InfoSAM: Fine-Tuning the Segment Anything Model from An Information-Theoretic Perspective

Yuanhong Zhang [*] [1] [2] **Muyao Yuan** [*] [1] [2] **Weizhan Zhang**[†] [1] [2] **Tieliang Gong** [1] [3] **Wen Wen** [1] [3] **Jiangyong Ying** [4] **Weijie Shi** [5]

Abstract

The Segment Anything Model (SAM), a vision foundation model, exhibits impressive zero-shot capabilities in general tasks but struggles in specialized domains. Parameter-efficient fine-tuning (PEFT) is a promising approach to unleash the potential of SAM in novel scenarios. However, existing PEFT methods for SAM neglect the domain-invariant relations encoded in the pre-trained model. To bridge this gap, we propose InfoSAM, an information-theoretic approach that enhances SAM fine-tuning by distilling and preserving its pre-trained segmentation knowledge. Specifically, we formulate the knowledge transfer process as two novel mutual information-based objectives: (i) to compress the domain-invariant relation extracted from pre-trained SAM, excluding pseudo-invariant information as possible, and (ii) to maximize mutual information between the relational knowledge learned by the teacher (pre-trained SAM) and the student (fine-tuned model). The proposed InfoSAM establishes a robust distillation framework for PEFT of SAM. Extensive experiments across diverse benchmarks validate InfoSAM's effectiveness in improving SAM family's performance on real-world tasks, demonstrating its adaptability and superiority in handling specialized scenarios. The code and models are available at InfoSAM project page.

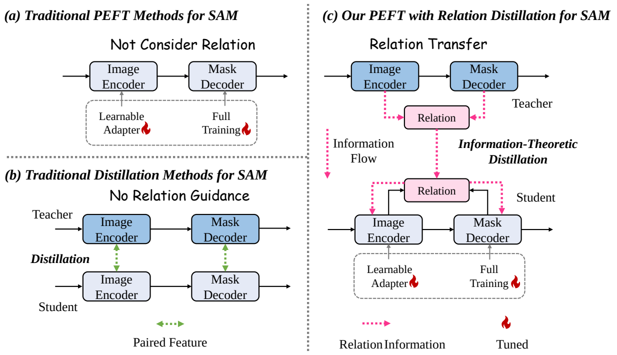


Figure 1: Comparing traditional PEFT and distillation paradigms with our proposed InfoSAM. (a) Existing PEFT methods for SAM directly adjust the trainable parameters of each module individually, often overlooking the cross-module relationships. (b) Traditional SAM distillation methods focus on model compression via paired feature alignment but lack relational guidance during projection training. (c) In contrast, our InfoSAM method enhances PEFT by incorporating information-theoretic distillation, enabling the transfer of domain-invariant relations from the pre-trained SAM to the fine-tuning student.

1. Introduction

Recently, the Segment Anything Model (SAM) (Kirillov et al., 2023; Ravi et al., 2022) emerged as a prominent foundation model for image segmentation. While SAM demonstrates exceptional zero-shot performance on generic object segmentation, it often struggles with domain-specific real-world segmentation tasks (Chen et al., 2023; Zhong et al., 2024). Therefore, Parameter-Efficient Fine-Tuning (PEFT) for SAM (Song et al., 2024; Zhang et al., 2024a; Peng et al., 2024b) has gained attention as a promising solution, significantly reducing the fine-tuning costs associated with SAM's large pre-trained parameter set. Existing PEFT methods for SAM primarily focus on fine-tuning the heavy image encoder (Song et al., 2024; Peng et al., 2024b) or aligning domain-specific features between the mask decoder and image encoder (Xiao et al., 2025). However, a

[*] Equal contribution [1] School of Computer Science and Technology, Xi'an Jiaotong University, Xi'an, China [2] Ministry of Education Key Laboratory of Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an, China [3] Shaanxi Province Key Laboratory of Big Data Knowledge Engineering, Xi'an Jiaotong University, Xi'an, China [4] China Telecom E-surfing Vision Technology Co., Ltd, Hangzhou, China [5] School of Electrical Engineering, Xi'an Jiaotong University, Xi'an, China. Correspondence to: [†] Weizhan Zhang <zhangwzh@xjtu.edu.cn>.

Proceedings of the 42nd International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

promising improvement avenue is overlooked: preserving the beneficial information in pre-trained models.

Notably, SAM follows an encoder-decoder architecture, where the mask decoder refines the image embeddings extracted by the image encoder to localize objects. Unified training or fine-tuning methods (Shu et al., 2025; Xiao et al., 2025) have demonstrated effectiveness within this framework. This suggests that preserving the implicit relationship between the encoder and decoder could be beneficial for model fine-tuning. This relationship may stem from extensive pre-training and be embedded in the feature distributions, making it delicate and easily disrupted by unrefined PEFT methods (Wang et al., 2024). We argue that this is because task-specific tuning tends to override or suppress the universal visual features learned during pre-training.

To enhance PEFT by leveraging implicit relationships, a natural approach is to extract these relationships from foundation models and inject them into fine-tuned models tailored for specific domains. However, not all implicit relationships are beneficial for downstream tasks—only the key domain-invariant relationships learned from across domains (Hoffman et al., 2018; Xu et al., 2022) contribute positively to every fine-tuned model. While knowledge distillation serves as a flexible bridge for transferring information between models (Gou et al., 2021), we propose to adopt a distillation approach between the pre-trained model and fine-tuned model to retain domain-invariant relationships.

Therefore, this brings us to two key challenges: *1) How can we extract the domain-invariant relationship from pre-trained foundation models? 2) How can we effectively transfer the extracted information to fine-tuned models?*

To address these challenges, we propose InfoSAM, a novel information-theoretical distillation method specifically designed for SAM PEFT. In order for the teacher to provide a good amount of the domain-invariant information, first we have to find out how this information could be quantified. To this aim, we introduce a robust and efficient Rényi's entropy-based quantification from information theory (Ahn et al., 2019) to measure such a relation. However, not all the relations in the pre-trained SAM are domain-invariant, there exists some pseudo-invariant information (e.g., color), which may negatively impact the generalization ability during the fine-tuning process (Li et al., 2022a). Therefore, **to address the first challenge**, we propose an attention-driven relation module specifically designed to extract critical structural patterns from the pre-trained SAM. By minimizing mutual information between the module's outputs and both encoder-decoder embeddings of SAM, it constructs an effective bottleneck that forces the module to maintain compressed yet domain-invariant representations. Furthermore, **to tackle the second challenge**, we effectively distill the valuable relational knowledge from pre-trained SAM to the fine-tuned SAM by maximizing the mutual information between their extracted relations. This ensures faithful propagation of compressed semantic dependencies, thereby facilitating a more effective fine-tuning process. Our experiments on SAM and SAM2, evaluated across 4 diverse domains and 8 datasets, show that InfoSAM achieves superior adaptation and segmentation performance.

Overall, our contribution can be summarized as follows:

- We present InfoSAM, the first information-theoretic framework for SAM adaptation, introducing an innovative distillation approach tailored for SAM PEFT to enhance performance in new scenarios.
- InfoSAM proposes novel dual complementary mechanisms for SAM adaptation: a relational bottleneck that strategically compresses task-irrelevant dependencies while preserving domain-invariant semantics, coupled with adaptive cross-model mutual information maximization ensuring provable preservation of essential structural knowledge.
- We conduct a comprehensive benchmark across diverse domains, including natural images, medical imaging, agriculture, and remote sensing. InfoSAM consistently demonstrates superior performance compared to other PEFT and distillation techniques across various downstream tasks.

2. Related Work

2.1. Parameter Efficient Fine-Tuning for SAM

Parameter-Efficient Fine-Tuning (PEFT) alleviates the challenges of task-specific deployment in large foundation models by fine-tuning only a minimal subset of parameters while keeping the majority frozen. Several prior works explore fine-tuning SAM for downstream tasks. SAM-Adapter (Chen et al., 2023) is one of the pioneering works applying the PEFT method to SAM, incorporating task-specific prompts for each adapter. SU-SAM (Song et al., 2024) presents a simple framework to efficiently fine-tune the SAM with Adapter or LoRA. SAM-COBOT (Peng et al., 2024b) boosts existing PEFT techniques for fine-tuning SAM through cross-block orchestration. BLO-SAM (Zhang et al., 2024a) finetunes SAM based on bi-level optimization, eliminating the need for manual prompts by a learnable prompt embedding. Conv-LoRA (Zhong et al., 2024) integrates ultra-lightweight convolutional parameters into LoRA, injecting image-related inductive biases into the plain ViT encoder.

However, the above methods overlook preserving pre-trained information in foundation models during fine-tuning. Our work explores enhancing fine-tuning methods for SAM

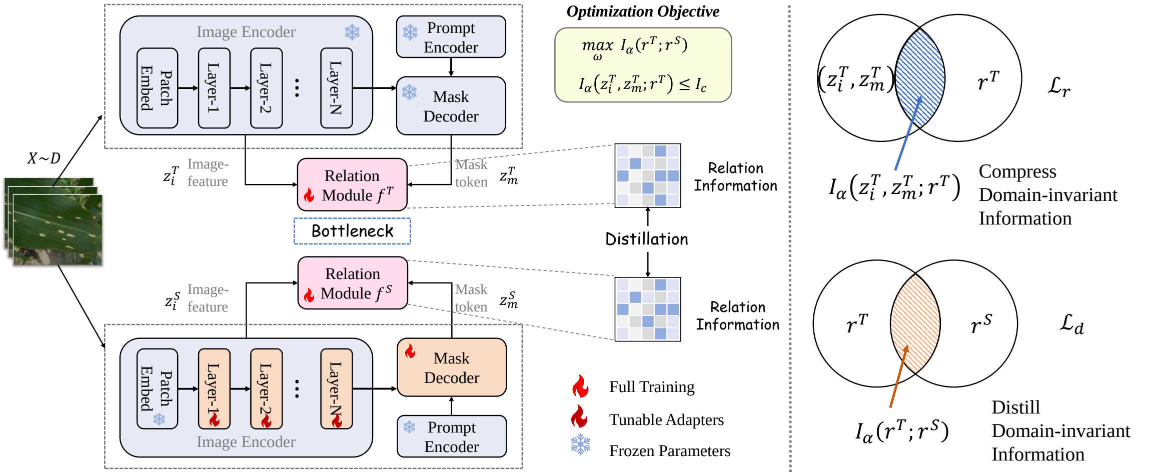


Figure 2: **The Flowchart of InfoSAM.** To leverage the domain-invariant relationships within modules from a well-trained foundation model (i.e., SAM) for enhancing PEFT. InfoSAM pioneers an information-theoretic framework for parameter-efficient SAM adaptation through two synergistic components: 1) Strategic compression of task-irrelevant dependencies while preserving domain-invariant feature relationships through optimized interaction between image embeddings z_i^T and mask tokens z_m^T (Eq.5), and 2) Cross-model mutual information maximization to ensure faithful knowledge transfer (Eq.6). The right Venn diagrams illustrate the information constraint from the optimization problem.

from a novel perspective: leveraging information-based distillation to maintain domain-invariant relationships.

2.2. Knowledge Distillation for SAM

Knowledge distillation (KD) (Gou et al., 2021) effectively transfers knowledge from a large, well-trained model (teacher) to a smaller or simpler one (student). When applying KD to SAM, most efforts focus on compressing and transferring representations for downstream tasks. MobileSAM (Zhang et al., 2023a) distills SAM's ViT encoder into a TinyViT, while TinySAM (Shu et al., 2025) uses full-stage KD. Other approaches distill SAM's semantic priors for tasks like medical segmentation (Dong et al., 2024; Shen et al., 2024) and image restoration (Zhang et al., 2024b). However, these methods focus on paired feature maps, neglecting the inter-module relationships within the teacher SAM. To address this, our approach utilizes information-theoretic principles to extract and transfer compact inter-module relationships to the student model.

2.3. Domain-invariant Information in SAM

The concept of domain-invariant information was first introduced in prior works on domain adaptive segmentation (DAS), which explored cross-domain invariant features such as edge and structural information (Hoffman et al., 2018). DAS aims to learn domain-invariant representations across multiple domains and follows two main approaches: (i) extraction and refinement of domain-invariant features,where methods like feature disentanglement (Chang et al., 2019) or analysis (Xu et al., 2022) decompose images into domain-invariant (e.g., shapes, edges) and domain-specific (e.g., textures, colors) components, aiming to enhance the former while suppressing the latter; (2) GAN-based domain-invariant feature generation, which employs adversarial training to align domains at different levels: image (Li et al., 2022b), feature (Ma et al., 2024), and output (Huang et al., 2022). For example, GLGAN (Ma et al., 2024) integrates multi-scale global and local features to improve cross-domain transferability in remote sensing.

SAM's large-scale pretraining encodes domain-invariant patterns for strong zero-shot generalization. Recent works leverage these universal visual patterns for downstream tasks (Peng et al., 2024a). However, these methods rely on complex designs or external data to learn representations. In contrast, we focus on preserving the domain-invariant information in pre-trained SAM for fine-tuning.

3. Preliminaries

3.1. Rényi's α-entropy and Mutual Information

In information theory, matrix-based Rényi's α-entropy provides a novel way to quantify single-variable information or interactions across variables directly from samples. Unlike Shannon entropy, it leverages the eigenspectrum of a Gram matrix in reproducing kernel Hilbert space (RKHS), avoiding costly distribution evaluations(Gong et al., 2022).

Definition 1. Let $\kappa : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ be an infinitely divisible positive kernel (Bhatia, 2006). Given $\{x_i\}_{i=1}^n \subset \mathcal{X}$, each x_i being a real-valued scalar or vector, and the Gram matrix K obtained from $K_{ij} = \kappa(x_i, x_j)$, a matrix-based analog to Rényi's entropy can be defined as:

$$\mathbf{S}_\alpha(\mathbf{A}) = \frac{1}{1-\alpha} \log_2 \left[\sum_{i=1}^{n} \lambda_i^\alpha(\mathbf{A}) \right] \tag{1}$$

where the kernel matrix $\mathbf{A}_{ij} = \frac{1}{n} \frac{K_{ij}}{\sqrt{K_{ii}K_{jj}}}$ is the normalized version of K and $tr(\mathbf{A}) = 1$. The $\lambda_i(\mathbf{A})$ denotes the ith eigenvalue of $\mathbf{A}$.

Definition 2. Given n pairs of samples $\{z_i = (x_i, y_i)\}_{i=1}^n$, and two positive definite kernels $\kappa_1 : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ and $\kappa_2 : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$. After computing the Gram matrix $\mathbf{A}$ and $\mathbf{B}$, a joint Rényi's entropy can be defined as:

$$\mathbf{S}_\alpha(\mathbf{A}, \mathbf{B}) = \mathbf{S}_\alpha \left(\frac{\mathbf{A} \circ \mathbf{B}}{\mathrm{tr}(\mathbf{A} \circ \mathbf{B})} \right) \tag{2}$$

where $(\mathbf{A} \circ \mathbf{B})$ denotes the Hadamard product between the matrices $\mathbf{A}$ and $\mathbf{B}$. The mutual information $\mathbf{I}_\alpha(\mathbf{A}; \mathbf{B})$ can be computed as:

$$\mathbf{I}_\alpha(\mathbf{A}; \mathbf{B}) = \mathbf{S}_\alpha(\mathbf{A}) + \mathbf{S}_\alpha(\mathbf{B}) - \mathbf{S}_\alpha(\mathbf{A}, \mathbf{B}) \tag{3}$$

The matrix-based Rényi's mutual information eliminates the need for high-dimensional probability density estimation of Shannon entropy, offering a more accurate and computationally efficient solution (Dong et al., 2023).

4. Methodology

4.1. Background and Notions

The overview of InfoSAM is illustrated in Fig. 2. Given a teacher model and a student model, we denote the pre-trained SAM as ϕ^T and the fine-tuned SAM as ϕ^S, which are parameterized by ω. Let $X \sim \mathcal{D}$ be an input sampled from the downstream dataset $\mathcal{D}$. The representations produced by $\phi^T(X)$ and $\phi^S(X)$ are defined as follows: The output features of the image encoder are denoted as z_i^T and z_i^S, where $z_i^T, z_i^S \in \mathbb{R}^{B \times H \times W \times D}$. Here, B is the batch size, H and W represent the height and width, respectively, and D is the dimension of the image embeddings. Similarly, the output tokens from the two-way transformer in the mask decoder are denoted as z_m^T and z_m^S, where $z_m^T, z_m^S \in \mathbb{R}^{B \times N \times D}$. These tokens encode the target mask information in a more abstract manner. Here, N represents the number of masks, and the output token shares the same dimension D as the image embeddings.

The goal of PEFT is to fine-tune $\phi^S(X; \omega)$ for adaptation to a new downstream task under the supervision of the teacher model ϕ^T, where ω denotes the trainable PEFT parameters. To enhance the PEFT process using the frozen pre-trained teacher SAM, the loss can be formulated as:

$$\omega^* = \arg\min_\omega \mathcal{L}(X, Y \mid T), \tag{4}$$

where T represents intermediate features extracted by the teacher model, capturing the relational information within SAM modules, and Y is the full dense label map. Following prior work, the task-specific loss function $\mathcal{L}(\cdot)$ is chosen as the structure loss (Zhong et al., 2024).

In this paper, rather than directly aligning paired representations between teacher and student (Zhang et al., 2023a; Shu et al., 2025), we leverage robust prior relational information from the pre-trained SAM to guide the PEFT process.

4.2. An Information View of SAM Distillation

Problem Formulation. Intuitively, the relationships between different modules in a well-trained foundation model are invaluable, as they are learned from extensive datasets. However, traditional PEFT methods for SAM, when fine-tuned to downstream tasks, risk disrupting these relationships. In this way, we need to address two key questions: how to capture critical relations from the pre-trained SAM and how to effectively transfer it to the fine-tuned model.

Firstly, by treating the teacher model as a mapping function, we argue that the critical relation information resides in multivariate mutual information $I_\alpha(z_i^T, z_m^T; r^T)$, where z_i^T, z_m^T and r^T represent the image embedding, mask token and relational interactions of the teacher, respectively. As such, $I_\alpha(z_i^T, z_m^T; r^T)$ quantifies how much information r^T can tell about (z_i^T, z_m^T). Crucially, this mutual information constitutes a learnable bottleneck that fundamentally constrains the knowledge transfer process. The bottleneck mechanism enforces selective attention by restricting the information flow to a compressed representation, where only the most salient teacher-student interactions can be preserved. This is particularly vital as not all relations in the pre-trained SAM are universally transferable. For example, invariant features (e.g., geometric outlines) that exhibit cross-domain consistency are effective, while pseudo-invariant features (e.g., color distributions) that carry domain-specific biases need to be suppressed (Li et al., 2022a).

To prioritize domain-invariant relations, we constrain the information flow via an upper bound I_c:

$$\mathbf{I}_\alpha(z_i^T, z_m^T;\, r^T) \le I_c \tag{5}$$

where $r^T = f^T(z_i^T, z_m^T; \theta)$ represents the teacher's relational mapping between image embeddings and mask tokens. The relation module is defined as $f^T(z_i^T, z_m^T; \theta) : \mathbb{R}^{B \times H \times W \times D} \times \mathbb{R}^{B \times N \times D} \to \mathbb{R}^{B \times N \times (H \cdot W)}$. The θ represents the parameters of the relation module. This compression forces the module to retain only essential information for distillation.

After that, we employ a distillation approach to transfer the core relationships by maximizing their mutual information:

$$\begin{aligned} &\max_{\omega} && \mathbf{I}_\alpha(r^T; r^S) \\ &\text{subject to} && \mathbf{I}_\alpha(z_i^T, z_m^T;\, r^T) \le I_c \end{aligned} \tag{6}$$

where $r^S = f^S(z_i^S, z_m^S; \theta)$ denotes the student's relation in fine-tuned SAM, with f^S sharing the same parameters as f^T. The information bottleneck principle operates through two coupled mechanisms: (i) *compression* via minimizing $\mathbf{I}_\alpha(z_i^T, z_m^T; r^T)$ to extract minimal sufficient statistics r^T from (z_i^T, z_m^T), and (ii) *distillation* via maximizing $\mathbf{I}_\alpha(r^T; r^S)$ to preserve maximal predictive information. The Lagrangian formulation explicitly implements this trade-off:

$$\max_{\omega} \quad \mathbf{I}_\alpha(r^T; r^S) - \beta \mathbf{I}_\alpha(z_i^T, z_m^T;\, r^T) \tag{7}$$

where β is a hyper-parameter for trade-off.

Compressing Intra-SAM Relations. To efficiently capture the relationship within pre-trained SAM, we propose an attention-based module designed for extraction, illustrated in Fig. 3. Given z_i^T, the output of the image encoder of SAM, and z_m^T, the mask token embedding, as the input of relation module f^T. It mainly uses a combination of attention mechanisms and residual connections.

First, both z_i^T and z_m^T are passed through a Layer Normalization step to stabilize the features. After that, z_m^T and z_i^T are linearly projected into a query vector $Q \in \mathbb{R}^{B \times N \times D}$ and a key vector $K \in \mathbb{R}^{B \times (H \cdot W) \times D}$, respectively:

$$\begin{aligned} Q &= W_Q \cdot LayerNorm(z_m^T), \\ K &= W_K \cdot LayerNorm(z_i^T) \end{aligned} \tag{8}$$

where $W_Q, W_K \in \mathbb{R}^{D \times D}$ are learnable projection matrices. The attention scores are computed by combining two components: the scaled dot product of Q and K and the residuals from the dot product of z_m^T and z_i^T. The scores are summed to produce the final attention map:

$$S_\alpha = \frac{QK^\top}{\sqrt{D}} + z_m^T \cdot z_i^{T\top} \tag{9}$$

where S_α is the attention score. To ensure consistency and comparability, α is flattened and normalized using ℓ_2-normalization, resulting in the final output of f^T, denoted as r^T. To encourage the relation encoding process to focus on domain-invariant information, the first loss for relation compression can be expressed as:

$$\begin{aligned} \mathcal{L}_r =& \mathbf{I}_\alpha(z_i^T, z_m^T; r^T) \\ =& \cancel{\mathbf{S}_\alpha(G_i^T, G_m^T)} + \mathbf{S}_\alpha(G_r^T) - \mathbf{S}_\alpha(G_i^T, G_m^T, G_r^T) \end{aligned} \tag{10}$$

where $G_i^T, G_m^T, G_r^T \in \mathbb{R}^{N \times N}$ are the Gram matrices induced by a batch of normalized features z_i^T, z_m^T, and the output of r^T with a polynomial kernel of degree 1. Notably, the teacher entropy term in this loss is excluded, as the teacher's weights remain fixed during PEFT.

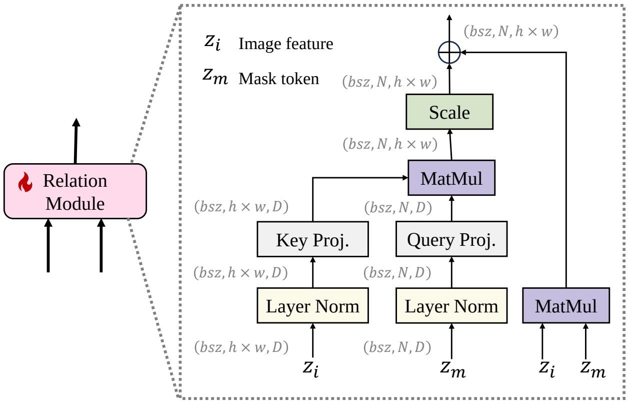


Figure 3: The architecture of attention-based relation module. It is designed to capture the relationship between image encoder and mask decoder, facilitating effective interaction between these components in SAM.

According to Eq.(1), computing eigenvalues of large matrices is computationally intensive (Kerr et al., 2009; Yu et al., 2019). To mitigate this, we set $\alpha = 2$, allowing us to compute matrix-based Rényi's α-entropy via the Frobenius norm: $|\mathbf{A}\|_F^2 = \text{tr}(\mathbf{A}\mathbf{A}^H) = \sum_{i=1}^n \lambda_i^2(\mathbf{A})$. Consequently, L_r can be reformulated as:

$$\mathcal{L}_r = -\log_2 \|G_r^T\|_F^2 + \log_2 \|G_{imr}^T\|_F^2 \tag{11}$$

where $G_{imr}^T = G_i^T \circ G_m^T \circ G_r^T$. The $\circ$ is Hadamard product. The first term in $\mathcal{L}_r$ acts as a spectral compression regularizer that constrains the relation module and encourages it to learn more compact and refined representations. The second term minimizes the joint entropy of the feature interactions across the image encoder, mask decoder, and relation module, effectively filtering spurious relationships and preserving domain-invariant interactions critical for cross-domain adaptation.

Maximizing Inter-SAM Relations. After extracting the essential relationships between the image encoder and the mask decoder, we transfer the relationships by minimizing their distance. A natural choice to accomplish this is by maximizing the mutual information between the two representations. While most existing works (Ahn et al., 2019; Kuang et al., 2023) focus on minimizing a lower bound of mutual information, we directly maximize the matrix-based Rényi's mutual information itself to avoid the expensive evaluation of underlying distribution for distillation loss:

$$\begin{aligned} \mathcal{L}_d =& -\mathbf{I}_\alpha(r^T; r^S) \\ =& -\mathbf{S}_\alpha(G_r^T) - \mathbf{S}_\alpha(G_r^S) + \mathbf{S}_\alpha(G_r^T, G_r^S) \end{aligned} \tag{12}$$

Similarly, G_r^T and G_r^S denote the Gram matrices corresponding to the student and teacher relations, respectively. We denote the $G_r^{TS} = G_r^T \circ G_r^S$, then the distillation loss can be expressed as:

$$\mathcal{L}_d = \log_2 \|G_r^T\|_F^2 + \log_2 \|G_r^S\|_F^2 - \log_2 \|G_r^{TS}\|_F^2 \quad (13)$$

Consistent with $\mathcal{L}_r$, the $\mathcal{L}_d$ also sets the entropy order α to 2 and utilizes the Frobenius norm for equivalent transformation. From this perspective, the components of $\mathcal{L}_d$ can be viewed as regularization terms (i.e., the first two terms) and a relation alignment (i.e., the third term) between two models, while $\log_2$ improving robustness to relations.

Overall, combining Eq.(11) and Eq.(13), the final objective of relation compression and transfer can be defined as:

$$\mathcal{L}_{info} = \lambda_1 * \mathcal{L}_r + \lambda_2 * \mathcal{L}_d \quad (14)$$

where λ_1 and λ_2 are hyper-parameters to trade-off between sufficiency (domain-invariant information transmitted from r^T to r^S) and minimality (the complexity of r^T). Further details and PyTorch-style pseudocode for InfoSAM are provided in Appendix A.2 and A.3.

4.3. Applying information theory to SAM

Overall Loss Function. Following previous works (Zhong et al., 2024), we incorporate the proposed information-theoretic distillation loss $\mathcal{L}_{info}$ with a structure loss $\mathcal{L}_{ce}$ (Fan et al., 2020b), which combines the weighted IoU loss and binary cross-entropy loss. The overall loss function is derived as:

$$\mathcal{L} = \mathcal{L}_{ce} + \mathcal{L}_{info} \quad (15)$$

Finally, we employ this new loss function for fine-tuning SAM. During fine-tuning, we first learn robust relations and then transfer this knowledge. The $\mathcal{L}_{info}$ regulates information flow between SAM's hierarchical representations, avoiding over-retention of low-level details while enhancing geometrically critical features. This aligns with the rate-distortion tradeoff in information bottleneck theory (Tishby & Zaslavsky, 2015), where information is compressed and then generalized.

5. Experiments

Settings. We conduct experiments using SAM (Kirillov et al., 2023) (with a ViT-B backbone) and SAM2 (Ravi et al., 2022) (with a Hiera-B+ backbone) with Adapter (Chen et al., 2022; Song et al., 2024), and LoRA (Hu et al., 2022) across four real-world domains: medical imaging, natural images, agriculture, and remote sensing. We fine-tune SAM's image encoder by adding adapters or LoRA, while fully training the decoder directly. We use a batch size of 4 and the Adam optimizer with an initial learning rate of 2×10^{-4}, utilizing a CosineAnnealing scheduler that decays to a final learning rate of 2×10^{-5}. All the methods are trained for 10 epochs with structure loss (i.e., the combination of weighted IoU loss and binary cross entropy loss) unless otherwise specified. During training, prompts are randomly selected from noised ground truth boxes and points at a 1:1 ratio. During evaluation, ground truth boxes are used as the default geometric input prompts to ensure a fair comparison and minimize randomness. More implementation details are provided in Appendix B.

Datasets. In the natural image domain, we focus on camouflaged object segmentation (Skurowski et al., 2018; Le et al., 2019; Fan et al., 2020a). For medical imaging, we investigate polyp segmentation (Bernal et al., 2015; Jha et al., 2020) and skin lesion segmentation (Codella et al., 2018). In agriculture and remote sensing, we use leaf disease segmentation (Rath, 2023) and road segmentation datasets (Mnih, 2013) as representative examples, respectively. For further details on the tasks and datasets, please refer to Appendix C.

To verify the effectiveness of our approach, we compare it with two categories of methods: PEFT methods and distillation methods.

PEFT Baselines. The PEFT baselines encompass three types of methods: the direct application of SAM, PEFT methods from the NLP or CV domain, and PEFT methods designed for SAM. These are as follows: 1) The zero-shot performance of the original SAM. 2) Fine-tune SAM's mask decoder only. 3) BitFit (Ben Zaken et al., 2022), which only fine-tunes bias terms in the pre-trained model. 4) AdaptFormer (Chen et al., 2022), which inserts the trainable bottleneck layers into the MLP block of the transformer. 5) LoRA (Hu et al., 2022) inserts trainable bottleneck layers parallel to the frozen linear weight. 6) HQSAM (Ke et al., 2024), which introduces a learnable high-quality output token and enhances mask details by fusing mask decoder features with both early and final ViT features. 7) SU-SAM (Song et al., 2024) presents a simple framework that efficiently fine-tunes the SAM using Adapter or LoRA. 8) ConvLoRA-SAM (Zhong et al., 2024) injects image-related inductive biases into the image encoder of SAM by integrating ultra-lightweight convolutional parameters into LoRA.

Distillation Baselines. In this study, we compare our method with the following baselines: 1) Logit-based distillation (Zhu et al., 2018). 2) single-layer paired feature distillation (i.e., PKD (Cao et al., 2022), PKT (Passalis et al., 2020)), which uses one-stage feature to distill knowledge, with MobileSAM (Zhang et al., 2023a) belonging to this category. 3) multiple-layers paired feature distillation (i.e., VID (Ahn et al., 2019), IBD (Kuang et al., 2023)), which utilizes multi-stage information to transfer knowledge, with each layer aligned separately. Similarly, TinySAM (Shu

Table 1: **Comparison of PEFT methods for SAM across various downstream segmentation tasks.** All results are based on the ViT-B backbone. "SAM": without adaptation. "decoder-only": directly fine-tuning the mask decoder of SAM.

METHOD	NATURAL IMAGES			MEDICAL				AGRICULTURE		REMOTE SENSING	
	CAMO			ISIC 2017		Kvasir		Leaf		Road	
	S_α ↑	E_ϕ ↑	F_β^ω ↑	Jac ↑	Dice ↑	S_α ↑	E_ϕ ↑	IoU ↑	Dice ↑	IoU ↑	Dice ↑
SAM	79.7± 0.02	88.8± 0.09	79.6± 0.01	61.0± 0.12	71.7± 0.14	71.4± 0.16	77.9± 0.17	37.6± 0.11	47.0± 0.16	7.2± 0.24	12.9± 0.29
decoder-only	84.9± 0.38	92.7± 0.34	81.8± 0.33	85.9± 0.34	92.2± 0.20	90.9± 0.05	95.2± 0.18	55.6± 1.12	68.8± 1.17	47.6± 0.47	64.1± 0.47
BitFit	87.5± 0.13	94.5± 0.08	85.3± 0.48	87.7± 0.14	93.2± 0.08	92.5± 0.12	96.3± 0.20	69.2± 0.67	80.3± 0.68	58.1± 0.06	73.1± 0.06
AdaptFormer	87.9± 0.10	94.8± 0.21	86.2± 0.19	87.6± 0.24	93.2± 0.15	93.3± 0.68	97.0± 0.81	75.0± 0.11	84.8± 0.08	61.1± 0.15	75.5± 0.12
LoRA	87.7± 0.59	94.6± 0.50	85.1± 0.64	87.8± 0.24	93.3± 0.13	93.0± 0.14	96.6± 0.11	71.4± 0.54	82.1± 0.62	59.0± 0.19	74.0± 0.17
Adapter	88.2± 0.44	94.8± 0.34	86.7± 0.92	87.7± 0.23	93.2± 0.16	93.4± 0.12	97.1± 0.15	74.4± 0.16	84.3± 0.28	60.5± 0.10	75.1± 0.08
HQ-SAM	85.1± 0.10	92.6± 0.10	81.0± 0.61	86.3± 0.32	92.4± 0.19	91.1± 0.50	95.5± 0.57	66.2± 0.44	77.8± 0.43	54.9± 0.16	70.6± 0.13
SU-SAM	88.3± 0.21	95.0± 0.22	86.2± 0.59	87.8± 0.18	93.2± 0.09	93.8± 0.02	97.5± 0.06	74.7± 0.53	84.5± 0.56	60.2± 0.26	74.8± 0.22
ConvLoRA-SAM	87.5± 0.39	94.5± 0.17	85.4± 0.41	87.7± 0.22	93.2± 0.11	92.9± 0.13	96.6± 0.28	71.4± 0.44	82.2± 0.37	59.6± 0.22	74.4± 0.20
LoRA+Ours	**88.3± 0.05**	**95.2± 0.00**	**85.8± 0.59**	**88.1± 0.08**	**93.5± 0.05**	**93.4± 0.11**	**96.8± 0.09**	**72.2± 0.06**	**82.8± 0.04**	**59.9± 0.20**	**74.6± 0.17**
Adapter+Ours	**88.6± 0.09**	**95.1± 0.05**	**87.1± 0.37**	**88.0± 0.05**	**93.4± 0.00**	**94.4± 0.12**	**97.9± 0.09**	**75.6± 0.27**	**85.2± 0.23**	**61.4± 0.30**	**75.8± 0.27**

Table 2: **Comparison of distillation methods for SAM fine-tuning across various domains.** "Teacher": SAM without adaptation. "Student": fine-tune SAM's image encoder by adding adapters, while fully training the decoder directly. All compared methods utilize student models with the same adapter-based structure.

METHOD	NATURAL IMAGES			MEDICAL				AGRICULTURE		REMOTE SENSING	
	CAMO			ISIC 2017		Kvasir		Leaf		Road	
	S_α ↑	E_ϕ ↑	F_β^ω ↑	Jac ↑	Dice ↑	S_α ↑	E_ϕ ↑	IoU ↑	Dice ↑	IoU ↑	Dice ↑
Teacher	79.7± 0.02	88.8± 0.09	79.6± 0.01	61.0± 0.12	71.7± 0.14	83.0± 0.10	88.8± 0.29	37.6± 0.11	47.0± 0.16	7.2± 0.24	12.9± 0.29
Student	88.2± 0.44	94.8± 0.34	86.7± 0.92	87.7± 0.23	93.2± 0.16	93.4± 0.12	97.1± 0.15	74.4± 0.16	84.3± 0.28	60.5± 0.10	75.1± 0.08
Logit	88.4± 0.08	94.9± 0.05	87.1± 0.22	87.2± 0.43	92.9± 0.29	93.2± 0.19	96.5± 0.19	73.0± 0.35	83.3± 0.29	50.9± 0.08	67.2± 0.06
PKD	87.0± 0.43	94.1± 0.23	84.3± 0.97	86.5± 0.26	92.5± 0.17	92.2± 0.25	96.0± 0.17	70.2± 1.15	81.1± 1.08	56.9± 0.61	72.2± 0.56
PKT	87.8± 0.40	94.5± 0.35	86.2± 0.46	87.4± 0.12	93.0± 0.07	93.7± 0.41	97.3± 0.53	74.2± 0.51	84.2± 0.52	60.7± 0.20	75.2± 0.16
IBD	85.2± 0.47	92.6± 0.35	82.4± 0.31	85.1± 0.74	91.7± 0.45	91.5± 0.14	95.3± 0.05	72.2± 0.12	82.7± 0.07	44.9± 0.18	61.5± 0.18
VID	87.9± 0.22	94.8± 0.34	86.3± 0.32	87.6± 0.44	93.1± 0.29	93.7± 0.16	97.4± 0.07	75.1± 0.08	84.9± 0.17	60.7± 0.19	75.4± 0.19
SemCKD	86.2± 0.16	93.5± 0.21	82.8± 1.54	85.4± 0.27	91.8± 0.19	92.4± 0.07	96.2± 0.03	72.0± 0.04	82.8± 0.10	53.5± 0.17	69.4± 0.17
ReviewKD	86.7± 0.07	94.0± 0.09	84.6± 0.63	85.5± 0.26	91.9± 0.15	92.4± 0.33	96.4± 0.26	72.6± 0.64	83.1± 0.47	57.3± 0.11	72.6± 0.11
TinySAM	83.7± 0.39	91.6± 0.31	81.1± 0.35	79.4± 1.12	87.8± 0.84	88.5± 0.31	93.5± 0.24	48.6± 1.14	61.0± 0.95	25.7± 1.19	39.6± 1.71
MobileSAM	87.1± 0.36	94.1± 0.27	85.1± 0.09	86.7± 0.13	92.6± 0.09	92.5± 0.12	96.3± 0.14	71.9± 0.30	82.6± 0.39	59.2± 0.09	74.1± 0.08
InfoSAM(Ours)	**88.6± 0.09**	**95.1± 0.05**	**87.1± 0.37**	**88.0± 0.05**	**93.4± 0.00**	**94.4± 0.12**	**97.9± 0.09**	**75.6± 0.27**	**85.2± 0.23**	**61.4± 0.30**	**75.8± 0.27**

et al., 2025) employs full-stage distillation. 4) cross-layer feature distillation (i.e., SemCKD (Chen et al., 2021a), ReviewKD (Chen et al., 2021b)), which utilizes knowledge from multiple layers of the teacher model to supervise the student, by leveraging diverse information extracted from these layers. Currently, no work in SAM has explored cross-layer fusion distillation for PEFT. InfoSAM is the first to address this.

We report the main experimental results on representative datasets from different domains. Additional experimental results are provided in Appendix D, and visualization results are available in Appendix G. All experiments are conducted three times to mitigate randomness, with both average values and standard errors reported.

5.1. Segment Anything Across Diverse Domains

We compare InfoSAM with two categories of methods: PEFT methods and distillation methods. The results are presented in Table 1 and Table 2, respectively. In Table 1, all PEFT methods outperform both the zero-shot performance and decoder-only fine-tuning, highlighting the importance of unified fine-tuning for SAM. Additionally, InfoSAM outperforms other PEFT techniques across various datasets from different domains. Compared to other PEFT methods, InfoSAM preserves the pre-trained, domain-invariant knowledge through information-based distillation, which proves effective in enhancing segmentation performance.

In Table 2, it is noteworthy that most distillation methods are detrimental during PEFT, leading to worse performance compared to fine-tuning without distillation. Specifically, TinySAM employs full-stage distillation, requiring the stu-

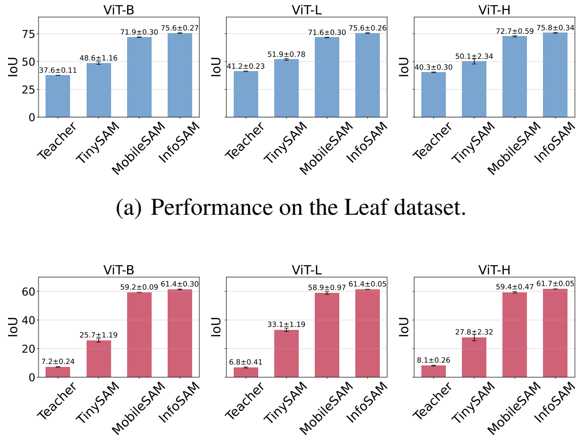
Table 3: **Comparison of PEFT methods and distillation methods with SAM2 across various domains.** All results are based on the Hiera-B+ backbone. "SAM2": without adaptation.

(a) PEFT Methods Comparison

METHOD	MEDICAL	AGRICULTURE	REMOTE SENSING
	S_α (Kvasir)	IoU (Leaf)	IoU (Road)
SAM2	87.1$\pm$ 0.12	42.7$\pm$ 0.32	6.9$\pm$ 0.13
decoder-only	93.2$\pm$ 0.07	71.8$\pm$ 0.58	48.5$\pm$ 0.47
BitFit	93.8$\pm$ 0.09	75.4$\pm$ 0.29	59.2$\pm$ 0.26
AdaptFormer	93.7$\pm$ 0.19	73.6$\pm$ 1.10	59.9$\pm$ 0.35
LoRA	93.7$\pm$ 0.10	75.9$\pm$ 0.40	60.8$\pm$ 0.32
Adapter	94.4$\pm$ 0.06	76.8$\pm$ 0.56	60.9$\pm$ 0.14
LoRA+Ours	**94.0**$\pm$ **0.09**	**76.1**$\pm$ **0.38**	**60.9**$\pm$ **0.05**
Adapter+Ours	**94.5**$\pm$ **0.17**	**77.3**$\pm$ **0.14**	**61.3**$\pm$ **0.05**

(b) Distillation Methods Comparison

METHOD	MEDICAL	AGRICULTURE	REMOTE SENSING
	S_α (Kvasir)	IoU (Leaf)	IoU (Road)
Teacher	87.1$\pm$ 0.12	42.7$\pm$ 0.32	6.9$\pm$ 0.13
Student	94.4$\pm$ 0.06	76.8$\pm$ 0.56	60.9$\pm$ 0.14
PKT	94.0$\pm$ 0.25	74.8$\pm$ 0.14	57.3$\pm$ 0.07
VID	94.1$\pm$ 0.47	77.2$\pm$ 0.37	61.1$\pm$ 0.38
ReviewKD	93.4$\pm$ 0.10	72.7$\pm$ 0.37	55.9$\pm$ 0.50
TinySAM	89.4$\pm$ 0.10	45.2$\pm$ 0.76	23.9$\pm$ 2.61
MobileSAM	93.3$\pm$ 0.15	74.1$\pm$ 0.35	52.3$\pm$ 0.46
InfoSAM2(Ours)	**94.5**$\pm$ **0.17**	**77.3**$\pm$ **0.14**	**61.3**$\pm$ **0.05**



(a) Performance on the Leaf dataset.

(b) Performance on the Road dataset.

Figure 4: **Performance of InfoSAM with larger teacher models** (i.e., ViT-L, ViT-H), while the student remains based on a ViT-B backbone. Each subfigure shows IoU metrics for different distillation methods on a specific dataset.

dent's features to fully mimic the teacher at every stage. However, this becomes catastrophic when the teacher performs poorly (e.g., achieving only 7.2% IoU on the Road dataset). In contrast, InfoSAM further enhances PEFT performance in this challenging scenario, likely due to the relation compression process during distillation, which ensures the student model learns only the essential information from the teacher model.

5.2. Extended Experiment with SAM2

Note that our method is orthogonal to model development, making it easily transferable to SAM2 backbones. As shown in Table 3, InfoSAM demonstrates consistent effectiveness with SAM2. This transferability is attributed to InfoSAM's foundation in information-theoretic derivation, which is structure-independent.

Table 4: **Ablation study results of two losses**: relation compression loss L_r and distillation loss L_d

L_r	L_d	MEDICAL	AGRICULTURE	REMOTE SENSING
		S_α (Kvasir)	IoU (Leaf)	IoU (Road)
		93.4	74.4	60.5
	✓	93.6 (+0.2)	75.2 (+0.8)	61.0 (+0.5)
✓	✓	94.4 (+1.0)	75.6 (+1.2)	61.4 (+0.9)

Table 5: **Effects of the Relation Module (RM).** Enhancing TinySAM and MobileSAM using RM.

MODEL	METHOD	AGRICULTURE	REMOTE SENSING
		IoU (Leaf)	IoU (Road)
TinySAM	w/o RM	48.6$\pm$ 1.14	28.7$\pm$ 1.69
	w RM	**50.3**$\pm$ **0.76**	**33.9**$\pm$ **0.32**
MobileSAM	w/o RM	71.9$\pm$ 0.30	59.2$\pm$ 0.09
	w RM	**73.8**$\pm$ **0.22**	**61.3**$\pm$ **0.35**

5.3. Distillation Across Models of Different Sizes

We also verify the effectiveness of InfoSAM when scaling the teacher model to larger sizes. As shown in Fig. 4, InfoSAM shows comparable improvements to distillation methods specifically designed for SAM compression. It indicates that InfoSAM is better suited for PFET, even in traditional large-to-small knowledge distillation scenarios.

5.4. Ablation Study

Ablation of Main Components. We conduct an ablation study to evaluate the impact of InfoSAM's components: relation compression loss $\mathcal{L}_r$ and relation distillation loss $\mathcal{L}_d$ on three datasets: Kvasir, Leaf, and Road. Using SAM with an adapter as the baseline (Table 4, row 1), row 2 introduces a fixed-dot relation module with mutual information-based distillation loss. Results show that incorporating simple relations improves performance, e.g., 0.8% on Leaf, and

Table 6: **Transferability of the Relation Module (RM).** InfoSAM-T represents the model trained with a pre-trained relation module from a different domain.

METHOD	FROZEN	MEDICAL	AGRICULTURE
	RM FROM	S_α (Kvasir)	IoU (Leaf)
InfoSAM	-	94.4$\pm$ 0.12	-
InfoSAM-T	Leaf	93.7$\pm$ 0.24	-
InfoSAM	-	-	75.6$\pm$ 0.30
InfoSAM-T	Kvasir	-	75.4$\pm$ 0.45

mutual information effectively transfers relational features. Combining both losses yields further gains, e.g., 1.4% on Leaf.

Effects of Relation Module (RM). To investigate the impact of the proposed relation module, we first conduct experiments to verify its effectiveness in enhancing various distillation methods. Specifically, we evaluate and compare the performance of two compact models, MobileSAM and TinySAM, with and without integrating the relation module, as shown in Table 5. We can observe a significant improvement with RM, e.g., 1.9% IoU on the Leaf dataset. These results suggest that the relation module can effectively capture and leverage high-level semantic information, thereby providing complementary benefits to existing distillation strategies during the fine-tuning stage.

Furthermore, Table 6 illustrates the effectiveness of domain-invariance dependencies in the relation module, we directly apply the module trained on one specific domain to another domain with entirely different knowledge. The results show that it still maintains satisfying results. Moreover, we conduct experiments to explore the nature of domain-invariant information. We use the Boundary F1 Score (Zhang et al., 2023b) to evaluate such universal patterns. The results show that our methods employing the relation module perform better in preserving structural edge features. More results and analysis are available in Appendix F.

6. Conclusion

We introduce InfoSAM, an information-theoretic tuning framework designed for SAM adaptation. From an information bottleneck perspective, we extract domain-invariant knowledge from the pre-trained SAM and inject it into the fine-tuned SAM to enhance adaptation efficiency. Specifically, we first propose an attention-based module to capture structural relations while minimizing mutual information to retain the most essential ones. These relations are then transferred by maximizing their mutual information. Extensive evaluations across eight segmentation datasets spanning diverse domains and tasks strongly validate the effectiveness of InfoSAM.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 62192781, 62172326, and 62137002; in part by the Project of China Knowledge Centre for Engineering Science and Technology; and supported by the Fundamental Research Funds for the Central Universities No. xxj032025002.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Ahn, S., Hu, S. X., Damianou, A., Lawrence, N. D., and Dai, Z. Variational information distillation for knowledge transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9163–9171, 2019.

Ben Zaken, E., Goldberg, Y., and Ravfogel, S. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 1–9, 2022.

Bernal, J., Sánchez, F. J., Fernández-Esparrach, G., Gil, D., Rodríguez, C., and Vilariño, F. Wm-dova maps for accurate polyp highlighting in colonoscopy: Validation vs. saliency maps from physicians. *Computerized medical imaging and graphics*, 43:99–111, 2015.

Bhatia, R. Infinitely divisible matrices. *The American Mathematical Monthly*, 113(3):221–235, 2006.

Boyd, S. Convex optimization. *Cambridge UP*, 2004.

Cao, W., Zhang, Y., Gao, J., Cheng, A., Cheng, K., and Cheng, J. Pkd: General distillation framework for object detectors via pearson correlation coefficient. *Advances in Neural Information Processing Systems*, 35:15394–15406, 2022.

Chang, W.-L., Wang, H.-P., Peng, W.-H., and Chiu, W.-C. All about structure: Adapting structural information across domains for boosting semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1900–1909, 2019.

Chen, D., Mei, J.-P., Zhang, Y., Wang, C., Wang, Z., Feng, Y., and Chen, C. Cross-layer distillation with semantic calibration. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 7028–7036, 2021a.

Chen, P., Liu, S., Zhao, H., and Jia, J. Distilling knowledge via knowledge review. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5008–5017, 2021b.

Chen, S., Ge, C., Tong, Z., Wang, J., Song, Y., Wang, J., and Luo, P. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022.

Chen, T., Zhu, L., Deng, C., Cao, R., Wang, Y., Zhang, S., Li, Z., Sun, L., Zang, Y., and Mao, P. Sam-adapter: Adapting segment anything in underperformed scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3367–3375, 2023.

Codella, N. C., Gutman, D., Celebi, M. E., Helba, B., Marchetti, M. A., Dusza, S. W., Kalloo, A., Liopyris, K., Mishra, N., Kittler, H., et al. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic). In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pp. 168–172. IEEE, 2018.

Dong, G., Wang, Z., Chen, Y., Sun, Y., Song, H., Liu, L., and Cui, H. An efficient segment anything model for the segmentation of medical images. *Scientific Reports*, 14 (1):19425, 2024.

Dong, Y., Gong, T., Yu, S., and Li, C. Optimal randomized approximations for matrix-based rényi's entropy. *IEEE Transactions on Information Theory*, 69(7):4218–4234, 2023.

Fan, D.-P., Ji, G.-P., Sun, G., Cheng, M.-M., Shen, J., and Shao, L. Camouflaged object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2777–2787, 2020a.

Fan, D.-P., Ji, G.-P., Zhou, T., Chen, G., Fu, H., Shen, J., and Shao, L. Pranet: Parallel reverse attention network for polyp segmentation. In *International conference on medical image computing and computer-assisted intervention*, pp. 263–273. Springer, 2020b.

Gong, T., Dong, Y., Yu, S., and Dong, B. Computationally efficient approximations for matrix-based rényi's entropy. *IEEE Transactions on Signal Processing*, 70:6170–6184, 2022.

Gou, J., Yu, B., Maybank, S. J., and Tao, D. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A., and Darrell, T. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pp. 1989–1998. Pmlr, 2018.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. Lora: Low-rank adaptation of large language models. *Proceedings of the International Conference on Learning Representations*, pp. 1–20, 2022.

Huang, J., Guan, D., Xiao, A., and Lu, S. Multi-level adversarial network for domain adaptive semantic segmentation. *Pattern Recognition*, 123:108384, 2022.

Jha, D., Smedsrud, P. H., Riegler, M. A., Halvorsen, P., De Lange, T., Johansen, D., and Johansen, H. D. Kvasir-seg: A segmented polyp dataset. In *MultiMedia modeling: 26th international conference, MMM 2020, Daejeon, South Korea, January 5–8, 2020, proceedings, part II 26*, pp. 451–462. Springer, 2020.

Ji, W., Li, J., Bi, Q., Liu, T., Li, W., and Cheng, L. Segment anything is not always perfect: An investigation of sam on different real-world applications, 2024.

Ke, L., Ye, M., Danelljan, M., Tai, Y.-W., Tang, C.-K., Yu, F., et al. Segment anything in high quality. *Advances in Neural Information Processing Systems*, 36, 2024.

Kerr, A., Campbell, D., and Richards, M. Qr decomposition on gpus. In *Proceedings of 2nd Workshop on General Purpose Processing on Graphics Processing Units*, pp. 71–78, 2009.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.

Kuang, H., Liu, H., Wu, Y., Satoh, S., and Ji, R. Improving adversarial robustness via information bottleneck distillation. *Advances in Neural Information Processing Systems*, 36:10796–10813, 2023.

Le, T.-N., Nguyen, T. V., Nie, Z., Tran, M.-T., and Sugimoto, A. Anabranch network for camouflaged object segmentation. *Computer vision and image understanding*, 184:45–56, 2019.

Li, B., Shen, Y., Wang, Y., Zhu, W., Li, D., Keutzer, K., and Zhao, H. Invariant information bottleneck for domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7399–7407, 2022a.

Li, J., Zi, S., Song, R., Li, Y., Hu, Y., and Du, Q. A stepwise domain adaptive segmentation network with covariate shift alleviation for remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2022b.

Ma, X., Zhang, X., Ding, X., Pun, M.-O., and Ma, S. Decomposition-based unsupervised domain adaptation for remote sensing image semantic segmentation. *IEEE Transactions on Geoscience and Remote Sensing*, 2024.

Miles, R., Yucel, M. K., Manganelli, B., and Saà-Garriga, A. Mobilevos: Real-time video object segmentation contrastive learning meets knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10480–10490, 2023.

Mnih, V. *Machine learning for aerial image labeling*. University of Toronto (Canada), 2013.

Passalis, N., Tzelepi, M., and Tefas, A. Probabilistic knowledge transfer for lightweight deep representation learning. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):2030–2039, 2020.

Peng, X., Chen, R., Qiao, F., Kong, L., Liu, Y., Sun, Y., Wang, T., Zhu, X., and Ma, Y. Learning to adapt sam for segmenting cross-domain point clouds. In *European Conference on Computer Vision*, pp. 54–71. Springer, 2024a.

Peng, Z., Xu, Z., Zeng, Z., Xie, L., Tian, Q., and Shen, W. 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*, pp. 3743–3752, 2024b.

Rath, S. R. Leaf disease segmentation dataset, 2023. URL https://www.kaggle.com/datasets/sovitrath/leaf-disease-segmentation-%with-trainvalid-split. Accessed: January 18, 2025.

Ravi, N., Gabeur, V., Hu, Y.-T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., et al. Sam 2: Segment anything in images and videos. *Proceedings of the International Conference on Learning Representations*, pp. 1–20, 2022.

Shen, Y., Li, J., Shao, X., Inigo Romillo, B., Jindal, A., Dreizin, D., and Unberath, M. Fastsam3d: An efficient segment anything model for 3d volumetric medical images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 542–552. Springer, 2024.

Shu, H., Li, W., Tang, Y., Zhang, Y., Chen, Y., Li, H., Wang, Y., and Chen, X. Tinysam: Pushing the envelope for efficient segment anything model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 20470–20478, 2025.

Skurowski, P., Abdulameer, H., Błaszczyk, J., Depta, T., Kornacki, A., and Kozieł, P. Animal camouflage analysis: Chameleon database. *Unpublished manuscript*, 2(6):7, 2018.

Song, Y., Zhou, Q., Lu, X., Shao, Z., and Ma, L. Simada: A simple unified framework for adapting segment anything model in underperformed scenes. *arXiv preprint arXiv:2401.17803*, 2024.

Tishby, N. and Zaslavsky, N. Deep learning and the information bottleneck principle. In *2015 ieee information theory workshop (itw)*, pp. 1–5. IEEE, 2015.

Wang, Z., Ji, K., Wang, D., and Cheng, F. Samcl: Empowering sam to continually learn from dynamic domains. *arXiv preprint arXiv:2412.05012*, 2024.

Wu, J., Wang, Z., Hong, M., Ji, W., Fu, H., Xu, Y., Xu, M., and Jin, Y. Medical sam adapter: Adapting segment anything model for medical image segmentation. *Medical image analysis*, 102:103547, 2025.

Xiao, A., Xuan, W., Qi, H., Xing, Y., Ren, R., Zhang, X., Shao, L., and Lu, S. Cat-sam: Conditional tuning for few-shot adaptation of segment anything model. In *European Conference on Computer Vision*, pp. 189–206. Springer, 2025.

Xu, Q., Yao, L., Jiang, Z., Jiang, G., Chu, W., Han, W., Zhang, W., Wang, C., and Tai, Y. Dirl: Domain-invariant representation learning for generalizable semantic segmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 2884–2892, 2022.

Yu, S., Giraldo, L. G. S., Jenssen, R., and Principe, J. C. Multivariate extension of matrix-based rényi's α-order entropy functional. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(11):2960–2966, 2019.

Zhang, C., Han, D., Qiao, Y., Kim, J. U., Bae, S.-H., Lee, S., and Hong, C. S. Faster segment anything: Towards lightweight sam for mobile applications. *arXiv preprint arXiv:2306.14289*, 2023a.

Zhang, L., Liang, Y., Zhang, R., Javadi, A., and Xie, P. Blo-sam: Bi-level optimization based finetuning of the segment anything model for overfitting-preventing semantic segmentation. In *Forty-first International Conference on Machine Learning*, 2024a.

Zhang, Q., Liu, X., Li, W., Chen, H., Liu, J., Hu, J., Xiong, Z., Yuan, C., and Wang, Y. Distilling semantic priors from sam to efficient image restoration models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25409–25419, 2024b.

Zhang, Y., Tian, S., Liao, M., Hua, G., Zou, W., and Xu, C. Learning shape-invariant representation for generalizable semantic segmentation. *IEEE Transactions on Image Processing*, 32:5031–5045, 2023b.

Zhong, Z., Tang, Z., He, T., Fang, H., and Yuan, C. Convolution meets loRA: Parameter efficient finetuning for segment anything model. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ezscMer8L0.

Zhu, X., Gong, S., et al. Knowledge distillation by on-the-fly native ensemble. *Advances in neural information processing systems*, 31, 2018.

A. Derivation of information-theoretic Losses

A.1. Multivariate Entropy and Mutual Information

Following Definition 2 and Eq.(2), we consider the matrix-based Rényi's α-order joint entropy for multiple variables (Yu et al., 2019).

Definition 3. Given a collection of n samples $\{s_i = (x_1^i, x_2^i, \cdots, x_k^i)\}_{i=1}^n$, where the superscript i denotes the sample index, each sample contains k ($k \geq 2$) measurements $x_1 \in \mathcal{X}_1, x_2 \in \mathcal{X}_2, \cdots, x_k \in \mathcal{X}_k$ obtained from the same realization, and the positive definite kernels $\kappa_1 : \mathcal{X}_1 \times \mathcal{X}_1 \mapsto \mathbb{R}$, $\kappa_2 : \mathcal{X}_2 \times \mathcal{X}_2 \mapsto \mathbb{R}$, $\cdots$, $\kappa_k : \mathcal{X}_k \times \mathcal{X}_k \mapsto \mathbb{R}$, a matrix-based analogue to Rényi's α-order joint-entropy among k variables can be defined as:

$$S_\alpha(\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_k) = S_\alpha\left(\frac{\mathbf{A}_1 \circ \mathbf{A}_2 \circ \cdots \circ \mathbf{A}_k}{\operatorname{tr}(\mathbf{A}_1 \circ \mathbf{A}_2 \circ \cdots \circ \mathbf{A}_k)}\right) \tag{16}$$

where $(\mathbf{A}_1)_{ij} = \kappa_1(x_1^i, x_1^j)$, $(\mathbf{A}_2)_{ij} = \kappa_2(x_2^i, x_2^j)$, $\cdots$, $(\mathbf{A}_k)_{ij} = \kappa_k(x_k^i, x_k^j)$, and $\circ$ denotes the Hadamard product.

Following Definition 2 and Definition 3, mutual information can be extended to measure interactions among multiple variables by grouping them into sets and treating each set as a single variable. It can be defined as:

$$\mathbf{I}_\alpha\left(\mathbf{A}_1, \cdots, \mathbf{A}_k; \mathbf{B}\right) = \mathbf{S}_\alpha\left(\mathbf{A}_1, \cdots, \mathbf{A}_k\right) + \mathbf{S}_\alpha(\mathbf{B}) - \mathbf{S}_\alpha\left(\mathbf{A}_1, \cdots, \mathbf{A}_k, \mathbf{B}\right) \tag{17}$$

where $\mathbf{A}_1, \cdots, \mathbf{A}_k$ and $\mathbf{B}$ denote the normalized Gram matrices.

A.2. Derivation of Relation Compression Loss $\mathcal{L}_r$ and Relation Distillation Loss $\mathcal{L}_d$

Relation Compression Loss $\mathcal{L}_r$. In this paper, according to Eq.(16) and Eq.(17), the mutual information of the compression process ($k = 2$) can be defined as:

$$\mathbf{I}_\alpha\left(\mathbf{A}_1, \mathbf{A}_2; \mathbf{B}\right) = \mathbf{S}_\alpha\left(\mathbf{A}_1, \mathbf{A}_2\right) + \mathbf{S}_\alpha(\mathbf{B}) - \mathbf{S}_\alpha\left(\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}\right) \tag{18}$$

In the relation compression loss $\mathcal{L}_r$, the matrices $\mathbf{A}_1$, $\mathbf{A}_2$, and $\mathbf{B}$ are normalized Gram matrices constructed from the image embeddings z_i^T, the mask embeddings z_m^T, and the relation module outputs r^T, respectively. To maintain consistency with the previously defined relation compression loss (refer to Eq.(10)), we replace $\mathbf{A}_1$, $\mathbf{A}_2$, and $\mathbf{B}$ with G_i^T, G_m^T, and $G_r^T \in \mathbb{R}^{N \times N}$, respectively, where all Gram matrices are computed using the polynomial kernel function defined as:

$$\kappa(x, y) = x^\top y, \tag{19}$$

where $\mathbf{x}, \mathbf{y} \in \mathbb{R}^N$ are input vectors. For instance, G_i^T is computed as the normalized Gram matrix:

$$G_i^T = \frac{\kappa(z_i^T, z_i^T)}{\operatorname{tr}(\kappa(z_i^T, z_i^T))}, \tag{20}$$

where $\operatorname{tr}(\cdot)$ denotes the trace of the matrix.

Furthermore, Rényi's α-order entropy is reformulated using its eigenvalue expansion (refer to Eq.(1)), leading to:

$$\begin{aligned}\mathcal{L}_r =& \mathbf{I}_\alpha(z_i^T, z_m^T; r^T) \\ =& \cancel{\mathbf{S}_\alpha(G_i^T, G_m^T)} + \mathbf{S}_\alpha(G_r^T) - \mathbf{S}_\alpha(G_i^T, G_m^T, G_r^T) \\ =& \frac{1}{1-\alpha}\log_2 \sum_{i=1}^{n} \lambda_i^\alpha\left(G_r^T\right) - \frac{1}{1-\alpha}\log_2 \sum_{i=1}^{n} \lambda_i^\alpha\left(G_{imr}^T\right)\end{aligned} \tag{21}$$

where G_{imr}^T is also normalized to have a trace of one. The teacher entropy term is excluded from this loss because the teacher's weights remain fixed throughout the training process. Substituting the marginal and joint entropy definitions from Definition 1 and Definition 3, $G_{imr}^S = G_i^T \circ G_m^T \circ G_r^T$. The $\circ$ is Hadamard product.

Since computing the eigenvalues of large matrices is typically computationally expensive during training (Kerr et al., 2009), we restrict the value of α to 2. This choice allows us to use the Frobenius norm as a proxy objective function. Notably, the

Frobenius norm has a connection with the eigenspectrum. Specifically, for a symmetric matrix $\mathbf{A}$, its Frobenius norm can be expressed as:

$$\|\mathbf{A}\|_F^2 = \operatorname{tr}(\mathbf{A}\mathbf{A}^H) = \sum_{i=1}^{n} \lambda_i(\mathbf{A}^2) = \sum_{i=1}^{n} \lambda_i^2(\mathbf{A}) \tag{22}$$

where $\operatorname{tr}(\cdot)$ denotes the trace operation. Since $\mathbf{A}$ is a symmetric matrix, $\sum_{i=1}^{n} \lambda_i(\mathbf{A}^2)$ is equivalent to $\sum_{i=1}^{n} \lambda_i^2(\mathbf{A})$ (Dong et al., 2023). Through this formulation, the Frobenius norm not only simplifies the computation but also retains an intrinsic connection to the matrix eigenspectrum, which is significant for both model training and theoretical analysis. Given G_r^T and G_{imr}^T is positive semi-definite, $\mathcal{L}_r$ can be reformulated as:

$$\begin{aligned}\mathcal{L}_r =& \frac{1}{1-2}\log_2 \sum_{i=1}^{n} \lambda_i^2\left(G_r^T\right) - \frac{1}{1-2}\log_2 \sum_{i=1}^{n} \lambda_i^2\left(G_{imr}^T\right) \\ =& -\log_2 \|G_r^T\|_F^2 + \log_2 \|G_{imr}^T\|_F^2\end{aligned} \tag{23}$$

The two terms in $\mathcal{L}_r$ serve distinct purposes. First, the term $-\log_2 \|G_r^T\|_F^2$ imposes spectral compression on the relation module. This ensures that the relation module focuses on the most discriminative features while suppressing irrelevant variations. Second, the term $\log_2 \|G_{imr}^T\|_F^2$ achieves joint entropy minimization across the image encoder, mask decoder, and relation module. It filters out spurious relationships induced by domain-specific noise while preserving domain-invariant interactions encoded in G_{imr}^T. This is critical for cross-domain adaptation, as it maintains consistency in feature interactions across different data distributions.

Relation Distillation Loss $\mathcal{L}_d$. Similar to $\mathcal{L}_r$, the goal of distillation is to maximize the mutual information of teacher-student relations. The mutual information of the distillation process can be defined as:

$$\begin{aligned}\mathbf{I}_\alpha\left(r^T; r^S\right) =& \mathbf{S}_\alpha\left(G_r^T\right) + \mathbf{S}_\alpha(G_r^S) - \mathbf{S}_\alpha\left(G_r^T, G_r^S\right) \\ =& \frac{1}{1-\alpha}\log_2 \sum_{i=1}^{n} \lambda_i^\alpha\left(G_r^T\right) + \frac{1}{1-\alpha}\log_2 \sum_{i=1}^{n} \lambda_i^\alpha\left(G_r^S\right) - \frac{1}{1-\alpha}\log_2 \sum_{i=1}^{n} \lambda_i^\alpha\left(G_r^{TS}\right)\end{aligned} \tag{24}$$

where G_r^S, G_r^T and $G_r^{TS} = G_r^T \circ G_r^S$ is the normalized Gram matrices. Then, applying Eq.(22) and set $\alpha = 2$, the distillation loss is:

$$\begin{aligned}\mathcal{L}_d =& -\mathbf{I}_\alpha\left(r^T; r^S\right) \\ =& -\frac{1}{1-\alpha}\log_2 \sum_{i=1}^{n} \lambda_i^\alpha\left(G_r^T\right) - \frac{1}{1-\alpha}\log_2 \sum_{i=1}^{n} \lambda_i^\alpha\left(G_r^S\right) + \frac{1}{1-\alpha}\log_2 \sum_{i=1}^{n} \lambda_i^\alpha\left(G_r^{TS}\right) \\ =& \log_2 \|G_r^T\|_F^2 + \log_2 \|G_r^S\|_F^2 - \log_2 \|G_r^{TS}\|_F^2\end{aligned} \tag{25}$$

Overall, the mutual information loss balances feature constraints by regularizing the student's relational feature complexity to prevent overfitting while aligning the student's and teacher's relational distributions for effective knowledge transfer. The three terms in the distillation loss serve distinct purposes: $\log_2 \|G_r^T\|_F^2$ captures the complexity of the teacher's relational features, guiding the teacher to learn rich and detailed representations rather than overly simplified ones; $\log_2 \|G_r^S\|_F^2$ regularizes the student's feature complexity, preventing overfitting by keeping the representations manageable; $-\log_2 \|G_r^{TS}\|_F^2$ Aligns the relational distributions between the teacher and student, ensuring the student learns the relationships between features as captured by the teacher.

A.3. Pseudocode of InfoSAM

Here, we summarize the core fine-tuning process for SAM using the proposed information-theoretic loss in Algorithm.1. Relationships between features across different network components are modeled and normalized to capture meaningful interactions. These interactions drive two key loss functions: a compression loss that strengthens the student network's ability to represent complex features and a distillation loss that aligns the student's relational understanding with the teacher's. This method emphasizes relational and structural learning to achieve better generalization and accuracy.

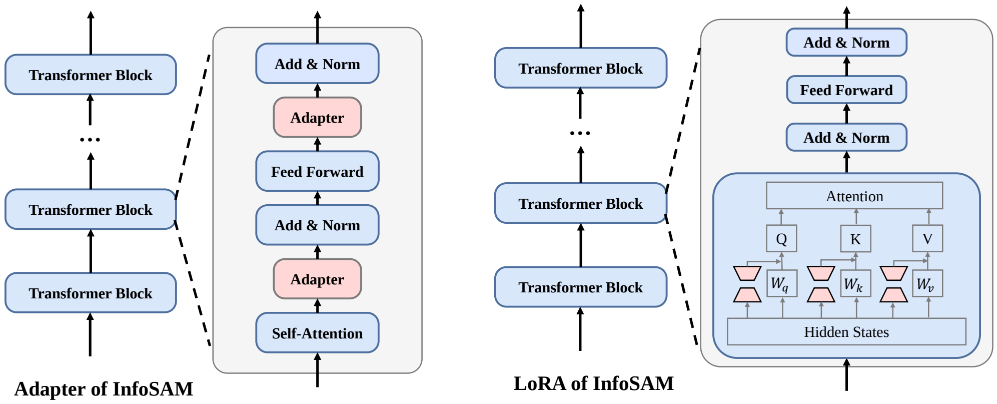
Algorithm 1 PyTorch-style pseudocode for InfoSAM

```
# F_t, F_s: Pre-trained SAM (teacher) and fine-tuned SAM (student)
# z_t_i, z_s_i: The output of the teacher and student image encoders
# z_t_m, z_s_m: The output tokens in the mask decoder of the teacher and student
# f_t, f_s: Teacher and student relation modules
# y_t, y_s: Teacher and student outputs
# y: Ground-truth labels
# Frob: Function for computing the square of the Frobenius norm

for x, y in loader:
    # Forward pass
    z_t_i, z_t_m, y_t = F_t(x)
    z_s_i, z_s_m, y_s = F_s(x)

    # Compute structure loss
    loss_ce = struct_loss(y_s,y)

    # Compute relations between image encoder and mask decoder
    f_s = f_t
    r_t = f_t(z_t_i, z_t_m)
    r_s = f_s(z_s_i, z_s_m)

    # Normalize the representations
    z_t_i_norm = F.normalize(z_t_i, p=2)
    z_t_m_norm = F.normalize(z_t_m, p=2)

    # Compute normalized Gram matrices for compression loss_r
    G_t_i = matmul(z_t_i_norm, z_t_i_norm.T)
    G_t_m = matmul(z_t_m_norm, z_t_m_norm.T)
    G_t_r = matmul(r_t, r_t.T)
    G_t_r_norm = G_t_r / trace(G_t_r)
    G_t_imr_norm = G_t_i * G_t_m * G_t_r / trace(G_t_i * G_t_m * G_t_r)

    # Compute normalized Gram matrices for distillation loss_d
    G_s_r = matmul(r_s, r_s.T)
    G_s_r_norm = G_s_r / trace(G_s_r)
    G_ts_r_norm = G_s_r * G_t_r / trace(G_s_r * G_t_r)

    # Compute relation compression loss_r and distillation loss_d
    loss_r = - log2(Frob(G_t_r_norm)) + log2(Frob(G_t_imr_norm))
    loss_d = log2(Frob(G_t_r_norm)) + log2(Frob(G_s_r_norm)) - log2(Frob(G_ts_r_norm))
    loss_info = lamda_1 * loss_r + lamda_2 * loss_d

    # The overall loss
    loss = loss_ce + loss_info

    # Optimization step
    loss.backward()
    optimizer.step()
```

B. Implementation Details

B.1. Architectures of Segment Anything Model (SAM)

SAM (Kirillov et al., 2023) consists of three key modules, i.e., image encoder, prompt encoder, and mask decoder. The image encoder is a heavy ViT-based network for image feature extraction. The prompt encoder is designed to capture positional information from geometric prompts (i.e., points, boxes, or masks) to generate prompt embeddings. The mask decoder, a two-way transformer module, combines image embeddings and prompt tokens to generate the final mask. The released model, trained on 11 million images and 1 billion high-quality masks, demonstrates impressive zero-shot capability in handling various conventional natural images. The latest SAM2 (Ravi et al., 2022) introduces a significant evolution over



(a) Adapter (b) LoRA

Figure 5: The architecture of InfoSAM with Adapter and LoRA

its predecessor by extending its capabilities to the domain of video segmentation. SAM2 replaces the backbone of SAM with Hiera backbones. And introduces an additional streaming memory component designed for processing video frames. However, recent studies(Wu et al., 2025; Ji et al., 2024) have revealed that SAM performs poorly in real-world segmentation tasks across domains such as medicine, agriculture, and remote sensing. Fine-tuning SAM for downstream tasks has been widely recommended.

B.2. Architectures of Adapter and LoRA

We implement InfoSAM with two of the most widely adopted PEFT methods: Adapter (Chen et al., 2022; Song et al., 2024) and LoRA (Hu et al., 2022). We fine-tune the image encoder of SAM and SAM2 by incorporating adapters or LoRA, while fully training the mask decoder. Notably, all training hyperparameters used for SAM and SAM2 remain the same.

The Adapter method introduces lightweight modules into the Transformer architecture by adding small projection layers between the original layers. Specifically, the adapter module consists of two linear transformations: a down-projection and an up-projection, with a non-linear activation layer (GELU in our implementation) applied between them. Let the input feature be $\mathbf{X} \in \mathbb{R}^{B \times L \times D}$, where B is the batch size, L is the sequence length, and D is the feature dimension. The adapter performs the following operations:

$$\mathbf{z} = \mathbf{X}\mathbf{W}_{\text{down}} \tag{26}$$

$$\mathbf{z}_{\text{act}} = \text{GELU}(\mathbf{z}) \tag{27}$$

$$\mathbf{h} = \mathbf{z}_{\text{act}}\mathbf{W}_{\text{up}} \tag{28}$$

where $\mathbf{W}_{\text{down}} \in \mathbb{R}^{D \times r}$ and $\mathbf{W}_{\text{up}} \in \mathbb{R}^{r \times D}$ are learnable weight matrices, and r denotes the bottleneck dimension, typically a small fraction of D. In our experiments, we set the bottleneck ratio r/D to 0.25. The final output of the adapter is computed as:

$$\mathbf{X}_{\text{out}} = \mathbf{X} + \mathbf{h} \tag{29}$$

with a residual connection preserving the original information. As illustrated in Fig. 5(a), two adapter modules are sequentially inserted between the attention and FFN layers.

LoRA (Low-Rank Adaptation) is another parameter-efficient method that integrates lightweight modules into the Transformer architecture. Similar to the adapter method, LoRA introduces trainable down-projection and up-projection matrices. However, as shown in Fig. 5(b), instead of being applied between layers, LoRA is added directly to the query ($\mathbf{q}$), key ($\mathbf{k}$), and value ($\mathbf{v}$) projections within the self-attention mechanism.

Given the input feature $\mathbf{X} \in \mathbb{R}^{B \times L \times D}$, the multi-head self-attention mechanism with H heads computes the query, key, and value as follows:

$$\mathbf{Q}_h = \mathbf{X}\mathbf{W}_{Q,h}, \quad \mathbf{K}_h = \mathbf{X}\mathbf{W}_{K,h}, \quad \mathbf{V}_h = \mathbf{X}\mathbf{W}_{V,h}, \quad h = 1, \dots, H \tag{30}$$

where $\mathbf{W}_{Q,h}, \mathbf{W}_{K,h}, \mathbf{W}_{V,h} \in \mathbb{R}^{D \times D_h}$, and $D_h = D/H$ is the dimensionality of each attention head.

LoRA introduces low-rank updates to each head's query, key, and value projections. Specifically:

$$\mathbf{Q}'_h = \mathbf{Q}_h + \Delta\mathbf{Q}_h, \quad \mathbf{K}'_h = \mathbf{K}_h + \Delta\mathbf{K}_h, \quad \mathbf{V}'_h = \mathbf{V}_h + \Delta\mathbf{V}_h \tag{31}$$

where the low-rank updates are defined as:

$$\Delta\mathbf{Q}_h = \mathbf{X}\mathbf{W}_{Q,h,\text{down}}\mathbf{W}_{Q,h,\text{up}}, \quad \Delta\mathbf{K}_h = \mathbf{X}\mathbf{W}_{K,h,\text{down}}\mathbf{W}_{K,h,\text{up}}, \quad \Delta\mathbf{V}_h = \mathbf{X}\mathbf{W}_{V,h,\text{down}}\mathbf{W}_{V,h,\text{up}} \tag{32}$$

Here: $\mathbf{W}_{Q,h,\text{down}}, \mathbf{W}_{K,h,\text{down}}, \mathbf{W}_{V,h,\text{down}} \in \mathbb{R}^{D \times r}$, $\mathbf{W}_{Q,h,\text{up}}, \mathbf{W}_{K,h,\text{up}}, \mathbf{W}_{V,h,\text{up}} \in \mathbb{R}^{r \times D_h}$. Note that the low-rank projection matrices are head-specific, and r is the bottleneck dimension shared across all heads, determining the rank of the adaptation. In our experiments, we set $r = 4$ for all the LoRA-based methods.

C. Tasks and Datasets

We benchmark InfoSAM across eight diverse benchmarks spanning four different domains, following Conv-LoRA (Zhong et al., 2024). The specific segmentation tasks and corresponding datasets utilized for benchmarking are elaborated below.

C.1. Natural Image

Camouflaged Object Segmentation. The task aims to detect objects hidden within complex or visually cluttered backgrounds, posing greater challenges compared to traditional object segmentation. We use three camouflaged object detection datasets: COD10K (Fan et al., 2020a), CHAMELEON (Skurowski et al., 2018), and CAMO (Le et al., 2019). COD10K contains 3,040 training and 2,026 testing samples, CHAMELEON provides 76 testing images, and CAMO includes 1,000 training and 250 testing images. The combined dataset of COD10K and CAMO training images is used, with 10% randomly split for validation, and testing is performed on all three datasets.

C.2. Medical Image

Polyp Segmentation. The task of polyp segmentation in gastrointestinal endoscopic images is critical for early colorectal cancer diagnosis and treatment planning, posing significant challenges due to the considerable variability in polyp shapes and sizes. We selecte two polyp segmentation datasets: Kvasir (Jha et al., 2020) and CVC-ClinicDB (also known as CVC-612) (Bernal et al., 2015). The Kvasir dataset consists of 1,000 images, while CVC-ClinicDB contains 612 publicly accessible images. (Fan et al., 2020b) splits the images into a 9:1 ratio for training and testing. Additionally, 20% of the training set is randomly selected as a validation set for use during training.

Skin Lesion Segmentation. The task involves identifying different types of skin lesions in medical images, playing a vital role in the early diagnosis and treatment of skin conditions, particularly skin cancer. However, it remains challenging due to ambiguous boundaries and color variations. We select the ISIC 2017 dataset (Codella et al., 2018) for skin lesion segmentation, which contains 2,000 images for training, 150 images for validation, and 600 images for testing.

C.3. Agriculture

Leaf Segmentation. The task focuses on identifying individual plant leaves in agricultural images, supporting automation in plant disease control and high-quality food production. We use the Leaf Disease Segmentation dataset (Rath, 2023), which includes 498 images for training and 90 for testing, with 20% of the training set randomly split for validation.

C.4. Remote Sensing

Road Segmentation. This task involves detecting road regions in images or video frames, which is essential for autonomous driving, traffic analysis, and urban planning. We use the Massachusetts Roads Dataset (Mnih, 2013), containing 1,107 images for training, 13 for validation, and 48 for testing. All the methods are trained for 20 epochs. During validation and testing, we use 5-point prompts instead of noisy ground-truth box prompts, as the complexity of road structures in remote sensing images is challenging to capture with box-based prompts.

D. Additional Experiment Results

Due to space constraints, experimental results that could not be included in the main body are provided here. These include additional results on more polyp segmentation and camouflaged object segmentation datasets using SAM, as well as the complete experimental results conducted on SAM2.

D.1. Additional Results with SAM

Polyp Segmentation: As shown in Table 7, InfoSAM exhibits superior performance in polyp segmentation on the Kvasir and CVC-612 datasets, outperforming both PEFT and distillation-based methods.

Table 7: Additional results of polyp segmentation.

METHOD	Kvasir S_α	Kvasir E_ϕ	Kvasir F_β^w	CVC-612 S_α	CVC-612 E_ϕ	CVC-612 F_β^w
SAM	83.0±0.10	88.8±0.29	79.7±0.06	87.8±0.19	94.5±0.19	85.8±0.34
decoder-only	90.9±0.05	95.2±0.18	89.1±0.48	92.9±0.14	97.4±0.26	89.5±0.63
BitFit	92.5±0.12	96.3±0.20	91.0±0.37	94.0±0.39	98.5±0.21	91.8±0.81
AdaptFormer	93.3±0.68	97.0±0.81	92.8±0.94	95.2±0.15	99.0±0.14	93.8±0.45
LoRA	93.0±0.14	96.6±0.11	91.8±0.57	94.3±0.31	98.7±0.17	92.3±0.62
Adapter	93.4±0.12	97.1±0.15	92.9±0.13	95.1±0.51	98.8±0.40	94.2±0.33
HQ-SAM	91.1±0.50	95.5±0.57	89.9±0.68	93.2±0.51	98.0±0.60	90.9±1.01
SU-SAM	93.8±0.02	97.5±0.06	**94.1±0.45**	95.3±0.42	98.9±0.39	94.4±0.65
ConvLoRA-SAM	92.9±0.13	96.6±0.28	92.2±0.46	94.4±0.05	98.8±0.14	92.6±0.63
Logit	93.2±0.19	96.5±0.19	92.9±0.11	95.1±0.30	99.0±0.34	93.8±0.59
PKD	92.2±0.25	96.0±0.17	91.9±0.17	94.2±0.25	98.5±0.13	92.3±0.40
PKT	93.7±0.41	97.3±0.53	93.8±0.43	95.3±0.19	99.1±0.18	94.3±0.36
IBD	91.5±0.14	95.3±0.05	89.9±0.74	93.1±0.13	97.4±0.15	90.0±0.52
VID	93.7±0.16	97.4±0.07	93.4±0.49	95.0±0.36	98.7±0.15	93.8±0.20
SemCKD	92.4±0.07	96.2±0.03	91.2±0.52	93.9±0.55	98.3±0.44	91.5±0.54
ReviewKD	92.4±0.33	96.4±0.26	91.6±0.65	94.0±0.59	98.4±0.49	92.2±1.40
MobileSAM	92.5±0.12	96.3±0.14	91.4±0.15	94.6±0.27	98.6±0.12	92.4±0.77
TinySAM	88.5±0.31	93.5±0.24	86.0±0.79	92.1±0.42	96.4±0.59	88.1±0.79
InfoSAM(Ours)	**94.4±0.12**	**97.9±0.09**	93.9±0.09	**95.3±0.09**	**98.9±0.09**	**94.3±0.15**

Camouflaged Object Segmentation: As illustrated in Table 8, InfoSAM achieves state-of-the-art performance in camouflaged object segmentation across the CHAMELEON, CAMO, and COD10K datasets, surpassing both PEFT and distillation-based approaches.

Table 8: Additional results of camouflaged object segmentation.

METHOD	CHAMELEON S_α	CHAMELEON E_ϕ	CHAMELEON F_β^w	CAMO S_α	CAMO E_ϕ	CAMO F_β^w	COD10K S_α	COD10K E_ϕ	COD10K F_β^w
SAM	80.4±0.14	88.9±0.10	74.5±0.13	79.7±0.02	88.8±0.09	79.6±0.01	83.5±0.02	92.5±0.03	79.2±0.01
decoder-only	87.0±0.20	93.5±0.41	80.0±0.74	84.9±0.38	92.7±0.34	81.8±0.33	87.1±0.10	94.4±0.11	79.9±0.13
BitFit	89.6±0.47	96.1±0.33	84.2±0.77	87.5±0.13	94.5±0.08	85.3±0.48	89.2±0.31	95.9±0.21	83.6±0.59
AdaptFormer	92.2±0.13	97.6±0.30	89.4±0.55	87.9±0.10	94.8±0.21	86.2±0.19	90.1±0.16	96.5±0.10	85.8±0.55
LoRA	90.4±0.48	96.5±0.25	85.5±0.59	87.7±0.59	94.6±0.50	85.1±0.64	89.8±0.17	96.3±0.06	84.9±0.40
Adapter	92.5±0.10	97.9±0.11	90.1±0.39	88.2±0.44	94.8±0.34	86.7±0.92	90.2±0.25	96.5±0.19	86.0±0.58
HQ-SAM	87.0±0.34	93.3±0.54	79.7±0.07	85.1±0.10	92.6±0.10	81.0±0.61	87.3±0.14	94.5±0.25	80.0±0.49
SU-SAM	92.3±0.28	97.9±0.27	**90.0±0.23**	88.3±0.21	95.0±0.22	86.2±0.59	90.2±0.15	96.5±0.09	86.0±0.45
ConvLoRA-SAM	90.6±0.34	96.6±0.19	86.0±0.71	87.5±0.39	94.5±0.17	85.4±0.41	89.8±0.23	96.2±0.21	84.9±0.61
Logit	91.9±0.47	97.4±0.46	89.2±0.55	88.4±0.08	94.9±0.05	87.1±0.22	90.4±0.12	96.6±0.10	86.1±0.31
PKD	90.7±0.40	96.2±0.64	87.5±1.10	87.0±0.43	94.1±0.23	84.3±0.97	89.7±0.17	96.3±0.16	85.4±0.43
PKT	92.2±0.40	97.7±0.50	90.0±0.73	87.8±0.40	94.5±0.35	86.2±0.46	90.3±0.16	96.6±0.10	86.2±0.50
IBD	86.6±0.29	93.0±0.60	79.5±0.77	85.2±0.47	92.6±0.35	82.4±0.31	87.9±0.13	94.8±0.18	81.4±0.37
VID	92.2±0.18	97.9±0.13	89.8±0.40	87.9±0.22	94.8±0.34	86.3±0.32	90.3±0.16	96.5±0.16	86.2±0.10
SemCKD	88.8±0.27	95.1±0.29	83.4±0.60	86.2±0.16	93.5±0.21	82.8±1.54	88.5±0.31	95.6±0.18	83.1±0.34
ReviewKD	90.3±0.30	96.5±0.19	85.6±0.02	86.7±0.07	94.0±0.09	84.6±0.63	89.4±0.23	96.0±0.10	84.3±0.42
MobileSAM	91.4±0.32	96.7±0.21	88.6±0.41	87.1±0.36	94.1±0.27	85.1±0.09	90.1±0.12	96.6±0.08	86.1±0.29
TinySAM	84.1±0.09	90.3±0.61	76.6±0.30	83.7±0.39	91.6±0.31	81.1±0.35	86.3±0.39	94.0±0.11	80.0±0.47
InfoSAM(Ours)	**92.6±0.14**	**98.0±0.25**	89.6±0.22	**88.5±0.05**	**95.1±0.05**	**87.1±0.09**	**90.5±0.05**	**96.7±0.09**	**86.2±0.22**

D.2. Additional Results with SAM2

The complete experimental results of InfoSAM with the SAM2 backbone are provided in Table 9 and Table 10. Maintaining the strong performance of SAM, InfoSAM achieves superior results with SAM2.

Table 9: Complete comparison results of PEFT methods with SAM2 across different domains.

Method	Medical			Agriculture		Remote Sensing	
	Kvasir			Leaf		Road	
	S_α	E_ϕ	F_β^ω	IoU	Dice	IoU	Dice
SAM2	87.1± 0.12	90.2± 0.06	85.2± 0.20	42.7± 0.32	53.3± 0.32	6.9± 0.13	12.4± 0.37
decoder-only	93.2± 0.07	96.6± 0.05	92.1± 0.41	71.8± 0.58	82.2± 0.58	48.5± 0.47	64.7± 0.49
BitFit	93.8± 0.09	97.0± 0.06	93.2± 0.17	75.4± 0.29	85.2± 0.26	59.2± 0.26	74.0± 0.25
AdaptFormer	93.7± 0.19	97.2± 0.42	93.3± 0.38	73.6± 1.10	83.7± 0.78	59.9± 0.35	74.6± 0.28
LoRA	93.7± 0.10	97.0± 0.07	93.2± 0.37	75.9± 0.40	85.5± 0.36	60.8± 0.32	75.3± 0.29
Adapter	94.4± 0.06	97.5± 0.09	93.8± 0.03	76.8± 0.56	86.2± 0.50	60.9± 0.14	75.4± 0.11
LoRA+Ours	**94.0**± **0.09**	**97.0**± **0.08**	**93.4**± **0.25**	**76.1**± **0.38**	**85.7**± **0.31**	**60.9**± **0.05**	**75.5**± **0.08**
Adapter+Ours	**94.5**± **0.17**	**97.4**± **0.16**	**94.0**± **0.16**	**77.3**± **0.14**	**86.6**± **0.08**	**61.3**± **0.05**	**75.8**± **0.05**

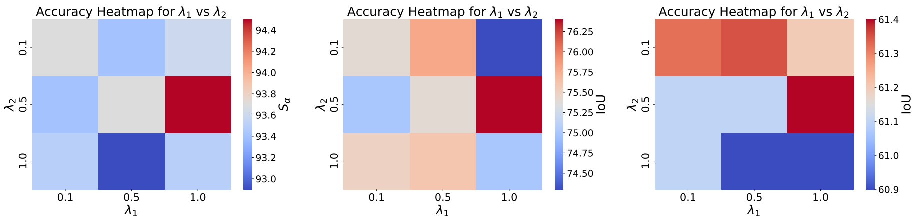
Table 10: Complete comparison results of distillation methods with SAM2 across various domains.

Method	Medical			Agriculture		Remote Sensing	
	Kvasir			Leaf		Road	
	S_α	E_ϕ	F_β^w	IoU	Dice	IoU	Dice
Teacher	87.1± 0.12	90.2± 0.06	85.2± 0.20	42.7± 0.32	53.3± 0.32	6.9± 0.13	12.4± 0.37
Student	94.4± 0.06	97.5± 0.09	93.8± 0.03	76.8± 0.56	86.2± 0.50	60.9± 0.14	75.4± 0.11
PKT	94.0± 0.25	97.2± 0.10	93.7± 0.40	74.8± 0.14	84.7± 0.20	57.3± 0.07	72.5± 0.04
VID	94.1± 0.47	97.2± 0.45	93.5± 0.45	77.2± 0.37	86.4± 0.26	61.1± 0.38	75.6± 0.30
ReviewKD	93.4± 0.10	97.0± 0.10	92.7± 0.34	72.7± 0.37	83.0± 0.36	55.9± 0.50	71.3± 0.53
MobileSAM	93.3± 0.15	96.7± 0.06	92.5± 0.69	74.1± 0.35	84.1± 0.18	52.3± 0.46	68.3± 0.38
TinySAM	89.4± 0.10	93.3± 0.32	86.3± 0.20	45.2± 0.76	56.1± 0.63	23.9± 2.61	36.5± 3.22
InfoSAM2(Ours)	**94.5**± **0.17**	**97.4**± **0.16**	**94.0**± **0.16**	**77.3**± **0.14**	**86.6**± **0.08**	**61.3**± **0.05**	**75.8**± **0.05**

E. Hyper-parameter Sensitivity Analysis

E.1. Analysis of the λ_1 and λ_2 in $\mathcal{L}_{info}$

In Fig. 6, we conduct a key hyper-parameter sensitivity study of λ_1 and λ_2 in balancing relation compression loss L_r and relation distillation loss L_d across three typical domains. Each sub-figure shows the heat map under different hyper-parameter settings, reflecting the change of loss function. It is recommended to view it in color display for best results. According to the accuracy heat map, we set $\lambda_1 = 1, \lambda_2 = 0.5$.



(a) Kvasir (b) Leaf (c) Road

Figure 6: Hyper-parameter sensitivity study of λ_1 and λ_2 in balancing L_r and L_d, with Kvasir, Leaf, and Road datasets (Best viewed in color).

E.2. Analysis of the α Parameter in Matrix-based Rényi's Entropy

In this paper, we set $\alpha = 2$ to compute matrix-based Rényi's α-entropy via the Frobenius norm. The core reasons for choosing $\alpha = 2$ in matrix-based Rényi's α-entropy are as follows: (i) The primary practical motivations are computational efficiency and alignment with prior works. By setting $\alpha = 2$, we enable direct computation of matrix-based Rényi entropy through Frobenius norm operations (see Eq.(11)), eliminating the necessity for eigenvalue decomposition. This optimization reduces time complexity from $O(n^3)$ to $O(n^2)$ (n represents the sample numbers) (Dong et al., 2023), substantially reducing computational costs while maintaining theoretical rigor, particularly advantageous for high-dimensional data analysis (Yu et al., 2019). Additionally, prior research has successfully applied Rényi entropy with $\alpha = 2$ in segmentation tasks (Miles et al., 2023), to align with the established practices in this field, we adopt $\alpha = 2$. (ii) For theoretical reasons, if the application requires emphasis on tails of the distribution (rare events) or multiple modalities (distributions with multiple peaks), α should be less than 2 and possibly approach to 1 from above. If the goal is to highlight the dominant mode (the most probable region), α should be greater than 2 to emphasize central tendencies. $\alpha = 2$ provides neutral weighting (Yu et al., 2019). Moreover, the Frobenius norm's differentiable and strongly convex properties guarantee rapid convergence in gradient-based optimization algorithms (Boyd, 2004).

Furthermore, in Table 11, we conduct an analysis to evaluate the performance of different α values ($\alpha = 1.01, 2, 3$). Following with prior work (Yu et al., 2019), we set $\alpha = 1.01$ to asymptotically approach Shannon entropy. The results indicate that $\alpha = 2$ achieves the highest verification accuracy while reducing computational overhead by an order of magnitude. This computational gain stems from its exclusive reliance on Frobenius norm operations, whereas $\alpha = 1.01$ or 3 require eigenvalue decompositions, which are computationally more expensive.

Table 11: Experiments of different α values in matrix-based Rényi's entropy.

METHOD	AGRICULTURE	REMOTE SENSING	COMPUTATION TIME
	IoU (Leaf)	IoU (Road)	ms
$\alpha = 1.01$	75.3$_{\pm 0.31}$	60.6$_{\pm 0.12}$	32.1$_{\pm 30.7}$
$\alpha = 2$	**75.6**$_{\pm \mathbf{0.27}}$	**61.4**$_{\pm \mathbf{0.30}}$	**1.2**$_{\pm \mathbf{0.3}}$
$\alpha = 3$	75.2$_{\pm 0.30}$	61.2$_{\pm 0.06}$	35.4$_{\pm 31.2}$

F. Deep Dive into the Relation Model

F.1. Understanding the Domain-invariant Information Encoded by the Relation Model

Many recent studies leverage SAM's pre-trained capabilities for downstream tasks by fine-tuning. However, when the fine-tuning data distribution is narrow, the model tends to overfit task-specific local features (Wang et al., 2024). We argue that this is mainly because task-specific optimizations will cover or suppress domain-invariant features learned during pre-training.

To substantiate this assumption, we have conducted experiments in Section 5.4 to illustrate that the extracted relation works (see Table 4) and is domain-invariant (see Table 6). In Table 4, the extracted relations boost other distillation methods (e.g., TinySAM) by 1.7%–5.2% IoU, indicating the preserved information's effectiveness. In Table 6, applying the RM trained on one domain to a completely different domain still preserves its effectiveness, suggesting that these transferable relations are domain-invariant and beneficial for fine-tuning.

We further explore the nature of domain-invariant information. We employ relations to represent domain-invariant information, which serves as an implicit yet generalizable characterization that may inherently encode various domain-agnostic properties. Here, we showcase and evaluate structural edge information using the Boundary F1 Score (BFS) (Zhang et al., 2023b). As shown in Fig. 7, InfoSAM with the relation module outperforms other fine-tuning baselines in boundary preservation, demonstrating that this implicit relational encoding effectively extracts richer structural edge features.

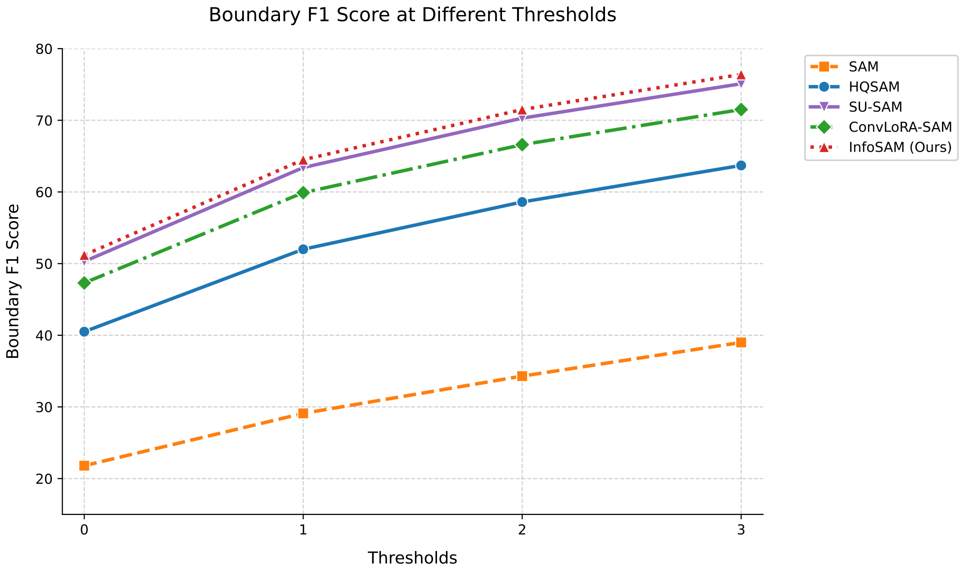


Figure 7: Boundary quality measured by the F1-score at various thresholds (0, 1, 2, and 3 pixels) on the Leaf dataset. The thresholds represent the allowable pixel distance: if the predicted boundary is within the threshold distance from the ground truth boundary, it is regarded as correct. A larger threshold provides a more tolerant evaluation.

Additionally, we visualize the relation maps extracted from the teacher and student models at various stages of training InfoSAM, ranging from the early to late epochs.

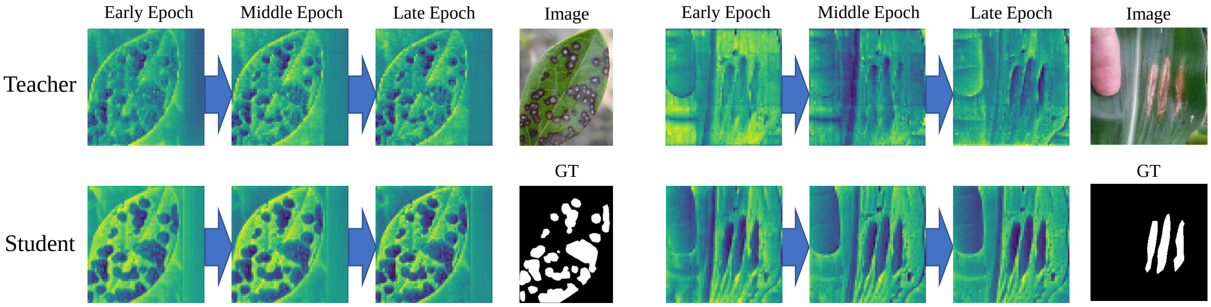


Figure 8: Relation maps evolve from early to late epochs. During training, the relation module gradually captures key information from the pre-trained teacher model, leading to improved performance of the student model.

F.2. Effectiveness of the Proposed Loss for Relation Model Learning

This section investigates the effectiveness of the proposed loss $\mathcal{L}_{\text{info}}$ in guiding the relation model to learn generalizable features while avoiding trivial solutions. As shown in Eq. (11) and Eq. (12), $\mathcal{L}_{\text{info}}$ includes regularization terms such as $\log_2 \|G^T_{\text{imr}}\|_F^2$, $\log_2 \|G^T_r\|_F^2$, and $\log_2 \|G^S_r\|_F^2$, which promote feature diversity and prevent it from converging to trivial solutions.

To further verify the effectiveness of these regularization terms, we conduct an ablation study to assess their impact both qualitatively (through the visualization of relation maps) and quantitatively (through performance on downstream tasks) in Fig. 9 and Table. 12, respectively: (i) For visualization performance, we visualize the relation maps and their corresponding statistical distributions evolving from early to late epochs. As shown in Fig. 9, without the regularization terms, the distribution of the relation maps becomes increasingly narrow during training, and the domain-invariant information

Table 12: Ablation study of regularization terms (RT) in $\mathcal{L}_{\text{info}}$.

METHOD	AGRICULTURE	REMOTE SENSING
	IoU (Leaf)	IoU (Road)
w/o RT	$74.6_{\pm 0.12}$	$59.6_{\pm 0.69}$
w RT	$\mathbf{75.6}_{\pm \mathbf{0.27}}$	$\mathbf{61.4}_{\pm \mathbf{0.30}}$

Table 13: Experiments of different Relation Model architectures. "Attn-n" represents the number of attention layers for RM.

METHOD	AGRICULTURE	REMOTE SENSING
	IoU (Leaf)	IoU (Road)
Dot Product	$75.2_{\pm 0.35}$	$61.0_{\pm 0.04}$
Linear	$74.9_{\pm 0.51}$	$59.3_{\pm 0.58}$
Attn-5	$75.4_{\pm 0.22}$	$61.4_{\pm 0.12}$
Attn-3	$75.4_{\pm 0.40}$	$\mathbf{61.7}_{\pm \mathbf{0.06}}$
Attn-1 (ours)	$\mathbf{75.6}_{\pm \mathbf{0.27}}$	$61.4_{\pm 0.30}$

captured by the relation maps becomes less distinct. In contrast, the RM trained with regularization terms maintains a broad relation distribution and a more representative relation map. (ii) For downstream performance shown in Table. 12, the regularization terms benefit our method by improving performance, as demonstrated by a 1.0% and 1.8% increase in IoU on the Leaf and Road datasets, respectively. Both results indicate that the proposed loss with regularization terms effectively extracts domain-invariant features, rather than domain-specific noise, thereby enhancing downstream performance and alleviating the problem of trivial solutions.

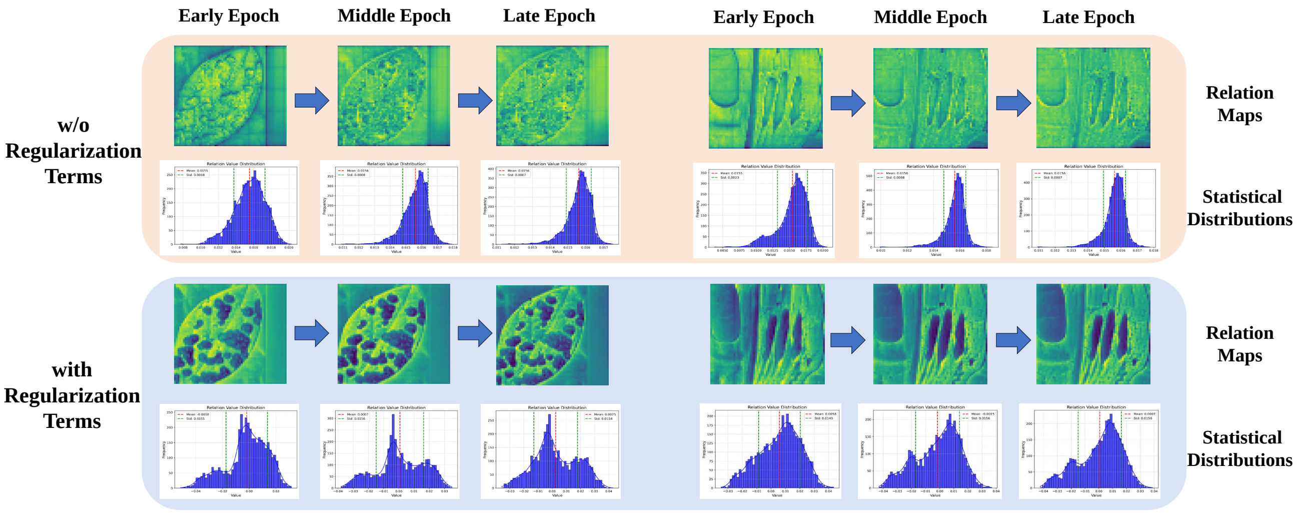


Figure 9: Evolution of relation maps and their statistical distributions over epochs, without and with the regularization term.

F.3. Exploring the Relation Model architectures

We conduct an analysis to compare different model architectures and explore the number of attention layers for relation module (RM). We compare direct dot product, a linear layer, multiple attention layers, and our proposed RM across multiple experiments on two distinct domains.

The experimental results show that: (i) attention-based RM outperforms other other architectures designs. This indicates that attention mechanism effectively assess the correlations between the input features (i.e., image and mask features), thereby adaptively filtering and enhancing the useful information (e.g., edge details) while reducing redundancy. (ii) If we stack an appropriate number of attention layers (e.g., 3 layers) in the RM can be beneficial for capturing key information. However, stacking too many (e.g., five layers) increases training difficulty and risks overfitting. In a nutshell, the current RM design is a trade-off between performance and computational overhead, and it effectively captures the relationships between image and mask features.

G. Visualization Results

We present visualization results of mask predictions across various datasets using different PEFT methods for SAM (SAM, HQSAM, SU-SAM, ConvLoRA-SAM, and InfoSAM). These results further demonstrate the superiority of our proposed InfoSAM.

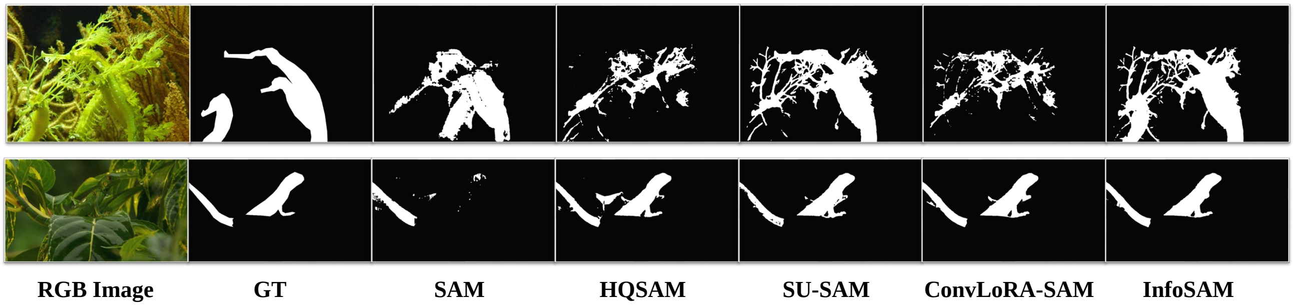


Figure 10: Visualization results on camouflaged object segmentation.

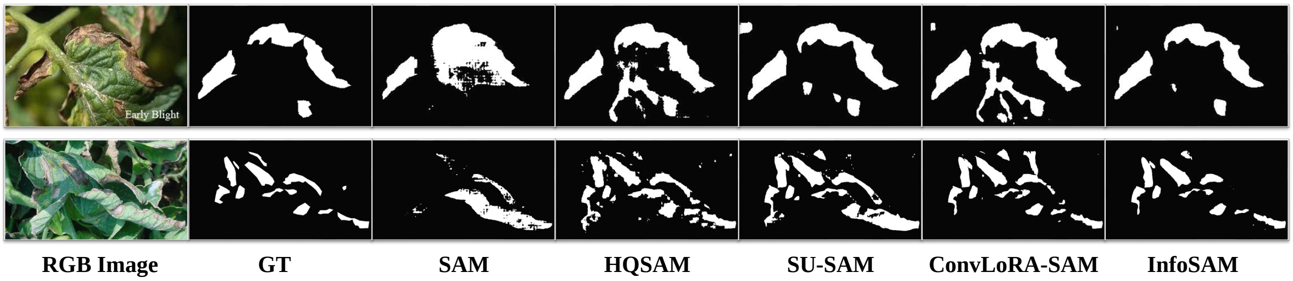


Figure 11: Visualization results on leaf disease segmentation.

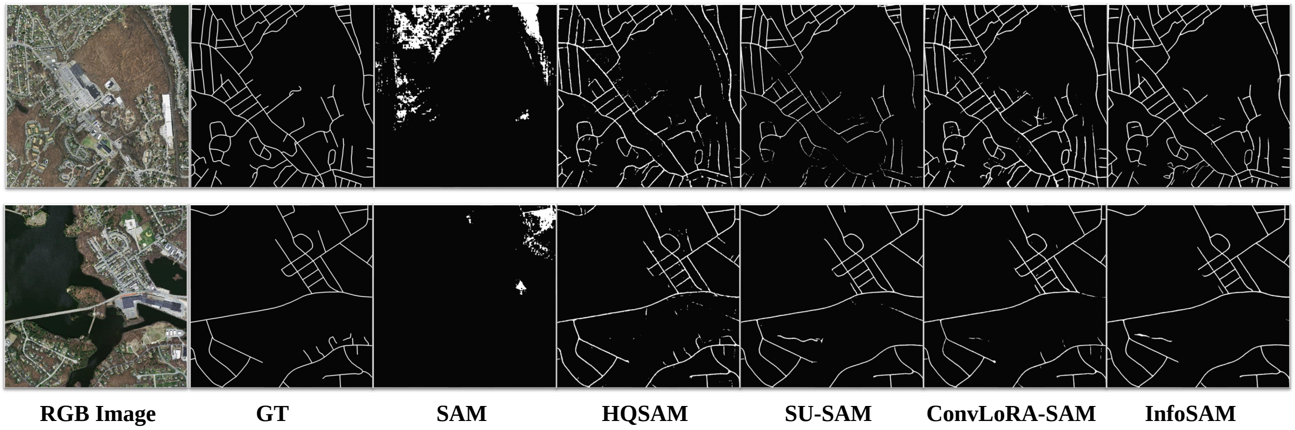


Figure 12: Visualization results on remote sensing road segmentation.