A Appendix

A.1 Time-Dependent Context Length

Figure 6 shows the variation of context lengths during the fine-tuning process. When $\mathcal{T}_{\text{target}}$ is reached, the total context length for the image and text branches stays constant. We use the warm-up strategy and hence the total context length does not change for the first few epochs. In the pruning process, we prune the context tokens in the text and image branches with the lowest score. The number of tokens removed per epoch can vary from the image branch to the text branch. Within the same branch, the context length can vary at various depths. Overall, Adapt encourages highly heterogeneous context lengths which can be difficult for the manually designed prompts. Adapt is able to automatically determine context lengths. The pruning processes on all 11 datasets are visualized in Figure 16.

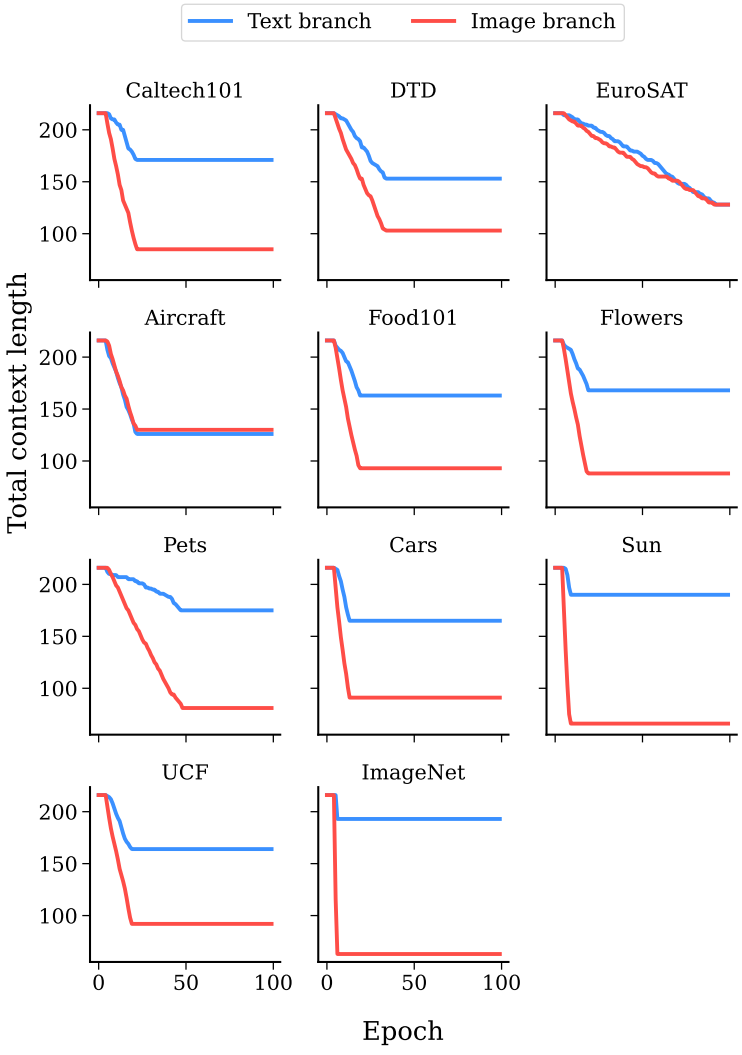

Figure 6: Variation of the total context length during the fine-tuning process on 11 datasets. The plateau in the initial stage is caused by the warm-up strategy. The second one is the regime in which the pruning is finished, *i.e.* $\mathcal{T}_{\text{effective}} = \mathcal{T}_{\text{target}}$.

A.2 Model Attention Using Prompts

We use Grad-CAM (Selvaraju et al., 2017) to examine the localized attention of the CLIP model using zero-shot CLIP, PromptSRC (state-of-the-art baseline) and Adapt. Figure 7 shows the comparison of the model attention. We find the difference in the localized attention between Adapt and zero-shot CLIP is remarkably larger than that between PromptSRC and zero-shot CLIP. Besides, Adapt pronouncedly improves the model's attention on salient features of image objects and avoids the distraction from the background.

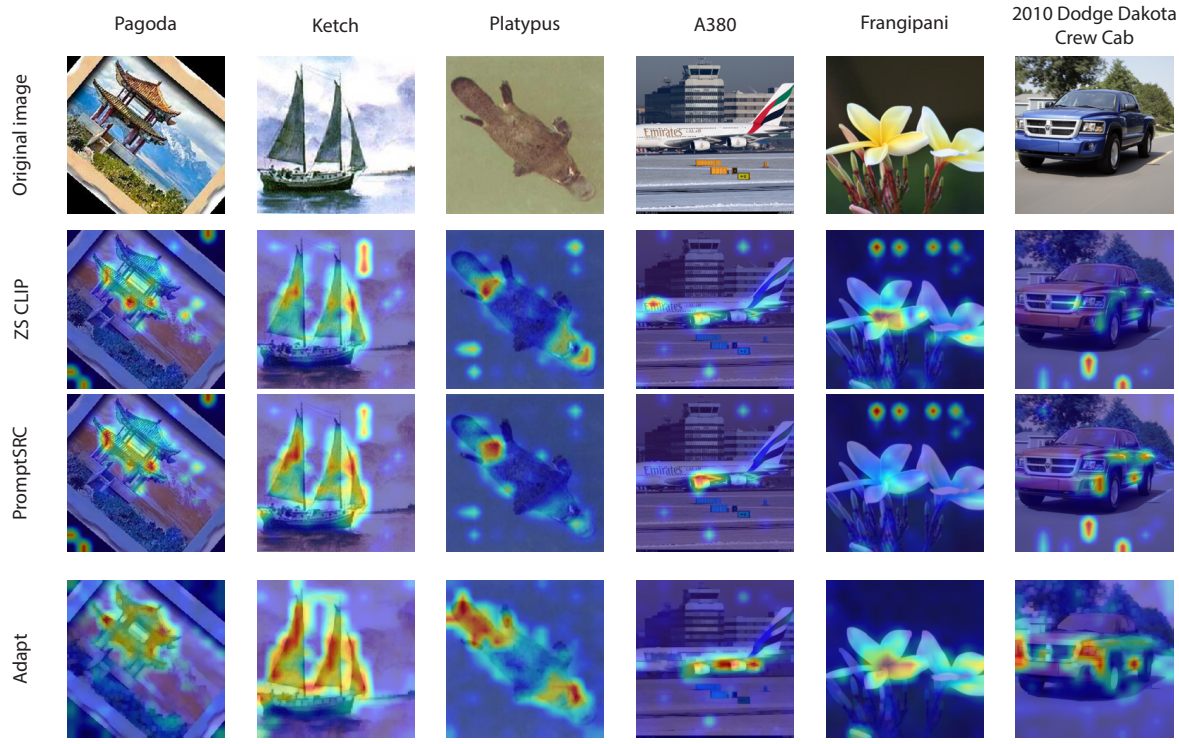

Figure 7: Grad-CAM (Selvaraju et al., 2017) visualization of the model's localized attention. The Adapt method helps the backbone model focus on the salient features of input images and avoid distractions from the background.

A.3 Datasets for Downstream Tasks

Table 2 shows statistics of 11 datasets used for the fine-tuning of the pre-trained CLIP model and text prompt templates used for prompting the vision language model. These datasets covering a wide range of tasks are commonly used as benchmarks for vision-language models. Of 11 datasets, only Caltech101 (Fei-Fei et al., 2004) and ImageNet (Deng et al., 2009) are generic object classification datasets. EuroSAT (Helber et al., 2019), FGVCAircraft (Maji et al., 2013), Food101 (Bossard et al., 2014), OxfordFlowers (Nilsback & Zisserman, 2008) are OxfordPets (Parkhi et al., 2012) are fine-grained classification datasets including images of sub-types under a common type. The text templates we use for 11 datasets are the same as ProGrad (Zhu et al., 2023).

A.4 Few-shot Learning

Figure 8 shows the few-shot learning results using $\{16, 8, 4, 2, 1\}$ shots. The pruning process of the Adapt method depends on the pruning rate r_p and the number of mini-batches n_k. When the number of shots is very small, $\mathcal{T}_{\text{target}}$ might not be reached due to the limited number of mini-batches. To ensure $\mathcal{T}_{\text{target}}$ is

Table 2: Details regarding 11 datasets and corresponding text prompts used for prompting vision-language models. We replace `[class]` with real class names and feed text prompts to text encoders for obtaining corresponding text embeddings.

Dataset	# Classes	Training size	Test size	Task	Text prompt template
Caltech101 (Fei-Fei et al., 2004)	100	4,128	2,465	Generic object classification	a photo of a [class]
ImageNet (Deng et al., 2009)	1,000	1.28M	50,000	Generic object classification	a photo of a [class]
Describable Textures (Cimpoi et al., 2014)	47	2,820	1692	Texture classification	a texture of [class]
EuroSAT (Helber et al., 2019)	10	13,500	8,100	Satellite image classification	a centered satellite photo of [class]
FGVCAircraft (Maji et al., 2013)	100	3,334	3,333	Fine-grained aircraft classification	a type of aircraft, a photo of a [class]
Food101 (Bossard et al., 2014)	101	50,500	30,300	Fine-grained food classification	a type of food, a photo of a [class]
OxfordFlowers (Nilsback & Zisserman, 2008)	102	4,093	2,463	Fine-grained flower classification	a type of flower, a photo of a [class]
OxfordPets (Parkhi et al., 2012)	37	2,944	3,669	Fine-grained pet classification	a type of pet, a photo of a [class]
StanfordCars (Krause et al., 2013)	196	6,509	8,041	Fine-grained car classification	a photo of a [class]
UCF101 (Soomro et al., 2012)	101	7,639	3,783	Action classification	a photo of a [class]
SUN397 (Xiao et al., 2010)	397	15,880	19,850	Scene classification	a photo of a [class]

reached, we decrease ξ_f and ξ_g as the number of shots decreases. We notice that as the number of shots decreases, the Adapt method does not maintain the best performance over all baselines. The cause might be related to the increasingly challenging optimization when the training data have a very limited size. The Adapt framework requires not only optimizing prompt weights but also the prompt configuration, *i.e.* the context length for each transformer block. It has been found that when a limited number of samples per class is available, the model complexity is a critical factor (Brigato & Iocchi, 2021). The performance can be improved by decreasing the model complexity. Besides, prompting methods generally perform poorly when only very limited data are available. New architecture design can boost the performance in the setting of very limited data (Hirohashi et al., 2024).

In addition to comparing test accuracies as shown in Table 1. We calculate the 95% confidence interval (CI) to statistically compare our method and the best baseline method PromptSRC. On the datasets where the Adapt method shows a tangible performance boost compared to the PromptSRC baseline (*e.g.* Aircraft dataset), we do not observe an overlapping in CI. The result shows that our approach shows the performance gain statistically.

Table 3: Performance comparison including CI. We use results over 3 runs to compute CI.

Method	Caltech101	DTD	EuroSAT	Aircraft	Food101	Flowers
PromptSRC	96.04 ± 0.17	72.75 ± 0.96	92.48 ± 0.17	50.86 ± 0.70	87.44 ± 0.05	97.62 ± 0.41
Adapt	96.57± 0.05	74.07 ± 0.79	93.53 ± 0.17	58.30 ± 0.73	86.67 ± 0.00	98.43 ± 0.14
Method	Pets	Cars	SUN	UCF	ImageNet	
PromptSRC	93.81 ± 0.40	83.79 ± 0.68	86.43 ± 0.12	77.24 ± 0.61	73.12 ± 0.22	
Adapt	92.67 ± 0.12	85.97 ± 0.12	86.43 ± 0.09	77.17 ± 0.38	73.20 ± 0.12	

A.5 Baseline Performance

We obtain the performance of PromptSRC (Khattak et al., 2023b), MaPLe (Khattak et al., 2023a), CoOp (Zhou et al., 2022b) and CoCoOp (Zhou et al., 2022a) from (Khattak et al., 2023b), that of VPT-Deep, VPT-Shallow, and UPT (Zang et al., 2022) from (Zang et al., 2022), that of LAMM from (Gao et al., 2024), that of linear probe approach and DAPT from (Cho et al., 2023). PLOT (Chen et al., 2022a) and ProGrad (Zhu et al., 2023) use ResNet50 (He et al., 2016) as the backbone of the image encoder. We replace the ResNet50 model with ViT-B/16 model and test the performance for a fair comparison.

A.6 Effect of Pruning Rate

The pruning rate r_p determines the rate of binary mask reaching $\mathcal{T}_{\text{target}}$. To study the effect of r_p on the final binary mask, we use the agreement ratio to characterize the agreement of the binary mask between two

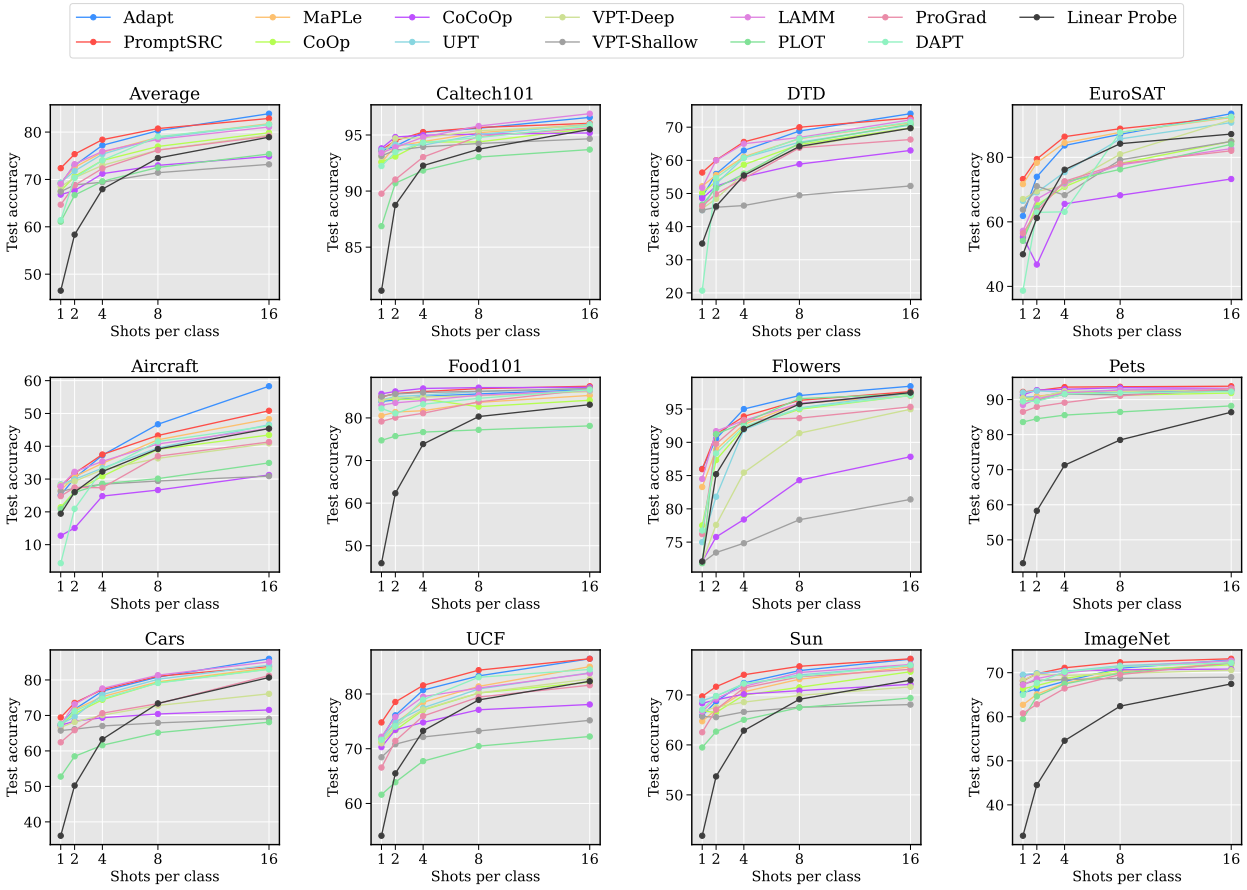

Figure 8: Few-shot learning in the image classification task on 11 datasets. The number of shots is $\{16, 8, 4, 2, 1\}$.

different pruning rates:

$$\text{Agreement ratio} \coloneqq \frac{1}{\ell \times \xi} \sum_{i=1}^{\ell} \sum_{j=1}^{\xi} \mathbb{I}\{\mathcal{M}_1(t)_{ij} = \mathcal{M}_2(t)_{ij}\} \,, \tag{11}$$

where $\mathbb{I}$ is the indicator function and $\mathbb{I}\{E\} = 1$ if the event E happens, otherwise $\mathbb{I}\{E\} = 0$. $\mathcal{M}_1(t)$ and $\mathcal{M}_2(t)$ are binary matrices using two different pruning rates. The agreement ratio is in the range $[0, 1]$. A higher agreement ratio indicates two binary masks are closer.

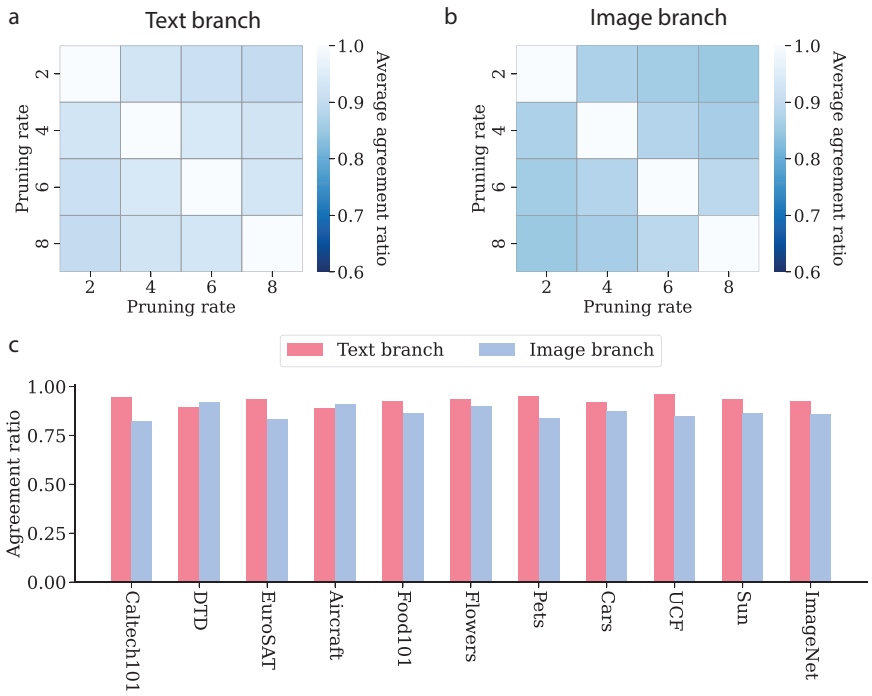

Figure 9: Agreement ratio of binary masks at the last epoch in the text and image branches. We visualize the average pair-wise agreement ratio using different pruning rates $r_p \in \{2, 4, 6, 8\}$ for (a) text branch (b) image branch. (c) The agreement ratio on each of 11 datasets between $r_p = 2$ and $r_p = 4$.

Figure 9 (a) and (b) show the average agreement ratio between pairs of pruning rates $r_p \in \{2, 4, 6, 8\}$. A fixed $\mathcal{T}_{\text{target}} = 256$ is used. The agreement ratio in the text branch is slightly higher than that in the image branch. Overall, the Adapt method is not sensitive to r_p. Using different r_p leads to similar binary masks. Figure 9 (c) shows the agreement ratio on each dataset using $r_p = 2$ and $r_p = 4$. On all 11 datasets, the agreement ratio remains on a high level. The result using different r_p indicates pruning one context token has a nearly negligible effect on the importance of remaining context tokens, which justifies the pruning strategy in the Adapt method. The Adapt method prunes context tokens based on saliency scores. Considering a scenario where r_p context tokens are pruned at the iteration t. If one removed context token can lead to an increase in the saliency score of another pruned context token, pruning r_p tokens at the same time might lead to the removal of important tokens. Results shown in Figure 9 exclude this possibility.

A.7 Pruning Process of Binary Masks

Figure 16 shows the pruning process of binary masks in the text and image branches for 11 datasets. The binary mask is initialized to be $\mathcal{M}(0) = \mathbf{1}_{\ell \times \xi}$. Given a binary mask $\mathcal{M}(t)$, the row dimension is related to the prompt depth ℓ while the column dimension is associated with the prompt width (also called context length). $\mathcal{M}(t)$ stays constant when $\mathcal{T}_{\text{target}}$ is reached. We use Snip to compute the score for each context tokens. The binary masks are highly heterogeneous: context lengths over prompt depth for different branches have

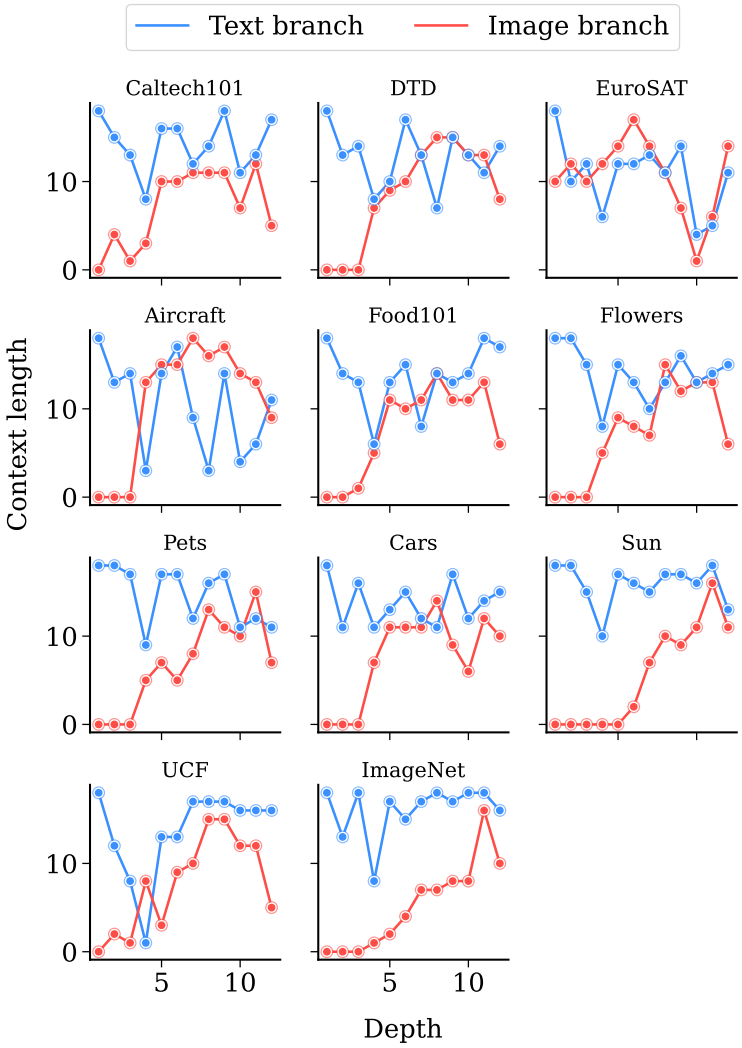

Figure 10: Context lengths at various depths after pruning, *i.e.* $\mathcal{T}_{\text{effective}} = \mathcal{T}_{\text{target}}$. The Adapt method generally prefers to prune context tokens on the image branch.

a large variation. The heterogeneity feature exhibits the strength of the automatic design of context lengths compared to manually designed prompts which tend to be homogeneous.

When $\mathcal{T}_{\text{target}}$ is reached, $\mathbf{P} \odot \mathcal{M}(t)$ on some datasets have a context length of 1 at a certain depth. Even though the pruning concentrated in the prompts for this layer, the performance is not negatively affected. In the network pruning, however, the concentrated pruning can lead to the layer collapse issue (Lee et al., 2019; Hayou et al., 2020). Considering an extreme case, if the weight of an entire layer is pruned, the model will have a disconnection issue. When pruning the prompt, there is no such issue. This indicates that pruning prompts can lead to highly heterogeneous prompt lengths.

After pruning, *i.e.* $\mathcal{T}_{\text{effective}} = \mathcal{T}_{\text{target}}$, we examine the context lengths at various depths. Figure 10 shows the context lengths and Figure 16 is the corresponding pruning process. Generally, the total context length on the text branch is higher than that in the image branch. Besides, the first layer on the text branch has the highest context length. Shallow prompting paradigms for vision-language models such as CoOp (Zhou et al., 2022b) insert prompt only to the text input for the text encoder of the vision-language model. This is consistent to the result shown in Figure 10: inserting prompts to the first layer on the text branch is important to boost the performance of pre-trained models.

Figure 11 shows context lengths at different depths after pruning. We do not observe a uniform context length pattern across datasets, which justifies the automatic pruning strategy.

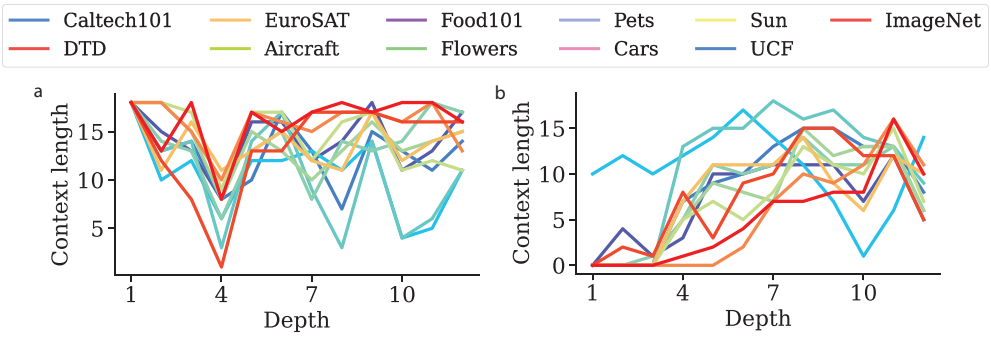

Figure 11: Context lengths across model depth after pruning for 11 datasets. Context lengths exhibit both dataset-dependent and depth-dependent behaviors. (a) Text branch; (b) Image branch.

A.8 Statistics of Reported Performance

Table 4 shows the standard deviation of the performance of the Adapt method over 3 runs. The number of shots is 16. We use $\mathcal{T}_{\text{target}} = 256$, $\xi_f = \xi_g = 18$, $l_f = l_g = 12$. We do not find a significant performance variation on any of 11 datasets.

Table 4: Standard deviation of the reported performance of the Adapt method over 3 runs.

Caltech101	DTD	EuroSAT	Aircraft	Food101	Flowers
96.57 ± 0.25	74.07 ± 0.39	93.53 ± 1.16	58.30 ± 0.87	86.67 ± 0.05	98.43 ± 0.28
Pets	Cars	Sun	UCF	ImageNet	
92.67 ± 0.08	85.97 ± 0.40	86.43 ± 0.17	77.17 ± 0.29	73.20 ± 0.26	

A.9 Performance Boost Compared to Zero-Shot CLIP

Figure 12 shows the performance boost of the Adapt method in comparison with zero-shot CLIP model. The performance boost correlates with the distribution shift of the downstream dataset compared to the pretraining dataset. EuroSAT is the out-of-distribution (OOD) dataset (Tsao et al., 2023) and has the largest performance boost of 55.96%.

A.10 Ablation Study

Target total context length $\mathcal{T}_{\text{target}}$ $\mathcal{T}_{\text{target}}$ is associated with the complexity of inserted prompts. Figure 13 (a) shows the performance variation when changing $\mathcal{T}_{\text{target}}$. When decreasing $\mathcal{T}_{\text{target}}$ from 256 to 128, the average test accuracy drops by 0.43% while the total context length after pruning becomes half of the original length. Continuing decreasing $\mathcal{T}_{\text{target}}$ leads to a pronounced degradation of the model performance. Increasing $\mathcal{T}_{\text{target}}$ from 256 to 512, however, results in the inferior performance.

Figure 14 shows the effect of $\mathcal{T}_{\text{target}}$ on the performance. Different from the results shown in Figure 13 (a), pruning rate is adjusted to ensure $\mathcal{T}_{\text{target}} = \mathcal{T}_{\text{effective}}$ when pruning is finished.

Heterogeneous prompting We compare the performance using homogeneous prompting to Adapt using heterogeneous prompting. For the homogeneous prompting, we use $\xi_f = \xi_g = 12$, and no pruning is applied. The Adapt method uses $\mathcal{T}_{\text{target}} = 240$ to ensure the total context length $\mathcal{T}_{\text{effective}}$ is the same as that of

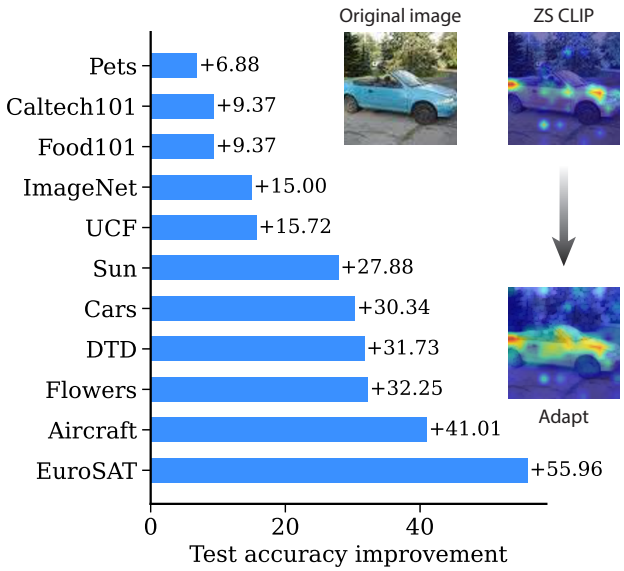

Figure 12: Performance improvement on 11 datasets compared to zero-shot CLIP model. The performance of the zero-shot CLIP model is greatly boosted by the Adapt method. The method helps the backbone model focus on salient features of input images. More examples of Grad-CAM (Selvaraju et al., 2017) visualization can be found in Appendix Figure 7. The significant performance enhancement indicates that knowledge distillation using KL divergence of the predicted probability distribution might be sub-optimal.

homogeneous prompting. Figure 13 (b) shows the performance comparison. Using heterogeneous prompts boosts performance.

Score computation We examine the effect of three different scoring functions: Snip (Lee et al., 2018), gradient norm, and l_2-norm. Snip considers both the gradient and magnitude of the prompt parameters. The comparison is shown in Figure 13 (c). Snip has the best performance. Overall, there is no remarkable difference among scoring functions.

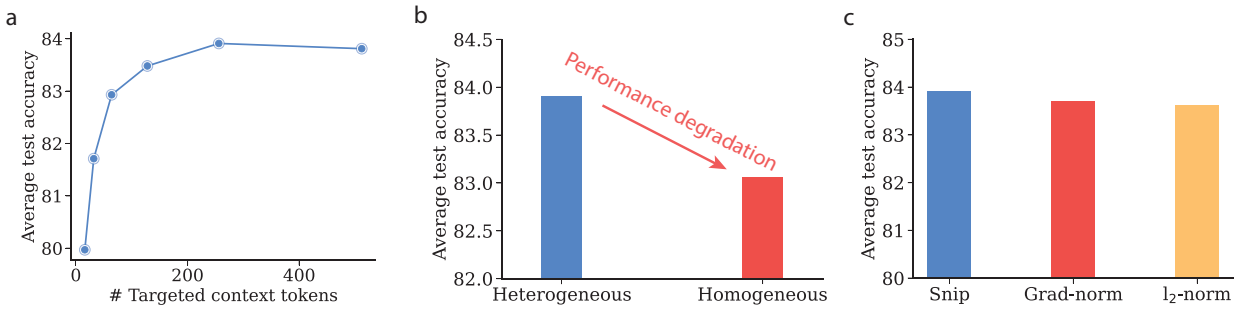

Figure 13: Ablation study on the effect of (a) target total context length $\mathcal{T}_{\text{target}}$. When $\mathcal{T}_{\text{target}} < 128$, $\mathcal{T}_{\text{target}}$ is not reached after pruning, *i.e.* $\mathcal{T}_{\text{effective}} < \mathcal{T}_{\text{target}}$. (b) heterogeneous prompting, and (c) scoring function.

A.11 Terminology

We use different terminologies in the prompt tuning. To ensure consistency in the terminology and improve the readability, we list a detailed explanation for all related terminologies:

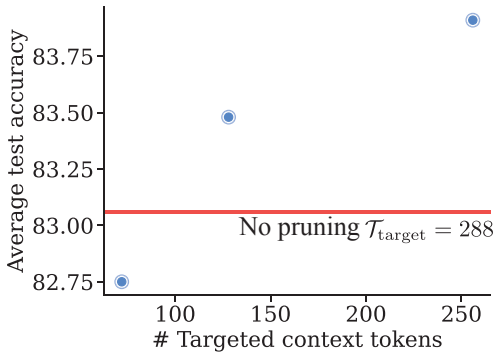

Figure 14: Effect of $\mathcal{T}_{\text{target}}$ on the performance. $\mathcal{T}_{\text{target}} \in \{72, 128, 256\}$ is examined. To ensure $\mathcal{T}_{\text{effective}} = \mathcal{T}_{\text{target}}$, we adjust the pruning rate accordingly. Horizontal line indicates the performance of Adapt without applying pruning and $\mathcal{T}_{\text{target}} = 288$.

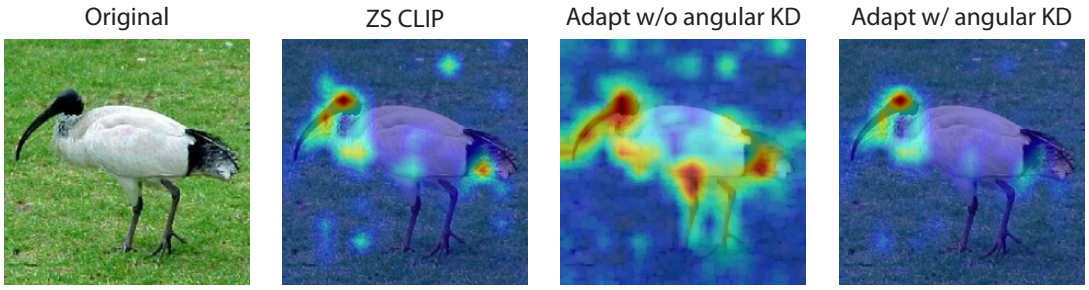

Figure 15: Grad-CAM visualization of zero-shot CLIP, Adapt method with and without angular KD.

- **Discrete prompting**: Discrete prompting is a common strategy in prompt engineering. It aims to search for an optimal prompt to elicit the potential of foundational models. The search space focuses on pure natural language words.
- **Continuous prompting**: continuous prompting, also called soft prompting, inserts differentiable context tokens in the prompt. Compared to discrete prompting that has a discrete search space, continuous prompting has a continuous search space. Hence, the optimized continuous context tokens might not correspond to any word tokens.
- **Shallow prompt tuning**: Shallow prompt tuning inserts context tokens only in the input to foundational models. Given a text prompt $[[w_1], \ldots, [w_k]]$, where $[w_i]$ is the word embedding for token w_i. One way of inserting shallow prompt is $[[e_1], \ldots [e_m], [w_1] \ldots, [w_k]]$, where $[e_i]$ is the inserted continuous prompting that is optimized during the training process.
- **Deep prompt tuning**: Deep prompt tuning inserts context tokens into not only the input to foundational models but also that to deeper model blocks. The deep prompting can be expressed by $\mathbf{x}^{(l)} = \mathcal{F}^{(l)}([\mathbf{x}^{(l-1)}, \mathbf{P}^{(l)}])$

A.12 Forgetting Issue

We empirically examine the forgetting issue in the fine-tuning process of the Adapt method. Figure 15 shows the comparison of the model's attention. Without applying angular knowledge distillation, the model's attention can be scattered, whereas zero-shot CLIP keeps centralized attention on salient features of the image object. We believe that the scattered attention is caused by the forgetting issue in the fine-tuning process.

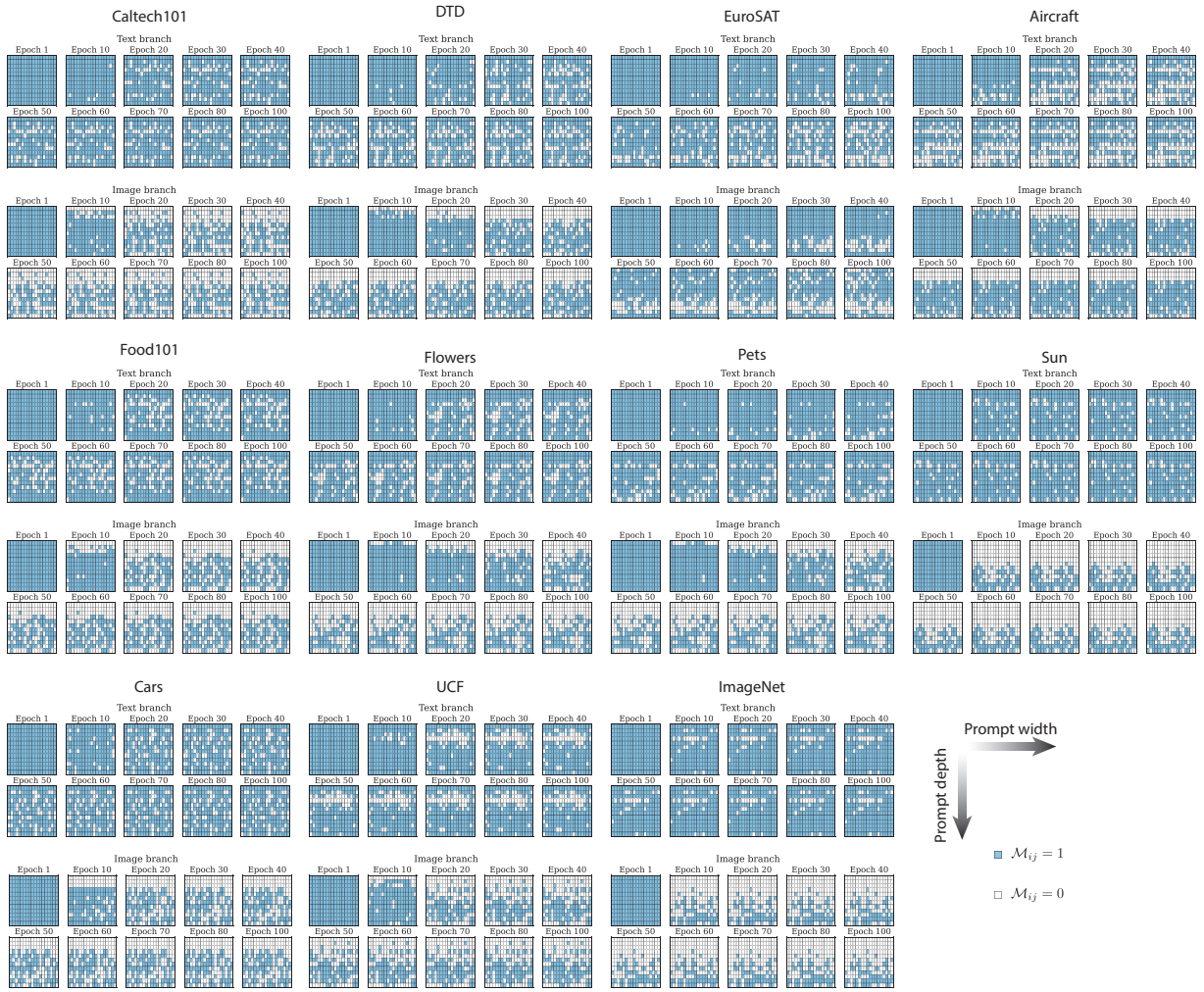

Figure 16: Pruning process of binary masks for the text and image branches for 11 datasets. $\mathcal{T}_{\text{target}} = 256$ and the warm-up epoch is 5. Owing to the warm-up process, $\mathcal{M}(t)$ at the epoch of 1 is same as the $\mathcal{M}(0)$, *i.e.* there is no pruning in inserted prompts.