Test-Time Distillation for Continual Model Adaptation

Supplementary Material

A. Pseudo-Codes of CoDiRe

Algorithm 1 presents the detailed pseudo-code for the CoDiRe framework, which addresses the two empirical pitfalls in our proposed Test-Time Distillation (TTD) paradigm.

Algorithm 1 The detailed procedures of CoDiRe

Require: Pretrained Target Model $f(\cdot;\theta_0)$, Frozen VLM Teacher $\mathcal{F}(\cdot)$, Unlabeled Test Data $\mathcal{X}_t^{\mathrm{T}}$, Reset Threshold γ_0, Anchor Update Step s, Reset Ratio α

Initialize current parameters $\theta_t \leftarrow \theta_0$
Initialize anchor parameters $\theta^{\text{anchor}} \leftarrow \theta_0$
Initialize previous parameters $\theta_{t-1} \leftarrow \theta_0$
for each time step $t = 1, 2, \ldots$ **do**
  Receive mini-batch $\{x_i\}_{i=1}^{N} \subseteq \mathcal{X}_t^{\mathrm{T}}$
  // 1. Distillation: Construct Blended Teacher
  Get logits: $z_i^{\text{tar}} = f(x_i;\theta_t)$, $z_i^{\text{tea}} = \mathcal{F}(x_i)$
  Compute MSP-based weight via Eq. (2):
    $\lambda_i = \frac{\exp(\max p_i^{\text{tea}})}{\exp(\max p_i^{\text{tea}})+\exp(\max p_i^{\text{tar}})}$
  Compute blended logits: $z_i^{\text{bt}} = \lambda_i \cdot z_i^{\text{tea}} + (1-\lambda_i)\cdot z_i^{\text{tar}}$
  Get blended probability: $p_i^{\text{bt}} = \sigma(z_i^{\text{bt}})$
  // 2. Rectification: Optimal Transport
  Compute marginal constraints m via pseudo-label voting
  Solve OT problem via Sinkhorn algorithm to get $\mathbf{P}^{rm}$ (Eq. 4-5)
  // 3. Optimization
  Compute Distillation Loss: $\mathcal{L}_{\text{Distill}}$ using p_i^{bt} (Eq. 3)
  Compute Rectification Loss: $\mathcal{L}_{\text{Rect}}$ using P^{rm} (Eq. 6)
  Compute Entropy Loss: $\mathcal{L}_{\text{Ent}}$ (Eq. 7)
  Total Loss: $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{Ent}} + \mathcal{L}_{\text{Distill}} + \mathcal{L}_{\text{Rect}}$
  Update parameters: $\theta_{t+1} \leftarrow \theta_t - \eta \nabla_{\theta_t} \mathcal{L}_{\text{total}}$
  // 4. Distribution-Aware Reset Mechanism
  Compute displacements: $\delta_t = \theta_{t+1} - \theta_t$, $\delta_t^{\text{anchor}} = \theta_t - \theta^{\text{anchor}}$
  Calculate cosine similarity: $\gamma = \cos(\delta_t, \delta_t^{\text{anchor}})$
  if $\gamma < \gamma_0$ **then** ▷ Domain shift detected
    Reset the last $\alpha\%$ layers of θ_{t+1} to θ_0
  end if
  if $t \bmod s == 0$ **then** ▷ Periodic anchor update
    Update anchor: $\theta^{\text{anchor}} \leftarrow \theta_{t+1}$
  end if
  $\theta_t \leftarrow \theta_{t+1}$
  $\theta_{t-1} \leftarrow \theta_t$
end for

B. MSP-Accuracy Bninning Experiment

This section presents an empirical validation of the reliability of Maximum Softmax Probability (MSP) as a confidence metric under distribution shifts. This experiment serves as the empirical foundation for the theoretical analysis presented in Appendix C, specifically supporting the error bound assumption in Eq. (12).

Experimental Setup. To rigorously assess the calibration of MSP, we conduct evaluations on two benchmark datasets: CIFAR-10-C and ImageNet-C. We pretrain ResNet-50 and ViT-B/16 as target models on CIFAR-10 and ImageNet, respectively, alongside the CLIP-ViT-L/14 teacher model. The evaluation encompasses all 15 corruption types at the highest severity level 5. For each model, we aggregate the prediction results across all corruption scenarios. We then discretize the MSP-based confidence scores into bins with an interval of 0.05. For each bin, we calculate the average accuracy of the samples falling within that confidence range.

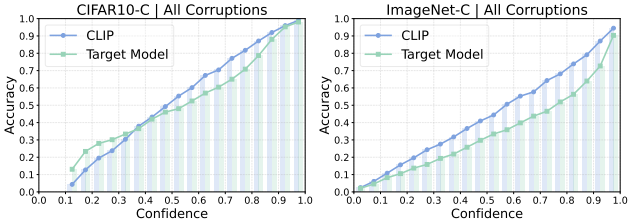

Figure 7. MSP-Accuracy Binning Plots on CIFAR-10-C and ImageNet-C. The plots aggregate results across all 15 corruption types. The x-axis represents the MSP-based confidence bins, and the y-axis represents the average accuracy within each bin. The strong alignment in the high-confidence region validates the assumption that high MSP implies high accuracy.

Results and Analysis. Figure 7 illustrates the relationship between confidence and accuracy for both the target model and CLIP. We observe a consistent, monotonic positive correlation between MSP and classification accuracy across both datasets and models. This indicates that MSP serves as an effective proxy for the model's correctness likelihood, even under severe distribution shifts.

Crucially, the results demonstrate that in the high-confidence regime, the models exhibit extremely high reliability. Specifically, as the confidence score approaches 1.0, the accuracy asymptotically approaches 100%. For instance, on CIFAR-10-C, samples with confidence scores above 0.95 achieve near-perfect accuracy for both models. This empirical observation strongly supports our theoretical premise: there exists a confidence threshold μ above

Table 7. **Comparison results under corruption scenarios on CIFAR-100-C with ResNet50 as backbone.** Classification accuracy of the standard CIFAR-100 → CIFAR-100-C online continual test-time adaptation task while continually adapting to different corruptions at the highest severity 5. The best and second-best results are highlighted in **bold** and underlined, respectively.

CIFAR-100-C ResNet50-BN	Venue	Noise			Blur				Weather				Digital				Avg.
		gauss.	*shot*	*impul.*	*defoc.*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*brit.*	*contr.*	*elastic*	*pixel*	*jpeg*	
	Time	→															
Source	-	10.06	11.67	6.67	31.87	17.52	37.58	36.61	39.01	27.27	29.71	60.98	16.87	41.92	19.26	41.79	28.59
BN Adapt	NIPS'20	32.13	32.92	28.77	59.37	34.97	56.05	59.35	46.87	46.29	49.44	62.46	58.11	46.09	47.99	40.60	$46.76_{\pm 0.01}$
Tent	ICLR'21	33.39	36.81	33.45	60.64	39.64	58.25	61.97	51.05	51.44	52.87	64.66	57.47	51.40	55.19	48.78	$50.47_{\pm 0.06}$
MEMO	NIPS'22	14.71	17.19	14.01	42.71	23.79	46.89	47.37	46.68	38.83	36.77	64.67	27.30	45.29	27.96	43.59	$35.85_{\pm 0.03}$
EATA	ICML'22	34.62	40.29	36.22	58.70	41.26	56.63	60.51	51.09	51.97	51.99	63.10	56.48	51.44	55.75	50.49	$50.70_{\pm 0.07}$
SAR	ICLR'23	33.64	37.25	33.33	59.05	38.27	56.15	59.34	48.98	49.04	50.65	62.21	55.52	49.28	52.59	46.67	$48.80_{\pm 0.15}$
DeYO	ICLR'24	37.07	45.81	39.81	58.39	42.80	56.43	60.03	50.95	51.82	51.97	61.09	55.36	50.82	54.86	49.51	$51.11_{\pm 0.51}$
CoTTA	CVPR'22	31.45	31.20	27.65	58.68	33.53	55.36	58.36	45.61	44.69	46.45	61.70	57.88	44.33	44.89	38.62	$45.36_{\pm 0.13}$
NOTE	NIPS'22	23.57	35.33	25.67	16.40	28.11	38.28	53.01	36.64	47.34	36.72	57.57	38.90	34.10	25.18	34.90	$35.45_{\pm 0.14}$
RoTTA	CVPR'23	27.09	26.39	25.46	49.02	27.59	47.82	52.86	39.05	33.46	40.71	56.34	36.37	41.80	40.11	38.20	$38.82_{\pm 0.19}$
SANTA	TMLR'23	33.60	37.42	33.60	60.39	39.17	57.04	60.43	50.29	50.40	51.61	64.27	56.87	49.85	53.12	47.61	$49.71_{\pm 0.33}$
ViDA	ICLR'24	32.25	33.38	29.70	59.64	36.06	57.14	60.28	47.74	47.49	50.84	63.30	58.93	48.18	50.44	43.98	$47.96_{\pm 0.09}$
CLIP	ICML'21	36.14	38.31	43.25	43.00	24.27	45.95	49.14	54.44	57.12	39.57	65.74	38.00	37.49	46.45	42.26	44.08
Ours	-	**51.70**	**58.43**	**57.45**	**66.69**	**47.08**	**66.37**	**70.56**	**66.22**	**68.37**	**61.79**	**76.91**	**64.91**	**57.93**	**65.11**	**59.68**	$\mathbf{62.61}_{\pm 0.13}$

which the probability of an incorrect prediction is negligible (bounded by a small α). This also justifies our strategy of utilizing high-MSP predictions to construct a reliable blended teacher signal.

C. Theoretical Analysis

Building upon the empirical observations in Appendix B, we provide a theoretical justification for the robustness of our proposed CoDiRe framework. Specifically, we analyze why the MSP-based dynamic weighting mechanism effectively filters out noise and prioritizes accurate predictions (experts) over incorrect ones (non-experts).

C.1. Premise and Problem Setup

Consider a set of M models $\{f_j\}_{j=1}^{M}$ operating on an input x with ground-truth label y. In the context of CoDiRe, this set typically comprises the target model and the VLM teacher (i.e., $M = 2$). Let $\text{Conf}_j(x) \in [0, 1]$ denote the confidence (MSP) of the j-th model. We define the set of *expert models* for a given sample x as $\mathcal{J}(x) = \{j \mid \arg\max f_j(x) = y\}$, and the set of *non-expert models* as $\mathcal{J}^c(x)$.

Our analysis rests on the empirical finding that high confidence is a reliable indicator of correctness. As demonstrated in Figure 7, when the confidence exceeds a certain threshold μ, the error rate is bounded by a negligible margin α. Formally:

Assumption 1 (Empirical Error Bound) *There exists a global confidence threshold $\mu \in (0, 1)$ and a small error bound $\alpha \approx 0$, such that for any model prediction with confidence Conf $\geq \mu$, the probability of error is bounded by:*

$$P(\hat{Y} \neq Y \mid \textit{Conf} \geq \mu) \leq \alpha. \tag{12}$$

C.2. Dynamic Weighting Analysis

The core mechanism of CoDiRe utilizes a soft weighting scheme (Eq. 2 in the main text) to fuse model outputs. We first establish that this mechanism inherently favors models with higher confidence.

Lemma 1 (Confidence-Driven Expert Dominance) *Assume that for a given input x, expert models exhibit higher confidence than non-expert models (Confidence Dominance), i.e., $\textit{Conf}_j(x) \geq \textit{Conf}_k(x)$ for all $j \in \mathcal{J}(x)$ and $k \in \mathcal{J}^c(x)$. Then, the weighting mechanism assigns strictly higher importance to the experts:*

$$\lambda_j(x) \geq \lambda_k(x). \tag{13}$$

Proof C.1 *Recall the weighting function $\lambda_i(x) = \frac{\exp(\textit{Conf}_i(x))}{Z}$, where Z is the normalization constant. Since the exponential function $g(z) = e^z$ is strictly monotonically increasing, the condition $\textit{Conf}_j(x) \geq \textit{Conf}_k(x)$ directly implies $\exp(\textit{Conf}_j(x)) \geq \exp(\textit{Conf}_k(x))$. Dividing by the positive constant Z preserves the inequality, yielding $\lambda_j(x) \geq \lambda_k(x)$.*

Lemma 1 confirms that our blending strategy effectively aligns the teacher signal with the more confident model. However, this relies on the *Confidence Dominance* assumption. Is it reasonable to assume experts are more confident? We answer this via a probabilistic analysis derived from Assumption 1.

Let E denote the event that a model is an expert ($\hat{Y} = Y$), N denote it is a non-expert, and H denote the event of high confidence (Conf $\geq \mu$). From Assumption 1, we have $P(N \mid H) \leq \alpha$, which implies $P(E \mid H) \geq 1 - \alpha$.

Lemma 2 (Probabilistic Justification) *The likelihood of an expert model producing a high-confidence prediction is*

Table 8. **Comparison results under corruption scenarios on CIFAR-10-C with ResNet26 as backbone.** Classification accuracy of the standard CIFAR-10 → CIFAR-10-C online continual test-time adaptation task while continually adapting to different corruptions at the highest severity 5. The best and second-best results are highlighted in **bold** and underlined, respectively.

CIFAR-10-C ResNet26-BN	Venue	Noise			Blur			Weather				Digital					Avg.
		gauss.	*shot*	*impul.*	*defoc.*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*brit.*	*contr.*	*elastic*	*pixel*	*jpeg*	
		Time ——————————————————————————>															
Source	-	26.89	33.22	30.54	59.83	48.12	62.12	61.96	74.51	60.68	60.55	89.90	44.77	73.25	38.57	69.08	55.60
BN Adapt	NIPS'20	60.70	63.02	53.90	82.69	58.89	80.18	82.25	74.83	74.57	79.31	85.97	82.28	71.08	73.42	64.53	$72.51_{\pm 0.05}$
Tent	ICLR'21	62.04	66.34	58.61	83.82	62.88	81.50	83.80	77.81	78.05	80.41	87.06	82.23	75.01	78.65	71.21	$75.29_{\pm 0.07}$
MEMO	NIPS'22	39.15	46.58	49.22	70.52	56.15	71.22	70.98	79.91	72.12	70.09	91.21	62.67	76.35	45.51	72.59	$64.95_{\pm 0.02}$
EATA	ICML'22	60.70	63.02	53.90	82.69	58.88	80.18	82.26	74.84	74.56	79.31	85.97	82.28	71.07	73.43	64.54	$72.51_{\pm 0.05}$
SAR	ICLR'23	61.84	66.34	58.28	84.01	62.92	81.91	84.17	78.18	78.66	81.12	87.60	82.93	75.86	79.18	71.96	$75.66_{\pm 0.07}$
DeYO	ICLR'24	64.18	71.82	64.57	85.01	67.37	83.28	86.03	81.50	82.75	83.51	89.87	86.96	78.27	83.55	75.69	$78.96_{\pm 0.22}$
CoTTA	CVPR'22	60.70	63.02	53.90	82.69	58.89	80.18	82.25	74.83	74.57	79.31	85.97	82.28	71.07	73.42	64.53	$72.51_{\pm 0.05}$
NOTE	NIPS'22	50.44	66.76	52.72	32.81	52.71	62.39	75.44	73.77	77.75	61.26	87.10	69.80	61.14	48.26	61.49	$62.26_{\pm 0.09}$
RoTTA	CVPR'23	56.20	59.40	51.21	76.48	56.55	76.48	80.28	70.90	62.35	71.13	80.82	50.34	64.02	62.88	59.24	$65.22_{\pm 0.25}$
SANTA	TMLR'23	63.84	69.78	62.34	83.91	64.95	80.98	83.26	79.02	79.20	80.39	87.62	80.93	75.60	79.67	74.79	$76.42_{\pm 0.34}$
ViDA	ICLR'24	60.73	63.19	54.26	82.92	59.36	80.47	82.58	75.37	75.27	79.78	86.33	82.39	72.28	74.55	66.01	$73.03_{\pm 0.05}$
CLIP	ICML'21	65.39	66.64	76.02	75.75	48.18	78.16	79.08	81.92	84.68	76.14	90.40	80.47	64.74	76.93	71.32	74.39
Ours	-	**76.21**	**82.83**	**81.18**	**88.67**	**74.43**	**88.07**	**90.43**	**88.97**	**90.68**	**88.89**	**94.57**	**91.70**	**83.43**	**89.02**	**83.16**	$\mathbf{86.15}_{\pm 0.01}$

significantly higher than that of a non-expert model. Specifically, the likelihood ratio satisfies:

$$\frac{P(H \mid E)}{P(H \mid N)} \geq \frac{(1-\alpha) \cdot P(N)}{\alpha \cdot P(E)}, \tag{14}$$

where $P(E)$ and $P(N)$ are the global accuracy and error rates of the model, respectively.

Proof C.2 *Applying Bayes' Theorem to expand the ratio:*

$$\frac{P(H \mid E)}{P(H \mid N)} = \frac{P(E \mid H)P(H)/P(E)}{P(N \mid H)P(H)/P(N)} = \frac{P(E \mid H) \cdot P(N)}{P(N \mid H) \cdot P(E)}. \tag{15}$$

Substituting the bounds derived from Assumption 1 ($P(E \mid H) \geq 1-\alpha$ and $P(N \mid H) \leq \alpha$):

$$\frac{P(H \mid E)}{P(H \mid N)} \geq \frac{(1-\alpha) \cdot P(N)}{\alpha \cdot P(E)}. \tag{16}$$

Remark. Given that $\alpha \approx 0$ in our experiments (as seen in the high-confidence bins of Figure 7), the term $\frac{1-\alpha}{\alpha}$ becomes extremely large. This indicates that $P(H \mid E) \gg P(H \mid N)$. In other words, statistically, high confidence is overwhelmingly generated by expert models. This provides a strong theoretical foundation for the Confidence Dominance assumption used in Lemma 1, validating the robustness of CoDiRe's blended teacher construction.

D. Experimental Details of Datasets, Baselines, and Hyperparameters

D.1. Datasets

In this paper, we evaluate the proposed method on widely recognized benchmarks that are designed to assess model robustness under distribution shifts. These benchmarks include CIFAR-10-C, ImageNet-C, Office-Home, and PACS. Moreover, we provide experimental results on the CIFAR-100-C dataset in Appendix F.1. They encompass two types of adaptation scenarios: corruptions and domain generalizations. Below, we provide an overview of each dataset.

CIFAR-10-C, CIFAR-100-C, and ImageNet-C. CIFAR-10-C, CIFAR-100-C, and ImageNet-C are derived from their respective base datasets, CIFAR-10, CIFAR-100, and ImageNet, by applying systematic corruptions. These datasets feature 15 corruption types, including Gaussian noise, shot noise, impulse noise, defocus blur, motion blur, zoom blur, frost, fog, brightness, contrast, elastic transformations, pixelation, and JPEG compression. Each corruption type is categorized into five severity levels, representing progressively more challenging distributional shifts. These datasets are widely used to benchmark model robustness against common corruptions and noise. An illustration of these corruptions is shown in Figure 8. We consistently choose the highest level 5 in this paper.

Office-Home. Office-Home is a domain adaptation dataset consisting of images from four domains: Art, Clipart, Product, and Real-World. It contains 65 object categories commonly encountered in office and home environments, such as "desk," "keyboard," and "backpack." The dataset is designed to evaluate domain adaptation methods under significant domain gaps, with a focus on generalizing from one domain to unseen domains. In this paper, we utilize "Art" as the source domain, with the other three domains as target domains.

PACS. PACS is a domain generalization benchmark that includes images from four distinct domains: Paintings, Artistic images, Cartoons, and Sketches. The dataset spans seven

Figure 8. **Illustration of ImageNet-C under 5 level of severity.** The dataset showcases 15 types of algorithmically generated corruptions across four categories: noise, blur, weather, and digital. Each corruption type is illustrated at five increasing levels of severity, demonstrating the progressive impact of these corruptions.

object categories shared across all domains, including "dog," "guitar," and "person." Each domain exhibits unique visual characteristics, leading to significant domain shifts. The primary evaluation task involves training on three domains and testing on the held-out domain, providing a rigorous assessment of a model's ability to generalize to unseen distributions. In this paper, we utilize "Art" as the source domain, with the other three domains as target domains.

D.2. Baselines

To validate the effectiveness of CoDiRe, we compare it against a comprehensive set of baselines, TTA methods, CTTA methods, VLM-TTA methods, and TTD methods. Below, we summarize these baselines, along with their configurations in this paper for reproducibility.

TTA Methods. TTA methods include:

- **Source (No Adaptation):** A baseline that directly uses the pre-trained model for inference on test data without any adaptation.
- **BN Adapt:** This method replaces Batch Normalization (BN) statistics with those computed from the current test batch, commonly referred to as Target Batch Normalization (TBN).
- **Tent:** Tent uses entropy minimization as a self-supervised loss to encourage the model to adapt to the target domain.
- **MEMO:** MEMO improves model robustness by averaging probabilities across multiple augmented views of the same input. After adapting to each batch, the model is recovered to the pre-trained, source version, making the adaptation process episodic. The augmentation size is set to 32, and the moving average rate for updating the teacher model is set to 0.999.
- **EATA:** EATA mitigates noisy gradients by introducing entropy-based filtering and weighting strategies. The entropy threshold E_0 is set to $\texttt{log}(K) \times 0.4$, where K is the number of classes, and the threshold ϵ for filtering redundant samples is set to 0.05. We use fisher regularizer here as default.
- **SAR:** SAR addresses noisy gradients in challenging scenarios such as small batch sizes and mixed distributions. It uses an entropy threshold $E_0 = \texttt{log}(K) \times 0.4$ and a

Table 9. **Comparison results under corruption scenarios on CIFAR-10-C with ViT-S/16 as backbone.** Classification accuracy of the standard CIFAR-10 → CIFAR-10-C online continual test-time adaptation task while continually adapting to different corruptions at the highest severity 5. The best and second-best results are highlighted in **bold** and underlined, respectively.

CIFAR-10-C VitSmall-LN	Venue	Noise			Blur				Weather				Digital				Avg.
		gauss.	*shot*	*impul.*	*defoc.*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*brit.*	*contr.*	*elastic*	*pixel*	*jpeg*	
	Time	→															
Source	-	66.53	71.67	82.21	94.18	77.43	89.42	95.08	94.50	92.65	87.15	97.11	89.97	87.27	75.56	84.51	85.68
Tent	ICLR'21	69.98	77.88	84.47	94.57	78.50	90.75	95.87	94.72	93.22	89.45	97.08	92.39	88.66	69.89	85.32	$86.85_{\pm 0.03}$
MEMO	NIPS'22	69.43	74.23	82.74	94.46	78.36	89.92	95.27	94.71	93.25	88.68	97.39	91.29	88.14	79.26	85.78	$86.86_{\pm 0.02}$
EATA	ICML'22	66.54	71.71	82.22	94.18	77.43	89.42	95.09	94.50	92.66	87.16	97.11	89.97	87.27	75.55	84.53	$85.69_{\pm 0.00}$
SAR	ICLR'23	70.06	78.20	84.61	94.63	78.42	91.01	95.88	94.80	93.45	89.44	97.07	92.29	88.76	67.49	85.12	$86.75_{\pm 0.05}$
DeYO	ICLR'24	74.77	81.81	86.89	**94.68**	77.87	92.12	95.95	95.04	94.11	92.48	96.73	95.61	87.22	85.67	83.78	$88.98_{\pm 0.03}$
CoTTA	CVPR'22	66.53	71.67	82.21	94.18	77.43	89.42	95.08	94.50	92.65	87.15	97.11	89.97	87.26	75.56	84.51	$85.68_{\pm 0.00}$
NOTE	NIPS'22	67.36	73.74	83.03	94.29	78.11	89.97	95.39	94.67	93.02	87.51	97.25	90.57	87.92	73.60	85.25	$86.11_{\pm 0.00}$
RoTTA	CVPR'23	66.62	71.64	83.32	94.34	76.20	90.14	95.11	94.03	90.94	83.74	96.46	89.08	85.78	72.30	71.71	$84.09_{\pm 0.12}$
SANTA	TMLR'23	66.03	70.73	83.47	94.29	77.63	89.86	95.65	94.67	92.29	86.79	96.84	89.77	88.78	75.34	83.17	$85.69_{\pm 0.06}$
ViDA	ICLR'24	66.57	71.50	82.53	94.24	77.34	89.60	95.17	94.51	92.76	87.40	97.23	90.56	87.58	75.97	84.69	$85.84_{\pm 0.06}$
DPCore	ICLR'25	81.07	84.61	**88.52**	94.54	82.01	91.76	95.65	94.79	94.37	91.59	96.83	94.84	89.17	92.21	86.01	$90.53_{\pm 0.04}$
CLIP	ICML'21	65.39	66.64	76.02	75.75	48.18	78.16	79.08	81.92	84.68	76.14	90.40	80.47	64.74	76.93	71.32	74.39
Ours	-	**83.55**	**87.63**	88.21	94.51	**84.26**	**93.39**	**96.62**	**95.99**	**95.52**	**93.20**	**97.71**	**96.24**	**90.90**	**93.49**	**88.22**	$\mathbf{91.96}_{\pm 0.04}$

model recovery threshold $e_m = 0.2$.

- **DeYO:** DeYO combines Pseudo-Label Probability Difference (PLPD) with entropy-based filtering for high-quality sample selection. The entropy threshold τ_{Ent}, PLPD threshold τ_{PLPD}, and Ent_0 are set to $\texttt{log}(K) \times 0.4$, $\texttt{log}(K) \times 0.5$, and 0.3, respectively. The other configuration is kept the same as the original paper.

CTTA Methods. CTTA methods include:

- **CoTTA:** A method for handling continuous domain shifts by leveraging temporal consistency and pseudo-label refinement. We set the augmentaion size to 32, the stochastic rate as 0.01, and the weight for the teacher model as 0.999. As we find that CoTTA is hyperparameter-sensitive, we set the confidence threshold to 0.62 in CIFAR-10-C, 0.52 in CIFAR-100-C, 0.1 in ImageNet-C, and 0.5 in other datasets.
- **NOTE:** Addresses temporally correlated test streams via Instance-Aware Batch Normalization (IABN) and Prediction-Balanced Reservoir Sampling (PBRS), which simulates i.i.d. sampling from non-i.i.d. streams using a memory bank.
- **RoTTA:** A method that tackles dynamic distribution shifts via robust statistics estimation and Category-balanced Sampling with Timeliness and Uncertainty (CSTU). We use the Adam optimizer here, and replace the standard Robust Batch Normalization (RBN) layers with learnable LN layers for ViT backbones.
- **SANTA:** A source anchoring network designed for target alignment in dynamic environments. As the original version of SANTA requires access to the source domain samples, we use a dynamically updated version of the prototype instead.
- **ViDA:** This approach focuses on homeostatic adaptation to balance stability and plasticity during continuous adaptation. All the configuration is kept the same as the original paper.
- **DPCore:** DPCore manages domain knowledge via a dynamic prompt coreset, utilizing Visual Prompt Adaptation (VPA) to align domains efficiently. We use the AdamW optimizer, with the prompt length set to 8 and the update threshold ρ set to 0.8.

VLM-TTA Methods. VLM-TTA methods include:

- **TPT:** A method that uses prompt tuning to improve generalization during test-time adaptation. We set the augmentation size to 64, and confidence threshold ρ to 0.1.
- **TDA:** A training-free approach that dynamically updates a cache with high-quality features from historical test data. All the configuration is kept the same as the original paper.
- **BoostAdapter:** An adapter-based approach that bootstraps VLM performance through regional feature refinement. All the configuration is kept the same as the original paper.
- **ZERO:** A training-free approach that simply zeroing out the Softmax temperature over augmented views and performing majority voting on confident predictions. All the configuration is kept the same as the original paper.

TTD Methods. TTD methods refer to the variants under our proposed Test-Time Distillation paradigm. They include:

- **Naive Ensemble:** A baseline that computes the inference output by simply averaging the logits of the target model and the frozen CLIP without updating any parameters.
- **BN Adapt w. NE:** This method performs BN Adapt (updating BN statistics on the test stream) on the target model, and uses the Naive Ensemble strategy for the final inference output.
- **Tent w. NE:** This method performs Tent (entropy mini-

Table 10. **Comparison results under corruption scenarios on ImageNet-C with ResNet26 as backbone.** Classification accuracy of the standard ImageNet → ImageNet-C online continual test-time adaptation task while continually adapting to different corruptions at the highest severity 5. The best and second-best results are highlighted in **bold** and underlined, respectively.

ImageNet-C ResNet26-BN	Venue	Noise			Blur			Weather				Digital					Avg.
		gauss.	*shot*	*impul.*	*defoc.*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*brit.*	*contr.*	*elastic*	*pixel*	*jpeg*	
	Time	→															
Source	-	2.06	3.04	2.58	11.66	6.86	10.74	20.72	10.56	15.38	15.94	51.78	2.60	11.20	17.76	31.36	14.28
BN Adapt	NIPS'20	11.59	13.47	11.45	10.81	12.80	20.07	32.41	27.76	28.21	38.63	58.96	7.86	38.17	38.75	26.79	$25.18_{\pm 0.07}$
Tent	ICLR'21	11.76	13.40	12.05	11.24	12.35	19.98	32.70	27.92	28.36	39.08	59.27	8.09	39.01	40.07	28.33	$25.57_{\pm 0.09}$
MEMO	NIPS'22	1.37	2.11	1.54	9.65	8.23	12.49	23.04	16.01	18.44	18.63	51.25	3.29	15.38	24.00	30.19	$15.71_{\pm 0.14}$
EATA	ICML'22	10.91	13.32	11.90	10.93	12.61	19.96	33.05	28.60	29.17	40.42	60.57	9.03	39.71	41.15	29.65	$26.07_{\pm 0.09}$
SAR	ICLR'23	11.15	13.03	11.33	10.77	12.37	19.89	32.89	28.17	28.67	38.45	59.84	8.51	39.28	39.97	27.61	$25.46_{\pm 0.14}$
DeYO	ICLR'24	11.47	13.20	11.86	10.71	12.55	19.29	33.11	28.97	29.81	42.01	59.39	10.15	39.78	41.93	32.65	$26.46_{\pm 0.14}$
CoTTA	CVPR'22	11.46	12.72	11.21	9.97	11.04	16.68	30.03	26.47	26.17	35.70	59.65	7.82	37.29	38.41	25.76	$24.03_{\pm 0.08}$
NOTE	NIPS'22	3.88	7.66	13.60	0.09	2.53	3.16	11.99	11.62	21.53	11.22	47.89	2.48	16.79	14.87	25.82	$13.01_{\pm 0.13}$
RoTTA	CVPR'23	9.42	14.21	13.89	9.47	11.83	16.97	30.91	13.66	23.61	30.81	56.82	3.01	25.55	15.44	25.58	$20.08_{\pm 0.36}$
SANTA	TMLR'23	11.20	13.30	11.40	11.39	12.28	19.77	32.58	28.19	29.27	38.81	59.45	8.05	38.45	39.91	28.06	$25.47_{\pm 0.12}$
ViDA	ICLR'24	11.88	13.13	11.89	10.87	12.19	19.70	33.18	27.82	29.22	40.23	59.92	9.55	39.45	41.25	30.15	$26.03_{\pm 0.04}$
CLIP	ICML'25	22.94	23.20	24.06	31.50	19.80	35.84	33.58	45.00	39.34	47.26	62.88	34.54	25.74	50.68	42.02	35.89
Ours	-	**27.36**	**29.55**	**30.10**	**35.59**	**25.77**	**42.13**	**43.97**	**52.17**	**48.17**	**57.73**	**72.38**	**39.71**	**41.95**	**60.57**	**52.39**	$\mathbf{43.97}_{\pm 0.04}$

mization) on the target model parameters, and utilizes the Naive Ensemble strategy for the final inference output.

- **Distill CLIP:** A direct distillation baseline where the target model is updated to match the predictions of the frozen CLIP model.

D.3. Hyperparameters

To ensure fair and reproducible comparisons, we use consistent experimental settings across all methods. Below, we summarize the hyperparameters for each dataset and scenario:

- **CIFAR-10-C and CIFAR-100-C:** We use ResNet50 as the default backbone of target model, and also provide the results of ResNet26 and ViT/S-16 in Appendix F. The learning rate is set to 1×10^{-4}.
- **ImageNet-C:** ViT-B/16 is used as the backbone of target model. Similarly, the results of ResNet26 and ResNet50 are also supplemented in Appendix F. The learning rate is set to 1×10^{-3} in ResNet architectures and 2.5×10^{-4} in ViT architectures.
- **PACS and Office-Home:** ResNet50 is used as the backbone of target model, with a learning rate of 1×10^{-4}.

E. Implementation Details

All experiments are conduced on the TTAB framework for fair evaluation, run on a single NVIDIA Tesla V100 GPU, and repeated three times with random seeds in the ranges [2022, 2023, 2024]. All the baselines' algorithmic settings (e.g., hyper-parameters) are set to their default values unless otherwise specified. As for our method, we set the reset threshold γ_0 to 0.25, step size s to 20 steps, and reset ratio α to 20(%). Unless otherwise specified, all experiments were conducted using SGD as the optimizer, with the batch size set to 64. Source code used in this paper is under the MIT License.

F. Additional Experiments and Analyses

To further validate the robustness and versatility of CoDiRe, this section provides additional comparative results across different datasets and variations in the target model's architecture. These experiments supplement the main paper's findings, offering a broader view of CoDiRe's performance against baseline methods under diverse conditions.

F.1. More Comparison Results

Comparison Results on CIFAR-100-C. We extend our evaluation to the CIFAR-100-C benchmark, which presents a more challenging corruption adaptation task due to its significantly larger number of classes compared to CIFAR-10-C. Table 7 details the performance of CoDiRe against various TTA and CTTA methods, using ResNet50 as the backbone. The results unequivocally demonstrate CoDiRe's superior robustness, maintaining high accuracy even in this complex scenario where baselines struggle to distinguish between fine-grained categories.

More Comparison Results on CIFAR-10-C. To verify the model-agnostic nature of CoDiRe, we assess its performance on CIFAR-10-C using different backbones for the target model. Table 8 shows the substantial gains with a ResNet26 backbone, while Table 9 presents results with a ViT-S/16 backbone. These outcomes confirm that CoDiRe is not reliant on specific architectures and consistently delivers impressive improvements across varying model capacities.

More Comparison Results on ImageNet-C. Similarly, we demonstrate the seamless scalability of our method on the large-scale ImageNet-C benchmark. Table 10 details

Table 11. **Comparison results under corruption scenarios on ImageNet-C with ResNet50 as backbone.** Classification accuracy of the standard ImageNet → ImageNet-C online continual test-time adaptation task while continually adapting to different corruptions at the highest severity 5. The best and second-best results are highlighted in **bold** and underlined, respectively.

ImageNet-C ResNet50-BN	Venue	Noise			Blur				Weather				Digital				Avg.
		gauss.	*shot*	*impul.*	*defoc.*	*glass*	*motion*	*zoom*	*snow*	*frost*	*fog*	*brit.*	*contr.*	*elastic*	*pixel*	*jpeg*	
	Time	→															
Source	-	18.32	18.88	17.70	16.72	8.70	19.02	25.76	27.64	30.00	35.22	63.36	20.94	12.16	25.58	47.88	25.86
BN Adapt	NIPS'20	10.12	12.36	10.18	9.70	10.10	20.28	34.46	35.50	38.06	53.18	65.88	16.28	36.66	34.82	30.74	$27.89_{\pm 0.03}$
Tent	ICLR'21	10.73	13.75	11.26	9.92	11.18	21.09	34.66	36.71	40.97	55.73	66.64	19.71	39.39	37.51	37.86	$29.81_{\pm 0.20}$
MEMO	NIPS'22	14.47	16.67	14.83	13.24	11.14	18.79	26.99	31.86	32.87	23.12	61.01	2.39	19.71	31.43	46.34	$24.32_{\pm 0.02}$
EATA	ICML'22	10.32	14.11	12.37	10.26	11.68	20.55	33.90	37.31	40.08	52.98	64.45	22.59	40.08	41.89	41.55	$30.28_{\pm 1.06}$
SAR	ICLR'23	11.15	13.83	11.09	9.79	11.03	19.90	34.83	37.44	39.73	53.85	66.01	19.29	38.79	38.21	36.01	$29.40_{\pm 0.01}$
DeYO	ICLR'24	9.44	14.06	12.40	10.52	12.42	21.28	35.22	38.02	42.10	54.40	64.54	24.72	41.92	42.34	41.76	$31.01_{\pm 0.03}$
CoTTA	CVPR'22	11.04	13.68	11.41	9.81	10.27	20.30	33.82	35.87	38.25	53.87	65.21	16.22	37.24	34.81	31.63	$28.23_{\pm 0.12}$
NOTE	NIPS'22	22.76	21.65	22.33	0.16	2.61	6.20	18.98	23.91	39.02	26.41	62.79	17.09	17.41	13.49	33.33	$21.88_{\pm 0.24}$
RoTTA	CVPR'23	12.13	11.94	11.91	1.59	4.89	11.87	32.90	30.25	38.86	45.83	64.10	18.98	25.49	26.25	33.68	$24.71_{\pm 0.73}$
SANTA	TMLR'23	10.42	14.21	10.65	9.64	10.97	20.87	35.18	37.12	40.59	55.15	65.75	18.38	39.03	36.79	36.01	$29.38_{\pm 0.08}$
ViDA	ICLR'24	10.65	13.67	10.45	10.08	10.77	20.97	34.21	36.42	38.95	54.42	65.89	17.83	37.65	34.49	32.71	$28.61_{\pm 0.12}$
CLIP	ICML'21	22.94	23.20	24.06	31.50	19.80	35.84	33.58	45.00	39.34	47.26	62.88	34.54	25.74	50.68	42.02	35.89
Ours	-	**27.57**	**31.39**	**32.01**	**33.74**	**27.13**	**42.20**	**48.11**	**56.02**	**55.20**	**64.28**	**75.19**	**44.03**	**50.61**	**62.15**	**57.81**	**47.16**$_{\pm 0.47}$

performance using ResNet26, and Table 11 provides results when employing ResNet50, complementing the ViT-B/16 results presented in the main paper. These results confirm CoDiRe's consistent superiority and stability, proving its effectiveness regardless of the backbone architecture size on challenging large-scale benchmarks. Moreover, the smaller the target model is, the greater it can benefit from distillation. This is quite natural phenomenon.