Preprints of the 23rd IFAC World Congress
Busan, Republic of Korea, August 23-28, 2026

Spot-catching Prompts: Efficient Vision-Language Prompt Learning ⋆

Mengmeng Zhang *,*** **Jing Wang** *,** **Fei-Yue Wang** *,**
Yisheng Lv *,**

* *School of Artificial Intelligence, University of Chinese Academy of Science, Beijing 100190, China*
** *the Faculty of Innovation Engineering, Macau University of Science and Technology, Macau 999078, China*
*** *Institute of Automation, Chinese Academy of Sciences, Beijing, China*
(e-mails: zhangmengmeng2022@ia.ac.cn, wangjing2014@ia.ac.cn, feiyue.wang@ia.ac.cn, yisheng.lv@ia.ac.cn)

Abstract: To enhance the few-shot learning capability of Vision-Language (V-L) models in downstream tasks, existing methods merely introduce additional modules. While this improves the model's predictive performance, it comes at the expense of increased computational cost. We propose an efficient visual-language prompt learning method, EfficientSCP (Efficient Prompt Learning by Catching Spot Features) to address this issue. EfficientSCP dynamically captures essential information from images and eliminates redundant information while performing prompt learning. This effectively improves both the generalization capability and computational efficiency of V-L models. We have conducted extensive experiments on 11 datasets, showing that our method outperforms previous methods. Specifically, EfficientSCP achieves average gains of +0.38%, +0.65%, +1.65%, and +3.94% over the state-of-the-art methods on novel class accuracy in accuracy-efficiency trade-off tasks.

Keywords: prompt learning, token merging, catching spot features

1. INTRODUCTION

Prompt learning, inspired by advances in Natural Language Processing (NLP), has become an effective strategy for adapting large pre-trained models to downstream tasks. Recent studies Shi and Lipani (2024); Han et al. (2023); Li et al. (2024); Hao et al. (2024) show that introducing a small set of learnable parameters while freezing the majority of the model significantly reduces fine-tuning cost. Despite these advantages, existing prompt learning methods Zhou et al. (2022b,a); Khattak et al. (2023a,b) still rely on large vision–language (V-L) backbones Radford et al. (2021); Jia et al. (2021); Gao et al. (2024); Geng et al. (2023); Zhai et al. (2023), leading to considerable computational overhead during training and inference.

To address this limitation, we draw inspiration from recent progress in multi-modal prompting and model acceleration Rao et al. (2021); Singh et al. (2024); Zhao et al. (2024); Liu et al. (2024); Huang et al. (2024); Ding et al. (2023) and propose EfficientSCP, a new V-L prompt learning framework designed to substantially enhance efficiency. Our method is motivated by two insights: (1) merging redundant visual tokens can effectively accelerate computation Bolya et al. (2022); Gu and Dao (2024); and (2) the vision and language branches in V-L models exhibit layer-wise correspondence Khattak et al. (2023a).

Leveraging this correspondence, EfficientSCP re-aligns learnable text prompts to the visual domain to generate vision dispatchers that selectively preserve critical spot features during token merging. In the language branch, we progressively introduce learnable prompts across layers to produce domain-adapted text embeddings. These embeddings are projected into the visual space and used as input-aware dispatchers that guide a spot-catching soft-match module, enabling selective protection of important visual cues while merging redundant tokens. This design maintains performance while accelerating both training and inference.

On the accuracy-efficiency trade-off evaluation benchmark across ten datasets, our method achieves average gains of +0.38%, +0.65%, +1.65%, and +3.94% over state-of-the-art methods across 11 datasets when merging 2, 4, 6, and 8 tokens in the vision branch, respectively. Furthermore, our method does not suffer a significant loss in generalization ability, making it an acceptable trade-off given the substantial savings in storage and the improvements in training and inference efficiency that we achieve.

In summary, the main contributions of this work include:

⋆ This work was partially supported by the National Natural Science Foundation of China (No.82372051); Beijing Natural Science Foundation (No.L222099); The Science and Technology Development Fund, Macao SAR (No. 0093/2023/RIA2, No. 0145/2023/RIA3, No. 0157/2024/RIA2); Capital's Funds for Health Improvement and Research (No.2024-2-4015); Sichuan Key-Area Research and Development Program(No.2024YFHZ0011)

©2026 the authors. Accepted by IFAC for publication under a Creative Commons Licence CC-BY-NC-ND

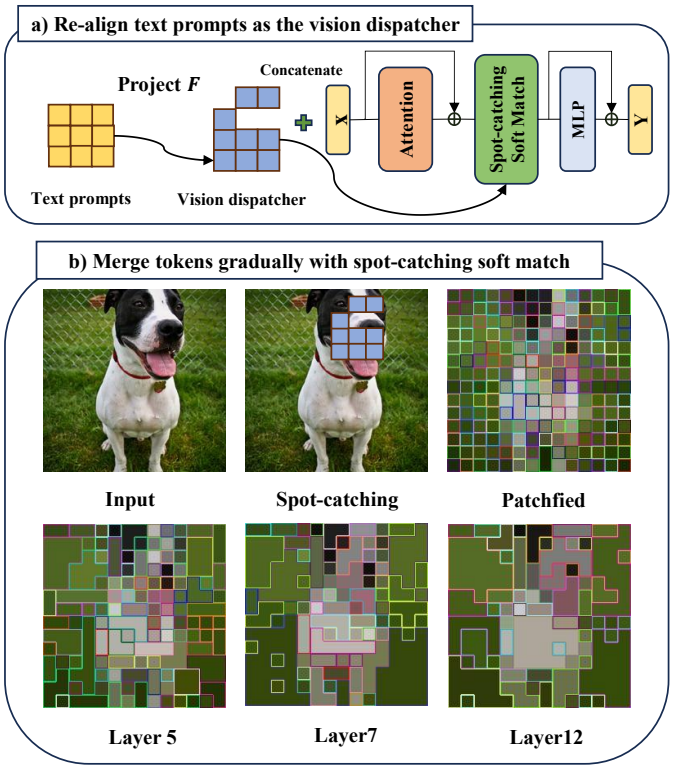


Fig. 1. EfficientSCP Algorithm Diagram.(a) is the generating process of spot-catching vision dispatcher. We use the dispatcher in the spot-catching soft match module for catching and protecting spot features in the vision branch. (b) Utilizing a spot-catching vision dispatcher to catch and protect critical features in the vision branch while merging the redundant tokens: for example, the dog's facial features are still well preserved, even with extensive tokens being merged.

- We propose a novel efficient vision-language prompt learning approach which improves both training and inference efficiency.
- We have introduced a spot-catching soft-match module. This module guides the re-alignment of text prompts and protects the spot features before merging redundant tokens.
- We have improved model efficiency during both training and inference processes, achieving an excellent accuracy-efficiency trade-off.

2. METHOD

In this section, we sequentially introduce the technical principles and details of EfficientSCP. EfficientSCP aims to enhance the performance and efficiency of V-L models in downstream tasks. Figure 2 illustrates our proposed method's architecture, EfficientSCP.

2.1 Text prompt learning

We incorporate learnable prompts from the initialization phase and introduce them incrementally at each layer. Compared to directly optimizing embeddings at the end, producing learnable text prompts at each layer facilitates a more nuanced and deeper understanding of downstream data by the language branch. These layer-wised generated text prompts, can effectively provide corresponding visual information when remapped to the visual representation domain. As a result, it facilitates the generation of a more refined dispatcher for matching spot features in the vision branch and eliminating redundancy.

We introduce c learnable prompts $\{p^i \in \mathbb{R}^{d_l}\}_{i=1}^{c}$ in the language branch to extract common visual features. The input embedding follows the form $[P_0, E_0]$, where $E_0 = [e^1, e^2, ..., e^{N_t}]$ corresponds to fixed input tokens and $P_0 = [p_0^1, ..., p_0^c]$ is the adaptive learnable prompts set in layer 0. N_t refers to the number of text tokens. The adaptive learnable prompts are further introduced into each transformer block of the language encoder(L_i) up to a specific depth J. In each encoder layer j, we introduce new learnable prompts set P_j to catch variable common features. After J^{th} transformer layer, the subsequent layers process previous layer prompts and the final text representation $t = \text{TextProj}(e_N^K)$ is computed, where N refers to the Nth layer, and K refers to the position of EOF token. We conducted an ablation study on the parameter J to explore the impact of J's configuration on the model's performance.

2.2 Vision dispatcher production

By replacing certain initialized tokens in the text branch with learnable prompts, we can obtain task-relevant shared feature embeddings for new data during the training process. However, since the inserted learnable prompts are shared across all categories, the resulting embeddings represent a generalized description of shared features applicable to all categories. To utilize the shared descriptions acquired in the text branch for capturing category-specific key information in the vision branch, we concatenate the text prompts P_j mapped to the visual representation space with the visual inputs $V_j = [v_j^1, v_j^2, ...v_j^{N_v}]$, where N_v refers to the number of vision tokens. The concatenated inputs are then fed into the existing visual self-attention module. By processing through the self-attention module, the mapped prompts are transformed into input-specific visual dispatchers and we separate them as vision dispatchers D_j. The following equation can express the entire process:

$$[V_j, D_j] = \text{self-attention}([V_j, F_j(P_j)])$$

F_j refers to the projection function of layer j.

2.3 Token Merging with Spot-catching

After obtaining the input-specific visual dispatchers, we can utilize these dispatchers to catch spot features and preserve them before merging redundant information. By calculating the cosine similarity between the regularized vision tokens V_j and the dispatchers D_j, we can obtain the relevance matrix $M_j = \text{sim}(V_j, \text{Norm}(D_j))$.

Then we compute the average along the rows of the relevance matrix M_j and sort the result to get the significance index vector s_j, which evaluates the significance index of vision tokens $V_j \in \mathbb{R}^{N_v \times d_v}$, as illustrated in Fig.2.

We select the top k vision tokens as protection targets based on the sorted vector s_j and merge the remaining vision tokens. Inspired by ToMe Bolya et al. (2022), we

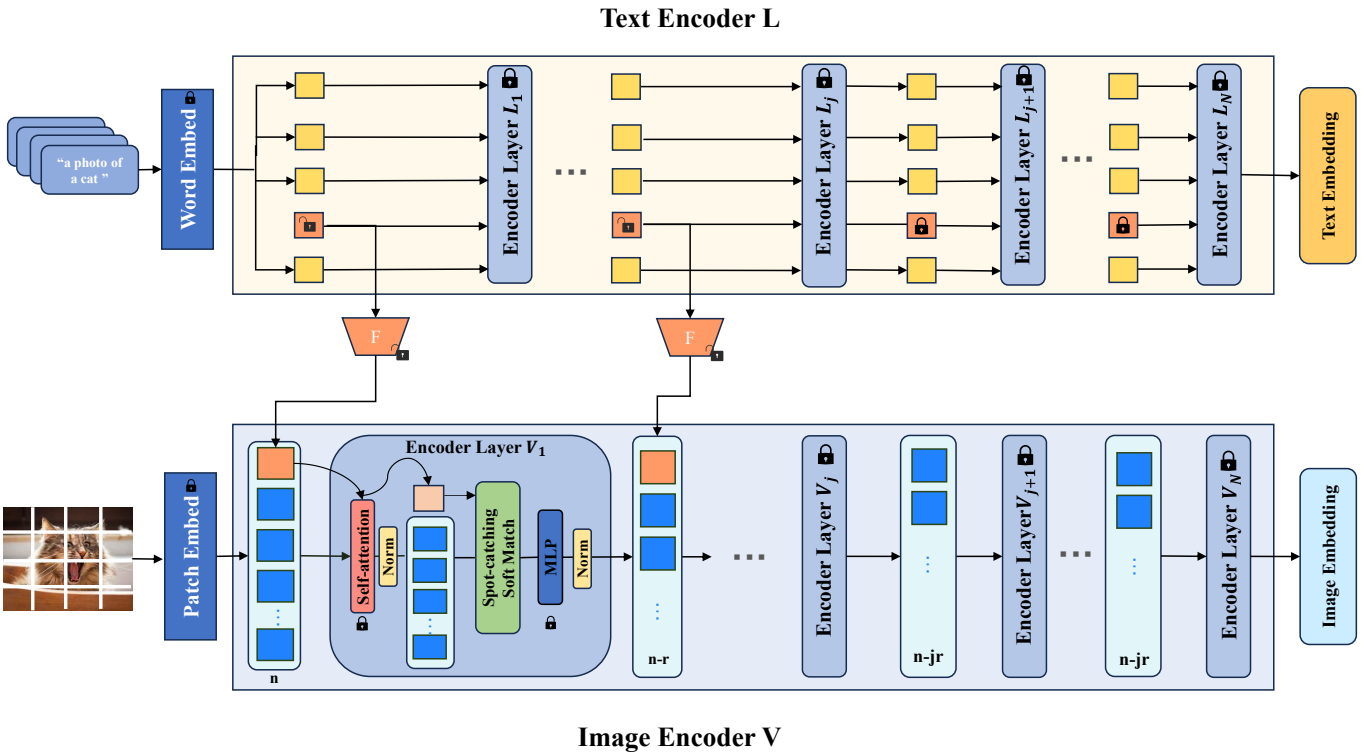


Fig. 2. Overall architecture of EfficientSCP. We first append learnable prompts in every layer of the language branch up to the Jth layer. Then we use the projection function F to re-align the learnable text prompts to visual representation space, and produce a visual dispatcher to merge r tokens in each layer by spot-catching soft match module. This helps achieve better model preference while improving training and inference efficiency.

utilize a bipartite soft matching method to merge the unprotected vision tokens. We first split the key value of these unprotected vision tokens into two parts, and find r most similar token pairs, then merge these tokens. To reduce the bias in attention computation caused by token merging, we also introduce proportional attention: $A = \text{softmax}(\frac{QK^T}{\sqrt{d}} + \log s)$, where s is a row vector containing the merging times of each token.

We do not employ a dynamic approach for token merging, which allows us to perform batch training and inference. Since the key information matched is carefully scrutinized, our method does not risk blurring or destroying important information.

2.4 Skip-loss

To ensure that dispatchers remapped to the visual representation space can effectively protect important features based on input specificity and enhance the model's performance, we have designed a skip-loss for these dispatchers. Collecting dispatchers layer by layer and calculating the cosine similarity between these dispatchers and the final text embeddings, we compute the cross-entropy loss for the input labels, thereby facilitating contrastive learning for the dispatchers.

The calculation formula for the final local loss is defined in Eq.(1), for it is the average of the local losses generated by the dispatchers across all J layers.

$$L_{\text{skip_loss}} = \frac{1}{J}\sum_{j=1}^{J} L_{\text{CE}}((proj_{v\to t}(D_j)\cdot t^T), \hat{y}) \quad (1)$$

Function $proj_{v\to t}$ transforms $D_j \in \mathbb{R}^{d_v}$ to $\mathbb{R}^{d_l}$. The overall loss to train the learnable prompts is the sum of local and global losses:

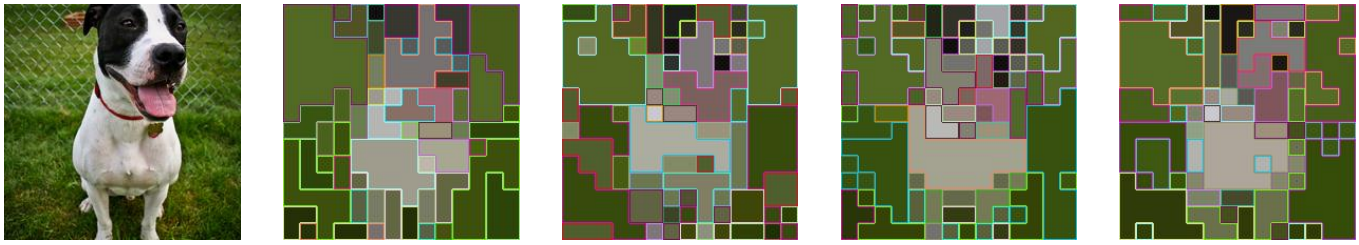
$$\begin{aligned} L_{\text{total_loss}} &= L_{\text{global_loss}} + L_{\text{skip_loss}} \\ L_{\text{global_loss}} &= L_{\text{CE}}(y, \hat{y}) \end{aligned} \quad (2)$$

Here, y is the output of the model, and $\hat{y}$ is the label of input. The optimization process of skip loss is synonymous with the adaptive adjustment of learnable prompts to re-align with the corresponding key information of images. We illustrate the protection of critical features by using a spot-catching token merging approach in Fig 1.

3. EXPERIMENTS

3.1 Benchmark configurations

We conduct an exhaustive empirical evaluation of our proposed approach on 11 datasets and provide a comparative analysis with existing methods. Our proposed EfficientSCP method primarily reveals that focusing on key parts of the input during the prompt learning process, as opposed to adding extra information, can more effectively enhance the efficiency of transferring Visual Language (V-L) models to downstream tasks while maintaining satisfactory performance. Therefore, in our experiments, we chose to compare with prompt learning methods that



(a) Prototype (b) CoOp (c) MaPLe (d) PromptSRC (e) Ours

Fig. 3. Visualization comparison between different methods: We randomly selected test images and further visualized the merging effects for each model. It is evident that models that merge solely based on token similarity do not achieve satisfactory results; they lose a significant amount of detail and may even introduce errors. For example, (b) CoOp model, the facial details of the dog are almost entirely lost. In (c) MaPLe, the dog's face includes the background of grass, and in (d) PromptSRC, the dog's facial features are misaligned. In contrast, after applying the same level of token merging, our model (e) preserves the facial details of the dog clearly and does not introduce incorrect environmental information.

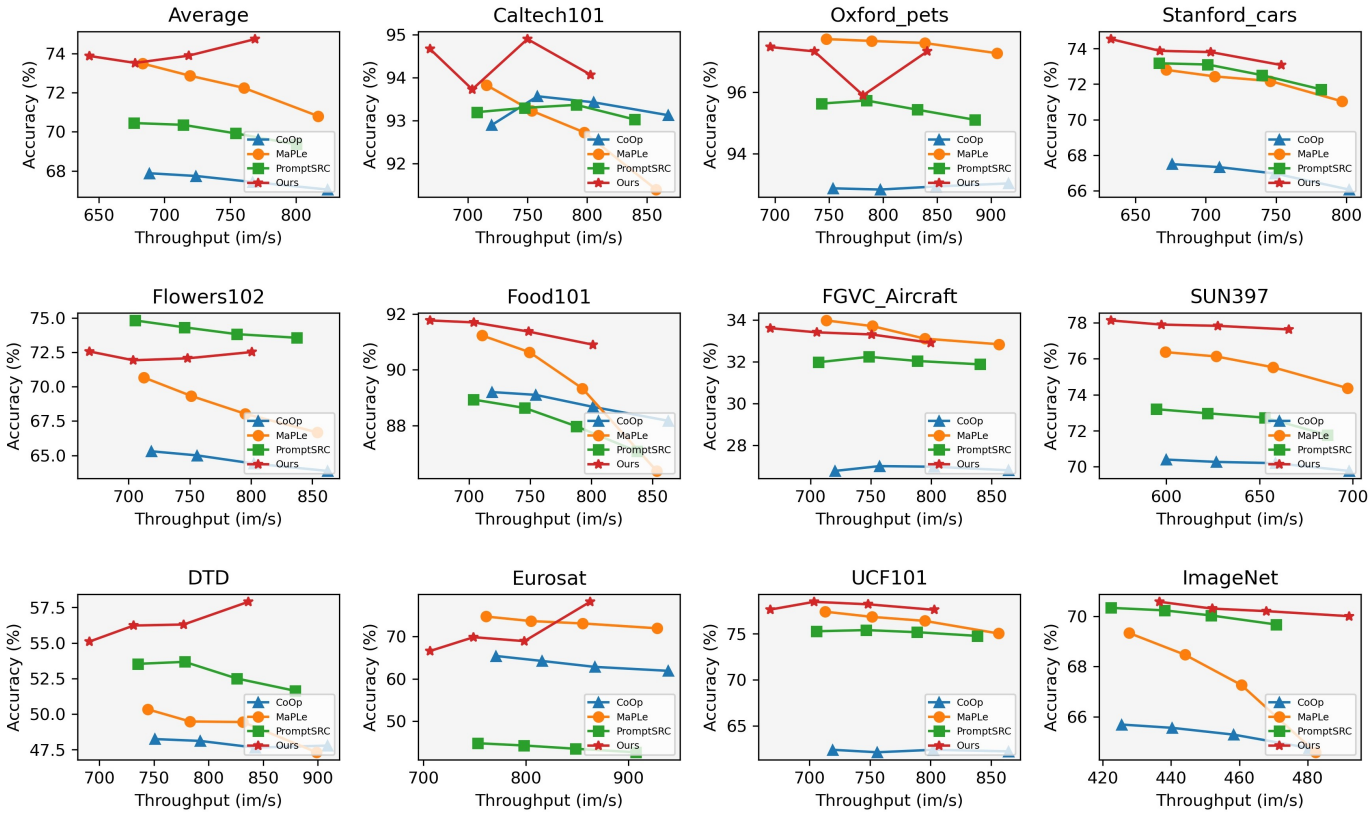


Fig. 4. EfficientSCP performance comparison on accuracy-efficiency trade-off evaluation. All the previous methods have incorporated the ToMe module and we compared the performance of various models under the conditions where 2, 4, 6, and 8 tokens are merged from the vision branch. The results show that our method significantly outperforms others across 11 datasets, and we achieve the best performance on 8 out of the 11 datasets.

have the same initial settings: CoOp, CoCoOp, MaPLe and PromptSRC. Among the aforementioned models with uniform initial settings, CoCoOp is incapable of batch training and testing, which we believe does not align with the principle of efficiency.

Accuracy-efficiency trade-off evaluation In order to test the performance of accuracy-efficiency trade-off of our approach, we augment the existing methods with a ToMe module and conduct a comparative analysis of their capabilities after the reduction of 2,4,6,8 tokens in vision branch.

Ablation on Spot-catching Window Size To investigate the impact of the spot-catching window size k on the model, we conducted a performance comparison on 11 datasets with the model configured to merge 8 tokens, evaluating different values of k such as 3, 5, 7, and 9 on 10 datasets except ImageNet.

Ablation on Merge-depth To explore the performance variation of the model with different numbers of token

merge layers, we conducted comparative experiments by setting the spot-catching window size to 8 and merging 8 tokens at each layer. We compared the results for merge layer counts j = 3, 6, 9, and 12 on 10 datasets except ImageNet.

Datasets: We follow CoOp Zhou et al. (2022b) and CoCoOp Zhou et al. (2022a), and evluate the performance of our approach on 11 image recognition datasets. These datasets cover multiple recognition tasks including ImageNet Deng et al. (2009) and Caltech101 Fei-Fei et al. (2004) which consists of generic objects; five fine-grained datasets, OxfordPets Parkhi et al. (2012), StanfordCars Krause et al. (2013), Flowers102 Nilsback and Zisserman (2008), Food101 Bossard et al. (2014), and FGVCAircraft Maji et al. (2013); SUN397 Xiao et al. (2010) for scene recognition, UCF101 Soomro et al. (2012) for action recognition, DTD Cimpoi et al. (2014) for texture classification, and EuroSAT Helber et al. (2019) which consists of satellite images.

3.2 Accuracy-Efficiency Trade-off Evaluation

We compare the accuracy-efficiency trade-off performance of our approach with CoOp Zhou et al. (2022b), MaPLe Khattak et al. (2023a), and PromptSRC Khattak et al. (2023b), and show the results in Fig 4. Overall, our method achieves the best trade-off between performance and efficiency in 8/11 datasets. Specifically, EfficientSCP achieves average gains of +0.38%, +0.65%, +1.65%, and +3.94% over the state-of-the-art methods on novel class accuracy in base-to-novel tasks, when merging 2, 4, 6, and 8 tokens in vision branch.

It is noteworthy that, aside from the datasets Flowers102, Oxford_pets, and FGVC_Aircraft, our model substantially outperforms the state-of-the-art on the remaining 8 datasets. Furthermore, as the degree of acceleration increases, the advantages of our method become more pronounced. Compared to previous token merge methods that directly merge tokens based on token similarity, our results demonstrate that our novel spot-catching soft-match module facilitates the maintenance of model performance.

Generally, as more information is aggregated, the loss of detail becomes increasingly significant. The performance of models that rely on these classification details will degrade. However, our proposed spots-catching soft-match module makes the aggregation process more targeted, allowing more effective details to be retained. This ensures better model performance while enhancing efficiency. Even during the process of eliminating redundant information, our model's performance can improve compared to lower removal rates. This is evident in datasets like Flower102 and Oxford Pets, where state-of-the-art performance had not been previously achieved.

3.3 Spot-catching window size k

The spot-catching window size k can influence the model's preference directly. It is an obvious conjecture that, the more protection is afforded, the fewer key micro-level details are lost due to merging, thereby enhancing the model's performance. We conduct experiments to test the performance of merging 8 tokens on the novel class at $k = 3, 5, 7, 9$ and present the results in Table 1.

Method	Base Acc.	Novel Acc.	HM
Merge(8)+$k = 3$	81.99	73.97	77.77
Merge(8)+$k = 5$	81.93	73.82	77.66
Merge(8)+$k = 7$	81.92	73.35	77.40
Merge(8)+$k = 9$	**82.06**	**75.21**	**78.49**

Table 1. Ablation on spot-catching window size k.

Method	Base Acc.	Novel Acc.	HM
Merge(8)+$j = 3$	**82.27**	73.91	77.87
Merge(8)+$j = 6$	81.90	74.23	77.88
Merge(8)+$j = 9$	82.06	**75.21**	**78.49**

Table 2. Ablation on prompting and merging depth j.

It can be seen that the value of k can affect the model's performance to a certain extent. However, contrary to our expectations, comparative experiments have demonstrated that a larger window size does not necessarily contribute to the protection of critical information. A window that is neither too large nor too small can even reduce the model's performance.

3.4 Merging Depth j

The number of learnable prompts inserted in the text branch can significantly affect the model's performance. Therefore, we conducted ablation experiments to determine the optimal number of layers j for text prompt insertion. In our study, we set $j = 3, 6, 9$ and evaluated the model's performance across 10 datasets. We present the results in Table 2. The results indicate that increasing the depth improves the model's performance, but it also increases computation time. After weighing the options, we chose a depth of 9.

4. CONCLUSION

Prompt learning adapts vision-language (VL) models to downstream tasks but often overlooks the large parameter count in VL models. Token merge methods enhance efficiency but can lead to the loss of critical details. Our work proposes an efficient prompt learning approach that catches spot features before merging redundant tokens. We demonstrate the importance of preserving critical information for enhanced model performance after token merge. Our EfficientSCP achieves state-of-the-art results in the accuracy-efficiency trade-off evaluation across multiple datasets.

5. ACKNOWLEDGEMENT

This work was partially supported by the National Natural Science Foundation of China (No.82372051); Beijing Natural Science Foundation (No.L222099); Capital's Funds for Health Improvement and Research (No.2024-2-4015); Sichuan Key-Area Research and Development Program(No.2024YFHZ0011).

REFERENCES

Bolya, D., Fu, C.Y., Dai, X., Zhang, P., Feichtenhofer, C., and Hoffman, J. (2022). Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*.

Bossard, L., Guillaumin, M., and Van Gool, L. (2014). Food-101–mining discriminative components with random forests. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VI 13*, 446–461. Springer.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. (2014). Describing textures in the wild. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 3606–3613.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. Ieee.

Ding, N., Tang, Y., Han, K., Xu, C., and Wang, Y. (2023). Network expansion for practical training acceleration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 20269–20279.

Fei-Fei, L., Fergus, R., and Perona, P. (2004). Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on Computer Vision and Pattern Recognition workshop*, 178–178. IEEE.

Gao, Y., Liu, J., Xu, Z., Wu, T., Zhang, E., Li, K., Yang, J., Liu, W., and Sun, X. (2024). Softclip: Softer cross-modal alignment makes clip stronger. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 1860–1868.

Geng, S., Yuan, J., Tian, Y., Chen, Y., and Zhang, Y. (2023). Hiclip: Contrastive language-image pretraining with hierarchy-aware attention. *ICLR*.

Gu, A. and Dao, T. (2024). Mamba: Linear-time sequence modeling with selective state spaces.

Han, C., Wang, Q., Cui, Y., Cao, Z., Wang, W., Qi, S., and Liu, D. (2023). E^ 2vpt: An effective and efficient approach for visual prompt tuning. *ICCV*.

Hao, T., Ding, X., Feng, J., Yang, Y., Chen, H., and Ding, G. (2024). Quantized prompt for efficient generalization of vision-language models. *ECCV*.

Helber, P., Bischke, B., Dengel, A., and Borth, D. (2019). Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7), 2217–2226.

Huang, W., Shen, Y., Xie, J., Zhang, B., He, G., Li, K., Sun, X., and Lin, S. (2024). A general and efficient training for transformer via token expansion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15783–15792.

Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., and Duerig, T. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, 4904–4916. PMLR.

Khattak, M.U., Rasheed, H., Maaz, M., Khan, S., and Khan, F.S. (2023a). Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19113–19122.

Khattak, M.U., Wasim, S.T., Naseer, M., Khan, S., Yang, M.H., and Khan, F.S. (2023b). Self-regulating prompts: Foundational model adaptation without forgetting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 15190–15200.

Krause, J., Stark, M., Deng, J., and Fei-Fei, L. (2013). 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision workshops*, 554–561.

Li, Z., Li, X., Fu, X., Zhang, X., Wang, W., Chen, S., and Yang, J. (2024). Promptkd: Unsupervised prompt distillation for vision-language models. *CVPR*.

Liu, T., Liu, X., Shi, L., Xu, Z., Huang, S., Xin, Y., and Yin, Q. (2024). Sparse-tuning: Adapting vision transformers with efficient fine-tuning and inference. *arXiv preprint arXiv:2405.14700*.

Maji, S., Rahtu, E., Kannala, J., Blaschko, M., and Vedaldi, A. (2013). Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*.

Nilsback, M.E. and Zisserman, A. (2008). Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, 722–729. IEEE.

Parkhi, O.M., Vedaldi, A., Zisserman, A., and Jawahar, C. (2012). Cats and dogs. In *2012 IEEE Conference on Computer Cision and Pattern Recognition*, 3498–3505. IEEE.

Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 8748–8763. PMLR.

Rao, Y., Zhao, W., Liu, B., Lu, J., Zhou, J., and Hsieh, C.J. (2021). Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34, 13937–13949.

Shi, Z. and Lipani, A. (2024). Dept: Decomposed prompt tuning for parameter-efficient fine-tuning. *ICLR*.

Singh, M.K., Yasarla, R., Cai, H., Lee, M., and Porikli, F. (2024). Tosa: Token selective attention for efficient vision transformers. *arXiv preprint arXiv:2406.08816*.

Soomro, K., Zamir, A.R., and Shah, M. (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., and Torralba, A. (2010). Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE conference on Computer Vision and Pattern Recognition*, 3485–3492. IEEE.

Zhai, X., Mustafa, B., Kolesnikov, A., and Beyer, L. (2023). Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11975–11986.

Zhao, W., Tang, J., Han, Y., Song, Y., Wang, K., Huang, G., Wang, F., and You, Y. (2024). Dynamic tuning towards parameter and inference efficiency for vit adaptation. *NeurIPS*.

Zhou, K., Yang, J., Loy, C.C., and Liu, Z. (2022a). Conditional prompt learning for vision-language models. *CVPR*.

Zhou, K., Yang, J., Loy, C.C., and Liu, Z. (2022b). Learning to prompt for vision-language models. *IJCV*.