Stabilizing Open-Set Test-Time Adaptation via Primary-Auxiliary Filtering and Knowledge-Integrated Prediction

Byung-Joon Lee[1]
lbj2001@skku.edu
Jin-Seop Lee[2]
wlstjq0602@skku.edu
Jee-Hyong Lee[†,1,2]
john@skku.edu

[1] Department of Computer Science and Engineering,
Sungkyunkwan University
[2] Department of Artificial Intelligence,
Sungkyunkwan University

Abstract

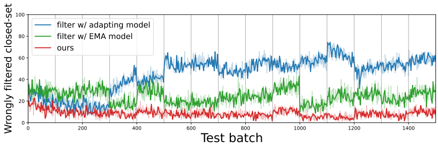
Deep neural networks demonstrate strong performance under aligned training-test distributions. However, real-world test data often exhibit domain shifts. Test-Time Adaptation (TTA) addresses this challenge by adapting the model to test data during inference. While most TTA studies assume that the training and test data share the same class set (closed-set TTA), real-world scenarios often involve open-set data (open-set TTA), which can degrade closed-set accuracy. A recent study showed that identifying open-set data during adaptation and maximizing its entropy is an effective solution. However, the previous method relies on the source model for filtering, resulting in suboptimal filtering accuracy on domain-shifted test data. In contrast, we found that the adapting model, which learns domain knowledge from noisy test streams, tends to be unstable and leads to error accumulation when used for filtering. To address this problem, we propose **Primary-Auxiliary Filtering** (**PAF**), which employs an auxiliary filter to validate data filtered by the primary filter. Furthermore, we propose **Knowledge-Integrated Prediction** (**KIP**), which calibrates the outputs of the adapting model, EMA model, and source model to integrate their complementary knowledge for OSTTA. We validate our approach across diverse closed-set and open-set datasets. Our method enhances both closed-set accuracy and open-set discrimination over existing methods. The code is available at https://github.com/powerpowe/PAF-KIP-OSTTA.

1 Introduction

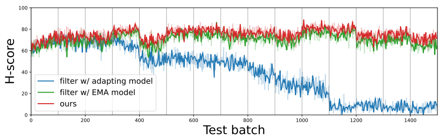
Deep neural networks have achieved outstanding performance across a wide range of tasks. However, this strong performance relies on the assumption that the training and test data are drawn from the same distribution [4]. In real-world scenarios, test data may undergo various domain shifts, such as changes in weather, brightness, or camera noise. To address this challenge, test-time adaptation (TTA) [15, 19, 20] aims to adapt the model to the test domain on the fly, during inference.

† Corresponding author.

(a) Percentage of wrongly filtered closed-set samples with UniEnt.



(b) Percentage of wrongly filtered closed-set samples with adapting, EMA, and our filter.



(c) H-score with adapting, EMA, and our filter.

Figure 1: Wrongly filtered closed-set samples and H-score on CIFAR100-C with Textures-C. The vertical dashed line marks domain shifts.

Most TTA studies have focused on the case where the classes in the training data and test data are identical (i.e., closed-set TTA) [1, 2, 6, 24]. However, it is common for unseen class data (i.e., open-set) to appear at test time [9]. In such scenarios, the TTA model continually learns incorrect information about closed-set classes, leading to error accumulation and degraded accuracy [8]. Furthermore, both open-set and closed-set data may appear with domain shifts, making it challenging to distinguish between them.

To address this problem, recent studies have proposed Open-set Test-Time Adaptation (OSTTA) methods that adapt to test data containing both closed-set and open-set samples [8, 10, 14, 21]. These methods estimate the confidence of each test sample, classify it as either open-set or closed-set, and leverage this categorization during adaptation.

While most OSTTA studies discard low-confidence data during adaptation, UniEnt [8] demonstrates that maximizing the entropy of such data can be effective. By increasing the entropy of open-set samples, the adapting model improves its ability to distinguish unknown instances by effectively learning what it does not know. Specifically, UniEnt utilizes a source pre-trained model to compute the similarity between each test sample and the source-domain class prototypes. Samples that closely align with known source classes are treated as reliable and optimized via entropy minimization, while those that appear dissimilar are considered unreliable and optimized via entropy maximization.

Incorporating domain knowledge into filtering is crucial, as domain-shifted test samples often exhibit low confidence when evaluated by a source model not trained on the target domain. However, UniEnt overlooks domain knowledge in filtering since its confidence scores rely solely on features from the fixed source model and class prototypes. As a result, it fails to exploit test-domain information and incorrectly filters many closed-set samples as open-set, as shown in fig. 1(a).

One way to leverage domain knowledge into filtering is to utilize the adapting model itself. However, relying on the adapting model's confidence for loss selection leads to significant error accumulation. This results in confirmation bias, where the adapting model reinforces its initial uncertainty by further reducing the confidence of low-confidence samples, as shown in fig. 1(b) and fig. 1(c).

We found that the exponential moving average (EMA) model of the adapting model helps prevent error accumulation, but fails to promptly capture domain shifts at the current time step, thereby resulting in suboptimal filtering performance. To address this, we propose **Primary-Auxiliary Filtering (PAF)**, which combines the adapting model and EMA model in a primary-auxiliary structure to incorporate domain knowledge while maintaining stability. The primary filter reflects test domain knowledge to achieve strong filtering performance, while the auxiliary filter ensures stability and prevents error accumulation caused by the primary filter. The two filters collaboratively categorize and adapt to the data through complementary soft and hard filtering strategies.

On the other hand, while typical TTA methods utilize either the adapting model [8, 17, 19, 21], the EMA model [20, 24], or their ensemble for inference [6], both models have limitations for inference in the OSTTA setting. Specifically, while our filtering reduces exposure to open-set data, the adapting model may still receive some open-set samples, potentially destabilizing the adapting model and impacting the EMA model. To address this, we propose **Knowledge-Integrated Prediction (KIP)**, which incorporates the source pre-trained model that remains unaffected by open-set samples. KIP combines the logits of the adapting model, EMA model, and source model, and adaptively assigns higher weights to models that exhibit greater confidence for each sample.

To demonstrate the effectiveness of the proposed method, we evaluate our approach across various open-set TTA scenarios. We use CIFAR10/100-C and ImageNet-C [4, 5] as closed-set datasets, while SVHN [16], Tiny-ImageNet [13], Places365 [25], and Textures [3] serve as open-set datasets. Whereas prior OSTTA studies focus on either closed-set accuracy or open-set discrimination, our method achieves substantial improvements on both objectives.

2 Primary-Auxiliary Filtering and Knowledge-Integrated Prediction

We first present the preliminaries in section 2.1, followed by the motivation in section 2.2. Then, we introduce our proposed methods, Primary-Auxiliary Filtering (PAF) and Knowledge-Integrated Prediction (KIP), in section 2.3 and section 2.4, respectively.

2.1 Preliminary

Let the source domain dataset with the label space $\mathcal{Y}_s = \{1, \cdots, \mathcal{C}_s\}$ be denoted as $\mathcal{D}_s = \{x_i, y_i\}_{i=1}^{N_s}$, and the test domain dataset be denoted as $\mathcal{D}_t = \{x_j, y_j\}_{j=1}^{N_t}$. If the label space of the test domain dataset $\mathcal{Y}_t$ is the same as $\mathcal{Y}_s$, it represents a standard closed-set TTA scenario. In contrast, open-set TTA refers to the setting where $\mathcal{Y}_s \subset \mathcal{Y}_t$, meaning that the label set of the test domain includes additional open-set classes.

For TTA, a source pre-trained model f_{θ_0} trained on $\mathcal{D}_s$ is given. The model deployed in the environment encounters streaming test batch $\mathbf{X}_t$ at time t for inference. Since the test batch does not have any labels, most TTA models minimizes the entropy of prediction as follows:

$$\min_{\theta_t} \mathcal{L}_{TTA}(x) = \sum_{x \in \mathbf{X}_t} H(f_{\theta_t}(x)), \tag{1}$$

where $H(\cdot)$ denotes the entropy. However, in the OSTTA scenario, the presence of open-set data makes it undesirable to increase the reliability of all data. To address this problem, existing OSTTA studies filter data based on confidence for test-time training as follows:

$$\min_{\theta_t} \mathcal{L}_{OSTTA}(x) = \sum_{x \in \mathbf{X}_t} \Big[\mathbb{I}[\text{conf}(x) > \tau] \cdot H(f_{\theta_t}(x)) - \alpha \cdot \mathbb{I}[\text{conf}(x) \leq \tau] \cdot H(f_{\theta_t}(x)) \Big], \quad (2)$$

where $\text{conf}(\cdot)$ is a function that measures the confidence of data, such as entropy or comparison with the source model, and τ is a confidence threshold. When $\alpha = 0$, the OSTTA method disregards low-confidence data [10, 14, 21], while for $\alpha > 0$, it maximizes the entropy of low-confidence data [8].

2.2 Motivation

While UniEnt has demonstrated the effectiveness of entropy maximization, we found that it relies on source knowledge for filtering and incorrectly identifies many closed-set samples as open-set. Furthermore, we observed that incorporating domain knowledge into training is prone to error accumulation. To validate this observation, we conduct experiments in a continual test-time adaptation setting, where CIFAR-100-C and Textures-C are used as the closed-set and open-set datasets, respectively. The model sequentially adapts to 15 different domains, and we update only the batch normalization layers. The domain changes every 100 batches, with transition points indicated by dashed lines in fig. 1. "Wrongly filtered closed-set" in fig. 1(a) and fig. 1(b) refers to the proportion of closed-set samples that were incorrectly assigned low confidence and trained with the entropy maximization loss.

Limitation for using fixed source model for measuring confidence. As mentioned earlier, existing methods rely on fixed source domain knowledge to measure $\text{conf}(\cdot)$. Figure 1(a) shows the proportion of closed-set samples that were incorrectly assigned low confidence and trained with the entropy maximization loss. Since they fail to incorporate test domain knowledge, they assign low confidence even to many closed-set samples. As a result, more than half of the closed-set data receive lower confidence scores than open-set data and are incorrectly trained with the entropy maximization loss.

Using the adapting model causes error accumulation. One simple way to utilize test domain knowledge for confidence measurement is to use the adapting model. We applied a strategy that selects the loss based on the confidence of the adapting model. Specifically, we designed a method that determines whether to apply entropy minimization or maximization using the entropy of the adapting model $f_{\theta_t}(x)$ as follows:

$$\mathcal{L}_{adapt}(x) = \sum_{x \in \mathbf{X}_t} \Big[\mathbb{I}\big[H\big(f_{\theta_t}(\tilde{x})\big) < \tau\big] \cdot H(f_{\theta_t}(\tilde{x})) - \mathbb{I}\big[H\big(f_{\theta_t}(\tilde{x})\big) \geq \tau\big] \cdot H(f_{\theta_t}(\tilde{x})) \Big], \quad (3)$$

where $\tilde{x}$ is an augmented view of x. Following the confidence threshold used in previous methods [17], we set $\tau = 0.4 \times \log(C)$, where C denotes the number of classes. Since this method directly utilizes the knowledge of the adapting model for filtering, it can immediately reflect the characteristics of each test domain. As shown in fig. 1(b), it achieves high filtering accuracy during the first 300 test batches. However, as more test batches accumulate, error accumulation occurs, leading to a significant drop in filtering accuracy. As a result, the model performance significantly degrades as training progresses, as shown in fig. 1(c).

EMA enables stable filtering but exhibits suboptimal performance. Inspired by Mean Teacher [18], which addresses a similar confirmation bias issue in semi-supervised learning, we found that the EMA model $f_t^{\text{EMA}}(x)$ obtained by aggregating the parameters of the adapting model can provide more stable filtering. For each test time step t, the EMA model $f_t^{\text{EMA}}(x)$ is updated as follows:

$$f_t^{\mathrm{EMA}}(x) = \beta \cdot f_{t-1}^{\mathrm{EMA}}(x) + (1-\beta) \cdot f_{\theta_t}(x), \tag{4}$$

where β is the decay rate that determines the weight of past parameters in the EMA model. Instead of using the adapting model, which causes error accumulation, we designed a strategy that utilizes the entropy of the EMA model as a confidence measure, as follows:

$$\mathcal{L}_{ema}(x) = \sum_{x \in \mathbf{X}_t} \Big[\mathbb{I}\big[H\big(f_t^{\mathrm{EMA}}(\tilde{x})\big) < \tau\big] \cdot H(f_{\theta_t}(\tilde{x})) - \mathbb{I}\big[H\big(f_t^{\mathrm{EMA}}(\tilde{x})\big) \geq \tau\big] \cdot H(f_{\theta_t}(\tilde{x})) \Big]. \tag{5}$$

The green graph in fig. 1(b) represents the filtering results of the EMA model, while the green graph in fig. 1(c) shows the overall performance. Since the EMA model aggregates the adapting model over multiple time steps, it reduces the rapid propagation of errors and provides more stable filtering.

However, unlike semi-supervised learning where Mean Teacher [18] is used, the EMA model in a continually shifting TTA environment accumulates knowledge from past domains. As a result, the EMA model still exhibits suboptimal filtering performance. In practice, the filtering behavior of the EMA model fluctuates significantly across different domains, and before error accumulation occurs (before 300 batches), its filtering performance is lower than that of the adapting model. To address this, we propose a method that leverages both the adapting model, which effectively reflects current domain knowledge but is prone to error accumulation, and the EMA model, which provides stable filtering. As shown in fig. 1(b) and fig. 1(c) (red graph), our approach maintains consistently high filtering performance across domains while preventing error accumulation over time.

2.3 Primary-Auxiliary Filtering

As aforementioned, using the adapting model or the EMA model individually for filtering has distinct limitations. Therefore, we propose **Primary-Auxiliary Filtering (PAF)**, which leverages both models for filtering. PAF is a simple yet effective method. First, we define the primary filter $F_{pr}(x)$ that utilizes the adapting model $f_{\theta_t}(x)$, and identify each sample as follows:

$$F_{pr}(x) = \mathbb{I}\Big[H(f_{\theta_t}(\tilde{x})) < \tau\Big]. \tag{6}$$

A sample x is identified as a potential open-set sample if $F_{pr}(x) = 0$. However, directly applying entropy maximization to these samples leads to error accumulation. To mitigate this issue, we perform an additional filtering step using the stable EMA model $f_t^{\mathrm{EMA}}(x)$ as an auxiliary filter. Similar to $F_{pr}(x)$, the auxiliary filter F_{aux} is defined as follows:

$$F_{aux}(x) = \mathbb{I}\Big[H(f_t^{EMA}(\tilde{x})) < \tau\Big], \tag{7}$$

where the only difference is that the EMA model is used instead of the adapting model. Finally, samples for which both filters output 0 are trained with entropy maximization. Meanwhile, samples with $F_{pr}(x) = 0$ but $F_{aux}(x) = 1$ are identified as potentially prone to error accumulation and are excluded from training.

Conversely, if $F_{pr}(x) = 1$, the sample x is identified as a closed-set instance. However, we found that applying double filtering with $F_{aux}(\cdot)$ may lead to overly conservative training or even underfitting. Nevertheless, relying solely on $F_{pr}(\cdot)$ may still result in incorrect training. To address this, we propose a soft filtering method that softly integrates $F_{aux}(\cdot)$ into training for samples identified as closed-set. Specifically, we integrate the confidence from the EMA mode f_t^{EMA} into entropy minimization by weighting the loss as follows:

$$w_{soft}(x) = \frac{1}{\exp[H(f_t^{\text{EMA}}(\tilde{x})) - \tau]}, \tag{8}$$

which assigns higher weights to high-confidence samples above the threshold [17]. The weight computed as above is multiplied by the entropy minimization loss. In other words, for data that $F_{pr}(\cdot)$ considers reliable, the influence on training increases as the EMA model assigns higher confidence. Conversely, if the entropy from the EMA model exceeds the threshold, a small weight is assigned to the loss. We further analyze the effects of soft and hard filtering. Finally, our proposed loss $\mathcal{L}_{PAF}(x)$ is defined as follows:

$$\begin{aligned} \min_{\theta_t} \mathcal{L}_{PAF}(x) = \sum_{x \in \mathbf{X}_t} \Big[& \mathbb{I}[F_{pr}(x) = 1] \cdot w_{soft}(x) \cdot H(f_{\theta_t}(\tilde{x})) \\ & - \alpha \cdot \mathbb{I}\big[(F_{pr}(x) = 0) \wedge (F_{aux}(x) = 0)\big] \cdot H(f_{\theta_t}(\tilde{x}))\Big]. \end{aligned} \tag{9}$$

With $\mathcal{L}_{PAF}(\cdot)$, we update only the batch normalization layers of the model. On the other hand, previous OSTTA studies [8, 21] output the open-set score of each sample to validate the open-set discrimination performance. Therefore, we output the energy score of the adapting model as the open-set score for fair comparison.

2.4 Knowledge-Integrated Prediction

As mentioned in section 2.2, the adapting model can respond quickly to distribution shifts, whereas the EMA model provides more stable and reliable predictions. Many existing TTA methods leverage either the adapting model [8, 17, 19, 21] or the EMA model [20, 24]. Notably, RMT [6] ensembles both models during inference to exploit these complementary strengths.

However, both models have limitations for inference in the OSTTA setting. Although PAF significantly improves filtering accuracy as shown in fig. 1(b), some samples are still incorrectly filtered. This may be attributed to the fact that the model is trained to suppress the confidence of certain closed-set logits, which degrades its predictive performance on closed-set samples near the decision boundary.

Although the source model $f_{\theta_0}(\cdot)$ is not adapted to the target domain, it is important to highlight that it has never encountered open-set samples. Therefore, we propose **Knowledge-Integrated Prediction (KIP)**, which calibrates the outputs of the source, adapting, and EMA models to produce more reliable predictions. We design a per-sample weighted prediction scheme, where the weight assigned to each model's output is determined by comparing its confidence to the average confidence across all models. If a model yields a lower confidence for a given sample, its importance in the ensemble is reduced relative to uniform averaging, and vice versa. Specifically, we define the weight $c_i(x)$ assigned to the i-th model as follows:

$$c_i(x) = \frac{1}{3} + \gamma \cdot (\max(p_i(x)) - \sum_{j=1}^{3} \frac{\max(p_j(x))}{3}). \tag{10}$$

Without loss of generality, we denote the probability vectors from the source, adapting, and EMA models as $p_1(x), p_2(x), p_3(x)$. The hyperparameter γ controls the sharpness of the weights. Since the upper bound of entropy depends on the number of classes, we instead

Table 1: Results of various methods on CIFAR10-C benchmarks with different open-set datasets. All values are averaged over 15 domains. **Bold** indicate the best results, underline indicate the second-best, and red numbers represent the improvement over the previous best.

	SVHN-C			TinyImageNet-C			Places365-C			Textures-C		
	ACC	AUR	H-S	ACC	AUR	H-S	ACC	AUR	H-S	ACC	AUR	H-S
Source	81.73	77.89	79.76	81.73	80.43	81.07	81.73	83.60	82.65	81.73	82.60	82.16
TENT	78.05	67.64	72.47	56.91	62.44	59.55	49.31	55.83	52.37	65.92	64.68	65.29
CoTTA	<u>85.23</u>	82.91	84.05	85.88	79.20	82.40	85.01	82.71	83.84	83.17	79.44	81.26
EATA	84.47	82.24	83.34	84.97	78.07	81.37	84.65	80.63	82.59	82.45	76.27	79.24
RoTTA	85.08	82.97	84.01	85.12	80.09	82.53	85.09	82.33	83.69	<u>84.80</u>	80.89	82.80
SoTTA	69.52	46.91	56.02	79.98	63.03	70.50	79.46	63.23	70.42	69.97	49.62	58.06
OSTTA	84.11	71.98	77.57	85.29	71.18	77.60	84.69	73.09	78.46	83.07	66.79	74.05
STAMP	82.95	73.27	77.81	<u>86.83</u>	79.72	83.12	<u>85.65</u>	80.19	82.83	81.48	66.09	72.98
UniEnt	84.56	90.21	87.29	85.19	84.43	84.81	84.74	89.42	87.02	82.36	85.60	83.95
UniEnt+	83.98	<u>92.94</u>	<u>88.23</u>	84.19	<u>85.53</u>	84.85	84.24	<u>90.77</u>	<u>87.38</u>	81.21	<u>88.36</u>	<u>84.63</u>
Ours	**87.49**+2.26	**97.66**+4.72	**92.30**+4.07	**88.26**+1.43	**90.61**+5.08	**89.42**+4.57	**88.35**+2.70	**94.24**+3.48	**91.20**+3.82	**87.80**+3.00	**97.71**+9.35	**92.49**+7.86

Table 2: Results of various methods on CIFAR100-C with different open-set datasets.

	SVHN-C			TinyImageNet-C			Places365-C			Textures-C		
	ACC	AUR	H-S	ACC	AUR	H-S	ACC	AUR	H-S	ACC	AUR	H-S
Source	53.25	60.55	56.67	53.25	68.91	60.08	53.25	66.92	59.31	53.25	62.37	57.45
TENT	20.76	67.69	31.77	20.91	56.87	30.58	19.85	59.50	29.77	22.86	61.13	33.28
CoTTA	55.44	74.07	63.42	57.61	68.97	62.78	55.21	71.94	62.47	52.13	65.01	57.86
EATA	52.81	70.79	60.49	54.25	68.00	60.35	53.69	70.81	61.07	49.75	57.88	53.51
RoTTA	59.74	75.53	66.71	60.00	70.74	64.93	60.05	70.14	64.70	59.27	72.33	<u>65.15</u>
SoTTA	39.48	65.74	49.33	41.52	61.25	49.49	43.20	60.01	50.24	52.31	57.32	54.70
OSTTA	<u>60.24</u>	75.68	67.08	61.60	71.20	66.05	60.35	72.72	65.96	<u>59.57</u>	66.43	62.81
STAMP	59.51	<u>92.61</u>	<u>72.46</u>	**64.25**	73.16	<u>68.42</u>	<u>62.89</u>	74.23	<u>68.09</u>	54.32	39.79	45.93
UniEnt	59.24	88.91	71.10	60.34	73.87	66.42	59.11	78.10	67.29	57.02	73.91	64.38
UniEnt+	58.74	91.03	71.40	60.24	<u>74.21</u>	66.50	58.90	<u>78.64</u>	67.35	56.58	<u>74.55</u>	64.33
Ours	**62.59**+2.35	**97.61**+5.00	**76.27**+3.81	<u>63.79</u>-0.46	**84.16**+9.95	**72.57**+4.15	**62.94**+0.05	**85.25**+5.98	**72.42**+4.33	**62.81**+3.24	**94.54**+19.99	**75.48**+10.33

use the maximum class probability, whose upper bound is independent of the number of classes. Finally, the aggregated logit $z_{kip}(x)$ is computed as:

$$z_{kip}(x) = \sum_{i=1}^{3} c_i(x) \cdot z_i(x), \tag{11}$$

where $z_i(x)$ denotes the logit of the i-th model. Since the source model was not trained with augmentations, it operates on the original input x rather than the augmented version $\tilde{x}$.

3 Experiments

3.1 Experiments Setup

Setting and Datasets. Following previous studies, we select three types of corrupted datasets as closed-set datasets. CIFAR10-C [4] is derived from the CIFAR10 test data with 15 types of corruption, containing 10 classes. Similarly, CIFAR100-C and ImageNet-C [4] are based on the CIFAR100 and ImageNet test datasets. Each corruption has an intensity level ranging from 1 to 5, and we use the highest level. For CIFAR benchmarks, we use SVHN-C [16], Tiny-ImageNet-C [13], Places365-C [25], and Textures-C [3] as open-set datasets. For ImageNet-C [5], we use Places365-C and Textures-C. All open-set data were resized to match the size of the closed-set data and had the same types of corruption applied as the closed-set. Following prior studies [8, 14, 21], we assume a continual OSTTA environment, where domains change continuously, and open-set and closed-set data are mixed at a 1:1 ratio. The model sequentially adapts to each of the 15 corruption domains, and no reset is performed throughout the adaptation process.

All models are provided by RobustBench [4] and are pre-trained on a clean dataset using AugMix augmentation. For CIFAR benchmarks, the WideResNet-40-2 [23] architecture is

Table 3: Results of various methods on ImageNet-C benchmarks with different open-set datasets.

	Places365-C			Textures-C		
	ACC	AUR	H-S	ACC	AUR	H-S
Source	28.22	67.86	39.86	28.22	71.55	40.48
TENT	41.28	65.91	50.77	35.12	47.30	40.31
CoTTA	41.41	72.64	52.75	40.81	70.05	51.57
EATA	<u>46.99</u>	78.95	58.91	43.42	65.62	52.26
RoTTA	44.15	72.14	54.78	45.13	77.60	57.07
SoTTA	2.66	50.49	5.05	2.59	40.46	4.87
OSTTA	47.05	75.62	58.01	45.03	61.73	52.07
STAMP	46.88	<u>80.11</u>	<u>59.15</u>	<u>47.49</u>	<u>80.59</u>	<u>59.76</u>
UniEnt	46.16	78.81	58.22	44.46	64.01	52.47
UniEnt+	45.82	78.75	57.93	44.23	63.50	52.14
Ours	**48.22**+1.17	**84.18**+4.07	**61.32**+2.17	**47.96**+0.47	**82.91**+2.32	**60.77**+1.01

Table 4: Ablation study of F_{pr}, F_{aux} and KIP.

F_{pr}	F_{aux}	KIP	ACC	AUR	H-S
✗	✗	✗	20.76	67.69	31.77
✓	✗	✗	26.27	82.98	39.91
✗	✓	✗	55.83	97.11	70.90
✓	✓	✗	58.27	97.61	72.98
✓	✓	✓	62.59	97.61	76.27

used. During TTA, a batch size of 200 and the Adam [12] optimizer with a learning rate of 0.001 are utilized. For ImageNet benchmarks, the ResNet-50 [11] architecture is used, with a batch size of 64 and the SGD optimizer with a learning rate of 0.00025. Note that SoTTA and STAMP are implemented with the SAM optimizer [7] as in the original paper, and their learning rates are maintained as specified in the respective studies.

Evaluation protocols. We evaluate both the classification capability on closed-set data and the discrimination ability for open-set data. The classification performance on closed-set data is measured using classification accuracy **(ACC)**. For open-set discrimination ability, we use the area under the ROC curve **(AUR)**, calculated based on the open-set scores generated by the model for each data point. Following prior studies, the energy score is used for all methods. To combine both metrics, we use the harmonic mean **(H-S)** of ACC and AUR. We compare our method with nine TTA methods, including open-set TTA approaches. The hyperparameters for each method are set as closely as possible to those in the original papers.

Implementation details. We use simple random crop and horizontal flip augmentations. The loss balancing hyperparameter α is set to 2.0 and 0.7 for CIFAR and ImageNet, respectively. Following Mean Teacher [18], we set β as 0.999. γ is set to 0.1. While several OSTTA methods employ dataset-specific thresholds, we set τ to $0.4 \times \log(C)$ across all settings, where C is the number of closed-set classes, following [17]. We later show that our method is robust to hyperparameter selection.

3.2 Results

CIFAR benchmarks. Table 1 shows the results of adding various open-set datasets to an environment where CIFAR10-C is used as the closed-set dataset. Existing methods achieve strong performance in only one of the two metrics. Generally, RoTTA and STAMP, which focus on stable learning for closed-set data, exhibit high ACC but lower AUROC. In contrast, UniEnt and UniEnt+ tend to have relatively lower ACC but higher AUROC. Nevertheless, our method consistently improves performance across all scenarios. In particular, when Textures-C was used as the open-set dataset, our method demonstrates a performance increase of 3.00%p in ACC and 9.35%p in AUROC.

Table 2 presents the evaluation results where CIFAR100-C serves as the closed-set dataset. Similar to CIFAR10-C, methods focused on stable learning exhibit high ACC but lower AUROC, while UniEnt shows relatively lower ACC but higher AUROC. On the other hand, our method achieves performance improvements across most cases. Notably, when Textures-C is used as the open-set dataset, our approach achieves 19.99%p higher AUROC than the best existing method. This indicates that our method effectively handles open-set data and allows the model to achieve strong discriminative capability.

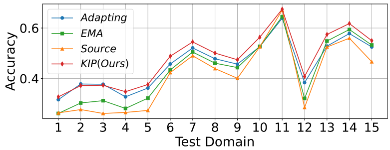


Figure 2: Accuracy of KIP and independent models used in KIP.

(a) CIFAR100-C + SVHN-C

Ent.Min	Ent.Max Soft	Ent.Max Hard
Soft	75.92	**76.27**
Hard	73.85	75.42

(b) ImageNet-C + Textures-C

Ent.Min	Ent.Max Soft	Ent.Max Hard
Soft	55.52	**60.77**
Hard	47.04	59.49

Table 5: Ablation study on soft and hard filtering

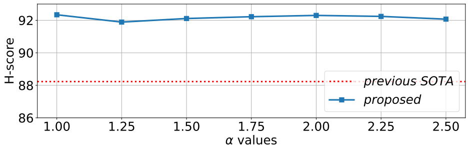


Figure 3: H-score with various loss-weighting hyperparameters α

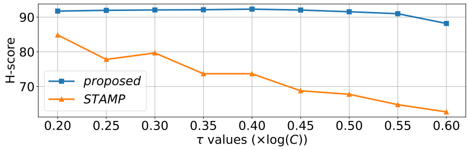


Figure 4: H-score with various entropy thresholds τ

ImageNet benchmarks. Table 3 presents the results of adding various open-set datasets to an environment where ImageNet-C is used as the closed-set dataset. ImageNet-C poses a more challenging task due to its 1000 classes and larger image size. As a result, UniEnt achieves relatively low AUROC, even when entropy maximization is applied to low-confidence samples. However, our method still achieves superior performance. Notably, in the environment where Places365-C is the open-set dataset, our approach significantly improves the H-score.

3.3 Further Analysis

Ablation study for F_{pr}, F_{aux}, and KIP. Table 4 shows the ablation study of the main components of our method, using CIFAR100-C as the closed-set and SVHN-C as the open-set dataset. Without any filtering, the model minimizes the entropy of open-set samples, leading to severe performance degradation. Using only F_{pr} leads to confirmation bias, while F_{aux} prevents error accumulation but fails to incorporate target-domain knowledge, resulting in lower accuracy. In contrast, combining both as a primary-auxiliary filter improves both metrics. Finally, integrating all components yields further improvements in both metrics.

Soft filtering and hard filtering. Table 5 presents the H-score on different filtering combinations. The best performance is achieved by applying soft filtering for entropy minimization and hard filtering for entropy maximization. Not rejecting samples during entropy maximization significantly degrades performance, while using hard filtering during entropy minimization leads to overly conservative learning and reduced accuracy.

Ablation study of models used in KIP. Figure 2 presents the accuracy of three models across different domains, along with the performance of KIP. ImageNet-C and Textures-C serve as the closed-set and open-set datasets, respectively. When each model is used independently, the performance remains suboptimal. In particular, learning from the open-set causes the performance of the adapting model to temporarily drop below that of the source or EMA model. In contrast, KIP effectively calibrates unstable knowledge and achieves a significant improvement. A detailed ablation and effectiveness analysis of KIP is presented in the appendix.

Hyperparameter sensitivity. Figures 3 and 4 present experiments on loss weighting hyperparameters α and threshold τ. CIFAR10-C and SVHN-C are used as the closed-set and open-set datasets, respectively. Our method consistently achieves higher H-scores than existing approaches across all settings and remains robust to the choice of both hyperparameters.

4 Conclusion

In this study, we propose PAF, which substantially mitigates error accumulation during sample filtering in OSTTA. We also introduce KIP, a simple and effective strategy to enhance inference in OSTTA. Unlike prior approaches that rely on extensive data augmentations [21] or require full model fine-tuning [20, 22], our method is efficient, requiring only a few forward passes and updating only batch normalization layers. Moreover, unlike replay buffer-based methods [10, 21, 22] that store test samples during inference, our approach operates without any external data buffer, making it more suitable for privacy-sensitive applications. Our method achieves strong performance on the CIFAR and ImageNet benchmarks.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (RS-2019-II190421, AI Graduate School Support Program(Sungkyunkwan University), 20%), (No.2022-0-01045, Self-directed Multi-modal Intelligence for solving unknown, open domain problems, 20%), (No.1711195788, Development of Flexible SW/HW Conjunctive Solution for on-edge self-supervised learning, 20%), (No.RS-2025-25442569, AI Star Fellowship Support Program (Sungkyunkwan Univ.), 20%) and (No.RS-2024-00360227, Developing Multimodal Generative AI Talent for Industrial Convergence, 20%).

References

[1] Dhanajit Brahma and Piyush Rai. A probabilistic framework for lifelong test-time adaptation. In *CVPR*, pages 3582–3591, 2023.

[2] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *CVPR*, pages 295–305, 2022.

[3] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, pages 3606–3613, 2014.

[4] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. In *NIPS workshop on Datasets and Benchmarks Track*, 2021.

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[6] Mario Döbler, Robert A Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *CVPR*, pages 7704–7714, 2023.

[7] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020.

[8] Zhengqing Gao, Xu-Yao Zhang, and Cheng-Lin Liu. Unified entropy optimization for open-set test-time adaptation. In *CVPR*, pages 23975–23984, 2024.

[9] Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. Recent advances in open set recognition: A survey. *PAMI*, 43(10):3614–3631, 2020.

[10] Taesik Gong, Yewon Kim, Taeckyung Lee, Sorn Chottananurak, and Sung-Ju Lee. Sotta: Robust test-time adaptation on noisy data streams. *NIPS*, 36, 2024.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[13] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS231N*, 7(7):3, 2015.

[14] Jungsoo Lee, Debasmit Das, Jaegul Choo, and Sungha Choi. Towards open-set test-time adaptation utilizing the wisdom of crowds in entropy minimization. In *ICCV*, 2023.

[15] Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *IJCV*, pages 1–34, 2024.

[16] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, page 4. Granada, 2011.

[17] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *ICML*, 2022.

[18] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *NIPS*, 30, 2017.

[19] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021.

[20] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *CVPR*, 2022.

[21] Yongcan Yu, Lijun Sheng, Ran He, and Jian Liang. Stamp: Outlier-aware test-time adaptation with stable memory replay. In *ECCV*, pages 375–392. Springer, 2025.

[22] Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *CVPR*, 2023.

[23] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016.

[24] Bowen Zhao, Chen Chen, and Shu-Tao Xia. Delta: Degradation-free fully test-time adaptation. In *ICLR*.

[25] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *PAMI*, 40(6):1452–1464, 2017.