Uncertainty-Calibrated Test-Time Model Adaptation without Forgetting

Mingkui Tan*, Guohao Chen*, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Peilin Zhao, and Shuaicheng Niu†

Abstract—Test-time adaptation (TTA) seeks to tackle potential distribution shifts between training and testing data by adapting a given model w.r.t. any testing sample. This task is particularly important when the test environment changes frequently. Although some recent attempts have been made to handle this task, we still face two key challenges: 1) prior methods have to perform backpropagation for each test sample, resulting in unbearable optimization costs to many applications; 2) while existing TTA solutions can significantly improve the test performance on out-of-distribution data, they often suffer from severe performance degradation on in-distribution data after TTA (known as catastrophic forgetting). To this end, we have proposed an Efficient Anti-Forgetting Test-Time Adaptation (EATA) method which develops an active sample selection criterion to identify reliable and non-redundant samples for test-time entropy minimization. To alleviate forgetting, EATA introduces a Fisher regularizer estimated from test samples to constrain important model parameters from drastic changes. However, in EATA, the adopted entropy loss consistently assigns higher confidence to predictions even when the samples are underlying uncertain, leading to overconfident predictions that underestimate the data uncertainty. To tackle this, we further propose EATA with Calibration (EATA-C) to separately exploit the reducible model uncertainty and the inherent data uncertainty for calibrated TTA. Specifically, we compare the divergence between predictions from the full network and its sub-networks to measure the reducible model uncertainty, on which we propose a test-time uncertainty reduction strategy with divergence minimization loss to encourage consistent predictions instead of overconfident ones. To further re-calibrate predicting confidence on different samples, we utilize the disagreement among predicted labels as an indicator of the data uncertainty. Based on this, we devise a min-max entropy regularization to selectively increase and decrease predicting confidence for confidence re-calibration. Note that EATA-C and EATA are different on the adaptation objective, while EATA-C still benefits from the active sample selection criterion and anti-forgetting Fisher regularization proposed in EATA. Extensive experiments on image classification and semantic segmentation verify the effectiveness of our proposed methods.

Index Terms—Out-of-Distribution Generalization, Test-Time Adaptation, Confidence Calibration, Catastrophic Forgetting.

1 Introduction

Deep neural networks (DNNs) have achieved excellent performance in many challenging tasks, including image classification [1], video recognition [2], [3], [4], [5], and many other areas [6], [7], [8]. One prerequisite behind the success of DNNs is that the test samples are drawn from the same distribution as the training data, which, however, is often violated in many real-world applications. In practice, test samples may encounter natural variations or corruptions (also called *distribution shift*), such as changes in lighting resulting from weather changes and unexpected noises resulting from sensor degradation [9], [10]. Unfortunately, models are often very sensitive to such distribution shifts and suffer severe performance degradation.

Recently, several attempts [11], [12], [13], [14], [15], [16] have been proposed to handle the distribution shifts by online adapting a model at test time (called *test-time adaptation*). Test-time training (TTT) [11] first proposes this pipeline. Given a test sample, TTT first fine-tunes the model via rotation classification [17] and then makes a prediction using the updated model. Without the need of training an additional self-supervised head, Tent [12] and MEMO [14] further leverage the prediction entropy for test-time adaptation, in which the adaptation only involves test samples and a trained model. Although recent test-time adaptation methods are effective at handling test shifts, in real-world applications, they still suffer from the following limitations.

Latency Constraints. Since TTA adapts a given model during inference, the adaptation efficiency is paramount in scenarios where latency is a critical factor. Previous methods, such as Test-Time Training (TTT) [11] and MEMO [14], often require performing multiple backward propagations for each test sample. However, the computation-intensive nature of backward propagation renders these methods impractical in situations where low latency is non-negotiable or computational resources are limited.

Forgetting on In-Distribution Samples. Prior methods often focus on boosting the performance of a trained model on out-of-distribution (OOD) test samples, ignoring that the model after test-time adaptation suffers a severe performance degradation (named *forgetting*) on in-distribution (ID) test samples (see Figure 3). An eligible test-time adaptation approach should perform well on both OOD and ID test samples simultaneously, since test samples often come from both ID and OOD domains in real-world applications.

Over-Confident Predictions. Existing methods like Tent [12] and SAR [18] primarily rely on test-time entropy minimization for model adaptation, which greedily enhances the model's confidence and minimizes the predictive uncertainty for test samples, without

- *Mingkui Tan, Guohao Chen, and Yaofo Chen are with the School of Software Engineering, South China University of Technology. Mingkui Tan and Guohao Chen are also with Pazhou Laboratory, Guangzhou, China. Email: mingkuitan@scut.edu.cn, {secasper, sechenyaofo}@mail.scut.edu.cn.*
- *Jiaxiang Wu is with XVERSE, China. The majority of this work was conducted while at Tencent AI Lab. Email: jiaxiang.wu.90@gmail.com.*
- *Yifan Zhang is with the School of Computing, National University of Singapore. Email: yifan.zhang@u.nus.edu.*
- *Peilin Zhao is with Tencent AI Lab, China. Email: masonzhao@tencent.com.*
- *Shuaicheng Niu is with the College of Computing and Data Science, Nanyang Technological University, Singapore. Email: shuaicheng.niu@ntu.edu.sg.*

** Authors contributed equally. † Corresponding author.*

TABLE 1
Characteristics of problem settings that adapt a trained model to a potentially shifted test domain. 'Offline' adaptation assumes access to the entire source or target dataset, while 'Online' adaptation can predict a single or batch of incoming test samples immediately.

Setting	Source Data	Target Data	Training Loss	Testing Loss	Offline	Online	Source Acc.	Prediction Uncertainty
Fine-tuning	✗	$\mathbf{x}^t, y^t$	$\mathcal{L}(\mathbf{x}^t, y^t)$	–	✓	✗	Not Considered	Not Considered
Continual learning	✗	$\mathbf{x}^t, y^t$	$\mathcal{L}(\mathbf{x}^t, y^t)$	–	✓	✗	Maintained	Not Considered
Unsupervised domain adaptation	$\mathbf{x}^s, y^s$	$\mathbf{x}^t$	$\mathcal{L}(\mathbf{x}^s, y^s) + \mathcal{L}(\mathbf{x}^s, \mathbf{x}^t)$	–	✓	✗	Maintained	Not Considered
Test-time training	$\mathbf{x}^s, y^s$	$\mathbf{x}^t$	$\mathcal{L}(\mathbf{x}^s, y^s) + \mathcal{L}(\mathbf{x}^s)$	$\mathcal{L}(\mathbf{x}^t)$	✗	✓	Not Considered	Not Considered
Fully test-time adaptation (FTTA)	✗	$\mathbf{x}^t$	✗	$\mathcal{L}(\mathbf{x}^t)$	✗	✓	Not Considered	Not Considered
EATA (ours)	✗	$\mathbf{x}^t$	✗	$\mathcal{L}(\mathbf{x}^t)$	✗	✓	Maintained	Not Considered
EATA-C (ours)	✗	$\mathbf{x}^t$	✗	$\mathcal{L}(\mathbf{x}^t)$	✗	✓	Maintained	Calibrated

distinguishing between model-induced and data-induced uncertainties. Consequently, even when the input data is naturally complex or highly corrupted (*i.e.*, with *irreducible* data uncertainty), the model is forced to make one-hot confident predictions where it should remain uncertain, leading to over-confident and potentially incorrect outputs. This phenomenon is particularly concerning in high-risk applications, such as autonomous driving [19] and medical diagnosis [20], posing potential safety risks.

To address the efficiency and forgetting issue, we have proposed an Efficient Anti-forgetting Test-time Adaptation (EATA) method consisting of a sample-efficient optimization strategy and a weight regularizer. EATA excludes unreliable samples characterized by high entropy values and redundant samples that are highly similar throughout the adaptation. In this case, we can reduce the total number of backward updates of test data streaming (improving efficiency) and enhance the model performance on OOD samples. Furthermore, EATA devises an anti-forgetting regularizer to prevent the important weights of the model from changing a lot during the adaptation, where the weights' importance is measured based on Fisher information [21] via a small set of test samples. With this regularization, the model can continually adapt to OOD test samples without performance degradation on ID test samples.

To mitigate overconfidence, we differentiate between various origins of uncertainty in TTA: 1) Reducible model uncertainty, which arises from not knowing the optimal model parameters to describe the data due to *insufficient training* [22]; 2) Irreducible data uncertainty that arises from inherent noise or variability in the data, *cannot be reduced by additional training* [23]. Based on their characteristics, we aim to reduce the model uncertainty at test time for domain adaptation, while accurately reflecting data uncertainty in model predictions to ensure confidence calibration.

To this end, we further devise EATA with Calibration, namely EATA-C. Specifically, EATA-C estimates model uncertainty by measuring the divergence between predictions from the full network and its randomly sampled sub-networks. By minimizing this divergence, our EATA-C reduces model uncertainty and promotes consistent, rather than overconfident, predictions for model adaptation during testing. Additionally, we introduce a data uncertainty indicator based on prediction disagreement, which effectively detects ambiguous samples near decision boundaries where conflicting predictions are more likely to occur. We then incorporate a min-max entropy regularizer to selectively adjust the prediction confidence based on this data uncertainty estimation. Note that EATA-C differs from EATA on the adaptation objective, while it still benefits from the active sample selection criterion and anti-forgetting Fisher regularization proposed in EATA. We summarize our main contributions as follows.

- We propose an Efficient Anti-forgetting Test-time Adaptation (EATA) method. Specifically, we reveal that test samples contribute differently to adaptation, and develop an active sample identification scheme to filter out non-reliable and redundant test samples from adaptation, thereby improving TTA efficiency. Moreover, we extend the label-dependent Fisher regularizer to test samples with pseudo label generation, which prevents drastic changes in important model weights and helps alleviate the issue of model forgetting on in-distribution test samples.
- We further introduce EATA with Calibration (EATA-C), which differentiates between reducible and irreducible uncertainty during testing to design a calibration-driven learning objective. Specifically, EATA-C estimates model uncertainty using the divergence between full network and sub-network predictions, incorporating a consistency loss to reduce this uncertainty for adaptation. Regarding data uncertainty, EATA-C leverages prediction disagreements and applies min-max entropy regularization to selectively adjust confidence for calibration enhancement.
- We demonstrate that our proposed EATA method improves both the performance and efficiency of test-time adaptation and also alleviates the long-neglected catastrophic forgetting issue. Our EATA-C further achieves better performance and calibration, with computational and memory efficiency comparable to EATA.

A short version of this work was published in ICML 2022 [24]. This paper extends our preliminary version from the following aspects: 1) We explore the calibrated test-time adaptation, which aims to provide calibrated predicting confidence that reflects the true likelihood of correctness during unsupervised adaptation; 2) To solve the over-confident issue, we develop a test-time consistency loss that leverages the reducible model uncertainty for calibrated uncertainty reduction, and devise a min-max entropy regularizer to re-calibrate predicting confidence based on the inherent data uncertainty; 3) We provide analyses about the impact of different uncertainty reduction strategies, empirically verifying that our consistency loss overcomes the issue of over-fitting and over-confident in entropy minimization loss when adapting to the test data; 4) We provide extensive new empirical evaluations on image classification and semantic segmentation tasks with various model architectures, demonstrating that EATA-C achieves substantially better performance and calibration over EATA, *e.g.*, improving accuracy by 6.5%, while reducing calibration error by relatively 64.9% on ImageNet-C dataset with ViT-Base [25].

2 Related Work

We divide the discussion on related works based on the different adaptation settings summarized in Table 1 and further review existing methods for model's uncertainty calibration.

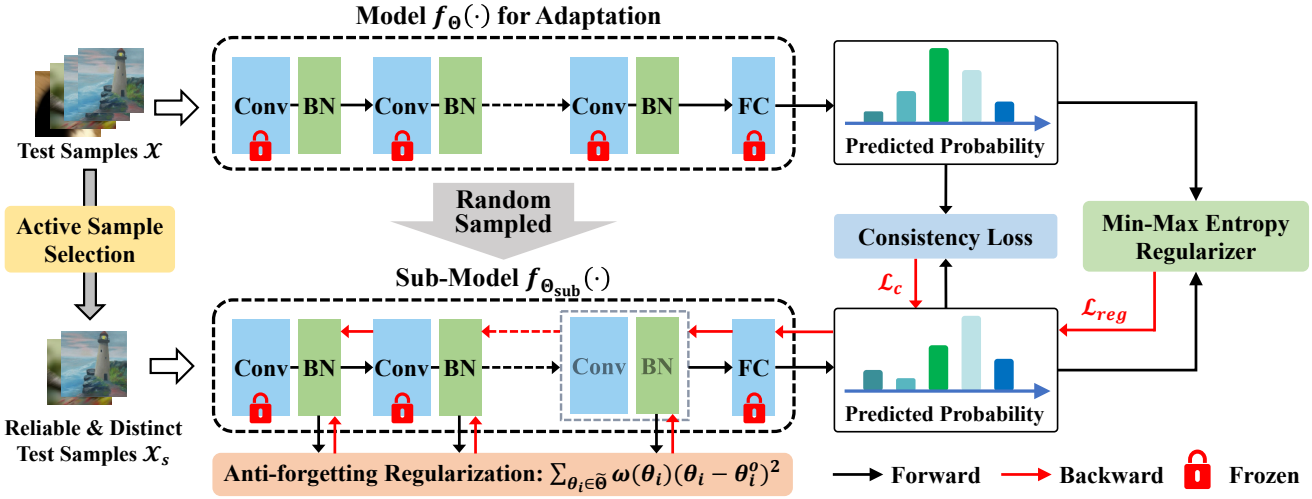


Fig. 1. An illustration of our proposed Efficient Anti-forgetting Test-time Adaptation with Calibration (EATA-C) method. During the test-time adaptation process, we update only the affine parameters of normalization layers in f_Θ and keep all other parameters frozen. Given a batch of incoming test samples $\mathcal{X}=\{\mathbf{x}_b\}_{b=1}^B$, we select the reliable and non-redundant ones $\mathcal{X}_s$ with an active sample selection criterion to conduct model update, thereby enhancing adaptation efficiency. These samples are then used for calculating the proposed unidirectional consistency loss to minimize the model uncertainty. Additionally, we devise a min-max entropy regularizer for confidence re-calibration based on the data uncertainty of each sample. Lastly, we introduce an anti-forgetting regularizer which prevents the important model parameters in Θ from changing too much.

Test-Time Adaptation (TTA) aims to improve model accuracy on OOD test data through model adaptation with test samples. Existing test-time training methods, *e.g.*, TTT [11], TTT++ [13], TTT-MAE [26], and MT3 [27], jointly train a source model via both supervised and self-supervised objectives, and then adapt the model via self-supervised objective at test time. This pipeline, however, necessitates both self-supervised head and test data in adaptation, while training such self-supervised head can be computation-consuming [26]. To address this, some methods have been proposed to adapt a model with only test data, including batchnorm statistics adaptation [28], [29], [30], prediction consistency maximization over different augmentations [31], and classifier adjustment [32]. Specifically, Tent [12] updates the model to minimize the entropy of predictions at test time. MEMO [14] further augments test samples for marginal entropy minimization to enhance robustness. Our work also alleviates the dependency on self-supervision heads and seeks to address the key limitations of prior works (*i.e.*, efficiency hurdle, catastrophic forgetting, and overconfidence) to make TTA more practical in real-world applications.

Continual Learning (CL) aims to help the model remember the essential concepts that have been learned previously, alleviating the catastrophic forgetting issue when learning a new task [21], [33], [34], [35], [36], [37]. In our work, we share the same motivation as CL and point out that test-time adaptation also suffers catastrophic forgetting (*i.e.*, performance degradation on ID test samples), which makes TTA approaches unstable to deploy. To conquer this, we propose a simple yet effective solution to maintain the model performance on ID test samples (by only using test data) and meanwhile improve the performance on OOD test samples.

Unsupervised Domain Adaptation (UDA). Conventional UDA tackles distribution shifts by jointly optimizing a source model on both labeled source data and unlabeled target data, such as devising a domain discriminator to learn domain-invariant features [38], [39], [40], [41]. To avoid access to source data, recently CPGA [42] generates feature prototypes for each category with pseudo-labeling. SHOT [43] learns a target-specific feature extractor by information maximization for representations alignment. Nevertheless, such methods optimize offline via multiple epochs and losses. In contrast, our method adapts in an online manner and selectively performs once backward propagation for one given target sample, which is more efficient during inference.

Uncertainty Calibration. A calibrated model refers to whose predicting confidence reflects the true likelihood of correctness. Post-training processing methods [44], [45], [46] re-calibrate a trained model by leveraging a labeled dataset within the target domain to estimate calibration error. In contrast, regularization-based methods [47], [48], [49], [50] introduce auxiliary objectives to improve calibration at the training phase. Recently, SB-ECE [51] proposes a differentiable estimation of calibration error as regularization to be jointly minimized. ESD [52] further reformulates the calibration objective in a class-wise manner to enhance calibration performance. Nevertheless, these methods necessitate labeled data from the source or target domain, which limits their applicability. Unlike these methods, we seek to improve calibration with only access to unlabeled test data in an online manner in TTA context.

3 Problem Formulation

Without loss of generality, let $P(\mathbf{x})$ be the distribution of training data $\{\mathbf{x}_i\}_{i=1}^N$ (namely $\mathbf{x}_i \sim P(\mathbf{x})$) and $f_{\Theta^o}(\mathbf{x})$ be a **base model** trained on labeled training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, where Θ^o denotes the model parameters. Due to the training process, the model $f_{\Theta^o}(\mathbf{x})$ tends to fit (or overfit) the training data. During the inference state, the model shall perform well for the in-distribution test data, namely $\mathbf{x} \sim P(\mathbf{x})$. However, in practice, due to possible distribution shifts between training and test data, we may encounter many out-of-distribution test samples, namely $\mathbf{x} \sim Q(\mathbf{x})$, where $Q(\mathbf{x}) \neq P(\mathbf{x})$. In this case, the prediction would be very unreliable and the performance is also very poor.

Test-time adaptation (TTA) [12], [14] aims at boosting the out-of-distribution prediction performance by doing model adaptation on test data only. Specifically, given a set of test samples $\{\mathbf{x}_j\}_{j=1}^M$, where $\mathbf{x}_j \sim Q(\mathbf{x})$ and $Q(\mathbf{x}) \neq P(\mathbf{x})$, one needs to adapt $f_\Theta(\mathbf{x})$ to improve the prediction performance on test data in any cases. To

achieve this, existing methods often seek to update the model by minimizing some unsupervised objective defined on test samples:

$$\min_{\tilde{\Theta}} \mathcal{L}(\mathbf{x}; \Theta),\ \mathbf{x} \sim Q(\mathbf{x}), \tag{1}$$

where $\tilde{\Theta} \subseteq \Theta$ denotes the free model parameters that should be updated. In general, the test-time learning objective $\mathcal{L}(\cdot)$ can be formulated as an entropy minimization problem [12] or prediction consistency maximization over data augmentations [14], *etc.*.

For existing TTA methods like TTT [11] and MEMO [14]), during the test-time adaptation, we shall need to compute one round or even multiple rounds of backward computation for each sample, which is very time-consuming and also not favorable for latency-sensitive applications. Moreover, most methods assume that all the test samples are drawn from out-of-distribution (OOD). However, in practice, the test samples may come from both in-distribution (ID) and OOD. Simply optimizing the model on OOD test samples may lead to severe performance degradation on ID test samples. We empirically validate the existence of this issue in Figure 3, where the updated model has a consistently lower accuracy on ID test samples than the original model.

Moreover, existing entropy-based test-time adaptation methods like Tent [12] and SAR [18] consistently encourage the model to produce one-hot highly confident predictions. However, in practice, input test samples can be naturally complex and severely corrupted [9], resulting in irreducible data uncertainty. Ideally, these samples should be predicted with relatively low confidence to reflect their ambiguity. Nevertheless, the data uncertainty is often overlooked by methods based on entropy minimization, causing the adapted model to produce highly confident predictions (*called overconfidence*), even when predictions should remain uncertain. Such misleading predictions raise potential safety concerns for real-world application scenarios. We empirically demonstrate the issue of overconfidence in Figure 11(a) and Table 2.

4 Uncertainty-Calibrated Efficient Anti-forgetting Test-time Adaptation

In this section, we first propose an **Efficient Anti-forgetting Test-time Adaptation (EATA)** method, which aims to improve the efficiency of test-time adaptation (TTA) and tackle the catastrophic forgetting issue brought by existing TTA strategies simultaneously. EATA consists of two strategies. **1)** *Sample-efficient entropy minimization* (c.f. Section 4.1) aims to conduct efficient adaptation relying on an active sample selection strategy. Here, the sample selection process is to choose only active samples for backward propagation and therefore improve the overall TTA efficiency (*i.e.*, less gradient backward propagation). To this end, we devise an active sample selection score, denoted by $S(\mathbf{x})$, to detect those reliable and non-redundant test samples from the test set for TTA. **2)** *Anti-forgetting weight regularization* (c.f. Section 4.2) seeks to alleviate knowledge forgetting by enforcing that the parameters, important for the ID domain, do not change too much in TTA. In this way, the catastrophic forgetting issue can be significantly alleviated. We illustrate EATA in Figure A in Supplementary.

To further address the overconfidence issue, we propose an **Efficient Anti-forgetting Test-time Adaptation with Calibration (EATA-C)** method. As shown in Figure 1, we introduce a new consistency-based test-time learning objective for model uncertainty reduction (c.f. Section 4.3), follow up a min-max entropy regularizer to re-calibrate the prediction uncertainty according to the inherent data uncertainty (c.f. Section 4.4).

4.1 Sample Efficient Entropy Minimization

For efficient test-time adaptation, we propose an active sample identification strategy to select samples for backward propagation. Specifically, we design an active sample selection score for each sample, denoted by $S(\mathbf{x})$, based on two criteria: 1) samples should be **reliable** for test-time adaptation, and 2) the samples involved in optimization should be **non-redundant**. By setting $S(\mathbf{x})=0$ for non-active samples, namely the unreliable and redundant samples, we can reduce unnecessary backward computation during test-time adaptation, thereby improving the prediction efficiency.

Relying on the sample score $S(\mathbf{x})$, following [12], [14], we use entropy loss for model adaptations. Then, the sample-efficient entropy minimization is to minimize the following objective:

$$\min_{\tilde{\Theta}} S(\mathbf{x})E(\mathbf{x};\Theta) = -S(\mathbf{x}) \sum_{y \in \mathcal{C}} f_\Theta(y|\mathbf{x}) \log f_\Theta(y|\mathbf{x}), \tag{2}$$

where $\mathcal{C}$ is the model output space. Here, the entropy loss $E(\cdot)$ is calculated over a batch of samples each time (similar to Tent [12]) to avoid a trivial solution, *i.e.*, assigning all probability to the most probable class. For efficient adaptation, we update $\tilde{\Theta} \subseteq \Theta$ with the affine parameters of all normalization layers.

Reliable Sample Identification. Our intuition is that different test samples produce various effects in adaptation. To verify this, we conduct a preliminary study, where we select different proportions of samples (the samples are pre-sorted according to their entropy values $E(\mathbf{x};\Theta)$) for adaptation, and the resulting model is evaluated on all test samples. From Figure 2, we find that: 1) adaptation on low-entropy samples makes more contribution than high-entropy ones, and 2) adaptation on test samples with very high entropy may hurt performance. The possible reason is that predictions of high-entropy samples are uncertain, so their gradients produced by entropy loss may be biased and unreliable. Following this, we name these low-entropy samples as reliable samples. Based on the above observation, we propose an entropy-based weighting scheme to identify reliable samples and emphasize their contributions during adaptation. Formally, the entropy-based weight is given by:

$$S^{ent}(\mathbf{x}) = \frac{1}{\exp\left[E(\mathbf{x};\Theta) - E_0\right]} \cdot \mathbb{I}_{\{E(\mathbf{x};\Theta) < E_0\}}(\mathbf{x}), \tag{3}$$

where $\mathbb{I}_{\{\cdot\}}(\cdot)$ is an indicator function, $E(\mathbf{x};\Theta)$ is the predicted entropy regarding sample $\mathbf{x}$, and E_0 is a pre-defined threshold. The above weighting function excludes high-entropy samples from adaptation and assigns higher weights to test samples with lower prediction uncertainties, allowing them to contribute more to model updates. Note that evaluating $S^{ent}(\mathbf{x})$ does not involve any gradient back-propagation.

Non-redundant Sample Identification. Although Eqn. (3) helps to exclude partial unreliable samples, the remaining test samples may still have redundancy. For example, given two test samples that are mutually similar and both have a lower prediction entropy than E_0, we still need to perform gradient back-propagation for each of them according to Eqn. (3). However, this is unnecessary as these two samples produce similar gradients for model adaptation.

To further improve efficiency, we propose to exploit the samples that can produce different gradients for model adaptation. Recall that the entropy loss only relies on final model outputs (*i.e.*, classification logits), we further filter samples by ensuring the remaining samples have diverse model outputs. To this end, one straightforward method is to save the model outputs of all

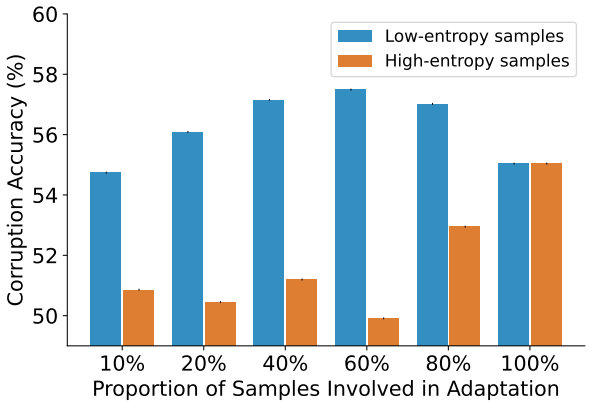


Fig. 2. Effect of different test samples in test-time entropy minimization [12]. We adapt a model on partial samples (top p% samples with the highest or lowest entropy values), and then evaluate the adapted model on all test samples. Results are obtained on ImageNet-C (Gaussian noise, level 3) and ResNet-50 (base accuracy is 27.6%). Introducing more samples with high entropy values into adaptation will hurt the adaptation performance.

previously seen samples, and then compute the similarity between the outputs of incoming test samples and all saved model outputs for filtering. However, this method is computationally expensive at test time and memory-consuming with the increase of test samples.

To address this, we exploit an exponential moving average technique to track the average model outputs of all seen test samples used for model adaptation. To be specific, given a set of model outputs of test samples, the moving average vector is updated recursively:

$$\mathbf{m}^t = \begin{cases} \bar{\mathbf{y}}^1, & \text{if } t = 1 \\ \alpha\bar{\mathbf{y}}^t + (1-\alpha)\mathbf{m}^{t-1}, & \text{if } t > 1 \end{cases}, \tag{4}$$

where $\bar{\mathbf{y}}^t = \frac{1}{n}\sum_{k=1}^{n}\hat{\mathbf{y}}_k^t$ is the average model prediction of a mini-batch of n test samples at the iteration t, and $\alpha \in [0, 1]$. Following that, given a new test sample $\mathbf{x}$ received at iteration $t > 1$, we compute the cosine similarity between its prediction $f_\Theta(\mathbf{x})$ and the moving average $\mathbf{m}^{t-1}$ (*i.e.*, $cos(f_\Theta(\mathbf{x}), \mathbf{m}^{t-1})$), which is then used to determine the diversity-based weight:

$$S^{div}(\mathbf{x}) = \mathbb{I}_{\{cos(f_\Theta(\mathbf{x}),\mathbf{m}^{t-1})<\epsilon\}}(\mathbf{x}), \tag{5}$$

where ϵ is a pre-defined threshold for cosine similarities. The overall sample-adaptive weight is then given by:

$$S(\mathbf{x}) = S^{ent}(\mathbf{x}) \cdot S^{div}(\mathbf{x}), \tag{6}$$

which combines both entropy-based (as in Eqn. 3) and diversity-based terms (as in Eqn. 5). Since we only perform gradient back-propagation for test samples with $S(\mathbf{x}) > 0$, the algorithm efficiency is further improved.

Remark. Given M test samples $\mathcal{D}_{test} = \{\mathbf{x}_j\}_{j=1}^{M}$, the total number of reduced backward computations is given by $\mathbb{E}_{\mathbf{x}\sim\mathcal{D}_{test}}[\mathbb{I}_{\{S(\mathbf{x})=0\}}(\mathbf{x})]$, which is jointly determined by test data $\mathcal{D}_{test}$, entropy threshold E_0, and cosine similarity threshold ϵ.

4.2 Anti-Forgetting with Fisher Regularization

In this section, we propose a new weighted Fisher regularizer (called anti-forgetting regularizer) to alleviate the catastrophic forgetting issue caused by test-time adaptation, *i.e.*, the performance of a test-time adapted model may significantly degrade on in-distribution (ID) test samples. We achieve this through weight regularization, which only affects the loss function and does not incur any additional computational overhead for model adaptation. To be specific, we apply an importance-aware regularizer $\mathcal{R}$ to prevent model parameters, important for the in-distribution domain, from changing too much during the test-time adaptation process [21]:

$$\mathcal{R}(\tilde{\Theta}, \tilde{\Theta}^o) = \sum_{\theta_i \in \tilde{\Theta}} \omega(\theta_i)(\theta_i - \theta_i^o)^2, \tag{7}$$

where $\tilde{\Theta}$ are parameters used for model update and $\tilde{\Theta}^o$ are the corresponding parameters of the original model. $\omega(\theta_i)$ denotes the importance of θ_i and we measure it via the diagonal Fisher information matrix as in elastic weight consolidation [21]. Here, the calculation of Fisher information $\omega(\theta_i)$ is non-trivial since we are inaccessible to any labeled training data. For the convenience of presentation, we leave the details of calculating $\omega(\theta_i)$ in the next subsection.

After introducing the anti-forgetting regularizer, the final optimization formula for EATA is formulated as:

$$\min_{\tilde{\Theta}} S(\mathbf{x})E(\mathbf{x};\Theta) + \beta\mathcal{R}(\tilde{\Theta}, \tilde{\Theta}^o), \tag{8}$$

where β is a trade-off parameter, $S(\mathbf{x})$ and $E(\mathbf{x};\Theta)$ are defined in Eqn. (2).

Measurement of Weight Importance $\omega(\theta_i)$. The calculation of Fisher information typically involves a set of labeled ID training samples. However, in our problem setting, we are inaccessible to training data and the test samples are only unlabeled, which makes it non-trivial to measure the weight importance. To conquer this, we first collect a small set of unlabeled ID test samples $\{\mathbf{x}_q\}_{q=1}^{Q}$, and then use the original trained model $f_\Theta(\cdot)$ to predict all these samples for obtaining the corresponding hard pseudo-label $\hat{y}_q$. Following that, we construct a pseudo-labeled ID test set $\mathcal{D}_F = \{\mathbf{x}_q, \hat{y}_q\}_{q=1}^{Q}$, based on which we calculate the fisher importance of model weights by:

$$\omega(\theta_i) = \frac{1}{Q}\sum_{\mathbf{x}_q \in \mathcal{D}_F} \left(\frac{\partial}{\partial\theta_i^o}\mathcal{L}_{CE}(f_{\Theta^o}(\mathbf{x}_q), \hat{y}_q)\right)^2, \tag{9}$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss. Here, we only need to calculate $\omega(\theta_i)$ once before performing test-time adaptation. Once calculated, we keep $\omega(\theta_i)$ fixed and apply it to any types of distribution shifts. Moreover, the unlabeled ID test samples are collected based on out-of-distribution detection techniques [53], [54], which are easy to implement. Note that there is no need to collect too many ID test samples for calculating $\omega(\theta_i)$, *e.g.*, 500 samples are enough for the ImageNet-C dataset. More empirical studies regarding this can be found in Figure 6.

4.3 Consistency-Based Uncertainty Minimization

As mentioned in Section 4.1, EATA conducts model adaptation by prediction entropy minimization. This strategy aims to reduce uncertainty in predictions and learn decision boundaries in low-density regions of the test samples [55], [56]. However, a persistent limitation of entropy minimization is its tendency to yield overly certain predictions, where the model is forced to make one-hot confident predictions even for ambiguous input data with irreducible data uncertainty. Thus, EATA may still result in overconfident predictions that do not accurately reflect the inherent data uncertainty. To address this, we further propose EATA with Calibration (EATA-C), shall be depicted in Sections 4.3 and 4.4.

Algorithm 1 The pipeline of proposed EATA and EATA-C.

Input: Test samples $\mathcal{D}_{test}$=$\{\mathbf{x}_j\}_{j=1}^{M}$, the trained model $f_\Theta(\cdot)$, ID samples $\mathcal{D}_F$=$\{\mathbf{x}_q\}_{q=1}^{Q}$, batch size B.

for a batch $\mathcal{X}$=$\{\mathbf{x}_b\}_{b=1}^{B}$ in $\mathcal{D}_{test}$ **do**
  Calculate predictions $\hat{y}$ for all $\mathbf{x} \in \mathcal{X}$ via $f_\Theta(\cdot)$.
  `For EATA:`
    Calculate sample selection score $S(\mathbf{x})$ via Eqn. (6).
    Update model ($\tilde{\Theta} \subseteq \Theta$) with Eqn. (8).
  `For EATA-C:`
    Select reliable and distinct samples $\mathcal{X}_s$ via Eqn. (15).
    Sample $f_{\Theta_{sub}}(\cdot)$ from $f_\Theta(\cdot)$ via stochastic depth [60].
    Calculate predictions $\hat{y}_{sub}$ for $\mathbf{x} \in \mathcal{X}_s$ via $f_{\Theta_{sub}}(\cdot)$.
    Compute consistency loss $\mathcal{L}_c(\mathbf{x})$ based on Eqn. (10)
    Calibrate confidence with entropy loss via Eqn. (12)
    Update model ($\tilde{\Theta} \subseteq \Theta$) with Eqn. (14).
end for

Output: The predictions $\{\hat{y}\}_{j=1}^{M}$ for all $\mathbf{x} \in \mathcal{D}_{test}$.

In EATA-C, we first propose a consistency-based loss to quantify and optimize the model uncertainty. Our method is inspired by MC Dropout [57], which has shown promising performance in estimating the model uncertainty through the divergence of multiple dropout-enabled predictions. In our context, considering adaptation efficiency, we define the model uncertainty as the KL divergence [58] between the full network prediction and randomly sampled sub-network prediction. Here, we use only two predictions and the former is indispensable since we select it as the final prediction. During TTA, we minimize this divergence to promote consistent predictions for model updates, rather than greedily increasing confidence which can result in overconfident outputs.

Consistency Loss. Formally, let $\hat{\mathbf{y}} = f_\Theta(\mathbf{x})$ be the prediction of the full network w.r.t. sample $\mathbf{x}$, and $\hat{\mathbf{y}}_{sub} = f_{\Theta_{sub}}(\mathbf{x})$ be that of the sub-network. The consistency loss is defined as follows:

$$\mathcal{L}_c(\mathbf{x}) = D_{KL}(\hat{\mathbf{y}}_{sub}, \hat{\mathbf{y}}_{fuse}), \tag{10}$$

$$\hat{\mathbf{y}}_{fuse} = (\hat{\mathbf{y}} + (1-p) \cdot \hat{\mathbf{y}}_{sub})/(2-p), \tag{11}$$

where $D_{KL}(\cdot||\cdot)$ denotes Kullback-Leibler divergence [58] and p is a constant for smoothing. Here, we calculate the divergence between $\hat{\mathbf{y}}_{sub}$ and $\hat{\mathbf{y}}_{fuse}$ (rather than $\hat{\mathbf{y}}$), since encouraging the sub-network to achieve the same performance as the full one is relatively hard. Thus, inspired by label smoothing [59], we softly fuse $\hat{\mathbf{y}}$ and $\hat{\mathbf{y}}_{sub}$ in Eqn. (11) for divergence optimization. Note that, during optimization, we conduct a unidirectional alignment from the sub-network to the full network, as the full network typically exhibits stronger generalization capabilities. To this end, we detach the gradient from $\hat{\mathbf{y}}$ and concentrate the optimization solely on $\hat{\mathbf{y}}_{sub}$. This strategy is designed to facilitate knowledge transfer from the full network to its sub-network, thereby enhancing the sub-network's performance while reducing the full networks' model uncertainty to adapt the network to the test domain.

Remark on Efficiency. Although consistency loss requires two forward passes from both the full network and the sub-network for each sample, the full network's forward pass is gradient-free without back-propagation and the sub-network's forward/backward pass is less computationally intensive. Moreover, we only perform sub-network's forward/backward passes on the selected reliable and non-redundant samples as outlined in Algorithm 1. As a result, the use of consistency loss remains efficient as shown in Table 7.

4.4 Calibrated Min-Max Entropy Regularization

In this section, we re-calibrate the model's prediction uncertainty in a manner that is sensitive to individual samples. This process involves categorizing samples into two distinct groups—'certain' and 'uncertain'—based on the aforementioned prediction consistency. This design is inspired by margin-based learning approaches [61], [62] which indicated that samples near decision boundaries are inherently more uncertain and have been well justified with theoretical guarantees. Specifically, we achieve categorization by comparing the predicted labels between the full network and a sub-network. Samples that exhibit mismatched predictions are deemed 'uncertain', suggesting their proximity to decision boundaries and high intrinsic data uncertainty. Note that unlike the consistency loss that measures model uncertainty, in which samples across the data space may yield low consistency loss, data uncertainty is reflected more prominently through prediction disagreements near the decision boundary (see Figure B for illustration). For identified uncertain samples, we aim to lower their predictive confidence by maximizing the prediction entropy, effectively acknowledging the model's lack of confidence in these cases. Conversely, for samples where predictions are consistent, labeled as 'certain', we conduct the opposite strategy, *i.e.*, boosting prediction confidence through entropy minimization. Formally, this min-max entropy regularization optimization problem is defined by:

$$\min_{\tilde{\Theta}_{sub}} C(\mathbf{x})E(\mathbf{x};\Theta_{sub}), \tag{12}$$

$$C(\mathbf{x}) = \begin{cases} 1, & \text{if } \arg\max(\hat{\mathbf{y}}) = \arg\max(\hat{\mathbf{y}}_{sub}), \\ -1, & \text{if } \arg\max(\hat{\mathbf{y}}) \neq \arg\max(\hat{\mathbf{y}}_{sub}), \end{cases} \tag{13}$$

where $\hat{\mathbf{y}}$ and $\hat{\mathbf{y}}_{sub}$ denote the prediction of the full and sub-network respectively, Θ_{sub} denotes the parameters of the sub-network and $\tilde{\Theta}_{sub} \subset \Theta_{sub}$ denotes parameters involved in model adaptation. Note that we only update the affine parameters of the sub-network considering efficiency as mentioned in Section 4.3.

Overall Objective. The methods proposed in Sections 4.3 and 4.4 are devised to address the overconfident issue in TTA, but still suffer from catastrophic forgetting when important model weights for the in-distribution domain are significantly modified during adaptation. Therefore, we jointly optimize the model with the anti-forgetting regularizer and further reduce the required backward computations with the active sample selection criterion in EATA. Then, the overall objective of EATA-C can be formulated as:

$$\min_{\tilde{\Theta}} S_c(\mathbf{x})\Big(\mathcal{L}_c(\mathbf{x}) + \alpha C(\mathbf{x})E(\mathbf{x};\Theta_{sub})\Big) + \beta\mathcal{R}(\tilde{\Theta}, \tilde{\Theta}^o). \tag{14}$$

where α and β are balance factors, $\mathcal{R}(\tilde{\Theta}, \tilde{\Theta}^o)$ is the fisher regularizer defined in Eqn. (7), and $S_c(\mathbf{x})$ is the joint indicator function in Eqn. (3) and Eqn. (5) to select reliable and non-redundant samples. To be specific, $S_c(\mathbf{x})$ is defined by:

$$S_c(\mathbf{x}) = S^{div}(\mathbf{x}) \cdot \mathbb{I}_{\{E(\mathbf{x};\Theta)<E_0\}}(\mathbf{x}). \tag{15}$$

We summarize the overall pipeline of our proposed EATA-C and EATA in Algorithm 1.

5 Experiments

Datasets and Models. We conduct experiments on three benchmark datasets for OOD generalization: ImageNet-C [9] (contains corrupted images in 15 types of 4 main categories and each type has 5 severity levels) and ImageNet-R [66] for image classification;

TABLE 2
Comparison with state-of-the-art methods on ImageNet-C with the highest severity level 5 regarding **Corruption Accuracy(%, ↑)** and **Expected Calibration Error(%,↓)**. "BN" and "LN" denote batch and layer normalization, respectively. The **bold** number indicates the best result and the underlined number indicates the second best result. All results are evaluated under the **lifelong adaptation scenario** except for Tent [12] and MEMO [14], which suffer severely from error accumulation. We use * and † to denote episodic and single-domain adaptation, respectively.

			Noise			Blur				Weather				Digital				Average		
Model	Method	Metric	Gauss.	Shot	Impul.	Defoc.	Glass	Motion	Zoom	Snow	Frost	Fog	Brit.	Contr.	Elastic	Pixel	JPEG	Avg.	#Forwards	#Backards
R-50 (BN)	Source	Acc.	1.8	3.0	1.7	18.2	10.1	13.4	20.8	14.0	22.1	21.9	58.7	5.3	17.6	22.1	37.5	17.88	50,000	0
		ECE	16.6	16.1	15.9	1.8	10.7	10.7	14.7	25.3	12.9	16.7	2.3	6.7	22.9	10.5	6.0	12.64		
	BN Adapt	Acc.	15.8	16.7	15.3	18.7	19.3	29.8	41.7	35.8	35.0	50.5	65.9	18.1	49.3	51.7	42.0	33.70	50,000	0
		ECE	1.1	0.8	1.0	3.0	1.3	0.8	3.4	1.1	1.0	5.4	1.4	7.6	4.3	3.8	4.8	**2.72**		
	Tent†	Acc.	28.0	30.1	28.1	29.9	29.5	42.2	49.7	46.2	41.5	57.7	67.1	30.0	55.7	58.3	52.5	43.11	50,000	50,000
		ECE	11.7	11.2	11.1	12.6	12.3	7.7	5.4	6.5	8.8	3.4	2.9	21.9	3.5	3.7	4.0	8.46		
	MEMO*	Acc.	6.8	8.5	7.5	20.5	13.4	19.8	25.8	22.1	27.7	27.6	60.9	11.3	24.4	32.2	37.9	23.09	50,000×65	50,000×64
		ECE	24.1	24.2	22.9	5.3	19.3	14.8	23.4	30.4	18.7	24.6	7.2	14.9	29.4	19.3	13.6	19.47		
	CoTTA	Acc.	19.9	31.8	35.3	30.4	34.4	40.2	43.2	39.3	38.6	47.7	51.8	36.0	43.5	46.7	43.1	38.80	152,315	50,000
		ECE	3.6	17.4	21.5	27.8	30.4	31.2	31.6	33.2	34.2	32.5	33.1	36.0	34.2	36.1	35.9	29.25		
	SAR	Acc.	29.6	38.4	37.8	31.5	32.8	41.4	48.6	42.9	40.2	53.3	63.7	37.7	53.0	56.3	52.3	43.96	85,964	68,145
		ECE	3.7	7.4	9.1	14.4	15.1	12.6	9.6	11.7	13.3	8.4	7.0	16.8	8.5	8.4	8.5	10.29		
	ROID	Acc.	36.7	38.6	35.9	29.1	28.8	40.6	46.5	49.5	41.8	55.6	65.3	43.5	53.6	56.1	52.8	44.95	80,380	80,380
		ECE	36.6	38.5	35.8	29.0	28.7	40.5	46.4	49.4	41.7	55.5	65.2	43.4	53.5	56.0	52.7	44.85		
	RDump	Acc.	35.6	34.4	36.1	32.7	34.6	45.5	51.6	50.3	44.1	59.9	66.8	46.3	56.8	59.1	54.7	47.23	50,000	26,375
		ECE	10.5	11.4	12.6	13.9	14.4	9.4	8.1	8.1	9.4	6.3	5.5	11.8	6.1	7.0	6.3	9.40		
	TEA	Acc.	18.5	0.2	0.1	0.1	0.2	0.2	0.1	0.1	0.1	0.1	0.2	0.1	0.2	0.1	0.1	1.36	50,000×23	50,000×22
		ECE	7.9	5.4	2.4	2.0	1.8	1.9	3.8	19.5	67.6	76.4	86.2	91.0	90.4	91.3	91.3	42.60		
	EATA (Ours)	Acc.	35.6	38.7	37.5	35.9	36.1	47.6	53.1	50.6	45.6	59.5	67.1	45.2	57.6	59.8	55.4	48.36	50,000	29,721
		ECE	10.5	13.4	14.7	18.5	18.5	14.7	12.8	14.1	16.1	11.6	10.4	18.6	13.3	13.1	13.9	14.28		
	EATA-C (Ours)	Acc.	37.2	40.9	39.9	36.6	37.1	48.5	52.7	51.9	47.2	60.4	67.0	49.0	57.6	60.1	56.1	**49.48**	83,312	33,312
		ECE	7.1	6.7	7.2	10.1	9.5	5.9	4.9	4.8	5.8	3.2	3.2	6.0	4.0	3.9	3.9	5.74		
ViT (LN)	Source	Acc.	12.9	17.6	11.7	34.4	27.7	43.7	36.2	43.4	45.4	52.8	73.3	45.5	37.9	54.7	60.2	39.84	50,000	0
		ECE	14.2	11.3	15.2	2.2	8.2	6.0	10.7	7.5	8.3	5.0	2.1	2.3	12.2	4.1	2.9	7.48		
	Tent†	Acc.	33.4	42.1	41.4	48.8	45.2	54.9	48.2	55.6	55.1	64.4	75.2	62.5	51.6	65.5	65.0	53.92	50,000	50,000
		ECE	24.1	10.1	11.7	8.6	10.4	7.5	12.4	8.0	8.0	4.8	2.4	5.2	10.0	4.3	4.1	8.78		
	MEMO*	Acc.	32.2	35.1	32.6	37.5	28.5	43.3	40.1	45.2	47.0	53.9	73.3	53.3	39.6	59.5	62.8	45.60	50,000×65	50,000×64
		ECE	33.0	32.5	33.2	36.7	45.7	41.0	47.2	39.8	38.8	33.7	20.0	26.7	48.5	30.4	27.2	35.63		
	CoTTA	Acc.	45.6	58.8	58.4	40.1	50.9	50.4	44.4	46.1	51.4	52.2	57.7	36.5	53.9	55.8	55.4	50.51	50,000×3	50,000
		ECE	9.0	15.8	21.0	29.3	28.3	29.1	34.5	29.8	29.6	30.0	29.2	36.3	29.0	31.2	31.8	27.59		
	SAR	Acc.	43.1	50.8	52.9	50.4	51.0	57.5	53.2	58.6	61.2	66.0	76.1	61.2	54.7	67.8	67.7	58.15	91,605	82,277
		ECE	8.0	8.4	9.5	9.1	11.4	9.6	13.0	9.9	9.6	7.8	4.5	10.0	13.8	7.5	7.5	9.30		
	ROID	Acc.	48.8	49.5	49.0	54.1	54.2	58.8	55.7	62.7	61.3	69.7	77.0	65.5	64.1	69.6	68.3	60.56	80,739	80,739
		ECE	48.7	49.4	48.9	54.0	54.1	58.7	55.6	62.6	61.2	69.6	76.9	65.4	64.0	69.5	68.2	60.46		
	RDump	Acc.	50.5	48.4	51.8	54.0	55.3	59.4	55.9	63.3	60.7	70.8	76.8	66.8	60.9	69.5	68.4	60.84	50,000	32,050
		ECE	10.6	9.7	10.5	8.9	9.4	7.9	10.0	6.9	7.4	5.4	3.2	6.7	7.6	5.1	5.0	7.62		
	TEA	Acc.	46.9	48.0	48.1	46.7	47.0	53.9	53.5	57.4	56.0	62.2	71.7	55.4	57.1	63.1	61.7	55.24	50,000×23	50,000×22
		ECE	10.6	13.7	13.7	14.5	14.4	11.3	11.6	9.4	10.2	7.8	3.9	10.5	9.8	7.3	7.4	10.40		
	EATA (Ours)	Acc.	50.5	55.6	56.0	54.9	56.3	61.1	59.8	64.3	64.0	70.1	77.4	65.5	63.1	70.4	69.5	62.57	50,000	36,688
		ECE	10.6	13.9	15.7	16.4	17.3	15.9	17.5	14.8	15.7	12.8	9.0	15.4	17.5	13.3	13.9	14.63		
	EATA-C (Ours)	Acc.	56.8	60.2	59.8	58.0	60.9	65.2	65.5	69.7	67.8	74.0	78.9	66.7	70.3	74.1	71.8	**66.65**	83,184	33,184
		ECE	5.2	5.1	5.9	7.1	6.2	5.6	5.9	5.0	5.0	4.1	3.3	5.4	4.7	4.2	4.7	**5.14**		

TABLE 3
Comparison on ImageNet-R. Results are evaluated in the single-domain adaptation scenario. We use * to denote episodic adaptation.

Model	Acc. (%)	ECE (%)	#Forwards	#Backwards
ResNet-50(BN)	38.0	17.7	30,000	0
• BN [29]	$40.4_{(+2.4)}$	$13.4_{(-4.3)}$	30,000	0
• Tent [12]	$42.3_{(+4.3)}$	$17.8_{(+0.1)}$	30,000	30,000
• MEMO* [14]	$41.9_{(+3.9)}$	$26.9_{(+9.2)}$	30,000×65	30,000×64
• CoTTA [63]	$42.4_{(+4.4)}$	$15.8_{(-1.9)}$	90,000	30,000
• SAR [18]	$42.7_{(+4.7)}$	$14.6_{(-3.1)}$	47,755	32,877
• ROID [64]	$\mathbf{48.6}_{(+10.5)}$	$48.1_{(+30.4)}$	48,303	48,303
• TEA [65]	$42.8_{(+4.8)}$	$14.2_{(-3.5)}$	30,000×23	30,000×22
• EATA (Ours)	$44.9_{(+6.9)}$	$16.7_{(-1.0)}$	30,000	5,417
• EATA-C (Ours)	$47.1_{(+9.1)}$	$\mathbf{13.3}_{(-4.4)}$	35,122	5,122
ViT(LN)	52.5	5.0	30,000	0
• Tent [12]	$54.2_{(+1.7)}$	$7.4_{(+2.4)}$	30,000	30,000
• MEMO* [14]	$57.5_{(+5.0)}$	$32.1_{(+27.1)}$	30,000×65	30,000×64
• CoTTA [63]	$56.4_{(+3.9)}$	$7.4_{(+2.4)}$	90,000	30,000
• SAR [18]	$55.0_{(+2.5)}$	$5.2_{(+0.2)}$	47,119	33,844
• ROID [64]	$62.2_{(+9.7)}$	$61.7_{(+56.7)}$	49,795	49,795
• TEA [65]	$60.1_{(+7.6)}$	$7.4_{(+2.4)}$	30,000×23	30,000×22
• EATA (Ours)	$58.2_{(+5.7)}$	$5.8_{(+0.8)}$	30,000	6,053
• EATA-C (Ours)	$\mathbf{64.2}_{(+11.7)}$	$\mathbf{3.9}_{(-1.1)}$	36,395	6,395

and ACDC [67] for semantic segmentation. We use ResNet-50 (R-50) [1] and ViT-Base (ViT) [25] for ImageNet experiments, and Segformer-B5 [68] for ACDC [67] experiments. The models are trained on ImageNet or CityScapes [69] training set with stochastic depth regularization [60] and tested on clean or OOD test sets.

Compared Methods. We compare with the following state-of-the-art methods. BN adaptation [29] updates batch normalization statistics on test samples. Tent [12] minimizes the entropy of test samples during testing. MEMO [14] maximizes the prediction consistency of different augmented copies regarding a given test sample. SAR [18] selects reliable samples for test time sharpness-aware entropy minimization. CoTTA [63] and DAT [70] minimize the cross entropy between the student network and its mean teacher during testing. RDump [71] periodically resets model parameters based on our EATA. TEA [65] employs energy-based contrastive learning with negative sample generation. ROID [64] minimizes the diversity-weighted soft likelihood ratio loss. We denote EATA without weight regularization in Eqn. (7) as **e**fficient **t**est-time **a**daptation **(ETA)**. More ablative methods can be found in Table 6.

Adaptation Scenarios. We conduct experiments under three adaptation scenarios: 1) *Episodic*, the model parameters will be reset immediately after each optimization step of a test sample or batch; 2) *Single-domain*, the model is online adapted through the entire evaluation of one given test dataset (*e.g.*, gaussian noise level 5 of ImageNet-C). Once the adaptation on this dataset is finished, the model parameters will be reset; 3) *Lifelong*, the model

is online adapted and the parameters will never be reset (as shown in Figure 3 (**Right**)), which is more challenging but practical.

Evaluation Metrics. 1) Clean accuracy/error (%) on original in-distribution (ID) test samples, *e.g.*, the original test images of ImageNet. Note that we measure the clean accuracy of all methods via **(re)adaptation**; 2) Out-of-distribution (OOD) accuracy/error (%) on OOD test samples, *e.g.*, the corruption accuracy on ImageNet-C; 3) Expected Calibration Error (ECE) [72] on OOD test samples, which measures the average discrepancies between model's confidence and accuracy within multiple confidence intervals; 4) The number of forward and backward passes during the entire TTA process. Note that the fewer #forwards and #backwards indicate less computation, leading to higher efficiency.

Implementation Details. For test time adaptation, we use SGD as the update rule, with a momentum of 0.9 and a batch size of 64. In EATA and ETA, the learning rate is set to 0.00025/0.001 for ResNet-50/ViT-Base on ImageNet, and 7.5×10^{-5} on ACDC, respectively (following Tent, SAR and CoTTA). In EATA-C, the learning rate is set to 0.005/0.1 for ResNet-50/ViT-Base on ImageNet, and 0.0005 on ACDC, respectively. The sub-network is obtained via stochastic depth regularization [60] with a drop ratio of 0.2/0.6 for ImageNet/ACDC. For both EATA and EATA-C, we use 2,000/20 samples for ImageNet/ACDC to calculate $\omega(\theta_i)$ in Eqn. (9).The effect and sensitivity of each hyperparameter is investigated in Section 5.3. More details are put in Supplementary.

5.1 Comparisons w.r.t. OOD Performance, Efficiency and Calibration Error

Results on ImageNet-C. From Table 2, our EATA and EATA-C consistently surpass existing approaches regarding adaptation accuracy, *e.g.*, the average accuracy of 48.4% (EATA) *vs.* 45.0% (ROID) on ResNet-50. Importantly, EATA yields a remarkable performance gain over its counterpart Tent, *e.g.*, $33.4\% \rightarrow 50.5\%$ on Gaussian Noise with ViT-Base, suggesting the significance of removing samples with unreliable gradients and tackling samples differently in the TTA process. Our enhanced method, EATA-C, further boosts adaptation accuracy by a large margin, which consistently outperforms TEA and ROID in all 15 corruption types over both ResNet-50 and ViT-Base, suggesting our effectiveness. Note that besides achieving strong OOD performance, EATA also alleviates the forgetting on ID samples (see Figure 3), showing the effectiveness of our anti-forgetting regularization without limiting the learning ability during adaptation (see also Table 6 for ablation).

In terms of computational efficiency, EATA requires only 29,721 backward passes on ResNet-50, which is much fewer than methods that require extensive data augmentations (*i.e.*, TEA at 50,000×22) or multiple optimization iterations (*e.g.*, SAR at 68,145 on ResNet-50). Compared with Tent (*e.g.*, 50,000 backward passes), EATA saves computation by excluding samples with high prediction entropy and redundant samples out of test-time optimization, resulting in higher efficiency. While our EATA-C uses additional forward passes, its forward passes with the full network are gradient-free, and the lightweight sub-network forward/backward passes are conducted only on the selected samples, maintaining overall computational efficiency comparable to EATA (see Table 7 for detailed results and discussions on wall-clock time and memory usage). Although optimization-free methods (such as BN adaptation) do not need backward updates, their applicability scope and OOD performances are limited.

Regarding calibration, existing methods consistently exhibit substantial calibration error (*e.g.*, ROID and CoTTA are 60.46% and 27.59% on ECE with ViT-Base), suggesting miscalibration as a prevalent issue in the unsupervised test-time adaptation. By filtering unreliable samples to reduce noisy learning signals, EATA improves calibration over Tent (*e.g.*, $11.7\% \rightarrow 10.5\%$ on Gaussian Noise with ResNet-50), though miscalibration is yet significant. By further decreasing reducible model uncertainty and reflecting data uncertainty in model predictions, our enhanced method, EATA-C, reduces the ECE of EATA by relatively 59.8% on ResNet-50 and 64.9% on ViT-Base, demonstrating the strong calibration effect of EATA-C across diverse datasets and architectures. In summary, while EATA-C improves performance and efficiency over the state of the art, our EATA-C further achieves high accuracy, well-calibrated prediction, and efficient computation simultaneously, establishing a new benchmark for test-time adaptation.

Results on ImageNet-R. From Table 3, EATA consistently achieves a favorable balance between performance and efficiency, significantly improving accuracy on both ResNet-50 and ViT-Base while requiring much fewer backpropagation steps. For instance, EATA improves accuracy from 42.8% (TEA) to 44.9% on ResNet-50, while reducing the backpropagation steps from 30,000×22 to 5,417. EATA-C further improves accuracy substantially (*e.g.*, by 6.0% over EATA on ViT-Base), while maintaining computational efficiency comparable to EATA. Importantly, EATA-C is the only method that reduces calibration error on both ResNet-50 and ViT-Base and uniquely lowers ECE on ViT-Base, suggesting the effectiveness of our calibration-driven objective in TTA.

Results on CityScapes-to-ACDC. Following [63], we evaluate our method on the semantic segmentation task in a lifelong adaptation scenario. From Table 4, while DAT initially achieves higher mIOU, it tends to overfit, leading to significant performance degradation in subsequent adaptations. In contrast, our EATA maintains a more stable performance compared to DAT and Tent, by filtering unreliable predictions and preventing drastic changes to important model parameters. Moreover, by replacing entropy minimization with our consistency maximization objective for more robust learning signals, EATA-C achieves state-of-the-art performance, surpassing EATA by 4.6% and CoTTA by 3.2% in mIoU over ten adaptation rounds. More critically, our EATA-C showcases consistent improvement over lifelong adaptation, increasing the average mIOU on four datasets from 59.8% (first round) to 62.3% (tenth round), further highlighting our long-term effectiveness.

5.2 Demonstration of Preventing Forgetting

In this section, we investigate the ability of our EATA in preventing ID forgetting during test-time adaptation. The experiments are conducted on ImageNet-C with ResNet-50. We measure the anti-forgetting ability by comparing the model's clean accuracy (*i.e.*, on original validation data of ImageNet) before and after adaptation. To this end, we first perform test-time adaptation on a given OOD test set, and then evaluate the clean accuracy of the updated model. Here, we consider two adaptation scenarios: the single-domain adaptation, and the lifelong adaptation. We report the results of severity level 5 in Figure 3 and put the results of severity levels 1-4 into Supplementary.

From Figure 3, our EATA consistently outperforms Tent regarding the OOD corruption accuracy and meanwhile maintains the clean accuracy on ID data (in both two adaptation scenarios), demonstrating our effectiveness. It is worth noting that the performance degradation in lifelong adaptation scenario is much more severe (see Figure 3 **Right**). More critically, in lifelong adaptation,

TABLE 4
Semantic segmentation results (mIoU in %) on the Cityscapes-to-ACDC lifelong test-time adaptation scenario. The model is continually adapted to the four adverse conditions for ten rounds without model reset. All results are evaluated based on the Segformer-B5 architecture. Following [63], we only show the results of the first, fourth, seventh, and last rounds due to page limits. Full results can be found in the supplementary material.

Round	1				4				7				10				All
Condition	Fog	Night	Rain	Snow	Fog	Night	Rain	Snow	Fog	Night	Rain	Snow	Fog	Night	Rain	Snow	Mean
Source	69.1	40.3	59.7	57.8	69.1	40.3	59.7	57.8	69.1	40.3	59.7	57.8	69.1	40.3	59.7	57.8	56.7
BN Stats Adapt	62.3	38.0	54.6	53.0	62.3	38.0	54.6	53.0	62.3	38.0	54.6	53.0	62.3	38.0	54.6	53.0	$52.0_{(-4.7)}$
Tent (lifelong)	69.0	40.2	60.0	57.3	66.6	36.6	58.9	54.2	64.6	33.4	55.9	51.6	62.5	30.4	52.6	48.7	$52.7_{(-4.0)}$
CoTTA	70.9	41.1	62.4	59.7	70.8	40.6	62.6	59.7	70.8	40.5	62.6	59.7	70.8	40.5	62.6	59.7	$58.4_{(+1.7)}$
DAT	71.7	44.6	63.8	62.2	68.0	42.0	60.9	59.4	66.1	40.6	59.8	57.8	63.8	39.6	58.2	55.4	$57.0_{(+0.3)}$
EATA	69.1	40.5	59.8	58.1	69.3	41.8	60.1	58.6	68.8	42.5	59.4	57.9	67.9	42.8	57.7	56.3	$57.0_{(+0.3)}$
EATA-C	71.0	44.3	63.1	61.1	72.0	47.3	64.9	63.8	71.8	48.2	64.2	64.2	72.0	48.7	64.3	64.1	**61.6**$_{(+4.9)}$

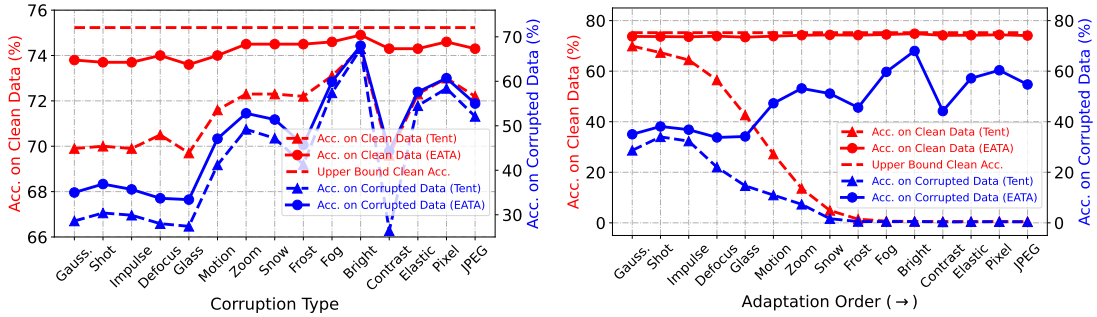


Fig. 3. Comparison of preventing forgetting on ImageNet-C (severity level 5) with ResNet-50. We record the OOD corruption accuracy on each corrupted test set and the ID clean accuracy (after OOD adaptation). In **Left**, the model parameters of both Tent and our EATA are reset before adapting to a new corruption type. In **Right**, the model performs lifelong adaptation and the parameters will never be reset, namely Tent (lifelong) and our EATA (lifelong). The upper bound clean accuracy is estimated with the source model without adaptation on corrupted OOD data, which does not suffer from forgetting. EATA achieves higher OOD accuracy and meanwhile maintains the ID clean accuracy.

both the clean and corruption accuracy of Tent decreases rapidly (until degrades to 0%) after adaptation of the first three corruption types, showing that Tent in lifelong adaptation is not stable enough. In contrast, during the whole lifelong adaptation process, our EATA achieves good corruption accuracy and the clean accuracy is also very close to the clean accuracy of the model without any OOD adaptation (*i.e.*, original clean accuracy, tested using Tent). These results demonstrate the superiority of our anti-forgetting Fisher regularizer in terms of overcoming the forgetting on ID data.

5.3 Ablation Studies

Effect of Components in $S(\mathbf{x})$ (Eqn. 6). Our EATA accelerates test-time adaptation by excluding two types of samples out of optimization: 1) samples with high prediction entropy values (Eqn. 3) and 2) samples that are similar (Eqn. 6). We ablate both of them in Table 5. Compared with the baseline $S(\mathbf{x})=1$ (the same as Tent), introducing $S^{ent}(\mathbf{x})$ in Eqn. (3) achieves better accuracy and fewer backwards (*e.g.*, 49.6% (37,636) *vs.* 33.4% (50,000) on level 5). This verifies our motivation in Figure 2 that some high-entropy samples may hurt the performance since their gradients are unreliable. When further removing some redundant samples that are similar (Eqn. 6), our EATA further reduces the number of back-propagation (*e.g.*, 37,636→28,168 on level 5) and achieves comparable OOD error (*e.g.*, 50.4% *vs.* 49.6%), demonstrating the effectiveness of our sample-efficient optimization strategy.

Effect of Components in EATA-C. Our EATA-C aims to achieve a favorable balance between accuracy, calibration, and efficiency. We conduct an ablation study to verify the effectiveness of each module as in Table 6. The results indicate the following findings: 1) *Consistency Loss*: Incorporating the consistency loss alone substantially enhances the source model's robustness and reduces ECE; 2) *Entropy Regularization*: The min-max entropy regularizer further calibrates prediction confidence and leads to a slight improvement in accuracy, *e.g.*, accuracy increases from 48.8% (Exp 10) to 49.0% (EATA-C), and ECE decreases from 5.1% to 4.6%; 3) *Fisher Regularization*: This anti-forgetting regularizer contributes to TTA stability, as in the lifelong TTA of Table 2 and Figure 3. In single-domain TTA, it also positively affects both ECE and accuracy, *e.g.*, ECE decreases from 5.4% (ETA-C) to 4.6% (EATA-C) and accuracy improves from 48.9% to 49.0%; 4) *Active Sample Selection*: By filtering out unreliable and redundant test samples, active sample selection significantly boosts computational efficiency while maintaining or improving accuracy, *e.g.*, accuracy increases from 48.5% (Exp 6) to 49.0% (EATA-C) while reducing the required backward passes by 35%. More discussions on wall-clock time and memory usage are provided in Table 7. These results collectively underscore the effectiveness of each component.

TABLE 5
Effectiveness of components in sample-adaptive weight $S(\mathbf{x})$ in EATA on ImageNet-C (Gaussian noise) with ResNet-50.

Method	Level 3		Level 5	
	Acc. (%)	#Backwards	Acc. (%)	#Backwards
Baseline ($S(\mathbf{x})$=1)	68.8	50,000	33.4	50,000
+$S^{ent}(\mathbf{x})$ (Eqn. 3)	70.7	45,302	49.6	37,636
+$S(\mathbf{x})$ (Eqn. 6)	**70.8**	**36,057**	**50.4**	**28,168**

Entropy Constant E_0 in Eqn. (3). We evaluate our EATA with different E_0, selected from $\{0.20, 0.25, 0.30, 0.35, 0.40, 0.45, 0.50, 0.55\}\times \ln 10^3$, where 10^3 is the class number of ImageNet. From Figure 4, our EATA achieves excellent performance when E_0

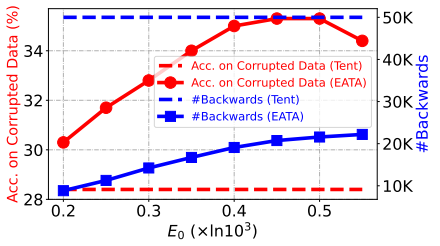


Fig. 4. Effect of different entropy margins E_0 in Eqn. (3). Results obtained on ImageNet-C(Gaussian, level 5) with ResNet-50.

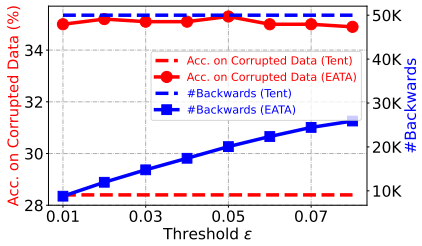


Fig. 5. Effect of different similarity threshold ϵ in Eqn.(5). Results obtained on ImageNet-C(Gaussian, level 5) with ResNet-50.

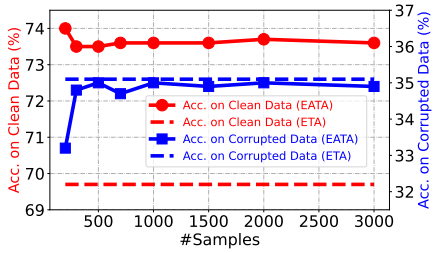


Fig. 6. Effect of #samples for calculating Fisher in Eqn. (9). Results obtained on ImageNet-C(Gaussian, level 5) with ResNet-50.

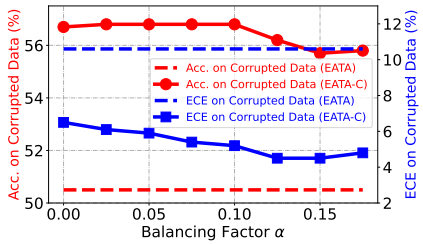


Fig. 7. Effect of different balancing factor α in Eqn. (8). Results obtained on ImageNet-C(Gaussian, level 5) with ViT-Base.

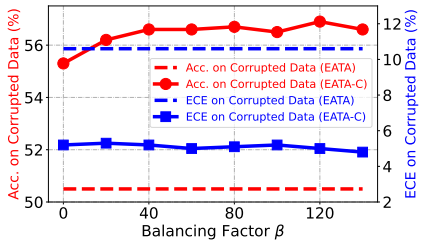


Fig. 8. Effect of different balancing factor β in Eqn. (8). Results obtained on ImageNet-C(Gaussian, level 5) with ViT-Base.

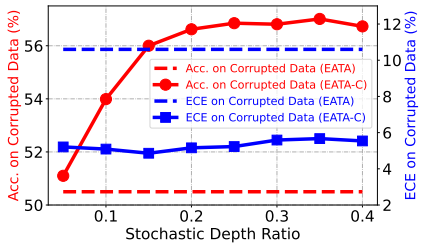


Fig. 9. Effect of different stochastic depth ratios in EATA-C. Results obtained on ImageNet-C(Gaussian, level 5) with ViT-Base.

TABLE 6
Effects of components in EATA-C. Results obtained on 15 corruptions of ImageNet-C (level 5) with ResNet-50 in single-domain TTA scenario, *i.e.*, **the model parameters are reset before adapting to a new domain**. **CL** denotes consistency loss. **ER** denotes min-max entropy regularizer. **FR** denotes Fisher regularizer. **SS** denotes active sample selection.

Experiment	CL	ER	FR	SS	Average Acc.	ECE	#Forwards	#Backwards
Source					39.8	7.5	50,000	0
1	✓				47.7	4.3	50,000×2	50,000
2		✓			44.0	5.2	50,000×2	50,000
3	✓	✓			48.6	4.3	50,000×2	50,000
4	✓		✓		47.5	3.7	50,000×2	50,000
5		✓	✓		43.4	4.2	50,000×2	50,000
6	✓	✓	✓		48.5	3.8	50,000×2	50,000
7	✓			✓	48.9	5.9	83,583	33,583
8		✓		✓	42.9	4.4	74,705	24,705
9 (ETA-C)	✓	✓		✓	48.9	5.4	82,882	32,882
10	✓		✓	✓	48.8	5.1	82,952	32,952
11		✓	✓	✓	42.3	3.8	74,226	24,226
12 (EATA-C)	✓	✓	✓	✓	49.0	4.6	82,492	32,492

belongs to $[0.4, 0.5]$. Either a smaller or larger E_0 would hamper the performance. The reasons are mainly as follows. When E_0 is small, EATA removes too many samples during adaptation and thus is unable to learn enough knowledge from the remaining samples. When E_0 is too large, some high-entropy samples would take part but contribute unreliable and harmful gradients, resulting in performance degradation. As larger E_0 leads to more backward passes, we set E_0 to $0.4\times \ln 10^3$ for efficiency-performance trade-off and fix the proportion of 0.4 for all other ImageNet experiments.

Similarity Threshold ϵ in Eqn. 5. We use ϵ to select diverse samples for TTA. From Figure 5, EATA maintains stable accuracy across a wide range of $\epsilon \in [0.01, 0.08]$, showcasing insensitivity, while a smaller ϵ removes significantly more samples and improves computational efficiency. We set ϵ=0.05 without careful tuning. More results on efficiency (*i.e.*, time and memory usage) of EATA and EATA-C with varying ϵ are provided in Table 7.

Number of Samples for Calculating Fisher in Eqn. (9). As described in Section 4.2, the calculation of Fisher information involves a small set of unlabeled ID samples, which can be collected via existing OOD detection techniques [54]. Here, we investigate the effect of #samples Q, selected from {200, 300, 500, 700, 1000, 1500, 2000, 3000}. From Figure 6, our EATA achieves stable performance with $Q \geq 300$, *i.e.*, compared with ETA, the OOD performance is comparable and the clean accuracy is much higher. These results show that our EATA does not need to collect too many ID samples, which are easy to obtain in practice.

Factor α for Entropy Regularizer in Eqn. 14. We directly set α=0.1 to align the magnitudes of consistency loss and entropy regularization loss for EATA-C without careful tuning. From Figure 7, increasing α within $[0, 0.1]$ effectively reduces more ECE while maintaining stable accuracy, verifying its efficacy. However, when α exceeds 0.1, the entropy regularization loss dominates the adaptation, which leads to a gradual decline in accuracy.

Factor β for Fisher Regularizer in Eqn. 14. From Figure 8, compared to ETA-C (*i.e.*, setting β=0), introducing the fisher regularizer consistently achieves better accuracy. Moreover, once activated, the performance of EATA becomes largely insensitive to β within the tested range of $[20, 140]$, highlighting its robustness.

Stochastic Depth Ratio for Obtaining Sub-Network. In Eqn. (10), We generate an extra prediction from the sub-network to measure model uncertainty, where the sub-network is obtained via stochastic depth [60] throughout the experiments. We evaluate the effect of stochastic depth ratio selected from {0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35 ,0.4}. As shown in Figure 9, our EATA-C achieves a satisfying performance-calibration trade-off when the ratio belongs to $[0.15, 0.25]$, where the full network consistently outperforms the sub-network while the sub-network retains sufficient capacity for learning. We fix the ratio to 0.2 for all other ImageNet experiments.

5.4 More Discussions

Efficiency Analysis of EATA and EATA-C. We evaluate the efficiency of our methods by including a more comprehensive comparison of time and memory usage for TTA, as in Table 7. The

TABLE 7
Efficiency comparison regarding wall-clock time and peak memory usage on ImageNet-C (Gaussian, severity level 5) with an A100 GPU.

Method	ResNet-50			ViT-Base		
	Acc. (%)	Time (s)	Mem. (MB)	Acc. (%)	Time (s)	Mem. (MB)
Source	1.8	54.6	771.7	12.9	55.7	816.6
Tent [12]	28.0	99.9	5417.6	33.4	106.1	7433.2
SAR [18]	29.6	149.0	5417.7	43.1	167.3	7433.2
ROID [30]	36.7	190.2	9315.3	48.8	181.6	12321.4
CoTTA [16]	19.9	241.0	12196.7	45.6	229.9	22296.5
TEA [65]	18.5	2266.1	15942.0	46.9	-	-
MEMO [14]	6.8	36329.2	8154.3	32.2	39918.8	11061.1
EATA (ϵ=0.02)	35.2	**93.0**	2285.2	50.7	**85.1**	4234.6
EATA-C (ϵ=0.02)	35.9	95.2	**2205.5**	55.0	88.8	**4014.4**
EATA (Ours)	35.6	106.5	3693.0	50.5	108.7	5887.6
EATA-C (Ours)	**37.2**	122.4	3358.7	**56.8**	114.9	5786.6

TABLE 8
Reliability of data uncertainty indicator. We report sub-model Acc. (%) on ImageNet-C(Gaussian, level 5) after adapting to B batches (batch size 64) with ViT-Base. "#Uncertain" are samples with disagreed predictions. "Indicator Acc." is the ratio of these samples misclassified by sub-model.

Metric	Source	EATA-C (B=150)	EATA-C (B=300)	EATA-C (B=450)	EATA-C (B=600)	EATA-C (B=750)
Model Acc. (%)	9.3	47.0	50.2	50.7	51.9	52.0
#Uncertain	26144	15697	14682	14149	13648	13722
Indicator Acc. (%)	97.1	90.0	88.9	89.3	88.6	89.1

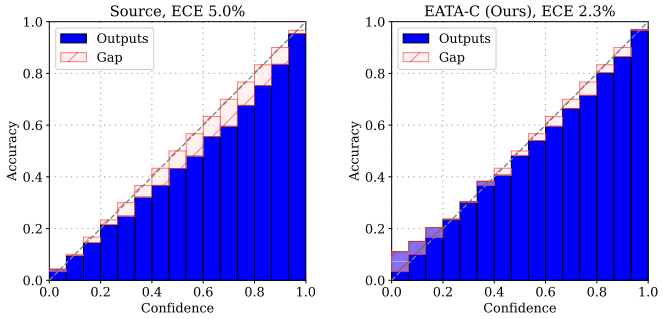


Fig. 10. Calibration comparison of each confidence interval on ImageNet-C(Fog, level 5) with ViT-Base. Results are visualized following [44].

TABLE 9
Comparison with Tent [12] w.r.t. corruption accuracy (%) with mixture of 15 corruption types on ImageNet-C with ViT-Base.

Severity	Source	Tent	EATA (ours)	EATA-C (ours)
Level=3	63.0	69.8$_{(+6.8)}$	71.1$_{(+8.1)}$	**72.4**$_{(+9.4)}$
Level=5	39.8	47.0$_{(+7.2)}$	58.2$_{(+18.4)}$	**60.4**$_{(+20.6)}$

results reveal the following: 1) ***Adaptation Time***: EATA and EATA-C require significantly less adaptation time than most baseline methods, while maintaining competitive or superior accuracy. For example, on ViT-Base, EATA-C improves accuracy from 48.8% (ROID) to 56.8% with a reduction in adaptation time from 181.6 seconds to 114.9 seconds. Moreover, by setting a stricter threshold ϵ to filter redundant test samples, EATA and EATA-C can be further accelerated while maintaining performance. For example, using ϵ=0.02, EATA on ViT-Base increases accuracy from 33.4% (Tent) to 50.7% while reducing adaptation time from 106.1 seconds to 85.1 seconds; 2) ***Memory Usage***: Our methods also demonstrate efficient memory utilization, where EATA and EATA-C consume substantially less memory compared to all competing TTA methods, *e.g.*, on ResNet-50, memory usage decreases from 5417.7MB (SAR) to 2205.5MB (EATA-C, ϵ=0.02) while accuracy increases from 29.6% to 35.9%. This memory reduction is achieved through our active sample selection strategy, which reduces the number of samples involved in backpropagation. Note that the current Pytorch implementation does not support instance-wise gradient computation, thus an ideal implementation should further speed up both EATA and EATA-C. See more discussions on our implementation details in Appendix X.

Effectiveness of Data Uncertainty Indicator. We evaluate the effectiveness of our data uncertainty indicator, *i.e.*, Eqn. (13), throughout TTA. The results are detailed in Table 8: 1) *Consistently High Indicator Accuracy*: Across various adaptation stages, our indicator continues to reliably identify uncertain samples, on which the sub-model's prediction is likely to be incorrect. Specifically, the indicator maintains around 90% accuracy, suggesting its effectiveness even in the early stage of adaptation. This reliability allows us to apply entropy maximization on these uncertain data to improve calibration without hindering adaptation; 2) *Reduced Uncertain Samples Over Time*: The model's initial poor performance mainly leads to a higher number of uncertain samples, including inherently difficult data for discrimination and data the model has yet to learn. However, model uncertainty quickly explains away during TTA (*i.e.*, within one-fifth of the data stream), leading to a stabilized number of uncertain samples that reflects irreducible data uncertainty.

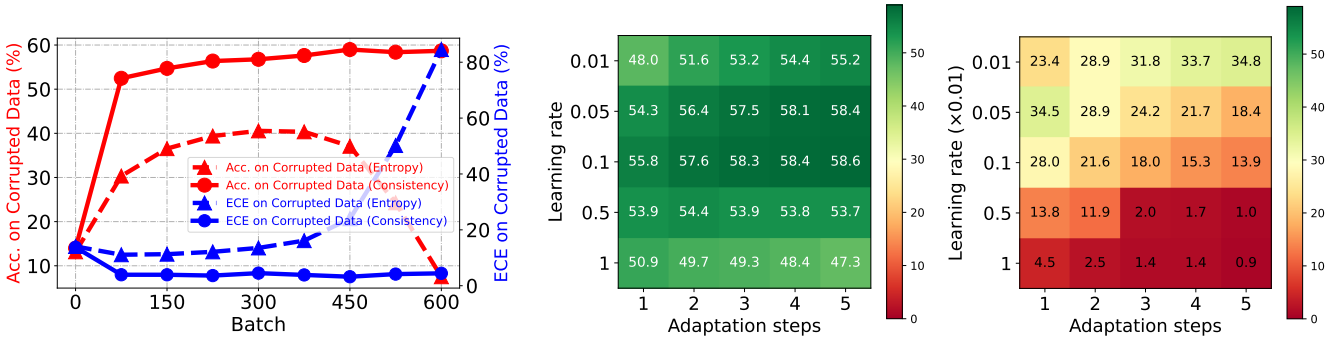
Additional Memory by Fisher Regularizer. Since we only regularize the affine parameters of normalization layers, EATA needs very little extra memory. For ResNet-50 on ImageNet-C, the extra GPU memory at run time is only 9.8 MB, which is much less than that of Tent with batch size 64 (5,675 MB).

Performance under Mixed-and-Shifted Distributions. We evaluate Tent and our EATA/EATA-C on mixed ImageNet-C (level=3 or 5) that consists of 15 different corruption types/distribution shifts (totaling 750k images). Results in Table 9 show the stability of EATA and EATA-C in large-scale and complex TTA scenarios.

Calibration Across Confidence Intervals. EATA-C aims to achieve a favorable balance among accuracy, calibration, and efficiency. In EATA-C, we discard high-entropy samples (termed active sample selection) mainly to improve computational efficiency. While some high-entropy samples might benefit from further calibration, they typically yield unreliable pseudo-labels, which can negatively impact the stability and effectiveness of TTA (see Table 6). Instead of directly calibrating on these samples, we show that focusing adaptation and calibration on only low-entropy samples can also improve the calibration on high-entropy ones, as in Figure 5.4, while improving TTA efficiency and effectiveness.

Advantage of Consistency Maximization over Tent [12]. We conduct a comprehensive comparison w.r.t. performance, calibration, and stability between the use of consistency loss, as defined in Eqn. (10), and Tent [12] to reduce uncertainty during testing. From Figure 11(a), we have the following observations: 1) Consistency loss consistently demonstrates superior performance and calibration throughout adaptation. 2) Consistency loss is more sample-efficient, where adapting with as few as 75 batches can significantly outperform Tent [12] that adapts with 300 batches. 3) Tent [12] shows rapid degradation in performance and calibration after convergence. In contrast, consistency loss maintains stable performance and calibration after convergence and continuously exhibits strong generalization throughout TTA.

We further evaluate the stability of consistency loss and Tent [12] across various combinations of learning rates and adaptation steps in Figures 11(b) and 11(c) following [73]. The



(a) Performance and calibration throughout adaptation (b) Consistency maximization via Eqn. (10) (c) Entropy minimization (Tent [12])

Fig. 11. Comparisons of performance, calibration and stability using consistency loss in Eqn. (10) or entropy minimization loss (*i.e.*, Tent) on ImageNet-C (Gaussian noise, severity level=5) with ViT-Base. In **Left**, we split the dataset into 80% and 20% slices to conduct adaptation and to evaluate the adapted model. We report the performance and the calibration of the adapted model every 75 batches with a batch size of 64. In the Middle and Right, we evaluate the stability of consistency loss and Tent under various combinations of learning rates and adaptation steps per batch. Consistency loss achieves substantially higher OOD performance and better stability while maintaining lower ECE.

results highlight that our consistency loss demonstrates remarkable stability and benefits from increased adaptation steps (*e.g.*, $55.8\% \rightarrow 58.6\%$, the best performance under 1 and 5 adaptation steps, respectively). In contrast, Tent [12] is highly sensitive to the combination of learning rate and adaptation steps, where its performance may deteriorate to as low as 1%, further indicating its tendency to overfit. These findings collectively underscore the superiority of our consistency loss regarding performance, calibration, and stability, making it a more robust choice for TTA.

6 Conclusion

In this paper, we have proposed an Efficient Anti-forgetting Test-time Adaptation method (EATA), to improve the performance of pre-trained models on a potentially shifted test domain. To be specific, we devise a sample-efficient entropy minimization strategy that selectively performs test-time optimization with reliable and non-redundant samples. This improves the adaptation efficiency and meanwhile boosts the out-of-distribution performance. In addition, we introduce a Fisher-based anti-forgetting regularizer into test-time adaptation. With this loss, a model can be adapted continually without performance degradation on in-distribution test samples. Moreover, we design EATA with Calibration (EATA-C) for test-time adapted model's confidence calibration. To this end, we present a consistency loss for calibrated model uncertainty reduction and a sample-aware min-max entropy regularization for confidence re-calibration, which improves the performance and calibration of test-time adaptation. Extensive experimental results on image classification and semantic segmentation demonstrate the effectiveness of our proposed methods.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
[2] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7794–7803.
[3] R. Zeng, H. Xu, W. Huang, P. Chen, M. Tan, and C. Gan, "Dense regression network for video grounding," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020.
[4] R. Zeng, W. Huang, M. Tan, Y. Rong, P. Zhao, J. Huang, and C. Gan, "Graph convolutional module for temporal action localization in videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
[5] P. Chen, D. Huang, D. He, X. Long, R. Zeng, S. Wen, M. Tan, and C. Gan, "Rspnet: Relative speed perception for unsupervised video representation learning," in *AAAI Conference on Artificial Intelligence*, vol. 1, 2021.
[6] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8789–8797.
[7] D.-P. Fan, T. Zhou, G.-P. Ji, Y. Zhou, G. Chen, H. Fu, J. Shen, and L. Shao, "Inf-net: Automatic covid-19 lung infection segmentation from ct images," *IEEE Transactions on Medical Imaging*, vol. 39, no. 8, pp. 2626–2637, 2020.
[8] G. Xu, S. Niu, M. Tan, Y. Luo, Q. Du, and Q. Wu, "Towards accurate text-based image captioning with content diversity exploration," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 637–12 646.
[9] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," in *International Conference on Learning Representations*, 2019.
[10] P. W. Koh, S. Sagawa, H. Marklund, S. M. Xie, M. Zhang, A. Balsubramani, W. Hu, M. Yasunaga, R. L. Phillips, I. Gao *et al.*, "Wilds: A benchmark of in-the-wild distribution shifts," in *International Conference on Machine Learning*, 2021, pp. 5637–5664.
[11] Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt, "Test-time training with self-supervision for generalization under distribution shifts," in *International Conference on Machine Learning*, 2020, pp. 9229–9248.
[12] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *International Conference on Learning Representations*, 2021.
[13] Y. Liu, P. Kothari, B. van Delft, B. Bellot-Gurlet, T. Mordan, and A. Alahi, "Ttt++: When does self-supervised test-time training fail or thrive?" in *Advances in Neural Information Processing Systems*, vol. 34, 2021.
[14] M. M. Zhang, S. Levine, and C. Finn, "Memo: Test time robustness via adaptation and augmentation," in *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021.
[15] Y. Zhang, B. Hooi, L. Hong, and J. Feng, "Self-supervised aggregation of diverse experts for test-agnostic long-tailed recognition," in *Advances in Neural Information Processing Systems*, 2022.
[16] Q. Wang, O. Fink, L. Van Gool, and D. Dai, "Continual test-time domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2022.
[17] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *International Conference on Learning Representations*, 2018.
[18] S. Niu, J. Wu, Y. Zhang, Z. Wen, Y. Chen, P. Zhao, and M. Tan, "Towards stable test-time adaptation in dynamic wild world," in *Internetional Conference on Learning Representations*, 2023.
[19] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang *et al.*, "End to end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.
[20] S. M. Anwar, M. Majid, A. Qayyum, M. Awais, M. Alnowami, and M. K. Khan, "Medical image analysis using convolutional neural networks: a review," *Journal of medical systems*, vol. 42, no. 11, pp. 1–13, 2018.

[21] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[22] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059.

[23] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[24] S. Niu, J. Wu, Y. Zhang, Y. Chen, S. Zheng, P. Zhao, and M. Tan, "Efficient test-time model adaptation without forgetting," in *International conference on machine learning*. PMLR, 2022, pp. 16 888–16 905.

[25] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.

[26] Y. Gandelsman, Y. Sun, X. Chen, and A. Efros, "Test-time training with masked autoencoders," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 29 374–29 385.

[27] A. Bartler, A. Bühler, F. Wiewel, M. Döbler, and B. Yang, "Mt3: Meta test-time training for self-supervised test-time adaption," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2022, pp. 3080–3090.

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

[29] S. Schneider, E. Rusak, L. Eck, O. Bringmann, W. Brendel, and M. Bethge, "Improving robustness against common corruptions by covariate shift adaptation," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 11 539–11 551.

[30] N. Reddy, M. Baktashmotlagh, and C. Arora, "Towards domain-aware knowledge distillation for continual model generalization," in *Winter Conference on Applications of Computer Vision*, 2024, pp. 696–707.

[31] F. Fleuret *et al.*, "Test time adaptation through perturbation robustness," in *Advances in Neural Information Processing Systems Workshop*, 2021.

[32] Y. Iwasawa and Y. Matsuo, "Test-time classifier adjustment module for model-agnostic domain generalization," in *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[33] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.

[34] D. Rolnick, A. Ahuja, J. Schwarz, T. P. Lillicrap, and G. Wayne, "Experience replay for continual learning," in *Advances in Neural Information Processing Systems*, 2019.

[35] M. Farajtabar, N. Azizan, A. Mott, and A. Li, "Orthogonal gradient descent for continual learning," in *International Conference on Artificial Intelligence and Statistics*, 2020, pp. 3762–3773.

[36] S. Niu, J. Wu, Y. Zhang, Y. Guo, P. Zhao, J. Huang, and M. Tan, "Disturbance-immune weight sharing for neural architecture search," *Neural Networks*, vol. 144, pp. 553–564, 2021.

[37] S. Mittal, S. Galesso, and T. Brox, "Essentials for class incremental learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2021, pp. 3513–3522.

[38] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *AAAI Conference on Artificial Intelligence*, 2018.

[39] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3723–3732.

[40] Y. Zhang, Y. Wei, Q. Wu, P. Zhao, S. Niu, J. Huang, and M. Tan, "Collaborative unsupervised domain adaptation for medical image diagnosis," *IEEE Transactions on Image Processing*, vol. 29, pp. 7834–7844, 2020.

[41] Y. Zhang, S. Niu, Z. Qiu, Y. Wei, P. Zhao, J. Yao, J. Huang, Q. Wu, and M. Tan, "Covid-da: deep domain adaptation from typical pneumonia to covid-19," *arXiv preprint arXiv:2005.01577*, 2020.

[42] Z. Qiu, Y. Zhang, H. Lin, S. Niu, Y. Liu, Q. Du, and M. Tan, "Source-free domain adaptation via avatar prototype generation and adaptation," in *International Joint Conference on Artificial Intelligence*, 2021.

[43] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation," in *International Conference on Machine Learning*, 2020, pp. 6028–6039.

[44] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *International conference on machine learning*. PMLR, 2017, pp. 1321–1330.

[45] M. P. Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 29, no. 1, 2015.

[46] J. Zhang, B. Kailkhura, and T. Y.-J. Han, "Mix-n-match: Ensemble and compositional methods for uncertainty calibration in deep learning," in *International conference on machine learning*. PMLR, 2020, pp. 11 117–11 128.

[47] A. Kumar, S. Sarawagi, and U. Jain, "Trainable calibration measures for neural networks from kernel mean embeddings," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2805–2814.

[48] S. Seo, P. H. Seo, and B. Han, "Learning for single-shot confidence calibration in deep neural networks through stochastic inferences," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9030–9038.

[49] S. Park, O. Bastani, J. Weimer, and I. Lee, "Calibrated prediction with covariate shift via unsupervised domain adaptation," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 3219–3229.

[50] X. Wang, M. Long, J. Wang, and M. Jordan, "Transferable calibration with lower bias and variance in domain adaptation," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 19 212–19 223.

[51] A. Karandikar, N. Cain, D. Tran, B. Lakshminarayanan, J. Shlens, M. C. Mozer, and B. Roelofs, "Soft calibration objectives for neural networks," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 29 768–29 779.

[52] H. S. Yoon, J. T. J. Tee, E. Yoon, S. Yoon, G. Kim, Y. Li, and C. D. Yoo, "ESD: Expected squared difference as a tuning-free trainable calibration measure," in *International Conference on Learning Representations*, 2023.

[53] W. Liu, X. Wang, J. Owens, and Y. Li, "Energy-based out-of-distribution detection," in *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[54] C. Berger, M. Paschali, B. Glocker, and K. Kamnitsas, "Confidence-based out-of-distribution detection: A comparative study and analysis," in *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging, and Perinatal Imaging, Placental and Preterm Image Analysis*. Springer, 2021, pp. 122–132.

[55] O. Chapelle and A. Zien, "Semi-supervised classification by low density separation," in *International workshop on artificial intelligence and statistics*. PMLR, 2005, pp. 57–64.

[56] J. Liang, R. He, and T. Tan, "A comprehensive survey on test-time adaptation under distribution shifts," *International Journal of Computer Vision*, pp. 1–34, 2024.

[57] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059.

[58] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[59] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[60] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, "Deep networks with stochastic depth," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*. Springer, 2016, pp. 646–661.

[61] M.-F. Balcan, A. Broder, and T. Zhang, "Margin based active learning," in *International Conference on Computational Learning Theory*. Springer, 2007, pp. 35–50.

[62] S. Khan, M. Hayat, S. W. Zamir, J. Shen, and L. Shao, "Striking the right balance with uncertainty," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 103–112.

[63] Q. Wang, O. Fink, L. Van Gool, and D. Dai, "Continual test-time domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 7201–7211.

[64] R. A. Marsden, M. Döbler, and B. Yang, "Universal test-time adaptation through weight ensembling, diversity weighting, and prior correction," in *Winter Conference on Applications of Computer Vision*, 2024, pp. 2555–2565.

[65] Y. Yuan, B. Xu, L. Hou, F. Sun, H. Shen, and X. Cheng, "Tea: Test-time energy adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2024, pp. 23 901–23 911.

[66] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo *et al.*, "The many faces of robustness: A critical analysis of out-of-distribution generalization," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8340–8349.

[67] C. Sakaridis, D. Dai, and L. Van Gool, "ACDC: The adverse conditions dataset with correspondences for semantic driving scene understanding," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[68] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 12 077–12 090.
[69] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.
[70] J. Ni, S. Yang, R. Xu, J. Liu, X. Li, W. Jiao, Z. Chen, Y. Liu, and S. Zhang, "Distribution-aware continual test-time adaptation for semantic segmentation," in *IEEE International Conference on Robotics and Automation*. IEEE, 2024, pp. 3044–3050.
[71] O. Press, S. Schneider, M. Kümmerer, and M. Bethge, "Rdumb: A simple approach that questions our progress in continual test-time adaptation," in *Advances in Neural Information Processing Systems*, vol. 36, 2024.
[72] M. P. Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 29, no. 1, 2015.
[73] H. Zhao, Y. Liu, A. Alahi, and T. Lin, "On pitfalls of test-time adaptation," in *International Conference on Machine Learning (ICML)*, 2023.
[74] Y. Wang, H. Wang, Y. Shen, J. Fei, W. Li, G. Jin, L. Wu, R. Zhao, and X. Le, "Semi-supervised semantic segmentation using unreliable pseudo-labels," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4248–4257.
[75] T.-Y. Pan, C. Zhang, Y. Li, H. Hu, D. Xuan, S. Changpinyo, B. Gong, and W.-L. Chao, "On model calibration for long-tailed object detection and instance segmentation," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 2529–2542.

Mingkui Tan is currently a Professor with the School of Software Engineering, South China University of Technology, Guangzhou, China. He received the Bachelor Degree in Environmental Science and Engineering in 2006 and the Master Degree in Control Science and Engineering in 2009, both from Hunan University in Changsha, China. He received the Ph.D. degree in Computer Science from Nanyang Technological University, Singapore, in 2014. From 2014-2016, he worked as a Senior Research Associate on computer vision in the School of Computer Science, University of Adelaide, Australia. His research interests include machine learning, sparse analysis, deep learning and large-scale optimization.

Guohao Chen is a Master student in the School of Software Engineering at South China University of Technology. He received his Bachelor Degree in the School of Software Engineering in 2022 from South China University of Technology in Guangzhou, China. His research interests are broadly in machine learning and mainly focus on inference-time learning. He has published papers in top venues, including NeurIPS and ICML.

Jiaxiang Wu is currently a senior researcher at XVERSE, China. Previously, he has worked at Tencent AI Lab. He received the B.E. degree in automation from Beijing Institute of Technology, and the Ph.D. degree in computer science from Institute of Automation, Chinese Academy of Sciences. His research interests include model compression, neural architecture search, distributed optimization, and protein structure prediction. He has published papers in top venues, including JMLR, PNAS, ICML, NeurIPS, ICLR, CVPR, and AAAI. He has been invited as a reviewer for top-tier conferences and journals, including ICML, NeurIPS, ICLR, CVPR, AAAI, IJCAI, TPAMI, and TNNLS.

Yifan Zhang obtained his Ph.D. degree in computer science at National University of Singapore. His research interests are broadly in machine learning, to solve domain shifts problems for deep learning. He has published papers in top venues, including NeurIPS, ICML, ICLR, CVPR, SIGKDD, ECCV, IJCAI, TIP, and TKDE. He has been invited as a reviewer for top-tier conferences and journals, including NeurIPS, ICML, ICLR, CVPR, ECCV, AAAI, IJCAI, TPAMI, TIP, IJCV, and TNNLS.

Yaofo Chen is recently a Post-doctoral Researcher with School of Future Technology at South China University of Technology. He received his Ph.D. degree in the School of Software Engineering in 2024 from South China University of Technology in Guangzhou, China. His research interests include neural architecture search and test-time adaptation. He has published papers in top venues, including ICML, ICLR, CVPR, AAAI, IEEE TCSVT and Neural Networks. He has been invited as a reviewer for top-tier conferences including ICLR, ICML, NeurIPS, CVPR, ICCV, ECCV and AAAI.

Peilin Zhao is currently a principal researcher at Tencent AI Lab in China. Previously, he worked at Rutgers University, A*STAR (Agency for Science, Technology and Research), and Ant Group. His research interests focused on machine learning and its applications. He has been invited to serve as area chair or associate editor at leading international conferences and journals such as ICML, TPAMI, etc. He received a bachelor's degree in mathematics from Zhejiang University, and a Ph.D. degree in computer science from Nanyang Technological University.

Shuaicheng Niu is currently a Research Fellow at Nanyang Technological University, Singapore. He received the Ph.D. degree from the South China University of Technology, China, in 2023. His research interests are broadly in machine learning and mainly focus on test-time computing and automated machine learning. He has published papers in top venues, including ICML, ICLR, NeurIPS, CVPR, ECCV, IJCAI, AAAI, TIP, and TKDE. He has been invited as a reviewer for top-tier conferences and journals, including ICML, ICLR, NeurIPS, CVPR, ICCV, ECCV, TPAMI, TIP, TNNLS, and IJCV.