SEARCH2ADAPT: TRAINING-FREE ADAPTER ARCHITECTURE SEARCH FOR SEGMENTATION FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent foundation models show remarkable success across general-domain vision tasks but often struggle in specialized domains with distinct visual characteristics. Existing adaptation methods typically rely on fixed adapter architectures that cannot account for domain-specific characteristics. We propose Search2Adapt, a framework that automatically discovers effective adapter architectures for segmentation foundation models through training-free search. Search2Adapt parameterizes each adapter by its operation type, configuration, and layer-wise insertion, and perturbs these variables to construct candidate adapters. Each candidate is scored by a zero-shot proxy based on gradient statistics with a sparsity penalty on layer-wise insertion, and the perturbation variables are iteratively updated toward high-scoring candidates through score-weighted aggregation. This enables domain-specific adapter search without training individual candidates. To evaluate Search2Adapt across specialized segmentation tasks, we conduct extensive experiments on camouflage, polyp, and nuclei segmentation datasets. Search2Adapt achieves performance competitive with or superior to fixed adapter designs across diverse domains, while sensitivity analysis confirms that adapter configuration substantially affects segmentation performance. These results demonstrate effective domain-specific adapter architecture discovery for segmentation foundation models.

1 INTRODUCTION

With the advancement of computational power, foundation models trained on large collections of general-domain datasets have been introduced. These models are typically trained on large collections of generalized datasets and demonstrate strong performance across diverse domains. However, foundation models often under-perform in specialized domains that the models have rarely seen during the training phase. To address this limitation, parameter-efficient fine-tuning (PEFT) has been proposed as an effective strategy for adapting foundation models to downstream tasks. Representative PEFT approaches include lora (Hu et al., 2022), vpt (Jia et al., 2022), and Adapter (Houlsby et al., 2019), which introduce lightweight learnable modules to enable efficient domain adaptation while largely preserving pretrained parameters. While effective, these methods typically rely on fixed architectural designs applied uniformly across diverse domains, limiting their ability to tailor architectural configurations to domain-specific characteristics.

For the segmentation task, the segment anything model (SAM) (Kirillov et al., 2023) is popular for being trained on massive scene image data and is able to perform zero-shot segmentation tasks on natural scene images. However, SAM still struggles with limitations in specialized domains, and several fine-tuning methods have been introduced. Among these, SAM-Adapter (Chen et al., 2023) and its subsequent variants, such as SAM2-Adapter (Chen et al., 2024b), have emerged as representative approaches that insert lightweight adapter modules into the image encoder for efficient domain adaptation. In parallel, several studies explore alternative adaptation mechanisms, including parameter space reconstruction (Peng et al., 2024) and domain-specific adapter designs (Wu et al., 2025; Yu et al., 2024). Since segmentation requires dense pixel-level prediction, designing appropriate adapter architectures to capture task-specific image features is particularly important for adapting SAM to specialized domains.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Several studies investigate PEFT adaptation from a model-specific perspective motivating the search for domain-specific adaptation architectures. In this context, neural architecture search (NAS) has been adopted as a promising approach for automatically identifying suitable architectural configurations for PEFT across different datasets and downstream tasks. Recent methods, such as NOAH (Zhang et al., 2024) and AutoPEFT (Zhou et al., 2024), show that exploring diverse architectural configurations can significantly enhance the adaptability and performance of foundation models by tailoring PEFT modules to individual domains. More recently, NAS-Adapter (Chen et al., 2026) introduces adapter architecture search for foundation models, demonstrating it applicability to medical imaging tasks. However, these approaches still rely on conventional search strategies, such as evolutionary algorithms or one-shot search paradigms, which require substantial computational cost during the search phase. Moreover, such search strategies are primarily designed to explore a broad range of candidate architectures, making it challenging to efficiently capture subtle variations in adapter configurations. Since small differences in architectural configurations can affect downstream prediction performance, a search strategy tailored to fine-grained adapter configurations is crucial for effective adapter architecture search.

More critically, recent NAS-based approaches for identifying PEFT adaptation methods suffer from several inherent limitations. First, minor architectural modifications can lead to substantial variations in candidate performance. Such local sensitivity can result in non-smooth search trajectories and unstable convergence, hindering reliable exploration of the search space. Second, many existing methods rely on training-based evaluation strategies, where the performance of candidate architectures is estimated through repeated optimization during the search phase. Such evaluations introduces substantial computational overhead for assessing candidate architectures, thereby increasing the overall search time. Finally, existing approaches provide limited consideration of adapter-specific search spaces and the hierarchical dependencies among architectural configurations. This can make fine-grained exploration of adapter architectures inefficient. Collectively, these challenges underscore the need for an efficient and domain-aware search framework that can identify appropriate adapter architectures without incurring prohibitive search costs.

To address these limitations, we introduce Search2Adapt, a principled framework for training-free adapter architecture search for foundation models. Search2Adapt employs a perturbation-based search strategy that enables smooth and controlled architectural modifications, promoting stable exploration of adapter configurations. The proposed method defines an adapter-specific search space encompassing diverse lightweight residual adaptation functions and hierarchically organizes their architectural configurations. Unlike conventional methods that rely on expensive training-based evaluation, Search2Adapt employs zero-cost proxies to efficiently estimate candidate performance. This training-free evaluation substantially reduces architecture evaluation overhead and search time. Although instantiated with SAM-Adapter, the proposed framework is not restricted to a specific foundation model or proxy metric, and its applicability is further validated across different combinations of foundation models and zero-cost proxies. Extensive experiments on camouflaged object detection and medical image segmentation datasets demonstrate that Search2Adapt-discovered architectures achieve competitive segmentation performance across diverse challenging domains. The code implementation is available for reproducibility. [1]

2 PRELIMINARIES

2.1 ADAPTER APPROACHES FOR SAM

The Segment Anything Model (SAM) (Kirillov et al., 2023) is trained on SA-1B dataset, which contains 1 billion masks and 11 million images collected from datasets such as COCO (Lin et al., 2014), LVIS (Gupta et al., 2019), ADE20K (Zhou et al., 2019), and Open Images V4 (Kuznetsova et al., 2020). Owing to its large-scale pre-training on general-domain images, SAM demonstrates strong segmentation capabilities across diverse natural image domains. However, its performance often degrades in specialized domains such as shadow detection, camouflaged object detection, and medical image segmentation, where substantial distribution gaps from general-domain representations exist.

[1] https://anonymous.4open.science/r/Search2Adapt/

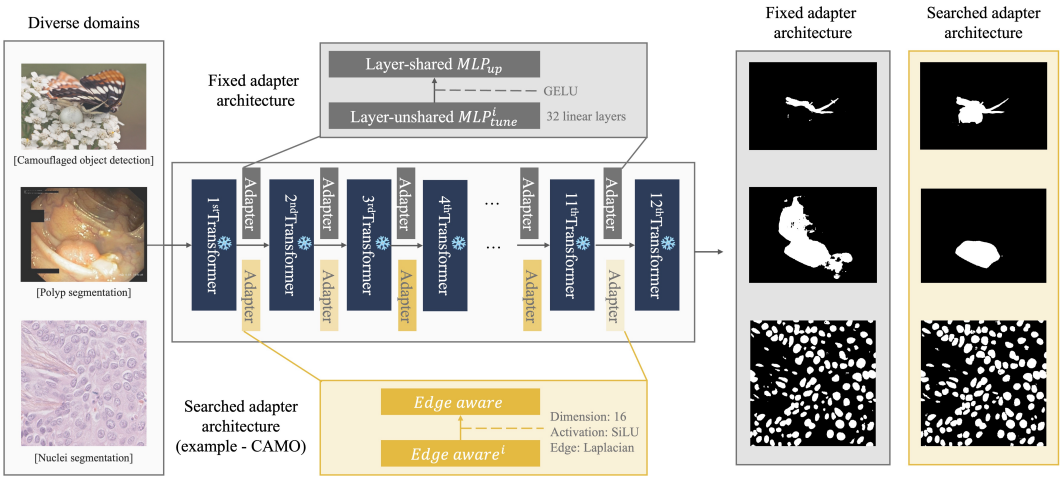

Figure 1: **Limitations of fixed adapter architectures and motivation for domain-specific adapter search.** Conventional methods, represented by SAM-Adapter, employ a fixed adapter architecture at every layer (gray) across diverse target domains, limiting their ability to accommodate domain-specific characteristics. In contrast, architecture search identifies domain-specific adapter structures, configurations, and layer-wise insertion patterns (yellow; example searched for CAMO), yielding more accurate masks in these examples (right; each domain uses its own searched adapter). Snowflakes denote frozon encoder layers.

To address these limitations, SAM-Adapter (Chen et al., 2023) introduces a lightweight adapter-based fine-tuning strategy for efficient domain adaptation. As illustrated in Fig. 1, the adapter architecture consists of two MLP components, a layer-shared MLP_{up} and a layer-specific MLP^i_{tune}, where i denotes the corresponding transformer layer. The MLP^i_{tune} component performs layer-specific feature adaptation with a GELU activation, while MLP_{up} is shared across layers to transform the adapted features. The adapter modules are incorporated into the transformer layers of the SAM image encoder. For example, when ViT-B is adopted as the image encoder, adapter modules are applied across its 12 transformer layers. This design provides a lightweight adaptation mechanism while maintaining a fixed adapter structure across different target domains. Building upon this adaptation paradigm, recent studies have further explored SAM-based adapter variants, including SAM2-Adapter (Chen et al., 2024b) and other lightweight adaptation frameworks, to improve specialization across diverse segmentation domains (Wu et al., 2025; Peng et al., 2024; Xiao et al., 2024; Chen et al., 2024a). These approaches demonstrate that the architectural design of adapter modules plays a critical role in adapting foundation models to domain-specific segmentation tasks. However, despite their simplicity and efficiency, fixed adapter architectures may be insufficient to accommodate diverse domain-specific characteristics, motivating the need for more flexible architectural adaptation.

2.2 Neural Architecture Search

Neural Architecture Search (NAS) aims to automatically discover optimal architecture for each dataset and downstream task. The search space defines the architectural components to be explored, while an appropriate search algorithm identifies high-performing architectures within the search space. Conventional NAS methods typically rely on reinforcement learning or evolutionary algorithms, but they often require high computational cost and long search time. To address this limitation, one-shot NAS has been proposed, which constructs a supernet and employs weight sharing among candidate architectures during the search process. Representative methods such as DARTS (Liu et al., 2018) and ENAS (Pham et al., 2018) reduce search costs by sharing network weights across candidate operations or architectures. However, imbalanced candidate training and discrepancies between shared-weight estimates and standalone performance can limit reliable architecture ranking. Recent studies have therefore focused on zero-shot NAS, which aims to estimate the potential performance of candidate architectures without training by using zero-cost proxies de-

rived from their intrinsic properties (Mellor et al., 2021; Li et al., 2023; Lin et al., 2021; Lee & Ham, 2024). This enables efficient architecture evaluation and ranking during the search phase without relying on training-based performance estimation.

2.3 Zero-cost Scoring Proxies

NASWOT (Mellor et al., 2021) introduces the concept of NAS without training. The core assumption is that the performance potential of a network can estimated before training using randomly initialized weights. Specifically, NASWOT constructs a kernel matrix K_H based on the Hamming distance d_H between the binary activation patterns of N input samples, where N_A represents the total number of activations in the network as shown in Eq. 2.

$$\mathbf{K}_H = \begin{pmatrix} N_A - d_H(\mathbf{c}_1, \mathbf{c}_1) & \cdots & N_A - d_H(\mathbf{c}_1, \mathbf{c}_N) \\ \vdots & \ddots & \vdots \\ N_A - d_H(\mathbf{c}_N, \mathbf{c}_1) & \cdots & N_A - d_H(\mathbf{c}_N, \mathbf{c}_N) \end{pmatrix} \tag{1}$$

$$s = \log |\mathbf{K}_H|. \tag{2}$$

ZiCo (Li et al., 2023) demonstrates that the mean and variance of gradients across multiple samples reflect the trainability of a network and introduces the ZiCo score to estimate the performance potential of candidate architectures. By exhibiting a strong correlation with final performance, the ZiCo score serves as a zero-cost indicator for efficiently estimating and ranking the performance of candidate architectures without training. Given a network with D layers and randomly initialized parameters θ, ZiCo computes the ratio between the mean and standard deviation of gradient magnitudes across input samples using a loss function L, as shown in Eq. 3.

$$\text{ZiCo} = \sum_{i=1}^{D} \log \left(\sum_{w \in \theta_i} \frac{\mathbb{E}_n \left[|\nabla_w L(X_n, y_n; \theta)| \right]}{\sqrt{\text{Var}_n \left(|\nabla_w L(X_n, y_n; \theta)| \right)}} \right), \tag{3}$$

where D denotes the number of network layers, θ_i represents the parameter of the i-th layers, and w denotes an individual parameter in θ_i. X_n and y_n represent the n-th input sample and its corresponding target, respectively. The expectation $\mathbb{E}_n[\cdot]$ and variance $Var_n(\cdot)$ are computed over the input samples.

In this work, we adopt ZiCo score due to its effectiveness and strong correlation with the model performance. Compared to NASWOT, which focuses on the expressivity. ZiCo targets the trainability of architectures by capturing statistical characteristics of gradient flow, which is particularly important for SAM adaptation. While conventional ZiCo employs an evolutionary algorithm for architecture search, we propose a perturbation-based exploration method combined with ZiCo scoring to efficiently identify optimal adapter architectures. Further details are provided in Sec.3.2.

3 Adapter Architecture Search

3.1 Hierarchical Adapter Architecture Search Space

Existing SAM-Adapter architectures typically employ a fixed lightweight MLP design across various domains. However, specialized segmentation tasks often exhibit diverse representational characteristics, making a single fixed adapter design suboptimal. To address this limitation, we construct an operation-conditioned search space composed of diverse lightweight residual adapter primitives designed to capture complementary feature characteristics. Each adapter primitive is associated with its own valid configuration space, including bottleneck dimensions, activation functions, gating functions, and operation-specific hyperparameters, enabling flexible yet parameter-efficient adaptation.

To parameterize the search space, we define an operation-level perturbation parameter α_{op} for adapter primitive selection, while the corresponding operation-specific configurations are represented by α_{config}^{op}. Unlike a globally shared configuration space, only the valid configuration parameters associated with each selected operation are activated during the search process, enabling

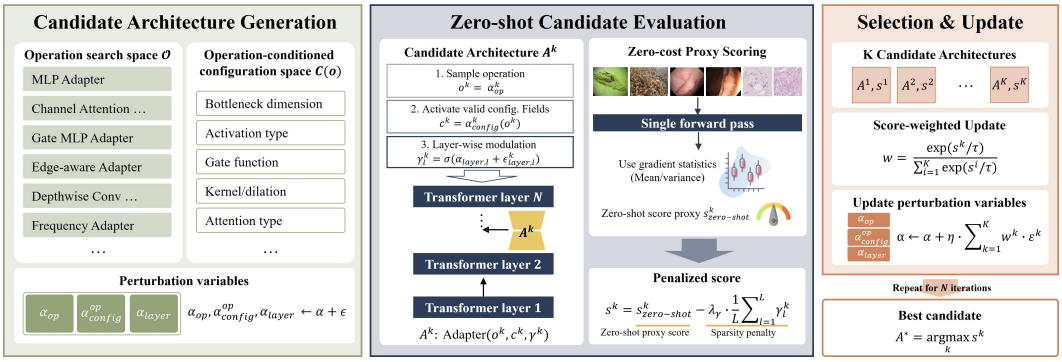

Figure 2: **Overview of the proposed Search2Adapt framework.** Search2Adapt hierarchically explores adapter operations, operation-specific configurations, and layer-wise insertion patterns through perturbation-based candidate generation (left). Each candidate is efficiently evaluated using the zero-shot proxy without parameter updates, and its score is regularized by a layer-wise sparsity penalty to encourage lightweight adaptation (middle). The resulting penalized scores guide score-weighted perturbation aggregation and iterative updates of the search variables (right). The highest-scoring architecture is finally selected and trained for the downstream task.

efficient exploration of diverse lightweight adapter structures while avoiding incompatible architectural configurations.

In addition, we introduce a layer-wise insertion parameter α_{layer} to control the contribution of adapter insertion across transformer layers. Unlike conventional SAM-Adapter frameworks that employ fixed insertion strategies, the proposed formulation jointly explores adapter operations, operation-specific configurations, and layer-wise insertion patterns to achieve lightweight yet effective domain adaptation. By adaptively controlling adapter insertion at each transformer layer, the search can identify effective insertion patterns while avoiding unnecessary adapter components. The perturbation parameter α_{op}, α^{op}_{config}, and α_{layer} are jointly optimized through the hierarchical perturbation strategy described in Sec. 3.2, enabling efficient exploration of the operation-conditioned search space without exhaustive combinatorial search.

3.2 Perturbation-based Search Algorithm

While the original ZiCo framework employs evolutionary search to explore candidate architectures, we adopt a hierarchical perturbation-based strategy tailored to adapter architecture search. Given an operation search space $\mathcal{O}$ and its operation-conditioned configuration space $\mathcal{C}(o)$, we parameterize adapter operations, operation-specific configurations, and layer-wise insertion patterns using $\alpha_{op}, \alpha^{op}_{config}, \alpha_{layer}$, respectively. These parameters hierarchically represent the candidate architecture while maintaining the parameter efficiency of SAM-Adapter. Over N search iterations, the parameters are progressively updated based on the performance of perturbed candidate architectures.

At each iteration, Gaussian perturbations $\epsilon^k_{op}, \epsilon^k_{config}, \epsilon^k_{layer}$ are independently sampled to generate K candidates. For each candidate k, the perturbed operation parameter selects an adapter primitive o^k, activating only its valid configuration space $\mathcal{C}(o^k)$, from which the configuration c^k is decoded. The layer-wise parameter is transformed into a continuous modulation coefficient as $\gamma^k_l = \mathrm{sigmoid}(\alpha_{layer,l} + \epsilon^k_{layer,l})$, controlling the residual insertion strength at the l-th transformer layer. The resulting candidate $\mathcal{A}^k = (o^k, c^k, \gamma^k)$ is evaluated by the zero-cost objective $\mathcal{S}(\mathcal{A}^k)$, and the search parameters are updated using the aggregated candidate scores. This process enables efficient exploration of operation-specific configurations while avoiding interference between incompatible adapter configurations.

The sampled adapter structure is shared across transformer layers, with the contribution controlled by the layer-wise modulation coefficients. Each candidate architecture is then evaluated using the ZiCo score, computed from gradient statistics of the inserted adapter modules through forward and

Algorithm 1 Hierarchical Perturbation Search for Adapter Architecture

Require: Operation search space $\mathcal{O}$ and operation-conditioned configuration space $\mathcal{C}(o)_{o\in\mathcal{O}}$, Iteration N, perturbation samples K
Ensure: Best adapter architecture $\mathcal{A}^*$

```
Initialize perturbation variables α_op, α_config, and α_layer
s* ← −∞, A* ← ∅
for t = 1 to N do
    for k = 1 to K do
        Sample perturbations ε^k_op, ε^k_config, ε^k_layer
        Select operation o^k from α_op + ε^k_op
        Activate valid configurations C(o^k)
        Decode configuration c^k from α_config^op + ε^k_config
        Compute layer modulation γ^k from α_layer + ε^k_layer
        Construct candidate architecture A^k = (o^k, c^k, γ^k)
        Evaluate s^k ← (A^k)
        if s^k > s* then
            A* ← A^k
            s* ← s^k
        end if
    end for
    Compute normalized weights {w^k}_{k=1}^K using Eq. 5
    Update α_mathrmop, α_config, α_layer using score-weighted perturbation aggregation
end for
return A*
```

backward propagation without parameter updates. The candidate score is defined as

$$\mathcal{S}(\mathcal{A}^k) = ZiCo(\mathcal{A}^k) - \lambda_\gamma \frac{1}{L}\sum_{l=1}^{L}\gamma_l^k, \tag{4}$$

where λ_γ controls the sparsity regularization for layer-wise adapter insertion. This penalized score is used for perturbation aggregation, enabling efficient candidate evaluation without additional training during the search process.

At each iteration, perturbations yielding higher penalized architecture scores contribute more strongly to score-weighted aggregation for updating the search parameters. Specifically, the normalized weight for the k-th candidate is computed as

$$w^k = \frac{\exp(\mathcal{S}(\mathcal{A}^k)/\tau)}{\sum_{j=1}^{K}\exp(\mathcal{S}(\mathcal{A}^j)/\tau)}, \tag{5}$$

where τ denotes the temperature controlling the concentration of the score-based weights. The search parameters are then updated by aggregating the sampled perturbations according to their normalized weights:

$$\alpha_x \leftarrow \alpha_x + \eta_x \sum_{k=1}^{K} w^k \epsilon_s^k. \qquad x \in \{op, config, layer\}, \tag{6}$$

where η_x denotes the update coefficients for each search parameter. Through this hierarchical perturbation-aggregation strategy, the proposed framework efficiently explores local architectural variations while progressively updating the search variables. The exploration-exploitation behavior is controlled by the perturbation scale σ_ϵ, the number of perturbation samples K, and the score-weighted update dynamics. The effects of these parameter settings are further analyzed in App A.3.2. Additional analysis of the search dynamics is provided in App. A.2 to examine the stability of the proposed perturbation-based search.

4 EXPERIMENTAL RESULTS

4.1 EXPERIMENTAL SETUP

4.1.1 MODEL SETUP

We conduct experiments using SAM (Kirillov et al., 2023) with the ViT-B image encoder (Dosovitskiy, 2020) pre-trained on SA-1B (Kirillov et al., 2023), and additionally evaluate SAM2 (Ravi et al., 2025) and MedSAM (Ma et al., 2024), with MedSAM used for medical image segmentation. The remaining settings follow SAM-Adapter (Chen et al., 2023). We use AdamW with an initial learning rate of 2×10^{-4} and cosine decay, incorporating high-frequency components F_{hfc} and patch embedding features F_{pe} following SAM-Adapter. All experiments are conducted on two NVIDIA GeForce RTX 4090 GPUs with 16GB of memory each.

4.1.2 DATASETS

We evaluate the proposed method on camouflaged object detection and medical image segmentation across six datasets. For camouflaged object detection, we use CAMO (Le et al., 2019) and COD10K (Fan et al., 2021), which contain 1,250 and 10,000 images, respectively. Following the evaluation protocol of Liu et al. (2023), we adopt S-measure (S_m) and mean E-measure (E_ϕ) as evaluation metrics. For medical image segmentation, Kvasir-SEG (Jha et al., 2019) and PolypGen (Ali et al., 2023) contain 1,000 and 1,537 images, while MoNuSeg (Kumar et al., 2019) and TNBC (Naylor et al., 2018) contain 30 and 11 histopathology images, respectively. Medical segmentation performance is evaluated using mean Dice and mIoU. Models are trained for 20 epochs on camouflaged object detection and 120 epochs on medical image segmentation.

4.1.3 HYPER-PARAMETER SETTINGS

The search uses $K = 5$ perturbation samples for $N = 20$ iterations. The perturbation scales $(\sigma_{op}, \sigma_{config}, \sigma_{layer})$ and update coefficients $(\eta_{op}, \eta_{config}, \eta_{layer})$ are set to $(0.1, 0.1, 0.5)$ and $(1.0, 1.0, 2.0)$, respectively. Larger values are assigned to the layer-wise parameters to encourage broader exploration of continuous insertion patterns. We set $\lambda_\gamma = 0.01$ and retain continuous layer-wise modulation coefficients of the highest-scoring architecture without binarization. Results are reported over three independent search runs as mean $\pm$ standard deviation. Additional hyperparameter analyses are provided in App. A.3.2.

4.1.4 COMPARISON MODELS

We compare Search2Adapt with SAM (Kirillov et al., 2023), SAM2 (Ravi et al., 2025), and MedSAM (Ma et al., 2024), as well as SAM-Adapter (Chen et al., 2023), SAM2-Adapter (Chen et al., 2024b), SAM-Parser (Peng et al., 2024), and Trans-SAM (Wu et al., 2025). SAM, SAM2, MedSAM are evaluated using bounding-box prompts derived from ground-truth connected components. Search2Adapt is evaluted with combinations of these backbones and two zero-cost proxies, ZiCo (Li et al., 2023) and NASWOT (Mellor et al., 2021). SAM2 with ZiCo is used for the main comparison in Tab. 1, with alternative combinations reported in Tab. 2.

4.2 RESULTS

The experimental results on camouflaged object detection, polyp segmentation, and nuclei segmentation are shown in Tab. 1. Overall, the results demonstrate the benefit of identifying appropriate adapter architectures for effective adaptation across diverse segmentation domains. Compared with existing adaptation methods that rely on predefined architectural configurations, Search2Adapt automatically discovers dataset-specific adapter architectures without requiring handcrafted architectural design.

For camouflaged object detection, Search2Adapt achieves competitive performance on both CAMO and COD-10K, with an S_m of 0.868 and 0.947, respectively. For polyp segmentation, the searched architectures achieve consistent segmentation performance across Kvasir-SEG and PolypGen despite their distinct visual characteristics. Similarly, for nuclei segmentation, Search2Adapt achieves comparable performance to existing adaptation methods on MoNuSeg and TNBC, demonstrating its

Table 1: **Quantitative comparison of Search2Adapt with existing adaptation methods across diverse segmentation domains.** We evaluate Search2Adapt using different foundation models (SAM, SAM2, MedSAM) and zero-cost proxies (NASWOT and ZiCo). Results are reported as mean ± standard deviation over three independent runs, with the best results highlighted in bold. All prompt-based foundation models are evaluated using the same ground-truth component prompting protocol.

Camouflaged Object Detection				
	CAMO		**COD-10K**	
Method	S_m	E_ϕ	S_m	E_ϕ
SAM(ICCV 2023)	0.819	0.811	0.932	0.425
SAM2(ICLR 2025)	0.892	0.890	0.959	0.453
SAM-Adapter(ICCV 2023)	0.767	0.797	0.892	0.909
SAM2-Adapter(ICLR 2025)	0.879	0.914	0.948	0.964
SAM-Parser(AAAI 2024)	0.799	0.806	0.930	0.937
Trans-SAM(KBS 2025)	0.734	0.823	0.768	0.733
Search2Adapt	0.868 ± 0.002	0.901 ± 0.002	0.947 ± 0.003	0.951 ± 0.001
Polyp Segmentation				
	Kvasir-SEG		**PolypGen**	
Method	Dice	mIoU	Dice	mIoU
SAM(ICCV 2023)	0.908	0.849	0.931	0.878
SAM2(ICLR 2025)	0.949	0.906	0.954	0.914
MedSAM(Nature 2024)	0.951	0.912	0.967	0.938
SAM-Adapter(ICCV 2023)	0.850	0.776	0.662	0.573
SAM2-Adapter(ICLR 2025)	0.914	0.856	0.813	0.754
SAM-Parser(AAAI 2024)	0.846	0.752	0.764	0.657
Trans-SAM(KBS 2025)	0.889	0.805	0.561	0.491
Search2Adapt	0.912 ± 0.002	0.836 ± 0.002	0.809 ± 0.009	0.755 ± 0.009
Nuclei Segmentation				
	MoNuSeg		**TNBC**	
Method	Dice	mIoU	Dice	mIoU
SAM(ICCV 2023)	0.883	0.791	0.917	0.848
SAM2(ICLR 2025)	0.892	0.805	0.924	0.859
MedSAM(Nature 2024)	0.723	0.568	0.820	0.698
SAM-Adapter(ICCV 2023)	0.800	0.667	0.786	0.657
SAM2-Adapter(ICLR 2025)	0.823	0.700	0.822	0.703
SAM-Parser(AAAI 2024)	0.750	0.602	0.635	0.489
Trans-SAM(KBS 2025)	0.652	0.484	0.496	0.347
Search2Adapt	0.823 ± 0.001	0.700 ± 0.002	0.829 ± 0.003	0.713 ± 0.004

applicability across datasets with different morphological characteristics. Qualitative segmentation results for each dataset are presented in Fig. 5.

These results highlight that adapter architecture itself represents an important design dimension for adapting segmentation foundation models. Although the searched architectures do not consistently outperform all foundation model and adaptation baselines, Search2Adapt replaces manual adapter design with an automated, domain-specific architecture discovery process while maintaining effective segmentation performance across diverse tasks. This suggests that adapting a foundation model can benefit not only from optimizing adapter parameters, but also from identifying an appropriate adapter architecture for the target domain. The small standard deviations across independent runs further indicate consistent downstream performance of the searched architectures.

4.3 Configuration Sensitivity under Perturbation-based Exploration

We conduct a fine-grained configuration sensitivity analysis on the TNBC dataset to investigate the importance of operation-specific configurations for domain-specific adaptation. We fix the edge-aware primitive as the base operation and independently vary a single configuration, including the

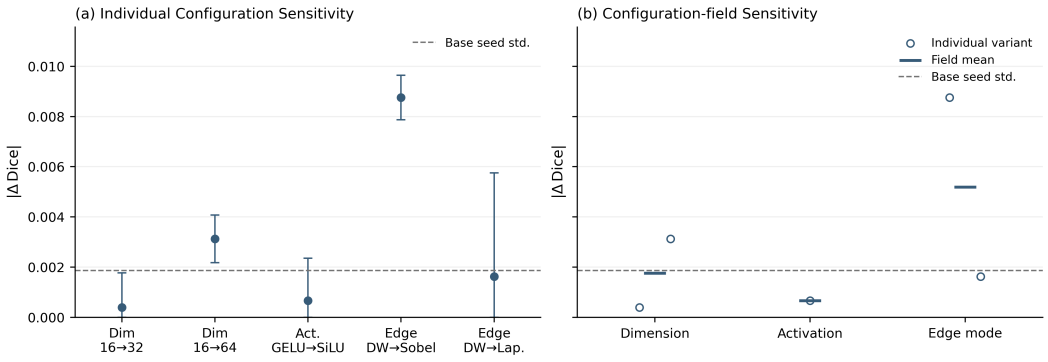

Figure 3: **Fine-grained adapter configuration sensitivity under local architectural perturbations on TNBC.** (a) Absolute Dice variation ($|\Delta\mathrm{Dice}|$) induced by changing a single configuration while keeping the edge-aware adapter operation fixed. The dashed line indicates the standard deviation of the base configuration across three training seeds. (b) Configuration-field sensitivity grouped by bottleneck dimension, activation function, and edge mode. Individual markers denote configuration variants, while horizontal bars indicate the mean sensitivity within each configuration field.

bottleneck dimension, activation function, and edge mode, while keeping the remaining architectural components unchanged. Experiments are conducted over three independent seeds to distinguish configuration-induced variations from training randomness.

As shown in Fig. 3, different architectural dimensions exhibit heterogeneous sensitivity even within the same adapter operation. While changes in the activation function and bottleneck dimension result in relatively minor variations, the edge-mode configuration exhibits a more pronounced effect. The largest variation reaches 0.0088 Dice and 0.0012 mIoU, approximately 4.6 and 5.0 times the standard deviation observed from repeated training, respectively. Although the absolute differences remain moderate, their magnitude relative to training variability indicates that fine-grained configuration choices can introduce systematic performance variations beyond training randomness. These results suggest that selecting an adapter primitive alone may overlook influential operation-specific configurations, motivating their explicit exploration within the proposed perturbation-based search.

5 Conclusion

This work advocates a shift in NAS from discovering architectures from scratch toward refining architectures around strong pre-trained anchors. Building upon this perspective, we propose Search2Adapt, a flexible adapter architecture search framework that enables domain-specific adaptation beyond conventional PEFT methods with fixed designs. Search2Adapt defines a targeted search space encompassing diverse adapter structures and layer-wise insertion patterns, and explores this space through a perturbation-based strategy guided by a zero-shot evaluation metric. By iteratively refining adapter architectures within the local neighborhood of a strong pre-trained baseline, Search2Adapt enables efficient and stable exploration while avoiding the high computational cost of global architecture search. Extensive experiments across diverse imaging domains demonstrate that the adapter architectures discovered by the proposed framework yield robust and competitive segmentation performance. Overall, Search2Adapt provides an effective and scalable solution for adapting foundation models across specialized domains. Although SAM is adopted as a representative segmentation foundation model, the proposed search framework can be extended to other foundation models and parameter-efficient adaptation framework.

AI use statement

In this work, we used generative AI tools to assist with experimental design, code implementation and refinement. We additionally used generative AI tools for language editing to improve the readability and grammatical correctness of the manuscript. All AI-assisted experimental designs and analyses were reviewed by the authors, and AI-assisted code was manually inspected and tested for

correctness before being used in the experiments. All manuscript content was further reviewed and verified by the authors. We take responsibility for the final content of this work, indicating text, claims, code, and artifacts produced with the aid of generative AI.

Ethics statement

This work uses publicly available benchmark datasets and does not involve the collection of new human-subject data. We follow the original licenses and usage protocols of the datasets and foundation models used in our experiments. We are not aware of additional ethical concerns specific to the proposed architecture search framework.

Reproducibility statement

To ensure reproducibility, we provide detailed descriptions of the adapter search space and hierarchical perturbation-based search strategy in Sec. 3, together with pseudocode for the complete search procedure. The experimental settings, including datasets, training configurations, search hyperparameters, and evaluation metrics, are described in Sec. 4.1. Additional ablation studies on the search hyperparameters are provided in App. A.3.2. All Search2Adapt experiments are conducted with three independent random seeds, and the implementation is publicly available through the repository[2].

References

Sharib Ali, Debesh Jha, Noha Ghatwary, Stefano Realdon, Renato Cannizzaro, Osama E Salem, Dominique Lamarque, Christian Daul, Michael A Riegler, Kim V Anonsen, et al. A multi-centre polyp detection and segmentation dataset for generalisability assessment. *Scientific Data*, 10(1): 75, 2023.

Cheng Chen, Juzheng Miao, Dufan Wu, Aoxiao Zhong, Zhiling Yan, Sekeun Kim, Jiang Hu, Zhengliang Liu, Lichao Sun, Xiang Li, et al. Ma-sam: Modality-agnostic sam adaptation for 3d medical image segmentation. *Medical Image Analysis*, 98:103310, 2024a.

Renqi Chen, Yuhui Cen, Hongbo Wang, Qiqi Pan, Shengjie Yan, and Jingjing Luo. Nas-adapter:: Adapting segmentation anything model with neural architecture search for medical image segmentation. 2026.

Tianrun Chen, Lanyun Zhu, Chaotao Ding, Runlong Cao, Yan Wang, Zejian Li, Lingyun Sun, Papa Mao, and Ying Zang. Sam fails to segment anything?–sam-adapter: Adapting sam in underperformed scenes: Camouflage, shadow, medical image segmentation, and more. *arXiv preprint arXiv:2304.09148*, 2023.

Tianrun Chen, Ankang Lu, Lanyun Zhu, Chaotao Ding, Chunan Yu, Deyi Ji, Zejian Li, Lingyun Sun, Papa Mao, and Ying Zang. Sam2-adapter: Evaluating & adapting segment anything 2 in downstream tasks: Camouflage, shadow, medical image segmentation, and more. *arXiv preprint arXiv:2408.04579*, 2024b.

Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Deng-Ping Fan, Ge-Peng Ji, Ming-Ming Cheng, and Ling Shao. Concealed object detection. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):6024–6042, 2021.

Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5356–5364, 2019.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

[2] https://anonymous.4open.science/r/Search2Adapt/

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Debesh Jha, Pia H Smedsrud, Michael A Riegler, Pål Halvorsen, Thomas De Lange, Dag Johansen, and Håvard D Johansen. Kvasir-seg: A segmented polyp dataset. In *International conference on multimedia modeling*, pp. 451–462. Springer, 2019.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European conference on computer vision*, pp. 709–727. Springer, 2022.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Neeraj Kumar, Ruchika Verma, Deepak Anand, Yanning Zhou, Omer Fahri Onder, Efstratios Tsougenis, Hao Chen, Pheng-Ann Heng, Jiahui Li, Zhiqiang Hu, et al. A multi-organ nucleus segmentation challenge. *IEEE transactions on medical imaging*, 39(5):1380–1391, 2019.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981, 2020.

Trung-Nghia Le, Tam V Nguyen, Zhongliang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. *Computer vision and image understanding*, 184:45–56, 2019.

Junghyup Lee and Bumsub Ham. Az-nas: Assembling zero-cost proxies for network architecture search. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5893–5903. IEEE, 2024.

Guihong Li, Yuedong Yang, Kartikeya Bhardwaj, and Radu Marculescu. Zico: Zero-shot nas via inverse coefficient of variation on gradients. *arXiv preprint arXiv:2301.11300*, 2023.

Ming Lin, Pichao Wang, Zhenhong Sun, Hesen Chen, Xiuyu Sun, Qi Qian, Hao Li, and Rong Jin. Zen-nas: A zero-shot nas for high-performance image recognition. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 337–346. IEEE, 2021.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.

Weihuang Liu, Xi Shen, Chi-Man Pun, and Xiaodong Cun. Explicit visual prompting for low-level structure segmentations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19434–19445, 2023.

Jun Ma, Yuting He, Feifei Li, Lin Han, Chenyu You, and Bo Wang. Segment anything in medical images. *Nature Communications*, 15(1):654, 2024.

Joe Mellor, Jack Turner, Amos Storkey, and Elliot J Crowley. Neural architecture search without training. In *International conference on machine learning*, pp. 7588–7598. PMLR, 2021.

Peter Naylor, Marick Laé, Fabien Reyal, and Thomas Walter. Segmentation of nuclei in histopathology images by deep regression of the distance map. *IEEE transactions on medical imaging*, 38 (2):448–459, 2018.

Zelin Peng, Zhengqin Xu, Zhilin Zeng, Xiaokang Yang, and Wei Shen. Sam-parser: Fine-tuning sam efficiently by parameter space reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 4515–4523, 2024.

Hieu Pham, Melody Guan, Barret Zoph, Quoc Le, and Jeff Dean. Efficient neural architecture search via parameters sharing. In *International conference on machine learning*, pp. 4095–4104. PMLR, 2018.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. In *International Conference on Learning Representations*, volume 2025, pp. 28085–28128, 2025.

Yanlin Wu, Zhihong Wang, Xiongfeng Yang, Hong Kang, Along He, and Tao Li. Trans-sam: Transfer segment anything model to medical image segmentation with parameter-efficient fine-tuning. *Knowledge-Based Systems*, 310:112909, 2025.

Aoran Xiao, Weihao Xuan, Heli Qi, Yun Xing, Ruijie Ren, Xiaoqin Zhang, Ling Shao, and Shijian Lu. Cat-sam: Conditional tuning for few-shot adaptation of segment anything model. In *European conference on computer vision*, pp. 189–206. Springer, 2024.

Yang Yu, Chen Xu, and Kai Wang. Ts-sam: Fine-tuning segment-anything model for downstream tasks. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2024.

Yuanhan Zhang, Kaiyang Zhou, and Ziwei Liu. Neural prompt search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019.

Han Zhou, Xingchen Wan, Ivan Vulić, and Anna Korhonen. Autopeft: Automatic configuration search for parameter-efficient fine-tuning. *Transactions of the Association for Computational Linguistics*, 12:525–542, 2024.

A Appendix

A.1 Theoretical Interpretation of Perturbation Search

The proposed perturbation-based search can be theoretically interpreted from three complementary perspectives related to local smoothness, variance reduction, and trust-region optimization, where $\mathcal{S}(\alpha)$ denotes the penalized architecture evaluation objective used during perturbation aggregation. First, under a local smoothness assumption, the zero-shot scoring proxy $\mathcal{S}(\alpha)$ admits a first-order Taylor approximation for small perturbations $\Delta\alpha$:

$$\mathcal{S}(\alpha + \Delta\alpha) \approx \mathcal{S}(\alpha) + \nabla_\alpha \mathcal{S}(\alpha)^\top \Delta\alpha. \tag{7}$$

This local linearization suggests that perturbation-based search operates within a predictable regime where performance changes remain approximately proportional to the perturbation magnitude. Second, sampling perturbation noise $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ from a continuous distribution can reduce the variance of architecture evaluation compared with discrete architecture sampling. Specifically, the variance $\mathrm{Var}\,\mathcal{S}(\alpha + \epsilon)$ decreases as perturbations concentrate around the current solution, leading to improved sample efficiency and more reliable gradient estimates. Third, the perturbation-based search can be interpreted as solving a trust-region optimization problem:

$$\max_{\|\Delta\alpha\|_2 \le \delta} \mathcal{S}(\alpha + \Delta\alpha). \tag{8}$$

This formulation explicitly constrains the magnitude of architectural updates within a trusted local region δ, preventing erratic jumps and ensuring convergence stability throughout the search process.

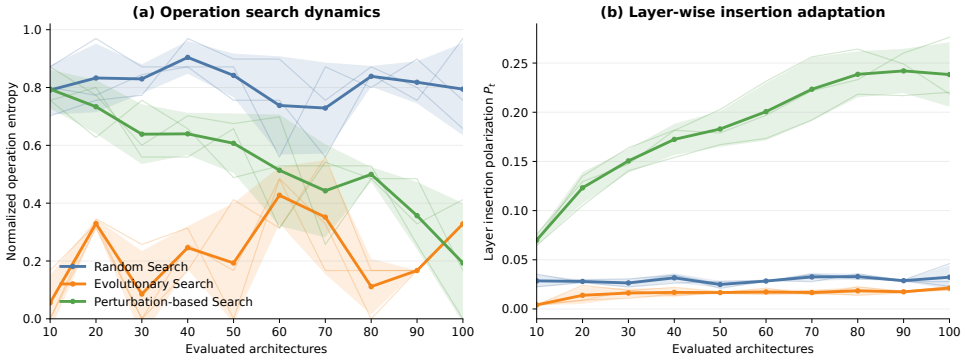

Figure 4: **Search dynamics under different architecture exploration strategies** (a) Normalized operation entropy over evaluated architectures. (b) Layer-wise insertion polarization during search. Solid lines and shaded regions indicate the mean and standard deviation across search seeds, respectively.

A.2 Analysis of Architecture Search Dynamics

We further analyze the search dynamics of random, evolutionary, and perturbation-based search under the same candidate evaluation budget from TNBC dataset. For a controlled comparison, all strategies operate on the same adapter search space under the same candidate evaluation budget, while differeing only in how architecture candidates are explored and selected. The evaluated architectures are divided into windows of 10 candidates to analyze the evolution of operation selection and layer-wise insertion throughout the search. As shown in Fig. 4(a), random search maintains relatively high operation entropy, whereas evolutionary search exhibits lower entropy with an early concentration on a narrower set of operations. In contrast, perturbation-based search shows an overall decrease in operation entropy with intermediate fluctuations, indicating a gradual transition from broad exploration toward score-guided concentration while retaining local re-exploration.

We further analyze layer-wise insertion using the polarization measure $\mathrm{P}_t = \frac{1}{24}\sum_{l=24}^{24} |\bar{\gamma}_{t,l} - 0.5|$, where $\bar{\gamma}_{t,l}$ denotes the mean modulation coefficient of layer l within the t-th evaluation window. A larger P_t represents a stronger deviation from the initial neutral insertion state of 0.5 As shown in Fig. 4(b), random and evolutionary search maintain relatively low polarization, whereas perturbation-based search progressively increases P_t throughout the search. Together, these results characterize the perturbation-based strategy as a structured search process that progressively concentrates operation choices while adapting layer-wise insertion strengths within the hierarchical adapter search space.

A.3 Ablation Studies

A.3.1 Generality across Foundation Models and Zero-cost Proxies

We conduct experiments with different combinations of foundation models and zero-cost scoring proxies to evaluate the generality of Search2Adapt. The proposed framework provides a flexible search mechanism that can be applied to different foundation models and proxy metrics without requiring a predefined adapter architecture. As shown in Tab. 2. Searh2Adapt discovers different adapter configurations depending on the backbone, proxy, and target dataset, while maintaining competitive segmentation performance across diverse domains. The detailed configurations of the searched architectures are provided in Tab. 3. Notably, different backbone-proxy combinations result in distinct adapter architectures even within the same target dataset, including variations in adpater operations and their operation-specific configurations. For example, Search2Adapt selects different adapter primitives and configurations for CAMO and COD-10K depending on the foundation model and scoring proxy. This observation indicates that the proposed search framework can flexibly adapt its architectural configuration to different search settings rather than relying on a universally predefined adapter design. These results demonstrate that Search2Adapt is not tied to a particular foundation model or zero-cost proxy, but can be instantiated with different backbone-proxy combinations

Table 2: **Evaluation of Search2Adapt across different foundation models and zero-cost proxies.** Different backbone-proxy combinations are evaluated across camouflaged object detection, polyp segmentation, and nuclei segmentation tasks to demonstrate the flexibility of the propose adapter architecture search framework. Results are reported as mean ± standard deviation over three independent search runs.

Camouflaged Object Detection					
		CAMO		**COD-10K**	
		S_m	E_ϕ	S_m	E_ϕ
SAM-Adapter		0.767	0.797	0.892	0.909
Foundation model	Zero-shot proxy	**Search2Adapt**			
SAM	NASWOT	0.814 ± 0.022	0.846 ± 0.027	0.821 ± 0.032	0.799 ± 0.045
SAM	ZiCo	0.834 ± 0.009	0.868 ± 0.011	0.812 ± 0.043	0.796 ± 0.021
SAM2	NASWOT	0.861 ± 0.008	0.892 ± 0.008	0.933 ± 0.002	0.947 ± 0.003
SAM2	ZiCo	0.868 ± 0.002	0.901 ± 0.002	0.947 ± 0.003	0.951 ± 0.001
Polyp Segmentation					
		Kvasir-SEG		**Polypgen**	
		Dice	mIoU	Dice	mIoU
SAM-Adapter		0.850	0.776	0.662	0.573
Foundation model	Zero-shot proxy	**Search2Adapt**			
SAM	NASWOT	0.891 ± 0.015	0.807 ± 0.017	0.782 ± 0.008	0.782 ± 0.008
SAM	ZiCo	0.898 ± 0.003	0.814 ± 0.004	0.678 ± 0.043	0.583 ± 0.063
SAM2	NASWOT	0.916 ± 0.002	0.841 ± 0.002	0.819 ± 0.007	0.762 ± 0.006
SAM2	ZiCo	0.912 ± 0.002	0.836 ± 0.002	0.809 ± 0.009	0.755 ± 0.009
MedSAM	NASWOT	0.900 ± 0.015	0.819 ± 0.020	0.713 ± 0.028	0.649 ± 0.031
MedSAM	ZiCo	0.884 ± 0.008	0.799 ± 0.011	0.677 ± 0.011	0.614 ± 0.012
Nuclei Segmentation					
		MoNuSeg		**TNBC**	
		Dice	mIoU	Dice	mIoU
SAM-Adapter		0.800	0.667	0.786	0.657
Foundation model	Zero-shot proxy	**Search2Adapt**			
SAM	NASWOT	0.829 ± 0.003	0.709 ± 0.004	0.822 ± 0.005	0.703 ± 0.007
SAM	ZiCo	0.792 ± 0.017	0.655 ± 0.022	0.815 ± 0.002	0.694 ± 0.003
SAM2	NASWOT	0.819 ± 0.002	0.694 ± 0.003	0.824 ± 0.005	0.705 ± 0.007
SAM2	ZiCo	0.823 ± 0.001	0.700 ± 0.002	0.829 ± 0.003	0.713 ± 0.004
MedSAM	NASWOT	0.800 ± 0.001	0.667 ± 0.002	0.778 ± 0.025	0.648 ± 0.029
MedSAM	ZiCo	0.785 ± 0.003	0.647 ± 0.005	0.785 ± 0.006	0.656 ± 0.008

for adapter architecture discovery. Overall, the results suggest that search-based adapter adaptation provides a more generalizable design strategy than relying on a single fixed adapter architecture across different foundation models and downstream domains.

A.3.2 Analysis of Hyper-parameter Settings

We conduct ablation experiments on hyperparameters including the number of search iterations and perturbation samples on the CAMO dataset. The results are shown in Tab. 4 and Tab. 5. The iteration number N controls how many times the perturbation-based search process is repeated, determining the number of iterative updates of the architecture parameters α. The experimental results show that $N = 20$ achieves the best segmentation performance, although increasing the number of iterations leads to additional search cost. Meanwhile, K controls how many perturbed architecture candidates are generated around the current adapter configuration within each iteration, enabling local exploration through score-weighted perturbation aggregation. The experimental results show that $K = 5$ achieves the best segmentation performance among the evaluated settings while requiring a moderate search cost. Notably, increasing either N or K does not monotonically improve segmen-

Table 3: **Adapter architecture discovered by Search2Adapt across different foundation models and zero-cost proxies.** The searched adapter operation and its configuration are reported for each backbone-proxy combindation on CAMO and COD-10K.

Dataset	Foundation Model	Zero-shot proxy	Adapter	Configuration
CAMO	SAM	NASWOT	gated mlp	dim: 128, gate: geglu
	SAM	ZiCo	channel attention	activation: gelu, attention type: eca, dim: 64
	SAM2	NASWOT	low rank	rank: 4
	SAM2	ZiCo	edge aware	activation: silu, dim:32, edge mode: laplacian
COD-10K	SAM	NASWOT	gated mlp	dim: 128, gate: geglu
	SAM	ZiCo	frequency	activation: silu, dim: 64, freq mode: laplacian
	SAM2	NASWOT	edge aware	activation: silu, dim:64, edge mode: sobel
	SAM2	ZiCo	frequency	activation: gelu, dim: 16, freq mode: token smoothing

Table 4: **Effect of the number of search iterations N on segmentation performance and search cost on CAMO.** GPU hours denote the computational cost required for architectural search.

Perturbation iterations N	S_m	E_ϕ	GPU hours
5	0.701	0.650	0.56
10	0.786	0.883	2.22
15	0.461	0.461	3.26
20	0.822	0.856	4.17

tation performance, indicating that an appropriate balance between perturbation diversity, iterative refinement, and search stability is important for effective architecture exploration.

Tab. 6 compares the performance of continuous layer-wise modulation and hard insertion strategies for adapter insertion. The continuous strategy adaptively controls the layer-wise insertion strength γ discovered during the search process, whereas the hard insertion strategy binarizes the insertion pattern by thresholding γ with $\tau = 0.5$. Across both datasets, continuous layer-wise modulation consistently achieves better segmentation performance than hard insertion. In addition, the continuous strategy achieves higher architecture scores on both CAMO and Kvasir-SEG, indicating that retaining the searched modulation coefficients provides a more effective representation of layer-wise adapter insertion than directly bianrizing them. These results suggest that continuous layer-wise modulation provides an effective adaptation mechanism for segmentation-oriented transformer architectures.

A.4 Visualization results of Various Domains

We visualize the segmentation results of the proposed method and comparison models in Fig. 5. While SAM (Kirillov et al., 2023) and SAM2 (Ravi et al., 2025) perform zero-shot segmentation without task-specific training, their prediction quality varies across different segmentation domains. SAM-Adapter Chen et al. (2023) and SAM2-Adapter (Chen et al., 2024b) adapt the foundation models to individual target domains through task-specific training with predefined adapter architectures. In contrast, Seach2Adapt additionally discovers adapter architectures according to the target domain rather than relying on a fixed architectural configuration. The qualitative results show that the searched adapters produce segmentation masks that generally follow the target structures across camouflaged object detection, polyp segmentation, and nuclei segmentation. Despite the substantial differences in visual and structural characteristics across these domains, the adapters discovered for each target dataset provide effective segmentation without relying on a universally fixed adapter ar-

Table 5: **Effect of the number of perturbation samples K on segmentation performance and search cost on CAMO.** GPU hours denote the computational cost required for architecture search.

Perturbation iterations N	S_m	E_ϕ	GPU hours
3	0.467	0.478	1.32
5	0.822	0.856	2.21
7	0.764	0.757	3.05
10	0.524	0.477	4.36

Table 6: **Ablation study of continuous and hard layer-wise adapter insertion.** We compare continuous layer modulation with binary insertion patterns on CAMO and Kvasir-SEG.

	CAMO			**Kvasir-SEG**		
	S_m	E_ϕ	Score	Dice	mIoU	Score
Continuous	0.822	0.856	0.655	0.895	0.811	0.642
Hard	0.686	0.722	0.591	0.737	0.588	0.589

chitecture. These results qualitatively support the role of domain-specific architecture discovery in adapting foundation models to heterogeneous segmentation tasks.

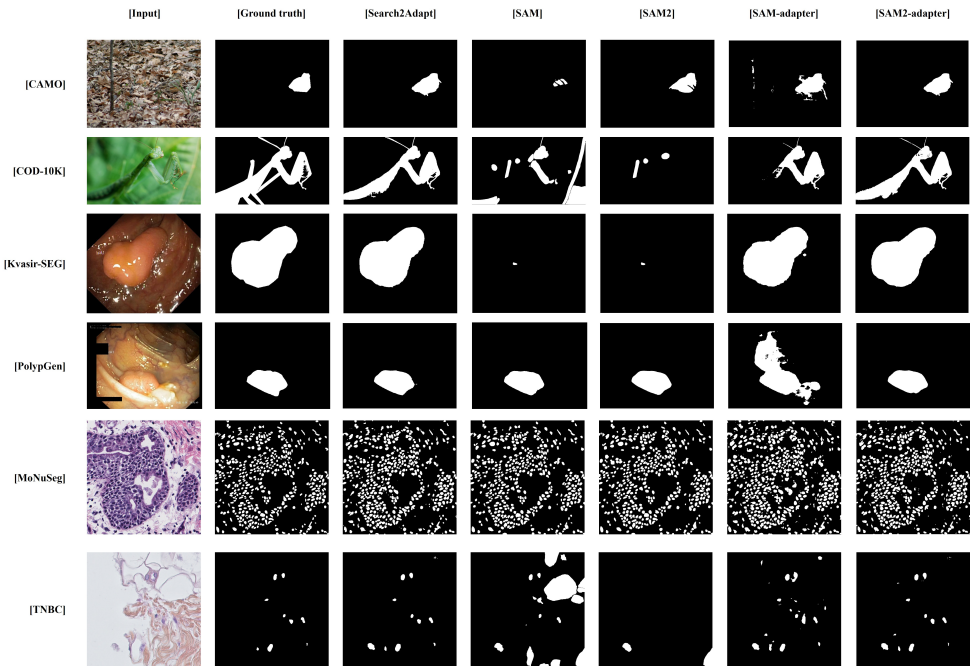

Figure 5: **Qualitative comparison of segmentation results across different target domains.** From left to right, the columns show the input image, ground truth, and segmentation predictions from Search2Adapt, SAM, SAM2, SAM-Adapter, and SAM2-Adapter. The examples cover camouflaged object detection (CAMO and COD-10K), polyp segmentation (Kvasir-SEG and PolypGen), and nuclei segmentation (MoNuSeg and TNBC). The searched adapter architectures produce segmentation masks that effectively capture target structures across diverse domain-specific visual characteristics.