A APPENDIX

A.1 DATASETS AND IMPLEMENTATION DETAILS

A.1.1 DATASETS

Datasets. We assess the transfer performance of our method across 9 fine-grained classification datasets and 4 out-of-distribution (OOD) datasets from ImageNet. Since our method primarily focuses on test-time model adaptation, our evaluation is exclusively based on the testing dataset across all these datasets.

• **Flower102** (Nilsback & Zisserman, 2008) is a widely used dataset consisting of 102 different categories of flowers. Each category consists of between 40 and 258 images. It is commonly employed for fine-grained image classification tasks.

• **OxfordPets** (Parkhi et al., 2012) is a dataset designed for pet image classification, containing a large variations in scale, pose, and lighting conditions. It contains images of 37 different pet breeds with roughly 200 images for each class.

• **Food101** (Bossard et al., 2014) is a dataset specifically curated for food recognition applications. It contains images of 101 different food categories, making it suitable for tasks related to food image classification and analysis.

• **Describable Textures Dataset (DTD)** (Cimpoi et al., 2014) is a texture dataset, which is consists of 5640 images. These images are classified into 47 distinct categories, inspired by human perception, with precisely 120 images allocated to each category.

• **StanfordCars** (Krause et al., 2013) is a dataset commonly used for fine-grained car classification tasks. This dataset contains 16,185 images of 196 classes of cars, which is split into 8,144 training images and 8,041 testing images.

• **Aircraft** (Maji et al., 2013) dataset contains 10,200 images of various aircraft, with 100 images for each of 102 different aircraft model variants, most of which are belonging to airplanes.

• **UCF101** (Soomro et al., 2012) is a widely recognized dataset for human action recognition, which consists of 13,320 video clips spanning 101 different human action categories. These 101 categories are further classified into 5 types, including Body motion, Human-object interactions, Playing musical instruments and Sports, Human-human interactions.

• **EuroSAT** (Helber et al., 2019) is a dataset and deep learning benchmark designed for land use and land cover classification. It is based on Sentinel-2 satellite images with 13 spectral bands and a total of 27,000 labeled and geo-referenced images with 10 distinct classes.

• **Caltech101** (Fei-Fei et al., 2004) dataset is composed of approximately 9,000 images with 101 object categories and a background category. Each object category contains approximately 40 to 800 images, with typical image sizes of 200-300 pixels.

• **SUN397** dataset encompasses 108,753 images spanning 397 categories, serves as a benchmark in scene understanding studies. Each category in this diverse collection is represented by a minimum of 100 images.

• **ImageNet** (Deng et al., 2009) dataset is a large-scale ontology of images built upon the backbone of the WordNet structure, designed to advance the field of computer vision. It spans 1000 distinct object classes and contains 1,281,167 training images, 50,000 validation images and 100,000 test images.

• **ImageNet-V2** (Recht et al., 2019) is a test set including natural images collected from various sources. This dataset is consisted of 10,000 images distributed across 1,000 distinct ImageNet categories.

• **ImageNet-Sketch** (Wang et al., 2019) dataset consists of 50000 images, 50 images for each of the 1000 ImageNet classes. These images are obtained by performing Google Image searches using the query "sketch of [standard class name]."

• **ImageNet-A** (Hendrycks et al., 2021b) is a challenging dataset containing real-world, unmodified, and naturally occurring examples that are misclassified by ResNet models. It includes 7,500 intentionally altered and corrupted images with 1,000 categories to assess the robustness of different models.

• **ImageNet-R** (Hendrycks et al., 2021a) collects images of ImageNet categories presented in artistic renditions, including a total of 30,000 images across 200 distinct ImageNet categories.

A.1.2 Implementation Details

Implementation Details. We apply in-context prompt learning on a pre-trained ViT-B/16 CLIP model. We minimize the semi-parametric objective loss to optimize the visual prompt for 1 step and cyclically alternate between optimizing the visual and prompt for 2 steps. All models are trained with in-context examples and test sample on a single NVIDIA GPU. We use the AdamW (Loshchilov & Hutter, 2017) optimizer with a learning rate of $5e^{-3}$ for all datasets. By default, we set weight parameter $\lambda = 0.4$. If not specifically emphasized, we adopt in-context prompt learning with only visual prompt as our default setting for all ablation studies. We select 5 in-context examples from a fixed subset of labeled data, which is composed by randomly sampling 1 sample from each category. The context samples only provide the task information to do on the test sample. They are usually from the same target dataset, while there is no other relationships between them, e.g., category. Token net is randomly initialized at the start of test-time adaptation and accumulatively updated across the entire evaluation process. For each test sample, P_{t} is initialized with prefix tokens derived from "a photo of a", which is then converted into a visual token. P_{v} is then initialized by the above learned P_{t}.

A.2 Additional Ablation Studies

In-context example selection: random *vs* definition. This experiment aims to evaluate two distinct approaches for selecting context examples to prompt each test sample: random and definition-based approaches. In the former approach, input-label pairs are randomly selected from candidate examples as context examples for each test sample, while the latter approach utilizes a common set of examples shared across all test samples once the context examples have been sampled. As depicted in Figure 6, the definition-based selection approach also exhibits significant fluctuations with different random seeds, and its performance consistently lags behind the random-based selection approach. The primary reason behind this discrepancy lies in the fact that definition-based examples cannot guarantee the provision of useful context information for all test samples, whereas random-based examples consistently provide each test sample with domain-specific information pertaining to the target distribution.

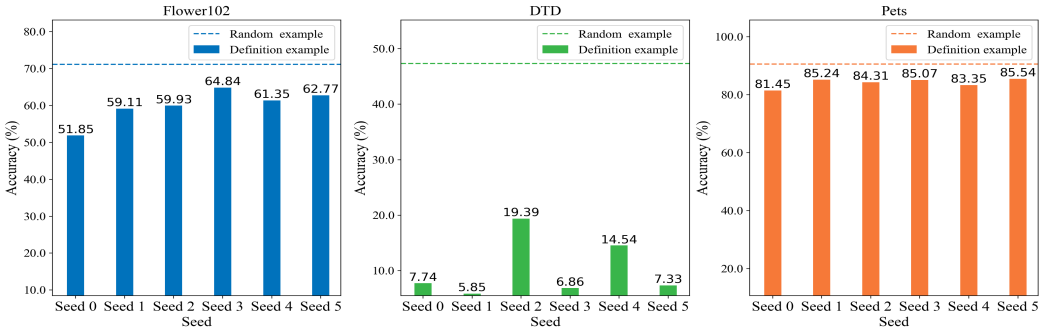

Figure 6: Experimental results w.r.t. different in-context example selection strategies.

Ground Truth Matters. To study the impact of correctly-paired inputs and labels in the in-context examples, referred as "ground truth input-label mapping", we evaluate the following three methods.

• **No example** is a typical test-time adaptation method that does not use any labeled data. A prediction is made via optimizing unsupervised objective involving unlabeled test sample.

• **Examples w/o labels** is the baseline that only uses input images without the corresponding labels, which indicates the model performance without looking at the label information.

Table 7: Comparative analysis of inference time and accuracy with existing TPT. The inference time (Infer. Time) is calculated in minutes.

Method	Flower102		Pets		Cars		Caltech101	
	Infer. Time (↓)	Top 1 acc. (↑)	Infer. Time (↓)	Top 1 acc. (↑)	Infer. Time (↓)	Top 1 acc. (↑)	Infer. Time (↓)	Top 1 acc. (↑)
TPT (Shu et al., 2022)	97.22	68.98	111.31	87.79	322.59	66.87	86.25	94.16
InCP (Ours)	**23.79**	**72.27**	**16.31**	**90.62**	**69.05**	**67.54**	**35.54**	**94.69**

• **Examples w/ gold labels** are employed in a typical in-context learning method with a set of labeled examples, which indicates the model performance with looking at relevant knowledge.

• **Examples w/ random labels** replace all gold labels with random labels, which are randomly sampled at uniform from label space on the testing data.

• **Examples w/ same labels** replace all gold labels with the same label, which is consistent with the label of test sample.

• **Examples w/ oracle labels** replace traditional in-context examples with the oracle examples, where labels are consistent with the label of test sample.

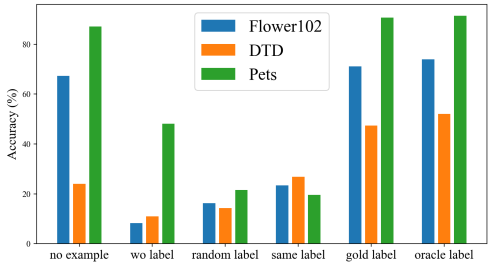

Figure 7: Results when using no-examples, examples w/o labels, examples w/ random labels, examples w/ same labels, examples w/ gold labels and examples w/ oracle labels in fine-grained classification task.

In Fig. 7, we present the recognition accuracy values obtained with different input-label strategies applied to in-context examples. It is evident that model performance is notably sensitive to the correctness of labels. Specifically, using correct gold labels yields better results than employing random labels. Furthermore, employing the same labels as the test sample leads to a significant improvement compared to using random labels. This observation indicates that having consistent labels provides the model with instructive information about the test samples. Moreover, employing examples with oracle labels achieves the upper bound in performance and outperforms the use of examples with gold labels. We attribute this phenomenon to the increased instructive information, which arises not only from the example labels but also from the example inputs themselves. These results underscore the substantial impact of in-context examples' labels on In-Context Learning (ICL) performance, aligning with the findings (Wu et al.).

Training strategy: In-context learning *vs* few-shot learning.

A conventional approach for leveraging the in-context examples dataset involves fine-tuning the prompt on a labeled dataset. As an alternative method, we implemented few-shot prompt learning using all available in-context examples and conducted a comparison with our InCP approach. As demonstrated in Figure 8, the image representations learned through the few-shot approach exhibit lower resolution in distinguishing between classes compared to our prompted representations. The few-shot approach might introduce more variance and noise into the feature space, and the learned features are specific to the task rather than the individual test sample. In contrast, our InCP method is specifically designed to learn an informative prompt for each test sample using in-context examples, enabling a more effective alignment between the sample and its associated examples.

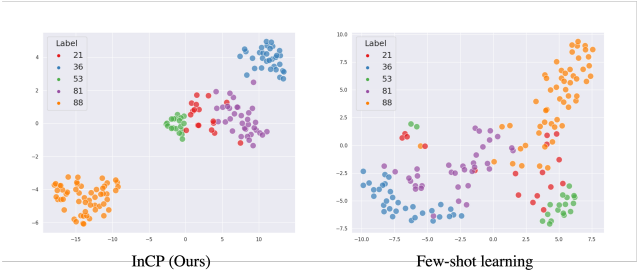

Figure 8: Comparison with few shot learning when using few in-context examples on Flower102.

Comparative analysis of inference time metrics with existing TPT. Table 7 reports the comparative analysis of inference time metrics with TPT Shu et al. (2022). All experiments are conducted on one 2080 Ti GPU, and inference time (Infer. Time) is calculated in minutes. The table shows that TPT Shu et al. (2022) needs significant inference time due to augmenting images by 64 times. In

Table 8: Comparison with CoOp and CoCoOp using the same examples.

Method	Flower102	DTD	Pets	Cars	Caltech101
	Top 1 acc.	Top 1 acc.	Top 1 acc.	Top 1 acc.	Top 1 acc.
CoOP (Zhou et al., 2022b)	66.10	30.97	82.77	60.20	90.26
CoCoOP (Zhou et al., 2022a)	67.23	31.72	83.14	59.78	90.43
Ours	**71.13**	**47.34**	**90.60**	**67.54**	**94.69**

Table 9: Study on task and instance-specific adaptations. w/o U/S/SS-obj represents without unsupervised/supervised/semi-supervised objective.

Method	Adaptation	Imagenet-R	ImageNet-S	Cars	Caltech101
		Top 1 acc.	Top 1 acc.	Top 1 acc.	Top 1 acc.
Ours w/o U-obj	Task	76.89	47.24	65.29	93.96
Ours w/o S-obj	Instance	75.73	44.51	59.08	93.10
Ours (SS-obj)	Task & Instance	**77.56**	**48.03**	**67.54**	**94.69**

contrast, our InCP only uses few in-context examples (i.e., 5) without any augmentation, requiring less inference time.

Comparison with CoOP and CoCoOP using the same examples. We provide CoOp/CoCoOp's results using the same examples as InCP in Table 8. Experimental results show that our InCP achieves better performance than CoOp and CoCoOp on the this setting.

Ablation studies on other datasets. To provide a more comprehensive evaluation, we perform additional ablation studies on other datasets, including ImageNet-R, ImageNet-S, Cars, and Caltech101. The results are detailed in Table 9 and Table 10.

Table 10: Accuracy of different visual prompt methods.

Method	Imagenet-R	ImageNet-S	Cars	Caltech101
	Top 1 acc.	Top 1 acc.	Top 1 acc.	Top 1 acc.
Patched prompt (Bahng et al., 2022)	70.62	43.69	65.14	91.68
Padded prompt (Jia et al., 2022)	68.54	40.48	55.93	89.61
Token prompt	70.39	40.10	57.22	85.60
Generic-language prompt	76.46	45.29	64.17	92.01
Ours (Language-aware)	**77.56**	**48.03**	**67.54**	**94.69**

A.3 Discussion

In-context learning allows large language models (GPT3 Brown et al. (2020), LaMMa Touvron et al. (2023)) to perform inference on unseen tasks by conditioning on in-context examples (a.k.a. prompt) without updating the model parameters. Inspired by this, existing works Wang et al. (2023b); Zhang et al. (2023); Wang et al. (2023a) explore "in-context learning" concept for vision model, in which the model is updated using in-context examples. Meanwhile, CLIP itself is not able to conduct in-context learning task. To equip CLIP with this ability, our InCP introduces learnable prompt for each test sample in test-time stage. In this way, the model can automatically understand the underlying task with in-context examples.

Comparison with few-shot learning. As few-shot methods, CoOP Zhou et al. (2022b) and CoCoOP Zhou et al. (2022a) fine-tune the prompt on ImageNet dataset using 16-shot training data per category and evaluate the generalization performance on downstream tasks. Our work primarily differs from few-shot methods in two main aspects, i.e., sample selection and quantity. (a) For sample selection, few-shot uses strict categories with specific number of samples, which are widely used in training stage. Differently, in-context learning has no constraint on category. The in-context samples in testing stage can either share the same category as the current test sample or the irrelevant category. It is also impractical to know the exact category of unlabeled test sample in advance. (b)

For sample quantity, few-shot learning requires a predefined number of samples from each category, while in-context learning uses a small, arbitrary set of labeled samples-commonly just five samples.

Comparison with semi-supervised learning. Semi-supervised learning Tarvainen & Valpola (2017); Sohn et al. (2020) typically incorporates labeled data during the training phase, amalgamating it with unlabeled data to fine-tune the model and improve its performance on unlabeled samples. Labeled data in semi-supervised learning often shares categories with the unlabeled data. In our method, there is no inherent relationship between in-context examples (labeled data) and the test sample, as they are both drawn from the same domain dataset. Our approach does not necessitate any category information about the test sample, distinguishing it from semi-supervised learning methods.