

Parameter-Free Fine-tuning via Redundancy Elimination for Vision Foundation Models

Jiahuan Long^{1, 2}, Tingsong Jiang^{2*}, Wen Yao^{2*}, Yizhe Xiong³,
Zhengqin Xu¹, Shuai Jia¹, Hanqing Liu^{1, 2}, and Chao Ma^{1*}

¹MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University,

²Defense Innovation Institute, Chinese Academy of Military Science,

³School of Software, Tsinghua University

Abstract

Vision foundation models (VFMs) have demonstrated remarkable capabilities in learning universal visual representations. However, adapting these models to downstream tasks conventionally requires parameter updates, with even parameter-efficient fine-tuning methods necessitating the modification of thousands to millions of weights. In this paper, we investigate the redundancies in the segment anything model (SAM) and then propose a novel parameter-free fine-tuning method. Unlike traditional fine-tuning methods that adjust parameters, our method emphasizes **selecting**, **reusing**, and **enhancing** pre-trained features, offering a new perspective on fine-tuning foundation models. Specifically, we introduce a channel selection algorithm based on the model’s output difference to identify redundant and effective channels. By selectively replacing the redundant channels with more effective ones, we filter out less useful features and reuse more task-irrelevant features to downstream tasks, thereby enhancing the task-specific feature representation. Experiments on both out-of-domain and in-domain datasets demonstrate the efficiency and effectiveness of our method in different vision tasks (e.g., image segmentation, depth estimation and image classification). Notably, our approach can seamlessly integrate with existing fine-tuning strategies (e.g., LoRA, Adapter), further boosting the performance of already fine-tuned models. Moreover, since our channel selection involves only model inference, our method significantly reduces GPU memory overhead.

Introduction

Vision foundation models (VFMs) (Kirillov et al. 2023; Ravi et al. 2024; Radford et al. 2021; Oquab et al. 2023) are large-scale and generalizable models designed to adapt to various visual tasks, such as image classification, detection, and segmentation. Among the VFMs, Segment Anything Model (SAM) (Kirillov et al. 2023; Ravi et al. 2024) significantly advances the research of vision foundation models on various visual tasks (Yang et al. 2023; Ren et al. 2024; Yu et al. 2023). Fine-tuning SAM can further unleash its potential in new specific tasks or scenarios, such as camouflage detection (Chen et al. 2023b) and medical image segmenta-

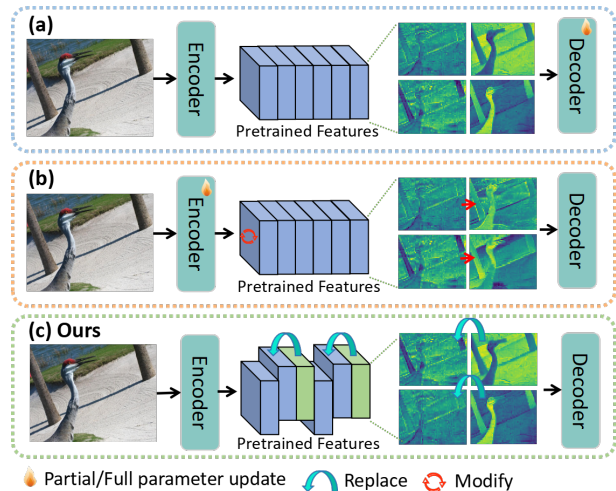


Figure 1: A comparison between our method and other fine-tuning methods. (a) shows a fine-tuning method that updates the decoder to align pre-trained features with the downstream task; (b) illustrates a fine-tuning method that updates the encoder to modify pre-trained features for downstream adaptation. In contrast, (c) depicts our method, which adapts pre-trained features to the downstream task by replacing specific redundant channels without any parameter updates.

tion (Zhang and Liu 2023; Ma et al. 2024), achieving superior performance than prior task-specific models.

Recently, parameter-efficient fine-tuning (PEFT) (Xiong et al. 2024; Hu et al. 2021; Houlsby et al. 2019; Han et al. 2023; Peng et al. 2024b) methods have become mainstream approaches for adapting VFMs to downstream tasks. Compared to full-parameter fine-tuning, PEFT updates only a small subset of parameters while keeping the majority of pre-trained parameters unchanged, significantly reducing computational overhead. For example, SAMed (Zhang and Liu 2023) customizes SAM for medical image segmentation using only 4.4% of SAM’s parameters via low-rank adaptation (Hu et al. 2021). Another approach, Sam-PARSER (Peng et al. 2024b), further reduces parameter usage to just 512 learnable parameters while still achieving substantial performance gains on medical datasets. As PEFT

*Corresponding authors.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Channel number	6	216	175	19	189
Baseline	50.6	50.6	50.6	50.6	50.6
Inactive (value = 0)	50.6 =	52.7 ↑	48.7 ↓	48.4 ↓	47.9 ↓

Table 1: Performance comparison of SAM on the PerSeg dataset with inactive channels.

methods continue to shrink the number of trainable parameters, a radical idea arises: *can a VFM be adapted to downstream tasks without modifying any model parameters?*

From the perspective of feature representation, existing works indicate that deep learning models exhibit significant feature redundancy (Li, Wen, and He 2023; Dalvi et al. 2020). This is largely because networks trained on large-scale datasets tend to learn universal features to generalize across different tasks, many of which may be irrelevant or even detrimental to specific downstream objectives. In particular, vision foundation models, designed to capture universal features from large-scale training data, retain substantial task-irrelevant features when adapted to smaller downstream datasets. To illustrate this point, we conduct a controlled experiment on the Segment Anything Model (SAM), in which we manually deactivate specific feature channels by setting their activation values to zero (i.e., effectively turning off those channels). As shown in Table 1, we observed that when some channels are inactive, the overall performance on PerSeg dataset remains unchanged (channel 6) and even improves (channel 216). This indicates that some channels in SAM are redundant or noisy for downstream tasks, as removing them does not impair performance (Refer to Figure 5 for visualizations of these redundant channels).

This observation naturally raises a question: **does replacing the redundant or even noisy channels with more effective ones improve the adaptability of foundation models for downstream tasks?** To this end, we propose a novel parameter-free fine-tuning paradigm to enhance model adaptability through feature selection, reuse, and enhancement. Specifically, we design a channel selection scheme based on the model’s output differences to separate effective and redundant channels. By selectively replacing redundant channels, our method removes ineffective features and reuses more relevant features to downstream tasks, thereby enhancing task-specific feature representation. Figure 1 illustrates how our method adapts VFMs to downstream tasks, in contrast to previous fine-tuning that updates parameters. The contributions of this work are summarized as follows:

- We identify the feature redundancy in VFMs and propose a parameter-free fine-tuning method for SAM and SAM2. Unlike parameter updating methods, our method focuses on selecting, reusing, and enhancing existing features.
- We introduce a channel selection algorithm based on output differences to search for optimal replacement pairs. This process only involves network inference over the search dataset, yielding little computational overhead compared to other fine-tuning approaches.
- Extensive experiments on nine diverse datasets demon-

strate that the proposed method is simple yet effective. It seamlessly integrates with existing fine-tuning strategies, further boosting the performance of already fine-tuned models.

Related Work

Vision Foundation Models. Segment anything model (SAM) (Kirillov et al. 2023; Ravi et al. 2024) is one of the most widely used foundational vision models for image segmentation (Kirillov et al. 2023). It generates object masks for images based on user-provided box or point prompts. SAM’s success in segmentation has inspired efforts to extend its application to more complex tasks beyond segmentation, such as tracking and detection (Yang et al. 2023; Ren et al. 2024; Yu et al. 2023). Similarly, other VFMs such as DINOv2 (Oquab et al. 2023) have demonstrated impressive performance on multiple vision tasks. This paper explores a novel parameter-free fine-tuning method to adapt these foundation models for downstream datasets.

Fine-tuning Strategies. Fine-tuning aims to adapt the pre-trained model’s weights to specific tasks by selectively updating its parameters. Existing methods can be categorized into full-parameter and parameter-efficient fine-tuning (PEFT) (Ding et al. 2023; Li et al. 2023; Tian et al. 2024). Full-parameter fine-tuning updates all model parameters, offering strong adaptability but incurring high computational costs. As pre-trained models continue to grow in size, PEFT methods, which update only a minimal subset of parameters, have gained importance. Many PEFT methods are inspired by Adapter (Houlsby et al. 2019), which adapts models by inserting adapter layers into transformer blocks as a few trainable parameters. Since then, various PEFT techniques have been developed, achieving favorable performance, including LoRA (Hu et al. 2021; Zhang and Liu 2023), SVD (Han et al. 2023; Peng et al. 2024b), DoRA (Liu et al. 2024), and Consolidator (Tianxiang Hao et al. 2023). Different from previous methods that adjust parameters, in this paper, we introduce a straightforward yet effective fine-tuning method by eliminating feature redundancies.

Channel Operations. Several works have explored channel operations during model training (Zhang et al. 2018; Wang et al. 2020). ShuffleNet (Zhang et al. 2018) introduces a channel shuffle operation, mixing features across groups to ensure cross-channel information fusion. Wang et al. (Wang et al. 2020) propose a channel-exchanging network, a multi-modal fusion framework that dynamically exchanges channels between sub-networks of different modalities. However, our method significantly differs from these channel shuffle methods in three aspects. (1) Instead of randomly swapping channels, our method selectively replaces redundant channels with more effective ones. (2) Channel shuffle is used during model training to generate more effective features, whereas our approach reuses the existing features in pre-trained models. (3) Our goal is to boost the adaptability of foundation models for downstream tasks, whereas channel shuffle methods aim to improve information fusion across multiple modalities or within grouped convolutions.

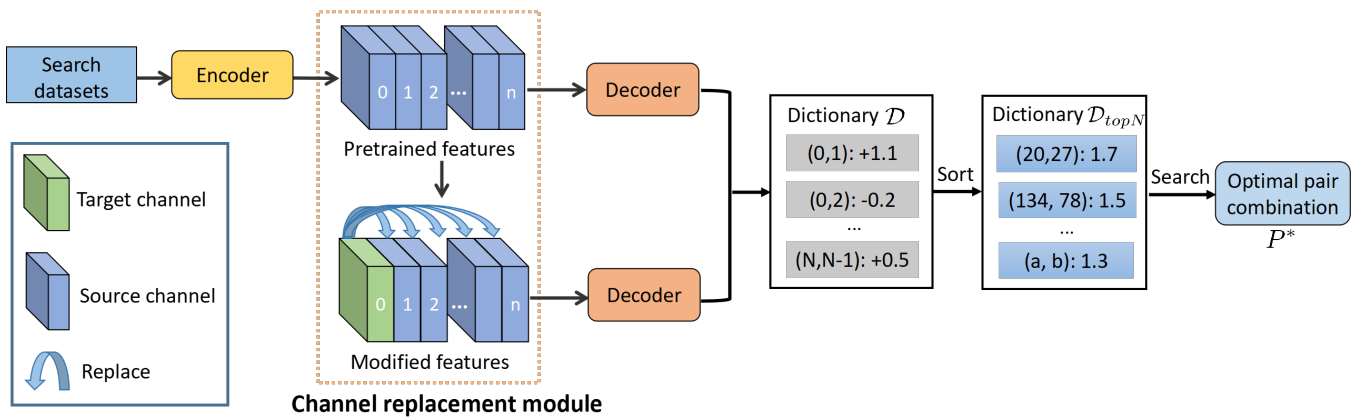


Figure 2: Pipeline of the search process for optimal replacement pair combination. First, we use a subset of the training dataset, referred to as the “search dataset,” which is fed into SAM’s encoder to obtain pretrained features. Next, in the channel replacement module, each source channel is sequentially replaced with a target channel, and the modified features are sent to the decoder. By comparing the decoded results from the original and modified features, we construct a dictionary \mathcal{D} that records each replacement pair and its corresponding output difference. After sorting the top N pairs in \mathcal{D} , we further explore the optimal replacement pair combination P^* within the subset \mathcal{D}_{topN} . By applying P^* , we replace the redundant channels with more effective ones, thereby enhancing the performance on downstream tasks.

Methodology

Problem Formulation

Large VFMs like SAM are trained on massive, diverse training datasets. While this diversity supports generalization, it also means some channels may carry task-irrelevant redundant features for specific downstream tasks. **Our goal is to search for the optimal replacement pairs in which the redundant channels can be replaced by the effective ones, thereby maximizing SAM’s performance on downstream tasks.** We determine a channel to be “redundant” if replacing it with another channel improves the model’s performance, and hence the replacement channel to be “effective.” This implies that the redundant and effective channels always come in pairs. Let P represent a single candidate replacement pair combination: $P = \{(i, j)_1, (i, j)_2, \dots, (i, j)_k \mid i, j, k \in \{1, 2, \dots, C\}\}$, where k represents the number of replacement pairs, and each (i, j) denotes replacing the feature map of the i -th channel with that of the j -th channel. C is the total number of channels. The objective function is defined as:

$$P^* = \arg \max_P \text{mIoU}(S, P), \quad (1)$$

where P^* is the optimal replacement pair combination that maximizes the mean Intersection over Union (mIoU) on the downstream dataset S .

However, enumerating all possible pair combinations is computationally infeasible. For instance, when $C = 256$, the number of combinations is 2^{C^2} , which would require an equivalent number of inferences on the training dataset to get an optimal solution. To address this, we propose several strategies for reducing computational overhead, enabling an efficient search instead of exhaustive enumeration.

Channel Selection Algorithm

Strategies for Reducing Search Overhead. To reduce the overhead of searching for the optimal replacement pair combination, we propose the following optimization strategies: (1) *Search based on output difference.* First, we calculate the output differences between the original model and the model applying a single replacement pair. We then select the top N pairs that yield the highest output improvements. The N pairs form a smaller search space, allowing us to evaluate various combinations of these pairs and find the optimal replacement pair combination. This method significantly reduces the number of inferences from 2^{C^2} to $C^2 + 2^N - 1$. (2) *Sample Reduction.* We reduce the original training dataset to a smaller “search dataset” containing only 50 images, which helps to reduce the model’s inference time. (3) *Feature Storage.* We pre-store the features of the search dataset generated by the encoder. During each inference, we modify these stored features and pass them to the decoder, avoiding repeated feature extraction by the encoder. The specific search process is as follows.

Search for Optimal Replacement Pairs. Given a search dataset S , the feature maps output by SAM’s encoder are represented as $X \in \mathbb{R}^{D \times C \times W \times H}$, where D is the total number of search images, C is the total number of channels, and the W and H are the width and height of the feature map, respectively. Define \mathcal{P} as the set of replacement pairs: $\mathcal{P} = \{(i, j) \mid i, j \in \{1, 2, \dots, C\}\}$, where (i, j) represents replacing the feature map of i -th channel with that of j -th channel. The transformed feature maps can be expressed as:

$$X'_{d,c,w,h} = X_{d,f_{i \rightarrow j}(c),w,h}, \quad (2)$$

where $f_{i \rightarrow j}(\cdot)$ is a mapping function that replaces channels according to the replacement pair (i, j) . Then, for each re-

Model	Backbone	Size (M)	Natural			Medical			Camou			Avg + Δ
			COCO	VOC	PerSeg	ISIC16	BUSI	KVASIR	CAMO	COD10K	CHAME	
SAM	ViT-B	91	41.20	38.88	50.60	57.33	51.11	55.84	44.96	52.52	49.79	49.14
	ViT-B*	91	51.31	56.33	71.68	65.21	58.06	63.08	48.10	55.19	53.78	58.08 + 8.94
	ViT-L	308	46.33	50.39	61.16	60.04	52.66	63.32	54.01	60.44	57.04	56.15
	ViT-L*	308	74.82	75.66	85.14	65.77	57.95	66.75	55.27	64.62	62.53	67.61 + 11.46
	ViT-H	636	46.46	49.38	62.50	61.37	50.31	62.26	52.10	61.23	54.22	55.54
	ViT-H*	636	54.08	59.18	66.94	63.57	55.33	68.69	54.58	65.82	57.94	60.68 + 5.14
SAM2	Hiera-T	39	56.53	58.16	66.67	52.18	54.82	59.32	53.98	58.50	55.47	57.29
	Hiera-T*	39	73.53	76.76	81.77	55.03	57.87	62.08	59.10	64.32	60.18	65.63 + 8.34
	Hiera-S	46	61.72	60.88	72.45	54.38	58.20	63.43	58.30	61.38	58.59	61.04
	Hiera-S*	46	75.41	77.46	84.76	57.78	61.92	66.12	61.91	68.48	64.45	68.69 + 7.65
	Hiera-B+	81	64.92	65.00	74.23	54.55	56.46	55.83	60.38	62.88	60.41	61.62
	Hiera-B+*	81	76.18	80.05	75.28	55.86	61.85	59.71	63.70	66.30	63.58	66.94 + 5.32
	Hiera-L	224	71.24	70.66	77.41	58.84	66.71	60.15	67.84	69.86	67.20	67.77
	Hiera-L*	224	78.01	83.06	86.27	64.39	69.63	64.49	71.07	73.42	71.49	73.53 + 5.76

Table 2: Performance improvement of our fine-tuning for SAM variants on downstream segmentation tasks. ‘‘Natural’’, ‘‘Medical’’ and ‘‘Camou’’ represent the natural, medical, and camouflage image datasets, respectively. ‘‘*’’ indicates the model of applying our fine-tuning method. ‘‘Size’’ represents the total parameter of models. ‘‘Avg + Δ ’’ represents the average mIoU along with the improvement over the baseline. Results demonstrate that the proposed method consistently improves segmentation accuracy across different SAM backbones.

placement pair (i, j) , we calculate the output difference:

$$\Delta \text{Acc}_{(i \rightarrow j)} = D(X') - D(X), \quad (3)$$

where $D(X)$ and $D(X')$ represent the SAM’s output using the original feature maps X and the transformed feature maps X' , respectively.

To identify the most effective replacement pairs, we compute the accuracy difference for every pair in \mathcal{P} , and store them in a dictionary \mathcal{D} :

$$\mathcal{D} = \{(i, j) : \Delta \text{Acc}_{(i \rightarrow j)}\}. \quad (4)$$

Next, we select the top N replacement pairs in \mathcal{D} with the highest $\Delta \text{Acc}_{(i \rightarrow j)}$ to form a new dictionary:

$$\mathcal{D}_{topN} = \{(i, j) : \Delta \text{Acc}_{(i \rightarrow j)}\}. \quad (5)$$

Finally, we evaluate all possible combinations in \mathcal{D}_{topN} on the downstream dataset to identify the optimal replacement pair combination P^* , and it can be represented as: $P^* = \{(i, j)_1, (i, j)_2, \dots, (i, j)_k \mid (i, j) \in \mathcal{D}_{topN}, k \in \{1, 2, \dots, N\}\}$. Compared to the exhaustive enumeration of all replacement pair combinations, our approach reduces the number of inferences from 2^{C^2} to $C^2 + 2^N - 1$, where C^2 represents the number of elements in \mathcal{D} , and $2^N - 1$ represents the number of non-empty combinations in \mathcal{D}_{topN} .

Figure 2 illustrates the complete search process for the optimal replacement pair combination. According to P^* , we identify redundant and effective channels in SAM for specific downstream tasks. By selectively replacing redundant channels with more effective ones, we filter out less useful features and reuse more relevant features, thereby enhancing task-specific feature representation.

Experiments

Experimental Setup

Datasets and Metrics. We evaluate our method across 9

datasets, covering different tasks of natural image segmentation (COCO (Lin et al. 2014), VOC (Everingham et al. 2010) and PerSeg (Zhang et al. 2023)), medical image segmentation (ISIC (Codella et al. 2018), BUSI (Al-Dhabyani et al. 2020) and KVASIR (Jha et al. 2020)) and camouflage detection (CAMO (Le et al. 2019), COD10K (Fan et al. 2022) and CHAME (Skurowski et al. 2018)). Each image is paired with a ground truth segmentation mask of a single object, along with corresponding prompts. Natural image datasets are used to assess SAM’s performance under in-domain scenarios, while medical and camouflage datasets are used to evaluate SAM’s performance under out-of-domain scenarios. All tasks use mean Intersection over Union (mIoU) as evaluation metrics.

Baselines. To validate the effectiveness of our method, we choose SAM (Kirillov et al. 2023) and SAM 2 (Ravi et al. 2024) as baseline models with different backbones for the segmentation task. The backbones of SAM include ViT-B, ViT-L, and ViT-H, whereas SAM 2 includes Hiera-T, Hiera-S, Hiera-B+, and Hiera-L. We also choose Dinov2 (Oquab et al. 2023) as baseline model for depth estimation and image classification tasks. To validate that our method can further improve the performance of already fine-tuned models, we introduce various parameter-efficient fine-tuning schemes including SAMed (Zhang and Liu 2023), SAM-Adapter (Chen et al. 2023a), SAM-PARSER (Peng et al. 2024b), SAM-COBOT (Peng et al. 2024a), DORA (Liu et al. 2024) and partial-parameter fine-tuning methods, including MedSAM (Ma et al. 2024), Encoder-only Tuning, and Decoder-only Tuning.

Implementation Details. For comparison to other fine-tuning, we employ the ViT-Base version of SAM (Kirillov et al. 2023) as our backbones, integrating point prompts for the prompt encoder input. For loss functions, we use the same combination of Dice loss and CE loss as in (Zhang

Method	Natural			Medical			Camou			Avg + Δ
	COCO	VOC	PerSeg	ISIC16	BUSI	KVASIR	CAMO	COD10K	CHAME	
<i>Partial-parameter Fine-Tuning</i>										
Decoder	69.02	75.81	90.14	80.57	78.23	75.83	63.45	65.68	63.81	73.61
Decoder + Ours	70.26	77.26	90.34	81.41	78.54	77.50	64.70	66.53	65.10	74.62 + 1.01
Encoder	51.98	72.21	89.41	85.51	78.61	75.94	58.64	50.65	52.03	68.33
Encoder + Ours	52.94	72.43	89.56	85.79	78.79	77.16	59.49	-	52.77	68.84 + 0.51
MedSAM (Ma et al. 2024)	65.19	74.28	89.94	83.78	78.51	75.84	59.66	65.59	59.57	72.48
MedSAM (Ma et al. 2024) + Ours	67.39	76.38	90.13	-	78.78	77.30	60.46	67.60	63.59	73.93 + 1.45
<i>Parameter-efficient Fine-Tuning</i>										
SAMed (Zhang and Liu 2023)	66.86	77.38	89.81	85.35	81.01	83.39	74.53	75.47	73.25	78.56
SAMed (Zhang and Liu 2023) + Ours	70.29	78.85	90.19	86.27	82.26	83.86	75.17	76.50	74.12	79.72 + 1.16
SAM-COBOT (Peng et al. 2024a)	68.96	82.30	91.20	86.31	78.44	83.59	71.41	73.88	72.47	78.73
SAM-COBOT (Peng et al. 2024a) + Ours	69.41	82.87	92.09	86.57	79.54	84.07	71.94	74.50	72.93	79.32 + 0.59
SAM-Adapter (Chen et al. 2023a)	65.76	74.24	80.48	83.59	72.83	70.77	67.79	71.20	69.39	72.89
SAM-Adapter (Chen et al. 2023a) + Ours	67.28	75.85	82.11	83.91	73.10	71.69	68.05	72.44	69.80	73.80 + 0.91
SAM-PARSER (Peng et al. 2024b)	63.14	67.24	55.82	71.62	57.66	62.69	53.30	60.99	56.20	60.96
SAM-PARSER (Peng et al. 2024b) + Ours	68.84	70.63	70.29	72.88	62.42	67.87	55.85	61.40	58.31	65.39 + 4.43
DoRA (Liu et al. 2024)	67.33	77.95	91.24	85.68	81.53	83.83	75.05	75.80	73.71	79.12
DoRA (Liu et al. 2024) + Ours	69.16	80.19	91.63	85.71	82.65	84.30	75.54	75.94	74.17	79.92 + 0.80

Table 3: Performance improvement for fine-tuned SAM on downstream segmentation tasks. All results are based on the ViT-Base version of SAM. “Decoder”: fine-tuning only decoder. “Encoder”: fine-tuning only encoder. “Avg + Δ ”: the average mIoU along with the improvement over the baseline. “-”: without any improvement. It shows that our method can serve as a plug-and-play module that further enhances the performance of fine-tuned models.

and Liu 2023; Peng et al. 2024a). For the search dataset, we randomly select 50 image samples from the training dataset. N in \mathcal{D}_{topN} is set to 10. For the training epoch, we set the total number of epochs to 25. Our training employs the Adam optimizer (Kingma and Ba 2014). The initial learning rate is set to 1.0×10^{-4} , and the weight decay is 5×10^{-5} with one image per mini-batch. All experiments are conducted on a computer equipped with four NVIDIA RTX 4090 GPUs. Please refer to the Appendix for more details.

Main Results

Benchmark Results. Table 2 quantitatively compares various SAM-variants on downstream segmentation tasks across natural, medical, and camouflage image datasets. The results validate the effectiveness of our proposed method in consistently enhancing the segmentation capability of SAM and SAM 2 across diverse datasets, including COCO (Lin et al. 2014), VOC (Everingham et al. 2010), PerSeg (Zhang et al. 2023), ISIC (Codella et al. 2018), BUSI (Al-Dhabyani et al. 2020), KVASIR (Jha et al. 2020), CAMO (Le et al. 2019), COD10K (Fan et al. 2022) and CHAME (Skurowski et al. 2018). Specifically, we observe that our method is effective across different backbones. It enhances the average mIoU of ViT-B, ViT-L, and ViT-H versions of SAM by 8.94, 11.46, and 5.14, respectively. Similarly, it enhances the average mIoU of Hiera-T, Hiera-S, Hiera-B+, and Hiera-L versions of SAM 2 by 8.34, 7.65, 5.32, and 5.76, respectively. Moreover, larger improvements are observed on natural image datasets. For example, on the ISIC16 medical dataset, the ViT-B version of SAM improves from 57.33 to 65.21 (+7.88), while on the VOC natural image dataset, it improves from 38.88 to 56.33 (+17.45). This suggests that our fine-tuning method is more effective on in-domain datasets (e.g., VOC) than out-of-domain datasets (e.g., ISIC16).

Enhancing fine-tuned Models. Table 3 compares the performance of various fine-tuning strategies for SAM, including Decoder-only Tuning, Encoder-only Tuning, MedSAM (Ma et al. 2024), SAMed (Zhang and Liu 2023), SAM-COBOT (Peng et al. 2024a), SAM-Adapter (Chen et al. 2023a), SAM-PARSER (Peng et al. 2024b), and DoRA (Liu et al. 2024), across nine datasets spanning three computer vision domains. In Table 3, we present both the original mIoU and the mIoU after applying our method to these fine-tuned models. The results demonstrate that our method effectively enhances these fine-tuned models, leading to consistent performance improvements across various tasks. For example, our method boosts SAMed, SAM-Adapter, SAM-PARSER and DoRA by 1.16, 0.91, 4.43 and 0.80, respectively, as well as Decoder-tuning and Encoder-tuning by 1.01 and 0.51. These findings suggest that feature redundancy persists in the models even after fine-tuning. By effectively identifying and removing this redundancy, our approach enhances the models’ ability to better adapt to downstream tasks.

Qualitative Results. Figure 3 presents segmentation examples that demonstrate the effectiveness of our method compared to leading fine-tuning techniques, including DoRA (Liu et al. 2024), and SAM-Adapter (Houlsby et al. 2019), across natural, medical and camouflage scenarios. In Figure 3, our method is compatible with existing fine-tuning across various scenarios, achieving a more precise segmentation. For instance, details such as the cat’s ear, the polyp’s contour, and the snake’s body align more closely with the ground truths.

Ablation Studies

Varying of Number of Replacement Pairs. To examine the relationship between the number of replacement pairs and performance, we evaluate different replacement pair

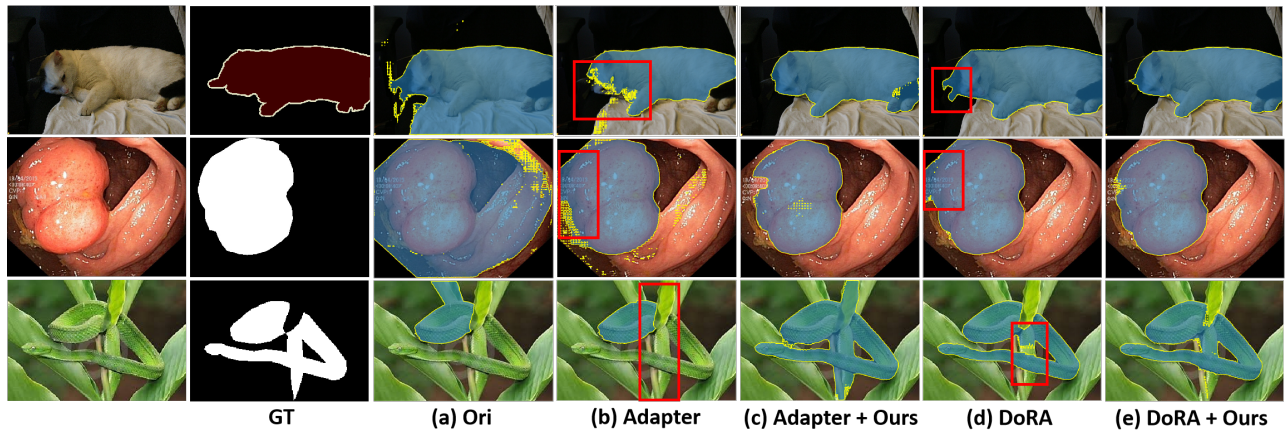


Figure 3: Qualitative comparison of various fine-tuning methods for SAM across natural, medical, and camouflage scenarios. Columns from left to right show the original input image (Input), the ground-truth segmentation mask (GT), the segmentation results from the original SAM (Base), and the results before and after applying our method to various fine-tuned methods (i.e., Adapter, DoRA). The results demonstrate that our method effectively enhances the performance of already fine-tuned models, producing more refined predictions that are closer to the ground truths, as highlighted in Red boxes. Refer to the Appendix for more visualizations.

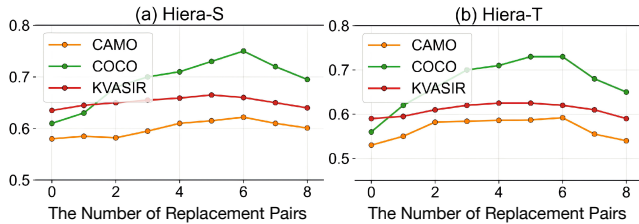


Figure 4: Performance Comparison of varying number of replacement pairs. This suggests that combining different replacement pairs can further improve segmentation performance.

combinations across various datasets (e.g., CAMO, COCO, KVASIR) using SAM’s Hiera-S and Hiera-T backbones, as shown in Figure 4. The results indicate that increasing the number of replacement pairs leads to improved performance. For example, on the COCO dataset (the green line), performance peaks when using six replacement pairs. This suggests that combining different replacement pairs effectively reduces redundancy in SAM.

Computational Overhead. Table 4 presents the computational overhead comparison of different fine-tuning methods. The results demonstrate that our method significantly reduces GPU memory usage compared to other fine-tuning approaches. While some PEFT methods (i.e., SAMed and SAM-PARSER) update only a small subset of parameters, they still require retaining the original computational graph for backpropagation, leading to high GPU memory consumption. In contrast, our method is entirely parameter-free, relying only on model inference, which minimizes memory usage and eliminates the need for additional parameters for adaptation.

Features of Optimal Replacement Pairs. To explore which

Method	GPU (GB)	Params.(K)
Encoder-only	34.6	89670
Decoder-only	13.7	4057
MedSAM (Ma et al. 2024)	34.7	93735
SAM-Adapter (Chen et al. 2023a)	28.5	3550
SAMed (Zhang and Liu 2023)	28.9	147
SAM-COBOT (Peng et al. 2024a)	29.1	148
SAM-PARSER (Peng et al. 2024b)	15.9	0.5
Ours	11.1	0

Table 4: Computational overhead for different fine-tuning strategies. Experiments are performed on the ViT-base version of SAM with input images at a resolution of 1024x1024 and a batch size of 4. It shows that our method requires less GPU usage and is entirely parameter-free compared to other fine-tuning.

feature types are redundant or effective, we compare channels in optimal replacement pairs on natural, medical, and camouflage scenarios. As shown in Figure 5, the features of effective channels have more discernible structures, well-defined edges, and textures, which are likely beneficial for segmentation tasks. In contrast, the features of redundant channels are generally more blurred and contain less structural information. They appear noisy and lack the clarity for accurate segmentation. Moreover, some certain channels exhibit consistent effectiveness or redundancy across various datasets. For example, Channel 19 shows consistent effectiveness, while Channels 20, 98, 162, and 226 show consistent redundancy across different scenarios. It indicates that some channels capture features with cross-domain generalizability.

Extension to Different Vision Tasks. To validate the effectiveness of our method beyond segmentation, we evaluate its performance on additional vision tasks, including image classification and depth estimation. Table 5 presents the

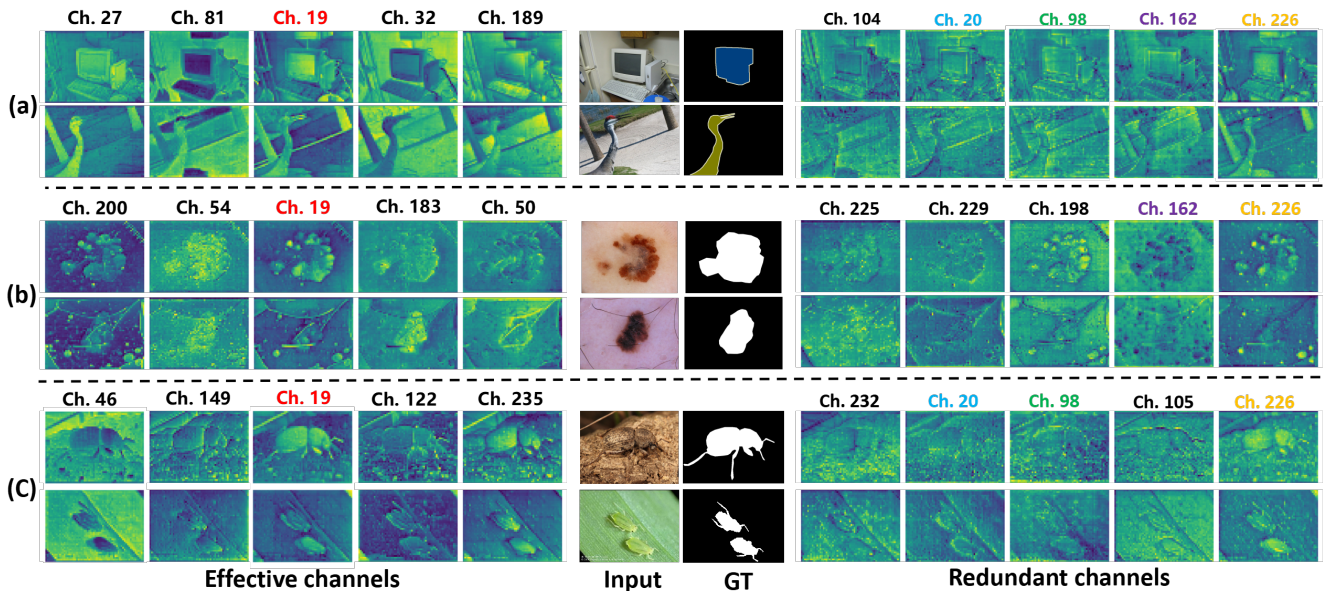


Figure 5: Comparison of effective and redundant channels in optimal replacement pairs for downstream tasks. “Effective channels” are used to replace the “Redundant channels” to improve task-specific performance. The colors in the feature map from green to yellow represent the response intensity from weak to strong. (a)-(c) showcase natural, medical and camouflage scenarios, respectively, showing effective channel features have more discernible structures, edges and textures compared to redundant channel features.

Model	Backbone	NYUv2			CIFAR
		MSE ↓	AbsRel ↓	δ_1 ↑	Acc. ↑
Dinov2	ViT-S	0.225	0.126	0.893	80.41
	ViT-S*	0.209	0.112	0.907	80.81
	ViT-B	0.210	0.110	0.900	88.08
	ViT-B*	0.193	0.095	0.916	88.49

Table 5: Effectiveness evaluation of Depth Estimation and Image Classification. This demonstrates the generalizability of our fine-tuning approach across depth estimation and classification tasks.

depth estimation results on the NYUv2 dataset (Silberman et al. 2012) and the image classification results on the CIFAR dataset (Krizhevsky, Hinton et al. 2009). Our method improves the accuracy of the ViT-Small and ViT-Base versions of DINOv2 from 80.41 to 80.81 and from 88.08 to 88.49, respectively. Similarly, it enhances the MSE, AbsRel, and δ_1 metrics from 0.210, 0.110, and 0.900 to 0.193, 0.095, and 0.916, respectively. These results demonstrate the effectiveness of our approach across different vision tasks (See Appendix for the visualization).

Discussion

Our method achieves performance gains without updating any parameters by exploiting a key observation: not all features in vision foundation models are equally useful for downstream tasks. Large VFMs like SAM are trained on massive, diverse training datasets and are thus equipped with rich but overcomplete representations. While this diversity

supports generalization, it also means many channels are not tailored for specific downstream tasks. Therefore, some channels may carry task-irrelevant or even distracting information—what we define as redundant. *From an intuitive perspective*, we use the replacing strategy to reconfigure the existing representation space. By replacing these redundant channels with effective ones already existing in the model, we enhance task-specific feature representation. This can be analogized to team optimization: if some team members are idle or misaligned with the current project goals, reallocating active contributors to fill their roles enhances team efficiency—without hiring anyone new.

Conclusion

In this paper, we propose a parameter-free fine-tuning method for adapting vision foundation models (VFMs) to downstream tasks. Unlike previous fine-tuning methods that update parameters, our approach introduces a new perspective by eliminating feature redundancies within VFMs. Experiments on both in-domain and out-of-domain datasets demonstrate that our method is simple yet effective, and it is also compatible with existing fine-tuning strategies to achieve additional performance gains. Since our method operates solely at the inference stage, it maintains a lower computational overhead. Future work could explore accelerating the search for optimal replacement pairs using traditional optimization techniques, such as particle swarm optimization. Additionally, given its strong generalization across multiple backbones and downstream tasks, we encourage researchers to apply our method to a broader range of models and vision tasks.

References

- Al-Dhabyani, W.; Gomaa, M.; Khaled, H.; and Fahmy, A. 2020. Dataset of breast ultrasound images. *Data in brief*.
- Chen, T.; Zhu, L.; Deng, C.; Cao, R.; Wang, Y.; Zhang, S.; Li, Z.; Sun, L.; Zang, Y.; and Mao, P. 2023a. Sam-adapter: Adapting segment anything in underperformed scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3367–3375.
- Chen, T.; Zhu, L.; Ding, C.; Cao, R.; Zhang, S.; Wang, Y.; Li, Z.; Sun, L.; Mao, P.; and Zang, Y. 2023b. SAM Fails to Segment Anything?—SAM-Adapter: Adapting SAM in Underperformed Scenes: Camouflage, Shadow, and More. *arXiv:2304.09148*.
- Codella, N. C.; Gutman, D.; Celebi, M. E.; Helba, B.; Marchetti, M. A.; Dusza, S. W.; Kallou, A.; Liopyris, K.; Mishra, N.; Kittler, H.; et al. 2018. Skin lesion analysis toward melanoma detection. In *IEEE 15th international symposium on biomedical imaging*.
- Dalvi, F.; Sajjad, H.; Durrani, N.; and Belinkov, Y. 2020. Analyzing redundancy in pretrained transformer models. In *Proceedings of the 2020 conference on empirical methods in natural language processing*.
- Ding, N.; Qin, Y.; Yang, G.; Wei, F.; Yang, Z.; Su, Y.; Hu, S.; Chen, Y.; Chan, C.-M.; Chen, W.; et al. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*.
- Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (VOC) challenge. *International journal of computer vision*.
- Fan, D.-P.; Ji, G.-P.; Cheng, M.-M.; and Shao, L. 2022. Concealed object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Han, L.; Li, Y.; Zhang, H.; Milanfar, P.; Metaxas, D.; and Yang, F. 2023. Svdiff: Compact parameter space for diffusion fine-tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Houlsby, N.; Giurghi, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*.
- Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv:2106.09685*.
- Jha, D.; Smedsrud, P. H.; Riegler, M. A.; Halvorsen, P.; De Lange, T.; Johansen, D.; and Johansen, H. D. 2020. Kvasir-seg: A segmented polyp dataset. In *MultiMedia modeling: 26th international conference*.
- Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv:1412.6980*.
- Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023. Segment anything. *arXiv:2304.02643*.
- Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.
- Le, T.-N.; Nguyen, T. V.; Nie, Z.; Tran, M.-T.; and Sugimoto, A. 2019. Anabranh Network for Camouflaged Object Segmentation. *Journal of Computer Vision and Image Understanding*.
- Li, J.; Wen, Y.; and He, L. 2023. Scconv: Spatial and channel reconstruction convolution for feature redundancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Li, Y.; Yu, Y.; Liang, C.; He, P.; Karampatziakis, N.; Chen, W.; and Zhao, T. 2023. Loftq: Lora-fine-tuning-aware quantization for large language models. *arXiv:2310.08659*.
- Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*.
- Liu, S.-Y.; Wang, C.-Y.; Yin, H.; Molchanov, P.; Wang, Y.-C. F.; Cheng, K.-T.; and Chen, M.-H. 2024. Dora: Weight-decomposed low-rank adaptation. In *Forty-first International Conference on Machine Learning*.
- Ma, J.; He, Y.; Li, F.; Han, L.; You, C.; and Wang, B. 2024. Segment anything in medical images. *Nature Communications*.
- Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F.; El-Nouby, A.; et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv:2304.07193*.
- Peng, Z.; Xu, Z.; Zeng, Z.; Xie, L.; Tian, Q.; and Shen, W. 2024a. Parameter efficient fine-tuning via cross block orchestration for segment anything model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Peng, Z.; Xu, Z.; Zeng, Z.; Yang, X.; and Shen, W. 2024b. Sam-parser: Fine-tuning sam efficiently by parameter space reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- 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*.
- Ravi, N.; Gabeur, V.; Hu, Y.-T.; Hu, R.; Ryali, C.; Ma, T.; Khedr, H.; Rädle, R.; Rolland, C.; Gustafson, L.; et al. 2024. Sam 2: Segment anything in images and videos. *arXiv:2408.00714*.
- Ren, T.; Liu, S.; Zeng, A.; Lin, J.; Li, K.; Cao, H.; Chen, J.; Huang, X.; Chen, Y.; Yan, F.; et al. 2024. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv:2401.14159*.
- Silberman, N.; Hoiem, D.; Kohli, P.; and Fergus, R. 2012. Indoor segmentation and support inference from rgb-d images. In *12th European Conference on Computer Vision*.
- Skurowski, P.; Abdulameer, H.; Błaszczak, J.; Depta, T.; Kornacki, A.; and Koziel, P. 2018. Animal camouflage analysis: Chameleon database. *Unpublished manuscript*.

Tian, C.; Shi, Z.; Guo, Z.; Li, L.; and Xu, C.-Z. 2024. Hydralora: An asymmetric lora architecture for efficient fine-tuning. *Advances in Neural Information Processing Systems*.

Tianxiang Hao; Hui Chen; Yuchen Guo; and Guiguang Ding. 2023. Consolidator: Mergeable Adapter with Grouped Connections for Visual Adaptation. In *The Eleventh International Conference on Learning Representations*.

Wang, Y.; Huang, W.; Sun, F.; Xu, T.; Rong, Y.; and Huang, J. 2020. Deep multimodal fusion by channel exchanging. *Advances in neural information processing systems*.

Xiong, Y.; Chen, H.; Hao, T.; Lin, Z.; Han, J.; Zhang, Y.; Wang, G.; Bao, Y.; and Ding, G. 2024. Pyra: Parallel yielding re-activation for training-inference efficient task adaptation. In *European Conference on Computer Vision*.

Yang, J.; Gao, M.; Li, Z.; Gao, S.; Wang, F.; and Zheng, F. 2023. Track anything: Segment anything meets videos. *arXiv:2304.11968*.

Yu, T.; Feng, R.; Feng, R.; Liu, J.; Jin, X.; Zeng, W.; and Chen, Z. 2023. Inpaint anything: Segment anything meets image inpainting. *arXiv:2304.06790*.

Zhang, K.; and Liu, D. 2023. Customized segment anything model for medical image segmentation. *arXiv:2304.13785*.

Zhang, R.; Jiang, Z.; Guo, Z.; Yan, S.; Pan, J.; Ma, X.; Dong, H.; Gao, P.; and Li, H. 2023. Personalize segment anything model with one shot. *arXiv:2305.03048*.

Zhang, X.; Zhou, X.; Lin, M.; and Sun, J. 2018. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.