explained in Appendix D.1. In both scenarios, BiTTA (with binary feedback) consistently outperformed SimATTA (with full-class labels), demonstrating that **binary feedback alone can drive more effective adaptation** than full-class supervision under limited interactions. With equal labeling cost, BiTTA achieved up to **32% higher accuracy**, highlighting its **superior cost-efficiency and robustness**.

In Figure 5, replacing an oracle feedback with a foundational model (GPT-4o) significantly degraded SimATTA's performance due to GPT-4o's high classification error (40% on average). This result underlines the current limitations of automatic feedback: while easily accessible, it is unreliable under distribution shifts. In contrast, our approach leverages **lightweight yet trustworthy binary feedback from an oracle**, offering a robust and cost-efficient alternative for real-world test-time adaptation.

We also evaluated BiTTA on broader distribution shifts in domain-wise (continual TTA (Wang et al., 2022)) and mixed-stream scenarios (Gui et al., 2024) for a domain generalization task. Table 2 shows that BiTTA achieved the highest average accuracy across all baselines, demonstrating gener-

Table 2: Accuracy (%) comparisons with TTA and active TTA baselines with binary feedback in PACS. The domain-wise data stream is a continual TTA setting, and the mixed data stream shuffled all domains randomly, where we report the cumulative accuracy at four adaptation points. Notation * indicates the modified algorithm to utilize binary-feedback samples. B: TTA with binary feedback. Results outperforming all other baselines are highlighted in **bold** fonts. Comparison with non-active TTAs and full-class active TTA are in Table 13 in Appendix C.

Label	Method	Domain-wise data stream →				Mixed data stream →			
		Art	Cartoon	Sketch	Avg	25%	50%	75%	100%(Avg)
-	Src	59.38 ±0.00	27.94 ±0.21	42.96 ±0.01	43.43 ±0.07	42.74 ±1.13	42.80 ±0.22	42.64 ±0.30	42.77 ±0.01
-	BN Stats	67.87 ±0.18	63.48 ±0.88	54.07 ±0.36	61.81 ±0.18	59.09 ±0.29	58.28 ±0.08	58.05 ±0.22	57.82 ±0.20
B	TENT*	71.26 ±0.44	67.71 ±0.89	51.57 ±1.73	63.51 ±0.33	**60.72 ±0.71**	58.86 ±0.53	57.54 ±0.33	56.21 ±0.15
B	EATA*	68.67 ±0.38	65.31 ±0.78	59.05 ±0.27	64.34 ±0.22	59.58 ±0.10	59.15 ±0.55	59.26 ±0.36	59.39 ±0.06
B	SAR*	67.95 ±0.20	63.66 ±0.81	55.35 ±0.39	62.32 ±0.12	59.17 ±0.14	58.57 ±0.21	58.42 ±0.06	58.41 ±0.08
B	CoTTA*	67.87 ±0.18	63.47 ±0.90	54.07 ±0.36	61.80 ±0.19	59.10 ±0.32	58.29 ±0.09	58.06 ±0.23	57.83 ±0.22
B	RoTTA*	64.73 ±0.20	55.14 ±1.91	56.05 ±0.72	58.64 ±0.50	55.50 ±1.30	52.68 ±0.64	51.45 ±0.56	50.10 ±0.33
B	SoTTA*	69.73 ±0.43	42.48 ±2.31	46.07 ±2.00	52.76 ±0.84	54.33 ±3.59	52.89 ±3.95	53.09 ±3.78	53.37 ±2.81
B	SimATTA*	55.83 ±16.69	59.68 ±7.98	72.40 ±4.51	62.63 ±9.63	59.34 ±2.78	63.81 ±0.68	67.09 ±0.34	69.21 ±0.11
B	BiTTA	**73.86 ±3.76**	**76.81 ±2.45**	**76.03 ±1.61**	**75.57 ±0.93**	59.65 ±0.70	**64.70 ±0.78**	**69.23 ±0.17**	**72.18 ±0.38**

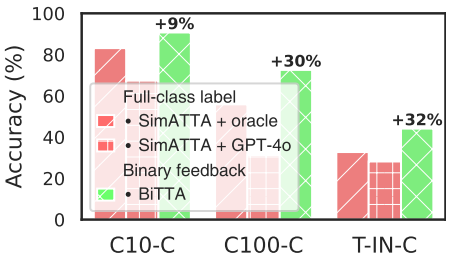


Figure 5: Accuracy (%) with full-class feedback (SimATTA) and binary-feedback (BiTTA) and under the equal total labeling cost. GPT-4o is used as a foundational model to provide a full-class label.

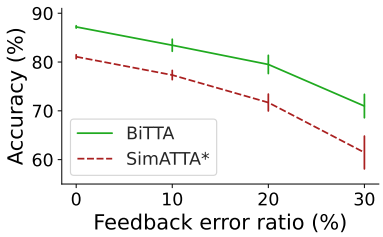


Figure 6: Accuracy (%) varying the feedback error in CIFAR10-C. Binary feedback is flipped (*correct* ↔ *incorrect*) when a feedback error occurs.

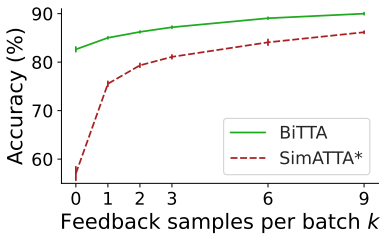


Figure 7: Accuracy (%) varying the number of binary-feedback samples per batch (k) in CIFAR10-C. Zero feedback sample is equivalent to conventional TTA.

alization beyond corruptions.

Overall, these results show that **TTA with binary feedback is a practical and scalable TTA paradigm**, and **BiTTA delivers state-of-the-art performance** under minimal supervision, outperforming full-class approaches in both accuracy and efficiency. Additional results, including large-scale and synthetic datasets (e.g., ImageNet-C (Hendrycks & Dietterich, 2019), ImageNet-R (Hendrycks et al., 2021), ColoredMNIST (Arjovsky et al., 2019), VisDA-2021 (Bashkirova et al., 2022), and DomainNet (Peng et al., 2019)) and challenging settings (e.g., non-iid and single-sample), are available in Appendix C.

Impact of feedback errors. We assumed the binary feedback provided by the oracle contained no labeling errors. In practice, user feedback might include labeling errors by shifting the binary feedback between *correct* and *incorrect*. We examine the impact of binary-feedback error compared with the active TTA baseline. Figure 6 shows that SimATTA has consistently low accuracy under labeling errors by relying on the noisy labeled samples without utilizing the unlabeled samples. In contrast, BiTTA combines many confident unlabeled samples with labeled samples to reduce the impact of labeling errors, thus outperforming SimATTA.

Impact of number of feedback samples. We evaluated how the number of binary-feedback samples per batch (k) influences adaptation performance. As illustrated in Figures 7 and 11, BiTTA maintains high accuracy even with sparse active samples. The performance improves as k increases, showcasing effective utilization of additional binary feedback. SimATTA shows a similar trend of increasing accuracy with more active samples, but the overall performance is consistently lower than BiTTA. This suggests that BiTTA can effectively leverage additional feedback while maintaining stability in low budgets, indicating its potential for deployment in scenarios with varying labeling budgets.

Synergistic effect of adaptation strategies. We compared BiTTA against its components: Binary Feedback-guided Adaptation (BFA) and Agreement-based self-Adaptation (ABA). In CIFAR10-C, BFA-only adaptation achieved 58.90% and ABA-only adaptation achieved 82.64%, whereas BiTTA achieved on average 87.20% accuracy, consistently outperforming entire continual corrup-

tions. The superior performance of the combined approach (BiTTA) indicates that BFA and ABA complement each other to achieve robust accuracy.

Uncertainty calibration of MC-dropout. Beyond accuracy, we also analyze the calibration quality of MC-dropout compared with deterministic softmax outputs. Reliable calibration is essential for effective sample selection in both BFA and ABA. We compute the expected calibration error (ECE) across CIFAR10-C corruptions and find that MC-dropout achieves a substantially lower ECE of 0.062, compared with 0.100 from softmax-based confidence, a 38% reduction in miscalibration. We further compare with various uncertainty estimation methods (e.g., augmentation, ensemble, and deterministic softmax confidence) in Table 6 (Appendix B), showcasing the importance of MC-dropout for the policy gradient modeling.

5. Related Work

Test-time adaptation. Test-time adaptation (TTA) improves model accuracy on distribution shift on the pre-trained model with only unlabeled test samples (Wang et al., 2021). Existing TTA focused on robust adaptation (Niu et al., 2023; Gong et al., 2022; Yuan et al., 2023; Wang et al., 2022; Boudiaf et al., 2022; Niu et al., 2022; Gong et al., 2023b; Park et al., 2024) across various types of distribution shifts (Niu et al., 2023; Gong et al., 2022; Wang et al., 2022; Gong et al., 2023b; Press et al., 2023). However, existing TTA methods suffer from adaptation failures during lifelong adaptation (Press et al., 2023), highlighting the need for a few-sample guide for robust adaptation. Active test-time adaptation (ATTA) (Gui et al., 2024) introduced a foundational analysis of active TTA setting, with a supervised learning scheme (SimATTA) using low-entropy source-like sample pseudo-labeling and active labeling from an incremental clustering algorithm. However, SimATTA is sensitive to the pre-trained model and selected active samples, as it does not leverage most unlabeled samples and only utilizes a few labeled samples. In contrast, TTA with binary feedback utilizes a large set of unlabeled samples while guiding adaptation with binary-feedback samples, performing stable adaptation.

Reinforcement learning for model tuning. Reinforcement learning (RL) has been successfully applied in various domains to incorporate non-differentiable rewards in the optimization process (Zoph & Le, 2017; Yoon et al., 2020; Ouyang et al., 2022; Fan et al., 2023; Black et al., 2024). For example, Zoph & Le (2017) and Yoon et al. (2020) employ the REINFORCE algorithm to use the accuracy of the validation dataset as a (non-differentiable) reward in neural architecture search or data valuation. In the domain of natural language processing, reinforcement learning with human feedback (RLHF) (Ouyang et al., 2022) has gained prominence for fine-tuning large language models, and Shankar et al. (2024) demonstrates how lightweight binary feedback (e.g., thumbs up/down) can effectively guide model behavior. Similar approaches have been explored in vision and multimodal research (Fan et al., 2023; Black et al., 2024; Le et al., 2022; Pinto et al., 2023). Recently, Reinforcement Learning with CLIP Feedback (RLCF) (Zhao et al., 2024) has been proposed to adapt vision-language models. RLCF relies on the pre-trained CLIP model as a reward function, which may not be available or suitable for all domains or tasks. In contrast, our approach provides a more general and flexible approach for test-time adaptation by effectively guiding the adaptation without relying on specific pre-trained models.

Active learning. Active learning (Cohn et al., 1994; Settles, 2009) involves an oracle (e.g., human annotator) in the machine learning process to enable efficient annotation and training. The active learning framework has been widely studied in active (source-free) domain adaptation (Ash et al., 2019; Prabhu et al., 2021; Li et al., 2022; Wang et al., 2023; Du & Li, 2024; Ning et al., 2021) and active TTA (Gui et al., 2024). Compared with active domain adaptation, active TTA focuses on the non-regrettable active sample selection on the continuously changing data stream without access to source data. Using binary feedback is related to the active learning with partial feedback problem (Hu et al., 2019), which seeks to recursively obtain partial labels until a definitive label is identified. Joshi et al. (2010) proposed an active learning setup where users compare two images and report whether they belong to the same category. In contrast, our approach leverages single-step binary feedback on the model's current batch sample output without requiring additional data. This simplifies the process and reduces the labeling effort.

6. Conclusion

We proposed test-time adaptation with binary feedback to address the challenge of adapting pre-trained models to new domains with minimal labeling effort. Our approach leverages binary feedback on the model predictions (*correct* or *incorrect*) from an oracle to guide the adaptation process, significantly reducing the labeling cost compared to methods that require full-class labels. Our method, BiTTA, uniquely combines binary feedback-guided adaptation on uncertain samples with agreement-based self-adaptation on confident samples in a reinforcement learning framework, balancing between a few labeled samples and many unlabeled samples. Through extensive experiments on distribution shift datasets, we demonstrated that BiTTA outperforms state-of-the-art test-time adaptation methods, showcasing its effectiveness in handling continuous distribution shifts. Overall, test-time adaptation with binary feedback represents a significant step forward in test-time adaptation, offering a practical balance between performance and labeling efficiency.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2025-02215122, Development and Demonstration of Lightweight AI Model for Smart Homes), Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2025-02263169, Detection and Prediction of Emerging and Undiscovered Voice Phishing), and Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. RS-2020-II201336, Artificial Intelligence Graduate School Program(UNIST)).

Impact Statement

TTA with binary feedback and BiTTA offer significant societal and practical benefits while presenting some important considerations for implementation. The use of binary feedback substantially reduces labeling costs, making active test-time adaptation more accessible to the end-users. This enables real-time adaptation in critical applications like autonomous driving, healthcare diagnostics, and medical applications; thereby improving safety, efficiency, and user experience. While the system enhances model performance in diverse and changing environments with minimal labeling effort, it requires careful consideration of potential risks, including privacy concerns in surveillance applications and biased decision-making in applications like targeted advertising.

References

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Ash, J. T., Zhang, C., Krishnamurthy, A., Langford, J., and Agarwal, A. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations*, 2019.

Bashkirova, D., Hendrycks, D., Kim, D., Liao, H., Mishra, S., Rajagopalan, C., Saenko, K., Saito, K., Tayyab, B. U., Teterwak, P., et al. Visda-2021 competition: Universal domain adaptation to improve performance on out-of-distribution data. In *NeurIPS 2021 Competitions and Demonstrations Track*. PMLR, 2022.

Black, K., Janner, M., Du, Y., Kostrikov, I., and Levine, S. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, 2024.

Boudiaf, M., Mueller, R., Ben Ayed, I., and Bertinetto, L. Parameter-free online test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020.

Chen, D., Wang, D., Darrell, T., and Ebrahimi, S. Contrastive test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Cohn, D., Ghahramani, Z., and Jordan, M. Active learning with statistical models. *Advances in neural information processing systems*, 7, 1994.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

Du, Z. and Li, J. Diffusion-based probabilistic uncertainty estimation for active domain adaptation. *Advances in Neural Information Processing Systems*, 2024.

Fan, Y., Watkins, O., Du, Y., Liu, H., Ryu, M., Boutilier, C., Abbeel, P., Ghavamzadeh, M., Lee, K., and Lee, K. Reinforcement learning for fine-tuning text-to-image diffusion models. In *Advances in Neural Information Processing Systems*, 2023.

Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, 2016.

Gong, T., Jeong, J., Kim, T., Kim, Y., Shin, J., and Lee, S.-J. NOTE: Robust continual test-time adaptation against temporal correlation. In *Advances in Neural Information Processing Systems*, 2022.

Gong, T., Jang, S. Y., Acer, U. G., Kawsar, F., and Min, C. Collaborative inference via dynamic composition of tiny ai accelerators on mcus. *arXiv preprint arXiv:2401.08637*, 2023a.

Gong, T., Kim, Y., Lee, T., Chottananurak, S., and Lee, S.-J. SoTTA: Robust test-time adaptation on noisy data streams. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b.

Gui, S., Li, X., and Ji, S. Active test-time adaptation: Theoretical analyses and an algorithm. In *International Conference on Learning Representations (ICLR)*, 2024.

Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., et al. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*, 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021.

Hong, J., Lyu, L., Zhou, J., and Spranger, M. MECTA: Memory-economic continual test-time model adaptation. In *The Eleventh International Conference on Learning Representations*, 2023.

Hu, P., Lipton, Z., Anandkumar, A., and Ramanan, D. Active learning with partial feedback. In *International Conference on Learning Representations*, 2019.

Jang, M., Chung, S.-Y., and Chung, H. W. Test-time adaptation via self-training with nearest neighbor information. In *The Eleventh International Conference on Learning Representations*, 2022.

Joshi, A. J., Porikli, F., and Papanikolopoulos, N. Breaking the interactive bottleneck in multi-class classification with active selection and binary feedback. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

Kim, Y., Yim, J., Yun, J., and Kim, J. Nlnl: Negative learning for noisy labels. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2019.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Le, N., Rathour, V. S., Yamazaki, K., Luu, K., and Savvides, M. Deep reinforcement learning in computer vision: a comprehensive survey. *Artificial Intelligence Review*, pp. 1–87, 2022.

Lee, J., Jung, D., Lee, S., Park, J., Shin, J., Hwang, U., and Yoon, S. Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In *The Twelfth International Conference on Learning Representations*, 2024a.

Lee, T., Chottananurak, S., Gong, T., and Lee, S.-J. AETTA: Label-free accuracy estimation for test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024b.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. M. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, 2017.

Li, X., Du, Z., Li, J., Zhu, L., and Lu, K. Source-free active domain adaptation via energy-based locality preserving transfer. In *Proceedings of the 30th ACM international conference on multimedia*, 2022.

Li, Y., Xu, X., Su, Y., and Jia, K. On the robustness of open-world test-time training: Self-training with dynamic prototype expansion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Liberis, E. and Lane, N. D. Differentiable neural network pruning to enable smart applications on microcontrollers. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 6(4):1–19, 2023.

Liu, W., Wang, X., Owens, J., and Li, Y. Energy-based out-of-distribution detection. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, 2020.

MacKay, D. J. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

Nado, Z., Padhy, S., Sculley, D., D'Amour, A., Lakshminarayanan, B., and Snoek, J. Evaluating prediction-time batch normalization for robustness under covariate shift. *arXiv preprint arXiv:2006.10963*, 2020.

Ning, M., Lu, D., Wei, D., Bian, C., Yuan, C., Yu, S., Ma, K., and Zheng, Y. Multi-anchor active domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Niu, S., Wu, J., Zhang, Y., Chen, Y., Zheng, S., Zhao, P., and Tan, M. Efficient test-time model adaptation without forgetting. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

Niu, S., Wu, J., Zhang, Y., Wen, Z., Chen, Y., Zhao, P., and Tan, M. Towards stable test-time adaptation in dynamic wild world. In *The Eleventh International Conference on Learning Representations*, 2023.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. In *Advances in neural information processing systems*, 2022.

Park, H., Hwang, J., Mun, S., Park, S., and Ok, J. MedBN: Robust test-time adaptation against malicious test samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., and Wang, B. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2019.

Pinto, A. S., Kolesnikov, A., Shi, Y., Beyer, L., and Zhai, X. Tuning computer vision models with task rewards. In *International Conference on Machine Learning*, 2023.

Prabhu, V., Chandrasekaran, A., Saenko, K., and Hoffman, J. Active domain adaptation via clustering uncertainty-weighted embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Press, O., Schneider, S., Kümmerer, M., and Bethge, M. Rdumb: A simple approach that questions our progress in continual test-time adaptation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Rusci, M., Capotondi, A., and Benini, L. Memory-driven mixed low precision quantization for enabling deep network inference on microcontrollers. *Proceedings of Machine Learning and Systems*, 2:326–335, 2020.

Saito, K., Kim, D., Sclaroff, S., and Saenko, K. Universal domain adaptation through self supervision. In *Advances in neural information processing systems*, 2020.

Sakaridis, C., Dai, D., and Van Gool, L. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126:973–992, 2018.

Settles, B. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

Shankar, S., Zamfirescu-Pereira, J., Hartmann, B., Parameswaran, A., and Arawjo, I. Who validates the validators? aligning llm-assisted evaluation of llm outputs with human preferences. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, 2024.

Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C. A., Cubuk, E. D., Kurakin, A., and Li, C.-L. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in neural information processing systems*, 2020.

Song, J., Lee, J., Kweon, I. S., and Choi, S. Ecotta: Memory-efficient continual test-time adaptation via self-distilled regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision, 2016.

Wang, D., Shelhamer, E., Liu, S., Olshausen, B., and Darrell, T. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.

Wang, F., Han, Z., Zhang, Z., He, R., and Yin, Y. Mhpl: Minimum happy points learning for active source free domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Wang, Q., Fink, O., Van Gool, L., and Dai, D. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

Yoon, J., Arik, S., and Pfister, T. Data valuation using reinforcement learning. In *International Conference on Machine Learning*, 2020.

Yuan, L., Xie, B., and Li, S. Robust test-time adaptation in dynamic scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Zhang, M., Levine, S., and Finn, C. Memo: Test time robustness via adaptation and augmentation. In *Advances in Neural Information Processing Systems*, 2022.

Zhao, S., Wang, X., Zhu, L., and Yang, Y. Test-time adaptation with clip reward for zero-shot generalization in vision-language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Zoph, B. and Le, Q. Neural architecture search with reinforcement learning. In *International Conference on Learning Representations*, 2017.

A. Discussion

Despite the promising results, the TTA with binary feedback setting and BiTTA method have limitations. First, the reliance on binary feedback, while reducing labeling effort, may still require substantial oracle involvement in scenarios with high data variability or rapid domain shifts. Although BiTTA robustly performs across various labeling scenarios (Figures 7, 11, and 12), future work may explore reducing oracle involvement by developing more advanced and dynamic sample selection strategies. Second, the computational overhead introduced by Monte Carlo dropout (Table 3) could be further reduced by efficient TTA (Hong et al., 2023; Song et al., 2023) and on-device machine learning (Liberis & Lane, 2023; Rusci et al., 2020; Gong et al., 2023a). Finally, although our algorithm robustly outperformed with feedback errors (Figure 6), designing a method for specifically handling noisy or incorrect feedback remains an area for future research.

B. Additional experiments

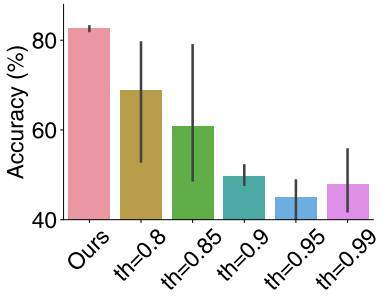


Figure 8: Accuracy (%) comparison with hard thresholding (th). Ours dynamically selects confident samples via agreement between the model and MC-dropout. Averaged over three random seeds.

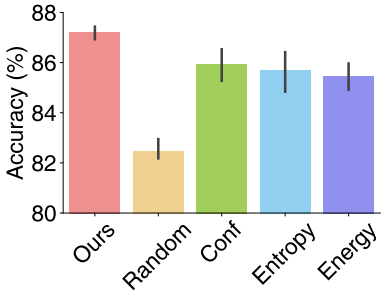


Figure 9: Accuracy (%) varying the binary-feedback sample selection strategy in CIFAR10-C. Ours leverages MC-dropout to select the most uncertain samples. Averaged over three random seeds.

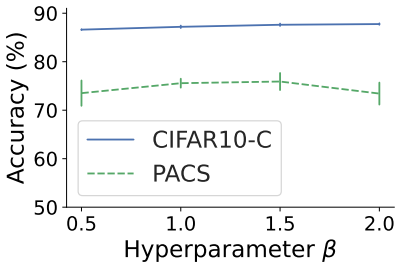


Figure 10: Accuracy (%) varying the balancing hyperparameter (β) in CIFAR10-C and PACS. Another balancing hyperparameter α is set to 1. Averaged over three random seeds.

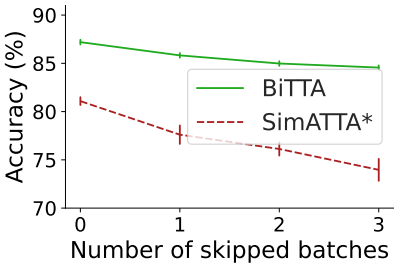


Figure 11: Accuracy (%) varying the labeling skip in CIFAR10-C. Averaged over three random seeds.

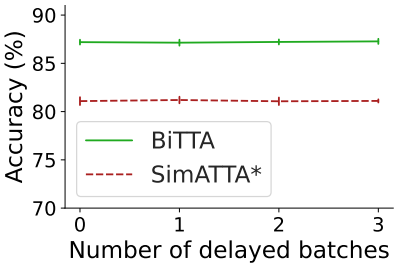


Figure 12: Accuracy (%) varying the labeling delay (labeled samples arrive after certain batches). Averaged over three random seeds.

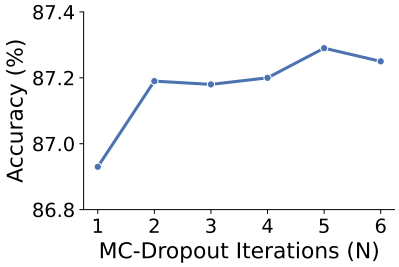


Figure 13: Accuracy (%) varying the number of dropouts in MC-dropout. Averaged over three random seeds.

Comparing prediction agreement with confidence thresholding. To assess the effectiveness of our prediction agreement method for confident sample selection, we compared it against fixed confidence thresholding approaches. We evaluated thresholds ranging from 0.8 to 0.99, with 0.99 being the value used in SoTTA (Gong et al., 2023b). Figure 8 illustrates the performance of these approaches on unlabeled-only TTA in the continual CIFAR10-C setting. Our prediction agreement method consistently outperformed all fixed thresholding approaches, which exhibited high variance and instability. This result demonstrates the superiority of our dynamic sample selection strategy, particularly in scenarios with continuously changing corruptions, highlighting the importance of adaptive confidence assessment in test-time adaptation.

Table 3: Average wall-clock time per batch (s) comparisons with TTA and active TTA baselines with binary feedback in Tiny-ImageNet-C. Notation * indicates the modified algorithm to utilize binary-feedback samples. Averaged over three random seeds.

	SrcValid	BN-Stats	TENT*	EATA*	SAR*	CoTTA*	RoTTA*	SoTTA*	SimATTA*	BiTTA
Avg.	0.18 ±0.12	0.33 ±0.20	1.03 ±0.35	0.98 ±0.39	1.02 ±0.38	26.63 ±5.40	1.68 ±0.27	1.25 ±0.16	45.45 ±13.50	4.19 ±0.06

Table 4: Average GPU memory consumption (MB). MECTA and gradient accumulation (GA) are applied to BiTTA. Notation * indicates the modified algorithm to utilize binary-feedback samples.

	SrcValid	BN-Stats	TENT*	EATA*	SAR*	CoTTA*	RoTTA*	SoTTA*	SimATTA*	BiTTA	+MECTA	+GA
Avg.	2081	2696	3246	3239	3244	2966	3038	3229	2824	8304	6724	2841

Impact of sample selection. We examined the impact of sample selection, including our MC-dropout certainty approach with random selection, maximum entropy (Saito et al., 2020), minimum confidence (Sohn et al., 2020), and minimum energy (Liu et al., 2020). In Figure 9, our method outperforms others by leveraging MC-dropout to estimate epistemic uncertainty. In contrast, naive methods may struggle with overconfidence in test-time scenarios, failing to prioritize samples that offer the most valuable information for model improvement. On the other hand, we observe that for large datasets (e.g., ImageNet-C/R and VisDA-2021 in Table 7), most predictions are already unconfident and incorrect, so random selection is sufficient to capture uncertain samples for effective adaptation.

Sensitivity to balancing hyperparameter α, β. We investigated the sensitivity of BiTTA to the balancing hyperparameter β while fixing $\alpha = 1.0$, which controls the trade-off between binary feedback-guided adaptation and agreement-based self-adaptation. Figure 10 illustrates the overall accuracy across various β values for both image corruption and domain adaptation datasets. The results demonstrate that BiTTA maintains consistent performance across a wide range of β values, indicating robustness to this hyperparameter choice. This stability suggests that BiTTA can effectively deploy across different scenarios without extensive hyperparameter tuning.

Impact of intermittent and delayed labeling. To further understand the impact of the annotator's labeling budget, we conduct an experiment scenario where annotators skip or delay the labeling of a few batches (e.g., labeling only 1 out of 4 consecutive batches). In Figures 11 and 12, we observe our BiTTA shows stable performance with minimal degradation, whereas the active TTA baseline (SimATTA) shows high accuracy degradation with batch skips and consistent suboptimal performance with delayed batches. The result supports the robustness of BiTTA in varying labeling scenarios, enabling a practical and scalable TTA.

Runtime analysis. To assess the practical applicability of BiTTA, we conducted a comprehensive runtime analysis by measuring the average wall-clock time per batch across different methods on the Tiny-ImageNet-C dataset. Our results in Table 3 show that BiTTA requires 4.19 ±0.06 seconds per batch, positioning it between simpler TTA methods (0.33-1.68s) and more complex approaches like CoTTA (26.63s) and SimATTA (45.45s). The runtime profile demonstrates that BiTTA achieves a favorable balance between computational cost and performance, particularly considering its significant accuracy improvements over faster baselines while maintaining substantially lower processing time than methods like SimATTA.

Memory analysis. We report the average GPU memory consumption across all CIFAR10-C corruptions (severity 5) in Table 4. Compared to baseline methods, BiTTA shows higher memory usage due to repeated forward passes for MC-dropout and the reinforcement learning formulation. To mitigate this, we evaluate two variants: BiTTA+MECTA (applying the memory-efficient continual test-time adaptation from MECTA (Hong et al., 2023)) and BiTTA+GA (using gradient accumulation to divide a batch into multiple mini-batches). BiTTA+GA reduces memory usage up to 60% while retaining BiTTA's performance, demonstrating a practical strategy for deployment in memory-constrained settings.

Impact of the number of epochs. To understand the BiTTA's performance under time-constrained environments, we examined how reducing training epochs affects adaptation accuracy on CIFAR10-C. We compared our standard 3-epoch configuration against reduced 1- and 2-epoch settings, adjusting learning rates proportionally (×3 and ×1.5) to compensate for fewer update steps. Results in Table 5 show that BiTTA maintains robust performance even with fewer epochs. This consistent performance across epoch configurations demonstrates that BiTTA can effectively adapt to distribution shifts even under stricter computational constraints, offering flexibility in real-world deployment scenarios where faster adaptation may be preferred.

Table 5: Accuracy (%) comparisons with varying epochs in CIFAR10-C (severity level 5). B: TTA with binary feedback. Averaged over three random seeds.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
B	BiTTA (epoch = 3)	76.78	84.24	78.75	87.51	77.39	88.38	91.36	89.42	90.72	90.30	94.65	92.62	86.15	92.42	87.24	**87.20**
B	· epoch = 1	76.92	84.29	78.61	86.99	77.20	88.36	91.51	89.31	90.58	90.30	94.51	92.70	85.77	92.08	87.50	87.11
B	· epoch = 2	76.30	84.01	78.80	87.66	77.30	88.43	91.56	89.16	90.61	90.37	94.52	92.61	85.83	92.33	87.75	87.15

Table 6: Accuracy (%) comparisons with replacing MC-dropout uncertainty estimation with (1) sample augmentation, (2) ensemble method, and (3) original confidence in CIFAR10-C (severity level 5). B: TTA with binary feedback. Averaged over three random seeds.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
B	BiTTA	76.78	84.24	78.75	87.51	77.39	88.38	91.36	89.42	90.72	90.30	94.65	92.62	86.15	92.42	87.24	**87.20**
B	· Augmentation	66.22	46.99	25.43	18.49	12.82	11.96	11.68	11.43	12.24	11.37	11.48	10.87	11.45	11.96	11.71	19.07
B	· Ensemble	74.08	81.60	75.57	88.16	74.15	88.50	91.02	87.93	89.71	88.92	94.05	92.01	83.28	89.83	83.64	85.50
B	· Confidence	74.00	82.61	76.54	87.12	75.13	87.83	90.92	88.14	89.90	89.20	94.33	92.28	84.08	90.93	85.54	85.90

MC-dropout configuration. We used 4 dropout inferences ($N = 4$) for policy estimation in BiTTA. To evaluate sensitivity to the number of MC-dropout inferences, we conducted an ablation study varying N from 1 to 6. As shown in Figure 13, BiTTA performs consistently well for all $N > 1$, demonstrating robustness to this hyperparameter. Even at $N = 1$, MC-dropout is still active, introducing stochasticity essential for both core components: (1) BFA relies on uncertainty to select feedback samples, and (2) ABA leverages agreement between stochastic and deterministic predictions. In contrast, removing dropout entirely leads to a 2.56% accuracy drop. Although gains from increasing N are modest, higher N improves uncertainty calibration, reducing the expected calibration error (ECE) from 0.142 at $N = 1$ to 0.064 at $N = 4$. We thus adopt $N = 4$ as a balanced choice. In practice, smaller N may be preferred for low-latency scenarios, while larger N may benefit reliability-critical deployments.

Replacing MC-dropout uncertainty estimation. To further understand the importance of MC-dropout uncertainty estimation, we compare BiTTA with replacing MC-dropout with augmentation-based estimation (as in Wang et al. (2022); Zhang et al. (2022)), ensemble-based estimation (Jang et al., 2022), and simple deterministic softmax confidence. Results in Table 6 suggest that augmentation-based uncertainty appears less stable and overfits in the early adaptation stage, leading to suboptimal performance.

C. Additional results

Results on additional datasets. We conduct an additional experiment to evaluate the scalability of BiTTA across various datasets covered in recent works (Lee et al., 2024a; Niu et al., 2023; Gui et al., 2024; Chen et al., 2022): ImageNet-C (Hendrycks & Dietterich, 2019), ImageNet-R (Hendrycks et al., 2021), ColoredMNIST (Arjovsky et al., 2019), VisDA-2021 (Bashkirova et al., 2022), and DomainNet (Peng et al., 2019). Results in Table 7 demonstrate a superior performance of BiTTA, especially on large-scale datasets such as ImageNet-C. The key insight is that BiTTA formulates both binary feedback and unlabeled sample adaptation as a single reinforcement learning objective, where the reward signals seamlessly guide the model's adaptation. Also, the use of MC-dropout provides a robust uncertainty estimate, while optimizing on MC-dropout prevents the TTA model from overfitting, therefore showing a stable adaptation in large-scale datasets.

Results on additional scenarios. Recent TTA works suggest a new scenario of (1) imbalanced/non-iid label distribution, where ground-truth labels are temporally correlated (Niu et al., 2023; Gong et al., 2022), (2) and batch size 1 (Niu et al., 2023). Note that SimATTA's clustering algorithm for sample selection is not applicable in scenarios where the memory capacity is limited to only one image. Experiment results on CIFAR10-C (Table 8) suggest the robustness of our method over imbalanced label distribution and batch size 1 by effectively utilizing reward signals from the binary feedback and unlabeled samples.

Comparison with recent baselines We incorporated new comparisons with recent TTA methods: DeYO (Lee et al., 2024a), OWTTT (Li et al., 2023), and TAST (Jang et al., 2022). As shown in Tables 9 and 10, BiTTA consistently

Table 7: Accuracy (%) comparisons with TTA and active TTA baselines in additional datasets. Notation * indicates the modified algorithm to utilize binary-feedback samples. Results outperforming all other baselines are highlighted in **bold** fonts. Averaged over three random seeds.

Dataset	SrcValid	BN-Stats	TENT*	EATA*	SAR*	CoTTA*	RoTTA*	SoTTA*	SimATTA*	BiTTA
ImageNet-C	14.43	26.88	0.93	30.87	35.15	26.80	22.55	36.02	17.50	**36.59**
ImageNet-R	33.05	35.08	29.10	37.14	36.64	35.02	34.35	31.00	35.63	**38.59**
VisDA-2021	27.36	26.46	20.38	27.82	27.41	26.46	27.23	27.71	22.80	**29.30**
DomainNet	54.82	54.41	18.80	59.49	57.78	54.40	56.41	54.82	58.41	**60.85**
ColoredMNIST	50.49	45.59	44.92	45.59	45.74	45.60	48.90	59.45	93.66	**96.75**

Table 8: Accuracy (%) comparisons with TTA and active TTA baselines with binary feedback in online CIFAR10-C (severity level 5) with additional scenarios. Notation * indicates the modified algorithm to utilize binary-feedback samples. B: TTA with binary feedback. Results outperforming all other baselines are highlighted in **bold** fonts. Averaged over three random seeds.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	24.01	30.91	22.36	55.00	53.44	66.99	63.74	78.01	68.41	73.92	91.34	34.30	76.77	46.26	73.05	57.70
-	BN-Stats	22.75	23.33	20.83	30.15	21.45	29.38	28.90	27.33	28.05	29.27	31.37	31.06	25.21	26.37	22.91	26.58
B	TENT*	20.00	21.27	19.56	26.77	19.19	26.54	25.76	24.94	24.66	26.50	28.03	26.66	22.14	23.88	20.98	23.79
B	EATA*	16.24	16.52	13.73	18.82	15.97	18.87	18.79	16.87	17.62	19.30	20.34	17.85	18.02	17.87	16.29	17.54
B	SAR*	22.95	23.57	21.36	30.06	21.44	29.52	28.81	27.38	28.10	29.48	31.40	30.69	24.90	26.46	23.31	26.63
B	CoTTA*	22.76	23.36	21.14	29.99	21.42	29.48	28.84	27.42	28.10	29.43	31.34	30.85	24.92	26.43	23.22	26.58
B	RoTTA*	41.83	44.60	37.97	58.54	41.14	57.40	57.79	52.54	51.86	56.87	62.27	53.20	48.41	50.65	44.84	50.66
B	SoTTA*	67.03	71.31	61.84	83.96	66.01	82.23	84.47	78.62	78.48	82.94	87.74	77.29	74.07	76.94	72.12	76.34
B	SimATTA*	59.05	68.67	44.43	84.96	67.46	83.36	84.99	81.75	82.87	83.83	89.11	72.28	76.15	81.90	73.41	75.62
B	BiTTA	**82.32**	**84.02**	**75.77**	**90.39**	**79.05**	**90.73**	**90.93**	**90.71**	**89.09**	**92.22**	**95.36**	**82.16**	**87.56**	**87.40**	**85.91**	**86.91**

(a) Imbalanced (non-iid) label distribution.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	25.96	33.19	24.71	56.73	52.02	67.37	64.80	**77.97**	67.00	74.14	**91.50**	33.90	**76.61**	46.38	**73.23**	57.70
-	BN-Stats	20.53	21.09	18.15	32.45	20.72	33.45	30.49	28.76	29.29	33.34	36.96	40.55	24.20	25.95	21.43	27.82
B	TENT*	10.50	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.04
B	EATA*	20.53	21.09	18.15	32.45	20.72	33.45	30.49	28.76	29.29	33.34	36.96	40.55	24.20	25.95	21.43	27.82
B	SAR*	20.56	21.12	18.29	32.51	20.86	33.59	30.67	29.12	29.51	33.68	37.52	41.15	24.70	26.57	21.98	28.12
B	CoTTA*	20.54	21.09	18.15	32.44	20.70	33.45	30.49	28.75	29.28	33.33	36.95	40.55	24.20	25.95	21.42	27.82
B	RoTTA*	11.70	10.23	10.03	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.01	10.14
B	SoTTA*	17.02	15.32	13.00	79.00	18.17	57.44	63.39	51.26	49.67	61.47	64.84	**50.27**	53.56	42.18	52.14	45.92
B	BiTTA	**62.14**	**64.01**	**55.13**	**82.07**	**59.64**	**79.22**	**83.26**	75.84	**71.26**	**81.92**	86.13	31.94	71.34	**73.80**	67.73	**70.17**

(b) Batch size 1.

outperforms these methods under equal binary feedback conditions.

Results on additional architectures. To further examine the applicability of BiTTA in various model architectures, we experimented with ResNet50 and ViT-Base. Table 11 shows the overall result, where BiTTA still outperformed the baselines in all corruptions, demonstrating the applicability of BiTTA over diverse models.

Comparison with original TTA and active TTA. In Tables 12 and 13, we compare BiTTA with original TTA (without binary-feedback samples) and original active TTA (with full-labeling) baselines. We first observe that comparing to TTA with binary feedback setting, unsupervised adaptation shows 1.14%p accuracy degradation on average; showcasing the importance of binary-feedback for guiding model adaptation. Also, the experiment results demonstrate the superior performance of BiTTA, even outperforming the active TTA baseline (SimATTA, (Gui et al., 2024)), showing the effectiveness of our RL-based adaptation with binary-feedback adaptation and agreement-based adaptation. We consider this the drawback of SimATTA's strategy of using source-like confident samples. Even with tuning the hyperparameters, the accuracy of source-like samples is highly dependent on the source-pretrained model. This results in noisy predictions, hindering its applicability in various datasets and scenarios.

Comparison with enhanced TTA. Following the setting of SimATTA (Gui et al., 2024), we compare BiTTA with an enhanced TTA setting, which is **unsupervised TTA baselines adapting on the fine-tuned model**, which is tuned with an

Table 9: Average accuracy (%) comparisons. Notation * indicates the modified algorithm to utilize binary-feedback samples. Averaged over three random seeds.

Label	Method	C10-C	C100-C	T-IN-C
B	DeYO*	84.41	61.30	40.67
B	TAST-BN*	75.53	29.13	17.05
B	BiTTA	**87.20**	**62.49**	**40.85**

Table 10: Average accuracy (%) comparisons under the OWTTT pre-trained model. Notation * indicates the modified algorithm to utilize binary-feedback samples. Averaged over three random seeds.

Label	Method	C10-C	C100-C
-	OWTTT	54.63	29.10
B	OWTTT*	31.24	3.39
B	BiTTA	**89.89**	**64.06**

Table 11: Accuracy (%) comparisons with TTA and active TTA baselines with binary feedback in CIFAR10-C (severity level 5) with additional architectures. Notation * indicates the modified algorithm to utilize binary-feedback samples. B: TTA with binary feedback. Results outperforming all other baselines are highlighted in **bold** fonts. Averaged over three random seeds.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	22.56	27.66	21.49	46.91	43.23	55.29	54.62	66.90	53.91	61.31	84.94	24.24	65.29	41.03	65.35	48.98
-	BN-Stats	60.20	62.13	55.50	82.21	58.39	80.01	81.65	75.67	73.78	78.92	86.14	81.86	69.56	73.34	67.23	72.44
B	TENT*	67.91	72.96	63.60	72.68	56.98	62.43	65.48	60.95	58.81	56.47	66.26	64.45	55.80	61.30	57.70	61.58
B	EATA*	75.19	80.89	73.29	81.65	67.68	76.30	79.09	75.80	77.09	76.19	82.23	79.64	68.67	74.07	70.10	75.62
B	SAR*	63.51	70.85	65.95	85.07	66.46	84.06	86.33	82.68	83.24	84.02	90.46	86.74	78.53	83.68	79.10	79.53
B	CoTTA*	60.20	62.13	55.50	82.20	58.40	80.01	81.65	75.68	73.78	78.92	86.14	81.87	69.55	73.36	67.20	72.63
B	RoTTA*	60.77	65.94	60.67	79.87	65.04	82.22	86.19	82.03	84.23	84.58	90.00	85.51	79.68	81.57	81.19	77.88
B	SoTTA*	71.06	80.72	73.98	82.02	67.78	79.96	83.85	81.16	81.96	80.95	87.10	82.77	74.12	78.02	75.80	78.29
B	SimATTA*	33.37	49.99	41.33	62.69	58.03	76.02	81.32	77.35	80.75	79.95	88.83	67.17	76.13	72.59	78.84	68.29
B	BiTTA	**75.72**	**83.25**	**78.58**	**85.41**	**75.75**	**86.14**	**89.82**	**87.28**	**89.55**	**88.83**	**93.67**	**92.04**	**84.93**	**91.91**	**88.38**	**86.08**

(a) ResNet50.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	39.84	41.82	34.89	55.54	56.59	58.42	60.38	65.70	57.74	43.48	72.40	20.83	68.84	66.77	66.92	54.01
B	TENT*	34.98	37.63	28.47	55.19	37.90	55.62	63.70	53.38	56.36	48.73	62.57	20.46	60.85	63.05	62.21	49.41
B	EATA*	**55.23**	61.37	51.56	63.45	62.71	69.75	74.29	67.57	68.86	58.67	72.04	29.22	68.66	70.93	69.05	62.89
B	SAR*	40.74	44.40	36.36	57.77	58.43	63.49	66.87	67.01	62.74	52.41	72.93	22.30	71.71	62.83	68.83	56.59
B	CoTTA*	39.84	41.83	34.89	55.54	56.56	58.43	60.38	65.68	57.69	43.46	72.36	20.82	68.90	66.77	66.94	54.01
B	RoTTA*	38.89	38.69	30.53	61.44	52.76	66.21	71.65	61.72	54.29	59.08	74.23	29.24	71.30	58.42	67.70	55.74
B	SoTTA*	54.59	62.22	51.71	63.62	61.67	67.45	72.72	66.28	67.09	58.19	70.80	30.92	67.18	70.71	67.34	62.17
B	SimATTA*	54.08	60.09	47.30	65.07	57.94	66.37	70.90	64.23	66.15	56.46	71.65	27.98	67.25	68.31	67.40	61.26
B	BiTTA	53.83	**62.40**	**52.23**	**66.77**	**63.88**	**72.27**	**77.66**	**69.97**	**72.40**	**65.10**	**76.07**	**30.70**	**73.80**	**74.84**	**72.65**	**65.64**

(b) ViT-Base.

equal amount of binary-feedback samples before the adaptation phase. In Table 14, we observe that BiTTA still outperforms over enhanced TTA baselines. The result necessitates the superiority of online adaptation on binary feedback samples.

D. Experiment details

We conducted all experiments with three random seeds [0, 1, 2] and reported the mean and standard deviation values. The experiments were mainly conducted on NVIDIA RTX 3090 and TITAN GPUs.

D.1. Settings

Dataset. We utilized the corruption dataset (CIFAR10-C, CIFAR100-C, Tiny-ImageNet-C (Hendrycks & Dietterich, 2019)) and domain generalization baselines (PACS (Li et al., 2017)). CIFAR10-C/CIFAR100-C/Tiny-ImageNet-C is a 10/100/200-class dataset of a total of 150,000 images in 15 types of image corruptions, including Gaussian, Snow, Frost, Fog, Brightness, Contrast, Elastic Transformation, Pixelate, and JPEG Compression. PACS is a 7-class dataset with 9,991 images in four domains of art painting, cartoon, photo, and sketch.

Table 12: Accuracy (%) and standard deviation comparisons with **original TTA and full-label active TTA baselines** in corruption datasets (severity level 5). F: Full-label feedback active TTA, B: TTA with binary feedback. Results that outperform all baselines are highlighted in **bold** font. Averaged over three random seeds.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	25.97	33.19	24.71	56.73	52.02	67.37	64.80	77.97	67.01	74.14	91.51	33.90	76.62	46.38	73.23	57.23
-	BN-Stats	66.96	69.04	60.36	87.78	65.55	86.29	87.38	81.63	80.28	85.39	90.74	86.88	76.72	79.33	71.92	78.42
-	TENT	74.34	77.30	65.86	74.12	54.40	58.08	58.89	53.49	50.45	46.76	48.23	40.65	34.78	34.37	29.62	53.42
-	EATA	76.45	77.33	64.70	77.51	62.31	71.91	78.34	75.29	75.24	78.56	84.68	83.19	68.81	70.97	67.18	74.16
-	SAR	67.94	69.45	62.82	**87.79**	66.18	86.31	87.38	81.63	80.28	85.39	90.74	86.88	76.72	79.33	71.98	78.72
-	CoTTA	66.97	69.04	60.37	87.78	65.55	86.30	87.38	81.63	80.27	85.39	90.74	86.88	76.72	79.33	71.92	78.42
-	RoTTA	65.21	71.11	64.77	85.11	69.73	87.44	89.95	86.05	86.60	87.98	92.73	88.00	82.53	85.49	81.11	81.59
-	SoTTA	74.59	81.22	74.55	84.74	71.41	83.33	87.86	83.68	84.63	85.51	90.34	83.09	78.87	82.88	77.99	81.65
F	SimATTA	73.89	82.45	73.36	79.97	72.14	84.13	88.95	86.22	89.01	87.94	92.81	85.21	80.94	85.93	83.97	83.13
B	BiTTA	**76.78**	**84.24**	**78.75**	87.51	**77.39**	**88.38**	**91.36**	**89.42**	**90.72**	**90.30**	**94.65**	**92.62**	**86.15**	**92.42**	**87.24**	**87.20**

(a) CIFAR10-C.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	10.63	12.14	7.17	34.86	19.58	44.09	41.94	46.34	34.22	41.08	67.31	18.47	50.36	24.91	44.56	33.18
-	BN-Stats	39.23	40.75	34.10	66.14	42.46	63.57	64.82	53.81	53.49	58.15	68.22	64.48	53.88	56.63	45.17	53.66
-	TENT	49.71	51.12	38.34	42.40	24.86	21.51	17.21	9.39	5.84	4.24	3.87	2.56	2.74	2.40	2.36	18.57
-	EATA	10.40	2.88	2.81	2.50	2.22	2.21	1.99	2.17	1.91	1.65	1.53	1.23	1.25	1.12	1.05	2.46
-	SAR	46.45	55.24	48.53	**66.27**	50.93	**65.35**	**68.49**	60.73	62.36	63.37	71.12	**69.48**	59.76	65.34	56.33	60.65
-	CoTTA	39.24	40.75	34.11	66.13	42.46	63.57	64.82	53.81	53.49	58.14	68.22	64.48	53.87	56.63	45.17	53.66
-	RoTTA	35.63	40.04	35.55	60.32	42.09	62.76	67.53	58.54	60.60	60.72	**71.58**	64.08	59.50	63.13	54.49	55.77
-	SoTTA	**52.31**	57.80	48.30	61.57	48.82	63.45	68.17	59.54	61.69	62.62	69.73	66.30	57.40	63.35	56.67	59.85
F	SimATTA	42.86	54.18	44.18	53.98	46.64	60.51	65.54	57.01	62.73	57.25	68.38	52.17	54.53	61.10	56.88	55.86
B	BiTTA	50.12	**58.34**	**52.07**	63.27	**52.70**	63.80	68.16	**62.65**	**65.39**	**63.79**	71.26	68.97	**63.93**	**69.45**	**63.38**	**62.49**

(b) CIFAR100-C.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
-	SrcValid	6.99	8.93	5.09	15.18	9.65	26.50	26.33	29.77	33.64	12.34	31.80	2.34	27.71	34.99	46.97	21.22
-	BN-Stats	31.45	33.28	23.55	32.33	22.30	44.30	45.04	38.89	42.64	29.97	46.55	8.46	43.70	52.53	49.50	36.30
-	TENT	35.97	33.92	18.12	8.67	2.93	2.84	2.57	2.35	1.87	1.86	1.86	1.33	1.57	1.63	1.58	7.94
-	EATA	34.53	36.80	26.46	**36.49**	25.69	47.83	48.33	41.88	44.98	35.83	49.62	6.86	44.86	53.79	50.95	38.99
-	SAR	33.35	38.03	28.94	35.83	**27.12**	47.13	48.39	41.36	45.09	36.79	50.24	**13.46**	46.45	52.44	50.52	39.68
-	CoTTA	31.45	33.29	23.54	32.35	22.27	44.33	44.99	38.94	42.67	29.99	46.57	8.67	43.74	52.58	49.45	36.32
-	RoTTA	31.13	34.94	25.71	31.74	25.01	46.18	47.47	41.40	45.13	31.38	48.01	8.92	45.07	50.77	49.69	37.50
-	SoTTA	**37.62**	**40.91**	**31.72**	33.55	26.75	41.50	44.84	37.72	41.42	38.75	47.04	7.46	34.88	44.08	45.04	36.89
F	SimATTA	23.70	33.82	26.11	23.55	23.36	40.16	43.41	30.22	41.84	26.42	40.72	2.88	41.37	49.21	52.85	33.31
B	BiTTA	34.84	39.88	28.56	35.37	26.65	**48.41**	**49.57**	**43.62**	**47.90**	**39.53**	**50.95**	12.27	**47.18**	**54.01**	**54.06**	**40.85**

(c) Tiny-ImageNet-C.

Source domain pre-training. We closely followed the settings and utilized the pre-trained weights provided by SoTTA (Gong et al., 2023b) and SimATTA (Gui et al., 2024). As the backbone model, we employ the ResNet18 (He et al., 2016) from TorchVision (TorchVision maintainers and contributors, 2016). For CIFAR10-C/CIFAR100-C/Tiny-ImageNet-C, we trained the model with the source data with a learning rate of 0.1/0.1/0.001 and a momentum of 0.9, with cosine annealing learning rate scheduling for 200 epochs. For PACS, we fine-tuned the pre-trained weights from ImageNet on the selected source domains for 3,000 iterations using the Adam optimizer with a learning rate of 0.0001.

Scenario. For the number of binary-feedback samples, we used $k = 3$ samples from a 64-sample test batch, accounting for less than 5% of the total data size. For the binary version of TTA baselines, we added cross-entropy loss (for correct samples) combined with complementary loss (for incorrect samples, (Kim et al., 2019)), maintaining an equal budget size to our method. To implement, we replace the original TTA loss $l_{\texttt{TTA}}$ with $l_{\texttt{TTA}} + l_{\texttt{CE}} + l_{\texttt{CCE}}$, where $l_{\texttt{CE}}$ is a cross-entropy loss on correct samples and $l_{\texttt{CCE}} = -\sum_{k=1}^{\texttt{num\_class}} y_k \log(1 - f_\theta(k|x))$ is the complementary cross-entropy loss (Kim et al., 2019) on incorrect samples.

For enhanced TTA, we used the same binary version loss with an SGD optimizer with a learning rate of 0.001 and a batch size of 64. The number of fine-tuning epochs was set to 150 for PACS, 150 for CIFAR-10, 150 for CIFAR-100, and 25

Table 13: Accuracy (%) and standard deviation comparisons with **original TTA and full-label active TTA baselines** in PACS. The domain-wise data stream is a continual TTA setting (Wang et al., 2022), and the mixed data stream shuffled all domains randomly, where we report the cumulative accuracy at each of the four adaptation points. F: Full-label feedback active TTA, B: TTA with binary feedback. Results outperforming all other baselines are highlighted in **bold** fonts. Averaged over three random seeds.

Label	Method	Domain-wise data stream				Mixed data stream			
		Art	Cartoo-	Sketch	Avg	25%	50%	75%	100%(Avg)
-	SrcValid	59.38 ±0.00	27.94 ±0.21	42.96 ±0.01	43.43 ±0.07	42.74 ±1.13	42.80 ±0.22	42.64 ±0.30	42.77 ±0.01
-	BN Stats	67.87 ±0.18	63.48 ±0.88	54.07 ±0.36	61.81 ±0.18	59.09 ±0.29	58.28 ±0.08	58.05 ±0.22	57.82 ±0.20
-	TENT	71.61 ±0.70	67.00 ±0.51	44.14 ±0.85	60.92 ±0.29	60.34 ±0.51	56.75 ±0.62	53.22 ±0.57	49.64 ±0.50
-	EATA	68.44 ±0.31	64.90 ±0.69	58.58 ±0.18	63.97 ±0.23	59.60 ±0.15	58.98 ±0.54	59.10 ±0.38	59.24 ±0.08
-	SAR	67.90 ±0.20	63.60 ±0.83	55.23 ±0.44	62.25 ±0.11	59.13 ±0.21	58.49 ±0.15	58.32 ±0.05	58.25 ±0.07
-	CoTTA	67.87 ±0.18	63.48 ±0.88	54.06 ±0.35	61.81 ±0.19	59.10 ±0.32	58.29 ±0.09	58.06 ±0.23	57.83 ±0.22
-	RoTTA	64.39 ±0.59	38.27 ±0.61	40.80 ±1.64	47.82 ±0.20	52.64 ±0.25	49.01 ±0.85	46.87 ±0.55	45.75 ±0.49
-	SoTTA	69.86 ±0.78	32.02 ±1.52	23.66 ±1.77	41.84 ±0.34	51.96 ±5.47	49.84 ±6.14	48.09 ±6.64	47.06 ±6.03
F	SimATTA	**77.13 ±0.76**	71.46 ±2.47	**78.80 ±0.53**	**75.80 ±0.74**	**68.27 ±1.24**	**72.67 ±0.45**	**75.41 ±0.30**	**77.47 ±0.44**
B	BiTTA	73.86 ±3.76	**76.81 ±2.45**	76.03 ±1.61	75.57 ±0.93	59.65 ±0.70	64.70 ±0.78	69.23 ±0.17	72.18 ±0.38

Table 14: Accuracy (%) comparisons with **enhanced TTA on fine-tuned model** (Gui et al., 2024) and TTA with binary feedback baselines on source model, in CIFAR10-C (severity level 5). Notation * indicates the modified algorithm to utilize binary-feedback samples. E: Enhanced TTA, B: TTA with binary feedback. Results outperforming all other baselines are highlighted in **bold** fonts. Averaged over three random seeds.

Label	Method	Noise			Blur				Weather				Digital				
		Gau.	Shot	Imp.	Def.	Gla.	Mot.	Zoom	Snow	Fro.	Fog	Brit.	Cont.	Elas.	Pix.	JPEG	Avg.
E	SrcValid	76.17	77.48	67.54	82.24	71.89	79.90	83.44	82.67	84.36	81.18	88.74	75.12	77.53	80.66	80.24	79.28
E	BN-Stats	77.90	79.66	71.76	86.52	73.53	85.26	86.77	84.66	85.27	84.07	90.10	86.70	79.39	84.76	78.98	82.36
E	TENT	77.52	76.94	63.79	68.35	52.67	56.00	55.58	52.93	49.02	45.02	43.94	33.46	32.12	31.39	29.27	51.20
E	EATA	77.18	75.32	64.66	70.73	58.46	64.62	70.22	68.00	68.34	67.35	75.81	69.52	62.93	69.02	64.28	68.43
E	SAR	77.90	79.66	71.76	86.52	73.53	85.26	86.77	84.66	85.27	84.07	90.10	86.70	79.39	84.76	78.98	82.36
E	CoTTA	77.90	79.66	71.77	86.52	73.53	85.26	86.77	84.66	85.27	84.06	90.09	86.71	79.39	84.76	78.98	82.36
E	RoTTA	78.93	81.00	74.28	86.56	75.45	86.18	88.63	86.85	87.71	86.73	91.36	88.06	82.41	87.19	82.42	84.25
E	SoTTA	**79.19**	81.45	74.23	82.67	70.73	81.99	85.41	82.78	83.69	85.02	89.40	84.41	78.41	83.44	78.94	81.45
B	SimATTA*	70.21	81.67	71.49	79.59	69.41	82.15	87.28	83.90	86.89	86.49	91.51	83.40	77.94	83.81	82.25	81.09
B	BiTTA	76.78	**84.24**	**78.75**	**87.51**	**77.39**	**88.38**	**91.36**	**89.42**	**90.72**	**90.30**	**94.65**	**92.62**	**86.15**	**92.42**	**87.24**	**87.20**

for Tiny-ImageNet-C. Note that the hyperparameters were selected to maximize accuracy on the test data stream, which is unrealistic since test data stream accuracy is not accessible during the fine-tuning process.

Comparison with active TTA. To compare BiTTA with full-label feedback methods, we propose two scenarios: (1) an equal labeling cost and (2) an equal number of active samples. To compare with an equal labeling cost, we formulate the labeling cost with Shannon information gain (MacKay, 2003) as $\log(p^{-1})$ where p is the probability of selecting a label. We assume the probability of each feedback strategy as $p = 2^{-1}$ (correct/incorrect) and $p = \texttt{num\_class}^{-1}$ (select in the entire class set). The final labeling cost for binary feedback is 1 for binary feedback and $\log(\texttt{num\_class})$ for full-label feedback. Therefore, we utilize $\log(\texttt{num\_class})$ times more feedback samples for TTA with binary feedback setting compared to active TTA.

D.2. TTA Baselines

TENT. For TENT (Wang et al., 2021), we utilize an Adam optimizer (Kingma & Ba, 2015) with a learning rate $LR = 0.001$, aligning with the guidelines outlined in the original paper and active TTA paper (Gui et al., 2024). The implementation followed the official code.[1]

EATA. For EATA (Niu et al., 2022), we followed the original configuration of $LR = 0.001$, entropy constant $E_0 = 0.4 \times \ln C$, where C represents the number of classes. Additionally, we set the cosine sample similarity threshold $\epsilon = 0.5$, trade-off parameter $\beta = 2,000$, and moving average factor $\alpha = 0.1$. The Fisher importance calculation involved 2,000

[1] https://github.com/DequanWang/tent

samples, as recommended. The implementation followed the official code.[2]

SAR. For SAR (Niu et al., 2023), we set a learning rate of $LR = 0.00025$, sharpness threshold $\rho = 0.5$, and entropy threshold $E_0 = 0.4 \times \ln C$, following the recommendations from the original paper. The top layer (layer 4 for ResNet18) was frozen, consistent with the original paper. The implementation followed the official code.[3]

CoTTA. For CoTTA (Wang et al., 2022), we set the restoration factor $p = 0.01$, and exponential moving average (EMA) factor $\alpha = 0.999$. For augmentation confidence threshold p_{th}, we followed the previous implementation (Gui et al., 2024) as $p_{th} = 0.1$. The implementation followed the official code.[4]

RoTTA. For RoTTA (Yuan et al., 2023), we utilized the Adam optimizer (Kingma & Ba, 2015) with a learning rate of $LR = 0.001$ and $\beta = 0.9$. We followed the original hyperparameters, including BN-statistic exponential moving average updating rate $\alpha = 0.05$, Teacher model's exponential moving average updating rate $\nu = 0.001$, timeliness parameter $\lambda_t = 1.0$, and uncertainty parameter $\lambda_u = 1.0$. The implementation followed the original code.[5]

SoTTA. For SoTTA (Gong et al., 2023b), we utilized the Adam optimizer (Kingma & Ba, 2015), with a BN momentum of $m = 0.2$ and a learning rate of $LR = 0.001$. The memory size was set to 64, with the confidence threshold $C_0 = 0.99$. The entropy-sharpness L2-norm constraint ρ was set to 0.5, aligning with the suggestion (Foret et al., 2021). The top layer was frozen following the original paper. The implementation followed the original code.[6]

SimATTA. We follow the original implementation of SimATTA (Gui et al., 2024). Since SimATTA queries active samples at a dynamic rate, we set the centroid increase number to $k = 3$ and limit the budget per batch to 3, ensuring an equal active sample budget compared to BiTTA. For the adaptation objective, we add the complementary loss (incorrect samples, (Kim et al., 2019)) to the original cross-entropy loss for correct samples. For CIFAR-10 and CIFAR-100, we performed a grid search to find the optimal hyperparameters. We found the optimal hyperparameters to be $LR = 0.0001/0.0001$, $e_h = 0.001/0.001$, and $e_l = 0.0001/0.00001$ for the CIFAR-10 and CIFAR-100 datasets, respectively. The implementation is based on the original code.[7]

BiTTA. During adaptation, we update all parameters, including BN stats, with an SGD optimizer with a learning rate/epoch of 0.001/3 (PACS), 0.0001/3 (CIFAR10-C, CIFAR100-C), and 0.00005/5 (Tiny-ImageNet-C) on the entire model. We applied weight decay of 0.05 to PACS and 0.0 otherwise. We applied stochastic restoration (Wang et al., 2022) in Tiny-ImageNet-C to prevent overfitting. We update batch norm statistics with momentum 0.3 on the unlabeled test batch, and freeze the statistics during adaptation, following Gui et al. (2024). We apply the dropout layer after residual blocks, following the previous work on TTA accuracy estimation (Lee et al., 2024b). With 4 dropout instances, we apply a dropout rate of 0.3 for small-scale datasets (e.g., CIFAR10-C, CIFAR100-C, PACS) and 0.1 for large-scale datasets (e.g., Tiny-ImageNet-C, ImageNet-C).

E. License of assets

Datasets. CIFAR10-C/CIFAR100-C (Creative Commons Attribution 4.0 International), and Tiny-ImageNet-C dataset (Apache-2.0). The license of PACS dataset is not specified.

Codes. Torchvision for ResNet18 (Apache 2.0), the official repository of TENT (MIT License), the official repository of EATA (MIT License), the official repository of SAR (BSD 3-Clause License), the official repository of CoTTA (MIT License), the official repository of RoTTA (MIT License), the official repository of SoTTA (MIT License), and the official repository of SimATTA (GPL-3.0 License).

[2] https://github.com/mr-eggplant/EATA
[3] https://github.com/mr-eggplant/SAR
[4] https://github.com/qinenergy/cotta
[5] https://github.com/BIT-DA/RoTTA
[6] https://github.com/taeckyung/sotta
[7] https://github.com/divelab/ATTA