TET: Introducing Energy To Test-Time Training

Julian Bibo
Universiteit van Amsterdam
julian.bibo@student.uva.nl

Jan Hutter
Universiteit van Amsterdam
jan.hutter@student.uva.nl

Brandon Li
Universiteit van Amsterdam
brandon.li@student.uva.nl

Henk Schaapman
Universiteit van Amsterdam
henk.schaapman@student.uva.nl

Abstract

Test-Time Adaptation (TTA) methods aim to improve model performance under distribution shifts. In particular, Test-time Energy Adaptation (TEA) integrates Energy-Based Model principles into the adaptation process. In this work, we attempt to validate TEA's methodology by reproducing its results. Additionally, we propose Test-time Energy Training (TET), a hybrid TTA approach inspired by TEA and Test-Time Training (TTT). To assess its performance, we benchmark it on corrupted datasets. Furthermore, we extend the application of TEA by successfully adapting it to a Vision Transformer. We are able to confirm TEA's strong performance. While TET is outperformed by TEA, it demonstrates the potential of incorporating pretraining-based objectives into TEA.

1 Introduction

TTA addresses distributional shifts that arise when pre-trained models are deployed in real-world environments, where the test (target) data distribution can differ significantly from the training (source) distribution. While current TTA methods adapt effectively to shifts in marginal distributions, they often fall short in correcting covariate shifts [13].

A potential solution is TEA [13], which incorporates EBMs [2] into the TTA framework and has shown promising performance. However, TEA is limited in several ways: it applies EBMs only in a post-training context, restricts adaptation to batch normalization (BN) layers, and is evaluated exclusively on WideResNet (WRN), a legacy architecture. As a result, the full potential of EBMs within TTA remains underexplored.

We aim to reproduce the results from the TEA paper [13] to validate their findings, and we propose two extensions to address its current limitations. For the first extension, we draw inspiration from TTT [9], which employs a self-supervised auxiliary task during pre-training for improved adaptation. We propose Test-time Energy Training (TET), a method that reformulates the auxiliary task as an unsupervised EBM objective, enabling feature-layer adaptation guided by energy-based signals that update all model parameters. Specifically, we aim to answer the following questions:

RQ 1: To what extent can we reproduce the results presented in the TEA paper [13]?

RQ 2: How does incorporating energy-based objectives during pre-training affect generalization at test time?

RQ 3: How does the TEA framework generalize to a modern architecture such as the Vision Transformer (ViT)?

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

2 Scope of Reproducibility

While the code of the original experiments is available, independent reproduction can help in the verification of results and ensure that the authors' results are not setup or hyperparameter dependent. The main contribution of the original paper [13] is the introduction of EBMs for online TTA. Their introduced TEA method achieves state-of-the-art (SOTA) generalization performance on several datasets and corruption types. In this study, we aim to reproduce the following claims made in the original paper:

Claim 1: SOTA performance. TEA achieves SOTA generalization performance across diverse models and datasets.

Claim 2: Target data modeling. TEA truly models test data distribution, enabling a model to generate samples representative to the test dataset, without provoking recollections from the train data.

3 Methodology

3.1 Energy Based Models

As mentioned in [2] and [13], EBMs reinterpret the output logits to assign a scalar to each input. This energy function is defined as follows: $E_\theta(x) = -\log \sum_y \exp(f_\theta(x)[y])$. Using this formula, any classifier can be treated as an EBM. We aim to minimize the energy of real inputs, and maximize that of fake inputs ("samples"). This is done using contrastive divergence [11], resulting in an energy-based loss function, $\mathcal{L}_{energy} = E_\theta(x_{test}) - E_\theta(\tilde{x})$. A detailed implementation of this loss can be found in [13].

3.2 Extensions

Test-time Energy Training. Our TET extension combines TEA's energy-based approach with the TTT [9] paradigm. Unlike in TEA, where energy loss is used only at test time, we add the aforementioned energy loss as a component to the training loss: $\mathcal{L}_{train} = \lambda_{cls}\mathcal{L}_{cls} + \lambda_{energy}\mathcal{L}_{energy}$, with λ_{cls} and λ_{energy} as scalars, essentially using the energy loss for the auxiliary task as done in TTT. At test time, the model is adapted like in TEA. That is, the model's weights are updated based on the sampling of fake images and subsequently computing the contrastive divergence. Our approach generates an energy landscape during training, with the aim of enabling easier energy-based adaptation at test time. We adapt all parameters using the combined loss during pretraining, and (like in TEA) only the BN layers at test-time. More details on TET and the used hyperparameters can be found in Appendix A. To enable comparison with other TTA methods that incorporate a pretraining stage, we include results for the WRN28-10-BN version of the TTT model [9], which uses a rotation-prediction auxiliary task, as an additional baseline.

Vision Transformer. We extend the TEA approach to ViTs, incorporating the Source, ETA, and EATA baselines. Using a ViT could lead to better performance as it generally outperforms traditional Convolutional Neural Networks (CNNs). The results demonstrate the generalization capabilities of TEA across different architectures and highlight the comparative performance of ViTs relative to ResNets. TET and TTT results are omitted in this section due to computational constraints and because our focus is primarily on the generalization behavior of the TEA method. The implementation provided in [1] is followed for the ViT-B/16 model. Similar to [1], pretrained weights on ImageNet-21K are finetuned on the first two datasets, CIFAR-10 and CIFAR-100.

4 Experimental Results

This section outlines the setup used in our experiments and the corresponding results. Our research follows the general implementation of [13]. Our hyperparameters for replicating the results can be found in Appendix A. More in-depth information about all other settings can be found in our GitHub repository[1].

[1] `https://anonymous.4open.science/r/dl2-B412/README.md`

Baselines. The performance of TEA is compared to eight state-of-the-art test-time adaptation methods which are either (1) Normalization based: BN [8] and DUA [10]; (2) Entropy-based: TENT [12], ETA [6], EATA [6], and SAR [7]; or (3) Pseudo-labeling-based: PL [4] and SHOT [5]. These eight TTA approaches are evaluated alongside Source, the unaltered WRN28-10-BN model [14].

Datasets. We utilize the same three datasets as found in the main results of [13]: CIFAR-10(C), CIFAR-100(C), and Tiny-ImageNet(C) [3], each having the same 15 corruption types as the original research [13].

Metrics. The metrics used throughout the experiments are average accuracy and mean Corruption Error (mCE). The precise definition of these metrics can be found in Appendix B.

4.1 Reproduction Study Results

The results of replicating the original experiments are shown in Table 1. The complete reproduction of TEA performance results are located in Appendix C. The claims below correspond with the claims listed in Section 2.

WRN-28-10		CIFAR-10(C)					CIFAR-100(C)					Tiny-ImageNet(C)				
		Clean	Corr Severity 5		Corr Severity 1-5		Clean	Corr Severity 5		Corr Severity 1-5		Clean	Corr Severity 5		Corr Severity 1-5	
		Acc(↑)	Acc(↑)	mCE(↓)	Acc(↑)	mCE(↓)	Acc(↑)	Acc(↑)	mCE(↓)	Acc(↑)	mCE(↓)	Acc(↑)	Acc(↑)	mCE(↓)	Acc(↑)	mCE(↓)
Source		94.77	56.48	100.0	73.45	100.0	80.64	35.29	100.0	51.92	100.0	**65.0**	12.69	100.0	20.79	100.0
Norm	BN	93.97	79.57	46.95	85.63	54.13	78.23	56.17	67.74	64.47	73.89	55.0	23.08	88.1	29.18	89.4
	DUA†	-	80.10	50.78	-	-	-	-	-	-	-	-	-	-	-	-
Pseudo	PL	93.67	38.54	141.21	65.9	128.45	78.28	24.46	116.75	47.54	109.09	5.2	1.09	113.29	1.58	124.25
	SHOT	93.98	81.64	42.19	86.87	49.47	78.31	61.96	58.79	68.46	65.59	53.0	24.57	86.4	30.55	87.68
Entropy	TENT	93.83	81.42	42.7	86.75	49.92	**78.38**	61.38	59.69	68.06	66.43	55.0	24.33	86.67	30.35	87.93
	ETA	93.97	79.59	46.91	85.63	54.11	77.45	57.62	65.5	65.24	72.29	54.4	23.66	87.44	29.77	88.66
	EATA	93.97	79.59	46.91	85.63	54.11	77.57	57.65	65.45	65.39	71.98	54.6	23.68	87.42	29.75	88.7
	SAR	93.97	80.7	44.34	86.18	52.06	77.27	58.14	64.7	65.5	71.75	59.2	**24.79**	**86.14**	**30.76**	**87.42**
Energy	TEA	**94.18**	**83.31**	**38.34**	**87.87**	**45.68**	78.37	**62.52**	**57.93**	**68.9**	**64.67**	53.0	24.55	86.42	30.43	87.84
Diff		+0.09	-0.03	-5.35	-0.01	-6.32	-2.51	-2.58	+1.75	-2.32	+0.93	+1.35	-7.12	-1.57	-9.53	-4.28

Table 1: Accuracy and mCE for various TTA methods compared to the Source model. Results marked with † are taken from [10]. The "Diff" row contains the differences between our results and the original paper [13]. Values in bold indicate the best score per metric.

Claim 1: SOTA performance. From the results in Table 1 a general increase in accuracy and decrease in mCE is visible when comparing TEA with other methods. Our results on the CIFAR datasets show slight deviations from the original findings. On Tiny-ImageNet, the discrepancies are more pronounced, with TEA failing to achieve the highest accuracy across both levels of corruption severity. Nevertheless, our results confirm that TEA performs comparably to current state-of-the-art methods, confirming the authors' first claim.

Claim 2: Target data modeling. To substantiate the claim that an adapted model truly learns the test data distribution, the authors included visualizations of test data distributions, including one for CIFAR-10 (Figure 4(iii) in [13]). Despite our success reproducing TEA's numerical results, we failed to reproduce an image that shows any recognizable pattern. We suspect the authors took some undocumented steps to produce the images. Because of this, we are unable to confirm the authors' second claim. Our best efforts in reproducing the CIFAR-10 image are included in Appendix D.

4.2 Extensions Results

Test-time Energy Training. As shown in Table 2(a), TET achieves slightly lower accuracy than TEA on the corruption set of CIFAR-10, and significantly lower accuracy on the corruption set of CIFAR-100. Furthermore, TTT achieves significantly lower accuracy on the corruption sets, compared to TEA, ETA, and EATA.

Vision Transformer. As can be seen in Table 2(b), the ViT Source model demonstrates stronger generalization across various corruption types as compared the WRN Source in (a), and achieves performance comparable to TEA when it is applied to the WRN model. Furthermore, we found that TEA generalizes less effectively across architectures compared to other TTA methods, yielding lower

Method		CIFAR-10(C)					CIFAR-100(C)				
		Clean	Corr Severity 5		Corr Severity 1-5		Clean	Corr Severity 5		Corr Severity 1-5	
		Acc(↑)	Acc(↑)	mCE(↓)	Acc(↑)	mCE(↓)	Acc(↑)	Acc(↑)	mCE(↓)	Acc(↑)	mCE(↓)
(a) WideResNet28-10-BN											
Source		**94.77**	56.48	100.0	73.45	100.0	80.64	35.29	100.0	51.92	100.0
TTT		87.60	72.20	63.87	78.24	81.99	58.51	35.48	99.71	43.23	118.05
Entropy	ETA	93.97	79.59	46.91	85.63	54.11	77.45	57.62	65.5	65.24	72.29
	EATA	93.97	79.59	46.91	85.63	54.11	77.57	57.65	65.45	65.39	71.98
Energy	TEA	94.18	**83.31**	**38.34**	**87.87**	**45.68**	**78.37**	**62.52**	**57.93**	**68.9**	**64.67**
	TET	93.56	83.17	38.67	87.53	46.99	74.32	57.06	66.36	63.81	75.27
(b) ViT-B/16											
Source		**98.46**	81.61	100.0	90.42	100.0	91.59	58.07	100.0	73.65	100.0
Entropy	ETA	**98.46**	81.00	99.79	90.43	99.92	**91.63**	63.37	87.37	76.43	89.42
	EATA	**98.46**	81.65	99.79	90.43	99.92	91.60	**65.00**	**83.49**	**77.22**	**86.43**
Energy	TEA	98.41	**86.23**	**94.24**	**92.55**	**77.78**	91.41	60.76	97.01	76.01	91.03

Table 2: Accuracy and mCE for various TTA methods compared to the Source model, alongside (a) our TET method applied to WRN, and (b) TEA applied on ViT-B/16.

accuracy on the corruption set of CIFAR-100 than both ETA and EATA. This signifies that TEA does not consistently achieve the highest accuracy with TTA on corruption sets when different model architectures are used.

5 Discussion

Our reproduction of CIFAR-10 and CIFAR-100 results closely aligns with the original findings [13], confirming that TEA outperforms other TTA methods. However, we were unable to reproduce TEA's performance on Tiny-ImageNet. Despite extensive efforts, we believe that suboptimal hyperparameters during the pretraining of the WRN may have prevented TEA from achieving top performance for this dataset. Some difficulties we encountered during our experiments are listed in Appendix E.

The TET evaluation results indicate that adding a pretraining stage to TTA methods does not consistently improve performance on the CIFAR-10 and CIFAR-100 corruption sets. Both TTT and TET showed reduced accuracy compared to TEA and other baseline methods. While TET shows a slight drop in performance, it still generalizes better under corruption than ETA and EATA, highlighting the potential of EBMs for robust learning. However, TET comes with notable limitations. As noted in the JEM [2] and TEA [13] papers, EBMs can be unstable and difficult to train, which we experienced to a certain degree. Attempts to update the feature extractor often resulted in unstable behavior, partly due to the uncertain quality of EBM-generated samples. While a thorough hyperparameter search could potentially allow TET to surpass TEA, the high computational cost of training makes TET less practical. In contrast, TEA offers a stable and easily deployable solution, making it more efficient and scalable in practice.

Furthermore, applying TEA to ViT is promising, as highlighted in Section 4.2. TEA performs best on the corrupted CIFAR-10 dataset, while EATA outperforms on CIFAR-100. We believe this reflects the increasing challenge of shaping a useful energy landscape as the number of classes grows. With more classes, the landscape becomes more complex and harder to optimize compared to entropy-based methods, when applied to the layer normalization (LN) layers in the ViT. These TTA techniques were originally designed for CNNs, where TEA leverages BN layers. However, in the context of ViTs, which use LN, the adaptation behaves differently. This opens up a new direction for research into how layer normalization influences generalization in ViTs. With the combination of TEA and ViT excelling on the CIFAR-10 dataset, we successfully demonstrated a proof of concept for applying a TTA method to a SOTA architecture.

To conclude, we were able to confirm the claim of the original authors regarding the performance of TEA, but could not do the same for the claim that the adapted model can generate samples representative to the test dataset. We demonstrated the potential of TET, a combination of TEA and TTT, although the performance fell behind the traditional TEA method. Finally, we explored applying TEA to a more recent and powerful architecture, the ViT. This effort resulted in performance gains for CIFAR-10.

References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[2] Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. *arXiv preprint arXiv:1912.03263*, 2019.

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

[4] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013.

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

[6] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 16888–16905. PMLR, 17–23 Jul 2022.

[7] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. *arXiv preprint arXiv:2302.12400*, 2023.

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

[9] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pages 9229–9248. PMLR, 2020.

[10] Yushun Tang, Ce Zhang, Heng Xu, Shuoshuo Chen, Jie Cheng, Luziwei Leng, Qinghai Guo, and Zhihai He. Neuro-modulated hebbian learning for fully test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3728–3738, June 2023.

[11] Tijmen Tieleman. Training restricted Boltzmann machines using approximations to the likelihood gradient. In *ACM Other conferences*, pages 1064–1071. Association for Computing Machinery, New York, NY, USA, July 2008.

[12] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

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

[14] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks, 2017.

A Training Hyperparameters

The two tables below contain the hyperparameters used for our experiments. Table 3 contains the hyperparameters regarding our reproduction and ViT experiments, and Table 4 contains the hyperparameters regarding the training of TET.

Model	**Dataset(s)**	**LR**	**WD**	**Opt.**	**Mom.**	**Epochs**	**Warmup**	**Batch**
WRN-28-10	CIFAR-10†	–	–	–	–	–	–	–
	CIFAR-100, TIN-200	0.10	5e-4	SGD	0.9	200	0	128
ViT-B/16	CIFAR-10, CIFAR-100	0.01	0	SGD	0.9	20 (10 000 steps)	500	512
	TIN-200	0.01	0	SGD	0.9	40 (20 000 steps)	500	512

Table 3: Hyperparameters used for training WRN-28-10 and for fine-tuning ViT-B/16. Experiments marked with † are done using pretrained weights imported from RobustBench.

Setting	**LR**	**WD**	**Opt.**	**Mom.**	**Epochs**	**Warmup**	**Batch**
Pre-training	0.1	0.0005	SGD	0.9	200	500 steps (on λ_{energy})	1024
Inference	0.01	0	SGD	0	-	None	200

Table 4: Hyperparameters used for our test-time energy training (TET) approach on the CIFAR-10 and CIFAR-100 datasets.

B Metrics

For evaluation on corruption datasets, we employ average accuracy and mean Corruption Error (mCE) [3, 13]. These metrics provide a comprehensive evaluation of a model's generalization in handling diverse distributions, thereby offering a multi-faceted perspective on model performance.

Average Accuracy. Average accuracy is the accuracy averaged over all severity levels and corruptions. Let C be the total number of corruptions, with each corruption having S severities. For some model f, let $\varepsilon_{s,c}(f)$ denote the top-1 error rate on the corruption c with severity level s averaged over the whole test set,

$$AvgAcc_f = 1 - \frac{1}{C \cdot S} \sum_{c=1}^{C} \sum_{s=1}^{S} \varepsilon_{s,c}(f).$$

Mean Corruption Error. mCE is a metric used to measure the performance improvement of model f compared to a baseline model f_0. We use the model without adaption as the baseline model. It is defined as

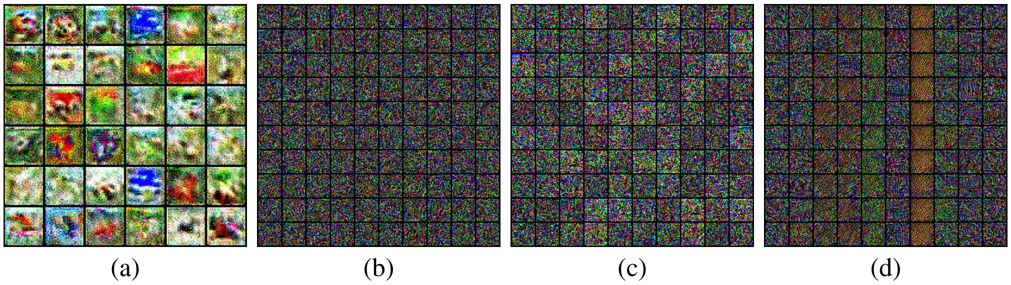
$$mCE_f = \frac{1}{C} \sum_{c=1}^{C} \frac{\sum_{s=1}^{S} \varepsilon_{s,c}(f)}{\sum_{s=1}^{S} \varepsilon_{s,c}(f_0)}.$$

C Complete Reproduction of TEA Results

WRN-28-10		CIFAR-10(C)					CIFAR-100(C)					Tiny-ImageNet(C)				
		Clean	Corr Severity 5		Corr Severity 1-5		Clean	Corr Severity 5		Corr Severity 1-5		Clean	Corr Severity 5		Corr Severity 1-5	
		Acc(↑)	Acc(↑)	mCE(↓)	Acc(↑)	mCE(↓)	Acc(↑)	Acc(↑)	mCE(↓)	Acc(↑)	mCE(↓)	Acc(↑)	Acc(↑)	mCE(↓)	Acc(↑)	mCE(↓)
Source		94.77	56.48	100.0	73.45	100.0	80.64	35.29	100.0	51.92	100.0	**65.0**	12.69	100.0	20.79	100.0
	Diff	+0.00	+0.01	-	+0.00	-	-1.15	-0.10	-	-0.20	-	+1.81	-8.52	-	-13.34	-
Norm	BN	93.97	79.57	46.95	85.63	54.13	78.23	56.17	67.74	64.47	73.89	55.0	23.08	88.1	29.18	89.4
	Diff	+0.00	+0.01	-5.70	+0.00	-5.87	-2.60	-3.89	+4.20	-3.64	+4.47	+9.96	-4.66	-5.32	-5.09	-11.56
	DUA†	-	80.10	50.78	-	-	-	-	-	-	-	-	-	-	-	-
Pseudo	PL	93.67	38.54	141.21	65.90	128.45	78.28	24.46	116.75	47.54	109.09	5.2	1.09	113.29	1.58	124.25
	Diff	-0.08	-12.88	+34.23	-6.72	+29.08	-2.24	-28.94	+44.63	-16.99	+33.80	-59.80	-14.94	+13.29	-19.21	+24.25
	SHOT	93.98	81.64	42.19	86.87	49.47	78.31	61.96	58.79	68.46	65.59	53.0	24.57	86.4	30.55	87.68
	Diff	+0.73	+6.87	-21.00	+4.52	-23.14	-2.21	+5.43	-9.22	+2.46	-7.69	+5.05	-4.57	-3.76	-9.46	-3.73
Entropy	TENT	93.83	81.42	42.70	86.75	49.92	78.38	61.38	59.69	68.06	66.43	55.0	24.33	86.67	30.35	87.93
	Diff	+0.17	+0.01	-5.43	+0.00	-6.25	-1.76	-1.71	+0.27	-1.41	-1.37	+15.46	-1.98	-8.85	-1.68	-16.56
	ETA	93.97	79.59	46.91	85.63	54.11	77.45	57.62	65.50	65.24	72.29	54.4	23.66	87.44	29.77	88.66
	Diff	+0.01	+0.01	-5.73	+0.00	-5.88	-3.20	-2.20	+0.98	-1.93	-0.11	+11.20	-3.62	-6.68	-3.69	-13.59
	EATA	93.97	79.59	46.91	85.63	54.11	77.57	57.65	65.45	65.39	71.98	54.6	23.68	87.42	29.75	88.70
	Diff	+0.01	+0.00	-5.71	-0.01	-5.87	-3.11	-2.59	+1.70	-2.09	+0.32	+11.18	-3.60	-6.67	-3.72	-13.54
	SAR	93.97	80.70	44.34	86.18	52.06	77.27	58.14	64.70	65.50	71.75	59.2	24.79	86.14	30.76	87.42
	Diff	+0.00	+0.93	-7.60	+0.35	-6.91	-3.57	-4.81	+5.33	-4.51	+5.76	+17.62	-3.42	-6.68	-3.84	-13.05
Energy	TEA	**94.18**	**83.31**	**38.34**	**87.87**	**45.68**	78.37	**62.52**	**57.93**	**68.9**	**64.67**	53.0	24.55	86.42	30.43	87.84
	Diff	+0.09	-0.03	-5.35	-0.01	-6.32	-2.51	-2.58	+1.75	-2.32	+0.93	+1.35	-7.12	-1.57	-9.53	-4.28

Table 5: Accuracy and mCE for various TTA methods compared to the Source model. Values in bold indicate the best score per metric. Results marked with † are taken from [10]. The "Diff" rows contain the differences between our results and the original paper [13].

D Data distribution Visualization



(a) (b) (c) (d)

Figure 1: Test distribution perception visualization: (a) a copy of Figure 4(iii) from [13]; (b) WRN-28-10 trained on CIFAR-10, no TEA adaptation (remains a noisy sample); (c) our reproduction created with WRN-28-10 trained on CIFAR-10, TEA adaptation on CIFAR-10 test set; (d) our reproduction created with WRN-28-10 trained on CIFAR-10, TEA adaptation on CIFAR-10 test set, with all layers adapted, not just BN layers, causing a sharp drop in validation accuracy.

E Difficulties

E.1 What was easy

The authors have provided code for the general framework of TEA, allowing for the replicability and interpretability of their novel method. Moreover, replicating the baselines required minimal effort, as they were readily available in the authors' repository.

E.2 What was difficult

Reproducing the results proved difficult, as the code for loading and configuring datasets—other than CIFAR-10—was not provided. Furthermore, as noted by the authors, the Source models for CIFAR-100 and Tiny-ImageNet had to be trained from scratch, but no training code was provided. Furthermore, We found that images of Tiny-ImageNet are downscaled from 64 × 64 to 32 × 32. This fact is not mentioned by the original research [13] nor by the research introducing the Source model [14]. Additionally, the code for both the baseline models and the TEA model did not shuffle the test samples. This omission led to degraded performance for all model adaptation techniques on Tiny-ImageNet, hence this behavior was changed in our experiments. Furthermore, the Source model achieved results comparable to those reported in [13] on Tiny-ImageNet when using the same learning rate scheduler as on CIFAR, whereas the original study did not apply a scheduler when pretraining the source model on this dataset.

Regarding the training of TET, the process of determining a proper training pipeline proved to be challenging. The pipeline consisted of many hyperparameters regarding the energy sampling process, the balancing of λ_{energy} and λ_{cls}, and more. TET training was expensive due to the complex training code with sampling and Langevin dynamics. This, in combination with our limited budget, possibly played a part in not being able to properly optimize the TET method for the WRN model, with performance not exceeding the baselines' performance as a result.