Controllable Federated Prompt Learning at Test Time

Supplementary Material

In the appendix, we elaborate on additional experimental details and present supplementary analysis that we did not include in the main paper. Appendix A provides more experimental setups and implementation details. Appendix B presents detailed experiments results of further study. Appendix C discusses a limitation of our work and outlines the potential avenues for future research that we aim to pursue.

A. Detailed Configuration and Experimental Setups

A.1. Datasets

We use the five benchmark datasets as described in the main paper: CIFAR100 [4], ImageNet [2] (together with its natural variants ImageNet-A [7], ImageNet-V2 [13], ImageNet-R [6]), Flowers-102 [12], Caltech-101 [3], and Food-101 [1]. Here we provide additional details on how the different test distributions (**Ori**, **Corr**, **OoC**, **Mix**, **A**, **V2**, **R**) are constructed for evaluation phase.

Synthetic shifts for CIFAR100, Caltech101, Flowers102, and Food101. The BRFL benchmark [8] creates synthetic shifted datasets only for CIFAR10 and ImageNet. Following the exact CIFAR10 protocol in BRFL, we extend the same construction to our test datasets (CIFAR100, Caltech101, Flowers102, Food101):

- **Ori (in-distribution).** The original local test split of each client obtained from the non-IID Dirichlet partition. This split matches the distribution of the client's training data.
- **Corr (corrupted).** We generate **Corr** by randomly sampling a corruption from 15 common corruptions and apply it to the test sample, setting the severity level of corruptions to highest 5 for all experiments.
- **OoC (out-of-client).** BRFL introduces out-of-client evaluation for CIFAR10; we follow exactly the same procedure here: for each client, we sample its **OoC** set from the test data of *other* clients (excluding its own local test samples), matching the size of its **Ori** split. We discuss the motivation for OoC evaluation in Appendix A.4. A.4.
- **Mix (mixture of test).** Following BRFL, **Mix** is constructed by combining equal-sized subsets from **Ori**, **Corr**, and **OoC**. This creates a more realistic mixture of in-distribution and shifted samples.

Synthetic shifts on ImageNet and our extension. For ImageNet, we follow the BRFL benchmark, which provides evaluation splits for **Ori**, **Mix**, and the natural distribution shift variants ImageNet-A, ImageNet-V2, and ImageNet-R. To enable out-of-client evaluation on ImageNet, which is not available in BRFL, we additionally construct an ImageNet **OoC** split using the same out-of-client sampling procedure described earlier for CIFAR100, Caltech101, Flowers102, and Food101.

A.2. Settings

Overall Experimental Pipeline of COTE. We simulate a federated environment with 20 clients, where each client holds a private local dataset partitioned under a non-IID Dirichlet distribution with concentration parameter $\alpha = 0.01$. This configuration yields *extreme heterogeneous* label distributions across clients, reflecting realistic personalization scenarios in federated systems.

The federated training phase is conducted for 100 communication rounds to obtain a global prompt model $\mathbf{P}^g$. Each client then fine-tunes this global prompt on its own local training data to derive a personalized prompt $\mathbf{P}_i^l$, which, together with the domain-neutral CLIP prompt $\mathbf{P}^c$, forms the deployed prompt set $\Pi_i = \{\mathbf{P}^g, \mathbf{P}_i^l, \mathbf{P}^c\}$. After deployment, no further communication or parameter exchange is performed.

For test-time evaluation, we follow the BRFL benchmark protocol [8] to construct shifted test distributions, including in-distribution, corrupted, out-of-client, natural shift, and mixed variants as described in Appendix A.1. Each client then performs local prompt adaptation on its unlabeled test data following the proposed controllable workflow guided by the alignment signal ψ in Equation 14. The image and text encoders remain frozen, and only the prompt parameters are updated. All compared methods share the same pretrained CLIP backbone and follow identical training and evaluation settings for fairness.

Dirichlet heterogeneity with $\alpha = 0.01$. Under a Dirichlet partition with concentration parameter $\alpha = 0.01$, the client data exhibit *extreme* label heterogeneity. For CIFAR100, which contains 100 classes and 20 clients, such a small α forces each client's label distribution to concentrate on only a few classes. In practice, each client typically receives *around five* dominant classes with substantial sample counts, while most of the remaining classes appear only sparsely or are entirely absent.

As shown in Figure A1, the heatmap provides a clear view of this sparsity structure. Each row corresponds to one client, and each column to one of the 100 classes. The intensity of each cell represents the number of samples assigned to that client-class pair. The resulting pattern prominently reveals highly localized bright regions that indicate the small subset of major classes for each client, while the vast majority of the grid remains near zero. This visual structure directly reflects the strong label skew introduced by the $\alpha = 0.01$ Dirichlet partition.

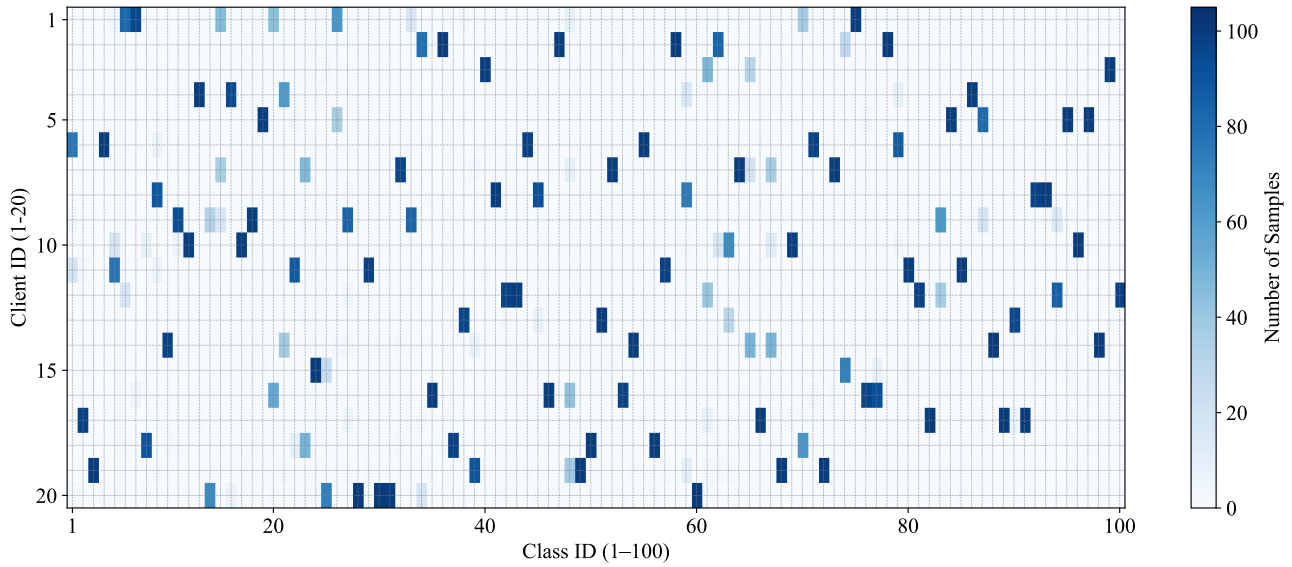


Figure A1. Per-class sample distributions for all 20 clients on CIFAR100 under a Dirichlet partition with concentration parameter $\alpha = 0.01$. Each row denotes a client and each column a class. The concentration of bright blocks indicates that each client is dominated by only a few classes, while most classes contain few or no samples. This heatmap highlights the extreme non-IID nature produced by the $\alpha = 0.01$ setting.

A.3. Implementation Details

Backbone and Model Specification. All experiments use the CLIP ViT-B/16 model as the backbone. The image and text encoders remain completely frozen during federated training, local personalization, and test-time adaptation. The global prompt $\mathbf{P}^g$ and local personalized prompts $\mathbf{P}_i^l$ each contain 16 learnable tokens (embedding dimension 512). The domain-neutral prompt $\mathbf{P}^c$ is a fixed hand-crafted template (`"a photo of a {class}"`), and is not learnable.

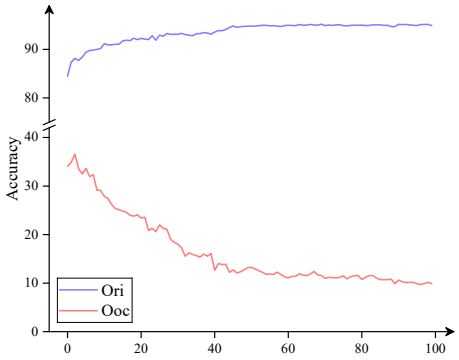
Federated Training Setup. We adopt the standard federated prompt learning configuration (PromptFL [5]). During federated training, the global prompt $\mathbf{P}^g$ is the only shared trainable component and is collaboratively optimized using federated averaging. Each client trains with the SGD optimizer using a learning rate of 2×10^{-3}, batch size 32, and 5 local epochs per communication round. A cosine learning-rate decay schedule is applied across 100 global rounds. We use a weight decay of 5×10^{-4} and apply gradient clipping with a maximum norm of 1.0. All learnable prompts are randomly initialized unless otherwise specified.

Prompt Tuning Configuration. After federated training, each client derives its personalized prompt $\mathbf{P}_i^l$ by fine-tuning locally on its private training data while keeping the backbone frozen. The global prompt $\mathbf{P}^g$, local prompt $\mathbf{P}_i^l$, and fixed domain-neutral prompt $\mathbf{P}^c$ together form the deployed prompt set Π_i for test-time adaptation. No additional parameters outside the prompt tokens are updated.

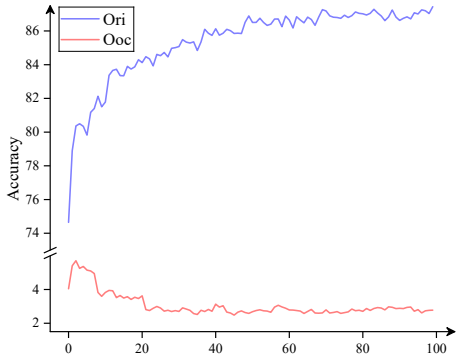
Test-time Adaptation Details. During test-time adaptation, each client updates only its prompt set Π_i using unlabeled test data. Samples are partitioned into high-confidence and low-confidence subsets using a confidence threshold $\tau_c = 0.7$ in Equation 16. For high-confidence samples $\mathcal{D}^{\text{high}}$, we perform 5 adaptation epochs with a learning rate of 1×10^{-4} and mini-batch size 32, using pseudo-label supervision guided by the alignment signal ψ in Equation 14. For low-confidence samples $\mathcal{D}^{\text{low}}$, we apply a sample-level test-time prompt tuning strategy for 1 epoch, initializing from $\mathbf{P}^c$ when $\psi > 0$ and from $\mathbf{P}_i^l$ when $\psi < 0$.

A.4. Rationale Behind OoC Evaluation.

In real-world privacy-sensitive visual systems such as multi-site infrastructure inspection or city-level camera deployments, models are often trained via federated learning to meet strict data-isolation requirements. Each site then performs local personalization using its own labeled data, allowing the model to better capture its local visual characteristics. While this personalization improves performance on inputs close to the site's historical distribution, its effectiveness is inherently limited by the narrow coverage of local data. By adapting the model to a restricted set of locally observed patterns, personalization may reduce the ability to handle broader variations naturally occurring across the federation. In practice, such situations arise when operational changes occur, for example temporary camera installation, emergency monitoring of adjacent areas, or sensor replacement, so that a site receives test-time images containing visual patterns not covered in its own training data. These patterns may be similar to variations that other sites could potentially observe, but they remain outside the personalized model's learned scope. To evaluate a model's generalization ability under such post-deployment distribution mismatch, we adopt this broader regime and refer to it as the out-of-client (OoC) setting in our evaluation. The OoC setting allows us to systematically test how well a personalized model handles inputs that fall beyond the distribution it has been adapted to. As illustrated in Figure A2a and Figure A2b, under the FedOTP personalization method the accuracy on in-distribution (Ori) samples steadily improves as training proceeds, whereas the accuracy on OoC samples drops markedly, showing that a model tailored to a limited local distribution can struggle when confronted with these broader forms of variation.



(a) Personalization performance on CIFAR100 dataset.



(b) Personalization performance on ImageNet dataset.

Figure A2. Behavior of the personalized federated model (FedOTP): while local personalization enhances in-distribution performance, it notably compromises generalization to unseen clients.

Our OoC construction is therefore a principled way to model this type of post-deployment shift. Instead of introducing an artificial test, OoC reflects the realistic case where a client receives samples statistically aligned with other clients' distributions rather than its own. This scenario aligns with the motivation of test-time adaptation, which aims to maintain model reliability under unpredictable and heterogeneous shifts. Incorporating OoC as a dedicated evaluation condition thus provides a comprehensive assessment of the robustness and generalization ability of TTFPL in practical federated environments.

B. Additional Results

B.1. Effect of Coefficient λ

Here, we provide the detailed results of Figure 3a in Section 5.4.

Table A1. Effect of coefficient λ on performance across CIFAR100 and ImageNet datasets.

λ	CIFAR100					ImageNet						
	Ori	Corr	OoC	Mix	Avg	Ori	A	V2	R	OoC	Mix	Avg
0.70	92.89	66.34	**42.34**	**67.13**	67.18	83.39	76.27	**98.44**	88.78	**36.60**	85.27	78.12
0.80	94.94	66.98	**42.34**	**67.13**	67.85	84.27	76.58	98.21	**89.27**	**36.60**	87.40	78.72
0.90	**96.16**	**79.13**	**42.34**	66.67	**71.08**	**93.10**	81.92	97.69	**89.27**	**36.60**	**90.05**	**81.44**
0.95	**96.16**	**79.13**	**42.34**	65.90	70.88	**93.10**	**82.36**	97.09	**89.27**	12.87	**90.05**	77.46

B.2. Effect of Weighting Factor ϕ

Here, we provide the detailed results of Figure 3b in Section 5.4.

Table A2. Effect of hyperparameter ϕ on performance across CIFAR100 and ImageNet datasets.

ϕ	CIFAR100					ImageNet						
	Ori	Corr	OoC	Mix	Avg	Ori	A	V2	R	OoC	Mix	Avg
0.00	**96.16**	79.01	**42.34**	65.87	70.84	**93.10**	**82.36**	97.09	**89.27**	24.83	**90.05**	79.45
0.25	**96.16**	**79.13**	**42.34**	66.05	70.92	**93.10**	82.07	97.76	**89.27**	35.49	**90.05**	81.29
0.50	**96.16**	**79.13**	**42.34**	66.67	**71.08**	**93.10**	80.43	97.76	**89.27**	**36.60**	**90.05**	81.20
0.75	**96.16**	68.01	**42.34**	67.03	68.39	86.39	77.51	97.86	**89.27**	**36.60**	89.45	79.51
1.00	93.00	66.34	**42.34**	**67.13**	67.20	83.62	76.27	**98.09**	89.16	**36.60**	85.59	78.22
auto	**96.16**	**79.13**	**42.34**	66.67	**71.08**	**93.10**	81.92	97.69	**89.27**	**36.60**	**90.05**	**81.44**

B.3. Additional Results on CIFAR100 under Different Heterogeneity Levels

We report additional results on CIFAR100 to further analyze the impact of client heterogeneity. For the extreme heterogeneity setting ($\alpha = 0.01$), we include additional baselines, including FOCoOp as a representative personalized federated prompt learning (PFPL) method and two recent test-time adaptation (TTA) approaches, COME and O-TPT (Table A3). We further evaluate all methods under two additional heterogeneity levels ($\alpha = 0.1$ and $\alpha = 1.0$), as shown in Table A4 and Table A5.

Under the extremely heterogeneous setting ($\alpha = 0.01$, Table A3), existing PFPL and TTA methods suffer significant degradation on OoC data, while COTE achieves the best performance with an average accuracy of 71.08%.

Table A3. Additional results on CIFAR100 dataset with $\alpha = 0.01$.

$\alpha = 0.01$	Ori	Corr	OoC	Mix	Avg
Personalized Federated Prompt Learning					
FOCoOp [10]	95.31	92.95	4.82	37.88	57.74
Test-time Adaptation					
COME [17]	94.89	51.50	9.51	53.66	52.39
O-TPT [14]	67.94	34.86	63.14	55.45	55.35
COTE	96.16	79.13	42.34	66.67	**71.08**

With moderate heterogeneity ($\alpha = 0.1$, Table A4), COTE consistently outperforms all baselines and achieves the best average accuracy of 72.07%.

Table A4. Results on CIFAR100 dataset with $\alpha = 0.1$.

$\alpha = 0.1$	Ori	Corr	OoC	Mix	Avg
Personalized Federated Prompt Learning					
PromptFL [5] + FT	90.12	64.54	43.86	66.17	66.17
FedOTP [9]	88.75	62.55	30.67	60.83	60.70
pFedMoAP [11]	86.90	56.60	31.36	58.48	58.34
FOCoOp [10]	88.83	84.40	21.84	48.58	60.91
Test-time Adaptation					
TENT [16]	88.49	24.18	47.40	57.62	54.42
TPT [15]	90.80	65.15	43.99	66.09	66.51
COME [17]	88.48	23.86	47.38	57.27	54.25
O-TPT [14]	77.59	47.15	66.55	63.49	63.70
COTE	90.41	65.61	65.58	66.69	**72.07**

Under relatively homogeneous settings ($\alpha = 1.0$, Table A5), the performance gap between methods becomes smaller. Nevertheless, COTE still achieves the best average accuracy of 66.65%.

Table A5. Results on CIFAR100 dataset with $\alpha = 1.0$.

$\alpha = 1.0$	Ori	Corr	OoC	Mix	Avg
Personalized Federated Prompt Learning					
PromptFL [5] + FT	76.88	41.14	76.26	65.06	64.84
FedOTP [9]	77.19	45.47	64.96	62.20	62.46
pFedMoAP [11]	72.70	37.16	58.81	55.87	56.14
FOCoOp [10]	75.82	65.69	55.59	66.08	65.80
Test-time Adaptation					
TENT [16]	77.68	8.97	76.10	62.04	56.20
TPT [15]	78.32	42.66	74.13	66.28	65.35
COME [17]	77.70	9.40	76.09	61.95	56.29
O-TPT [14]	77.88	42.40	77.62	66.02	65.98
COTE	80.20	45.55	74.19	66.64	**66.65**

B.4. Results on Runtime

The runtime comparison is reported in Table A6. Overall, COTE achieves a favorable trade-off between efficiency and performance. Compared with PFPL methods such as FedOTP, pFedMoAP, and FOCoOp, COTE requires substantially less runtime since it performs adaptation only at test time instead of repeated federated optimization. Among TTA methods, batch-wise approaches (e.g., TENT and COME) are faster but achieve much lower accuracy under distribution shifts. Sample-wise methods such as TPT and O-TPT improve robustness but incur higher computational cost due to per-sample optimization. In contrast, COTE maintains competitive runtime while achieving the best overall performance.

Table A6. Runtime comparison of PFPL and TTA methods on CIFAR100.

Methods	Runtime (hours)	Acc (Avg)
Personalized Federated Prompt Learning		
FedOTP [9]	3.82h	60.41
pFedMoAP [11]	18.33h	59.32
FOCoOp [10]	8.40h	57.74
Batch-wise Test-time Adaptation		
TENT [16]	0.25h	51.99
COME [17]	0.25h	52.39
Sample-wise Test-time Adaptation		
TPT [15]	1.97h	64.17
O-TPT [14]	2.22h	55.34
COTE	1.32h	**71.08**

C. Limitations and Future Works

While the proposed TTFPL framework demonstrates its effectiveness across diverse distribution shifts, our work opens up a broader design space rather than exhaustively exploring every possibility. As a first attempt to unify federated prompt learning with test-time adaptation, our method deliberately adopts a simple and modular formulation. This design highlights the generality, extensibility, and practicality of the framework, but also leaves several promising directions for future research.

First, the current instantiation focuses on a straightforward combination of MoDA-based alignment and confidence-aware adaptation. Although simple, this demonstrates that even minimal components can already yield strong performance. More

sophisticated adaptation strategies, such as adaptive prompt composition, meta-learned update rules, or other effective ways, could further unlock the potential of the framework.

Second, our current instantiation operates purely with textual prompts on the language side of the VLM. An interesting extension is to incorporate visual prompts, for example by inserting learnable prompt tokens into the vision encoder or jointly learning text-visual prompts at test time. Such multi-branch prompt designs may allow clients to better capture domain-specific visual patterns while still benefiting from shared semantic priors in the text space.

Third, although our current framework already combines batch-wise updates for high-confidence samples and sample-wise refinement for low-confidence ones, the structure of test-time adaptation can be further extended. In real federated deployments, test inputs often arrive in an irregular or evolving manner rather than in fixed batches. A promising direction is to explore more dynamic test-time adaptation strategies that operate in online or incremental settings. Such extensions may allow the model to respond more smoothly to non-stationary data streams.

Overall, our work provides a general, conceptually simple, and highly extensible paradigm for combining federated learning and test-time adaptation. We believe it establishes a foundation for a wide range of future explorations, architectures, and applications.

References

[1] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European Conference on Computer Vision*, pages 446–461. Springer, 2014. 1

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 248–255. Ieee, 2009. 1

[3] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. *Computer Vision Image Understanding*, 106(1):59–70, 2007. 1

[4] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2066–2073. IEEE, 2012. 1

[5] Tao Guo, Song Guo, Junxiao Wang, Xueyang Tang, and Wenchao Xu. Promptfl: Let federated participants cooperatively learn prompts instead of models–federated learning in age of foundation model. *IEEE Transactions on Mobile Computing*, 23(5):5179–5194, 2023. 2, 4, 5

[6] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 1

[7] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021. 1

[8] Liangze Jiang and Tao Lin. Test-time robust personalization for federated learning. *arXiv preprint arXiv:2205.10920*, 2022. 1

[9] Hongxia Li, Wei Huang, Jingya Wang, and Ye Shi. Global and local prompts cooperation via optimal transport for federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12151–12161, 2024. 4, 5

[10] Xinting Liao, Weiming Liu, Jiaming Qian, Pengyang Zhou, Jiahe Xu, Wenjie Wang, Chaochao Chen, Xiaolin Zheng, and Tat-Seng Chua. Focoop: Enhancing out-of-distribution robustness in federated prompt learning for vision-language models. In *International Conference on Machine Learning*, pages 37528–37554. PMLR, 2025. 4, 5

[11] Jun Luo, Chen Chen, and Shandong Wu. Mixture of experts made personalized: Federated prompt learning for vision-language models. *arXiv preprint arXiv:2410.10114*, 2024. 4, 5

[12] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008. 1

[13] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pages 5389–5400. PMLR, 2019. 1

[14] Ashshak Sharifdeen, Muhammad Akhtar Munir, Sanoojan Baliah, Salman Khan, and Muhammad Haris Khan. O-tpt: Orthogonality constraints for calibrating test-time prompt tuning in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19942–19951. IEEE, 2025. 4, 5

[15] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems*, 35:14274–14289, 2022. 4, 5

[16] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. 4, 5

[17] Qingyang Zhang, Yatao Bian, Xinke Kong, Peilin Zhao, and Changqing Zhang. Come: Test-time adaption by conservatively minimizing entropy. In *International Conference on Learning Representations*, 2025. 4, 5