Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

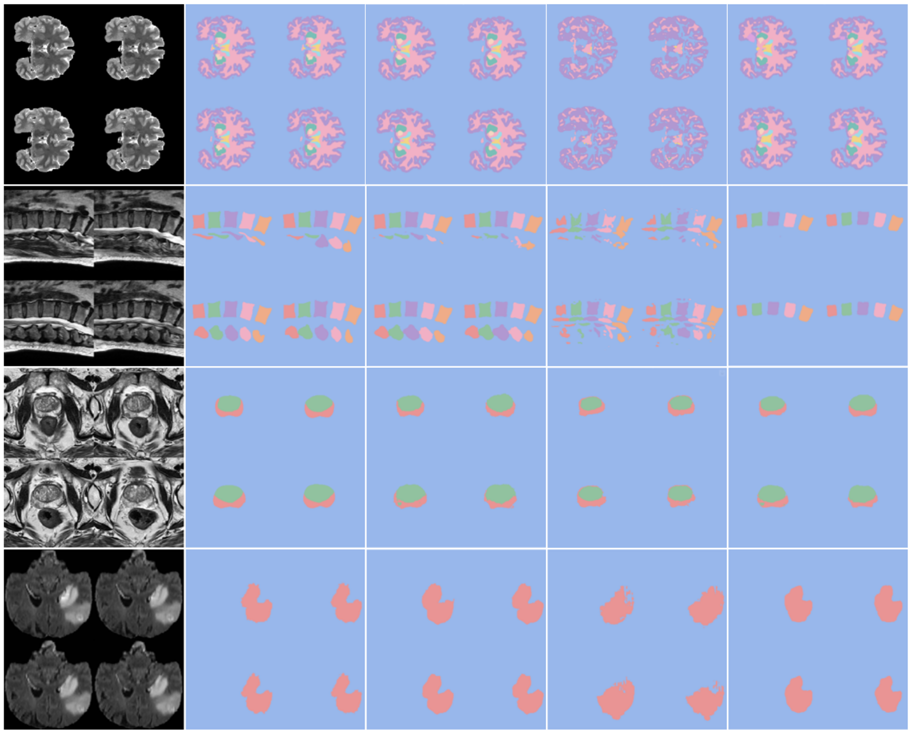
Computer
Vision
Lab

# Training and Tuning Strategies for Foundation Models in Medical Imaging

Master's Thesis

Faruk Kerem Cekmeceli

Computer Vision Laboratory
Department of Information Technology and Electrical Engineering



Advisors: Dr. Anna Susmelj, Dr. Ertunc Erdil
Supervisor: Prof. Dr. Ender Konukoglu

June 17, 2024

# Abstract

In this thesis, we explore advanced training and tuning strategies for foundation models in medical imaging, focusing on their generalization and adaptation capabilities. The primary challenge addressed is the development of robust segmentation models that can generalize well across different medical imaging datasets and anatomical structures. We employed various foundation models, including Dino, SAM, MedSam, and MAE, with state-of-the-art decode heads and fine-tuning strategies to achieve this goal. Notably, we introduce HQHSAM, a novel decode head combining elements of HSAM and HQSAM, which demonstrates superior performance in medical segmentation applications. Our findings indicate that foundation models, particularly with the HQHSAM decode head, significantly enhance domain generalization performance. Additionally, parameter-efficient fine-tuning techniques such as Rein and Ladder fine-tuning were crucial in optimizing model performance. These results underscore the potential of foundation models in improving the accuracy and robustness of medical image segmentation and offer promising avenues for domain adaptation, providing a solid foundation for future research in this domain.

# Acknowledgements

I would like to express my deepest gratitude to my supervisors, Dr. Anna Susmelj and Dr. Ertunc Erdil, for their invaluable support, guidance, and supervision throughout this project. Their expertise and encouragement have been instrumental in the completion of this thesis. I also extend my sincere thanks to Professor Ender Konukoglu for his insightful feedback and suggestions.

I am grateful to Guney Tombak, a PhD student from the lab, for his support and for the many informal and useful discussions we had. Special thanks to Meva Himmetoglu, another PhD student from the lab, for her help and guidance on brain tumor datasets. I would also like to acknowledge all the staff at the ETH Computer Vision Laboratory for providing access to the necessary hardware and infrastructure.

Finally, I would like to mention the use of AI tools, including ChatGPT, which assisted me in refining my ideas and structuring my writing.

Thank you all for your contributions and support.

Faruk Kerem Cekmeceli

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Focus of this Work

Medical image segmentation is a critical task in the field of medical imaging, essential for various diagnostic and therapeutic applications. Traditional models often struggle with generalization, particularly when applied to new datasets or anatomical structures that differ from their training data. This thesis aims to address these limitations by leveraging foundation models, which are pre-trained on large and diverse datasets, for medical image segmentation. Our research investigates the effectiveness of different decode heads and fine-tuning strategies to enhance the performance and generalization of these models across multiple medical imaging datasets.

## 1.2 Scope of the Work

This thesis focuses on the evaluation and optimization of foundation models, including DinoV2 [25], Sam [18], MedSam [21], and Mae [11], for medical image segmentation tasks. We implement state-of-the-art decode heads such as HQHSAM and test various fine-tuning strategies like Rein [32] and Ladder fine-tuning [5]. The models are evaluated on datasets from four anatomies: brain, prostate, lumbar spine, and brain tumor. Our goal is to identify the optimal configurations that maximize both in-domain and domain generalization performance, while also exploring their potential for domain adaptation.

## 1.3 Thesis Organization

The thesis is structured as follows:

- **Chapter 1:** Introduction, providing an overview of the research focus and scope.
- **Chapter 2:** Related Work, reviewing existing literature on medical image segmentation, foundation models, and fine-tuning techniques.
- **Chapter 3:** Materials and Methods, detailing the data loaders, encoder/decoder models, fine-tuning strategies, and metrics used in our experiments.
- **Chapter 4:** Experiments and Results, presenting the findings from our evaluations, including decode head selection, fine-tuning strategy assessments and domain adaptation analysis.
- **Chapter 5:** Discussion, analyzing the results and their implications for future research.

- **Appendices:** Supplementary materials, including detailed experiment tables, additional data and figures.

Through this research, we aim to demonstrate the potential of foundation models to significantly improve the robustness and accuracy of medical image segmentation, paving the way for more reliable and generalizable diagnostic tools in medical imaging. Furthermore, we highlight the potential of our models for semi-supervised domain adaptation, enabling their application across diverse clinical environments.

# Chapter 2

# Related Work

Medical image segmentation, transformer-based architectures, and vision foundation models for segmentation tasks have attracted considerable attention in the literature over the past years. In the following sections, we explore key advancements in these areas. We begin with Established Models for Medical Imagery Segmentation, reviewing foundational techniques. Next, Transformers for Semantic Segmentation highlights the impact of transformer architectures. We then delve into Domain Adaptation and Generalization, emphasizing techniques to enhance model robustness across different domains. Self-Supervised Learning and Foundation Models discusses leveraging unlabeled data for model training. Parameter Efficient Fine-Tuning Techniques examines methods to adapt models efficiently, and finally, Data Enhancement Techniques covers strategies like data augmentation and rare class sampling to improve segmentation performance.

## 2.1 Established Models for Medical Imagery Segmentation

In the realm of medical image segmentation, several models have been established as benchmarks due to their efficiency and accuracy. Three prominent models in this field are U-Net [26], Swin U-Net [4] and nnU-Net [15].

**U-Net** [26] is a convolutional neural network architecture designed for biomedical image segmentation. The architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. The contracting path follows the typical architecture of a convolutional network, with repeated application of convolution and max-pooling layers to reduce the spatial dimensions and increase feature complexity. The inherent inductive bias of convolutions, which assumes that local patterns and spatial hierarchies are important, allows U-Net to effectively learn and generalize from spatial structures in the data. The expanding path involves up-sampling and convolution layers, which restore the original image resolution and combine features from the contracting path through skip connections. These skip connections help in retaining high-resolution features that are crucial for accurate segmentation. U-Net has been widely adopted due to its efficiency and effectiveness in segmenting complex structures in medical images.

**Swin U-Net** [4] is an architecture that combines the strengths of U-Net and the Swin Transformer for medical image segmentation. To understand Swin U-Net, it is important to first grasp the concepts of self-attention [30], Vision Transformers (ViTs) [8], and Swin Transformers [19].

Self-attention is a mechanism that allows neural networks to dynamically weigh the importance of different input features. In the context of image processing, it enables the model to focus on various parts of an image with varying degrees of attention, capturing long-range dependencies and contextual relationships more effectively than traditional convolutional operations.

Vision Transformers (ViTs) apply the self-attention mechanism directly to image patches. Unlike convolutional neural networks (CNNs) that use convolutions to process the entire image, ViTs divide the image into fixed-size patches and treat each patch as a token in a sequence. The self-attention mechanism is then applied to these tokens, allowing the model to learn relationships between different parts of the image. Although ViTs have shown impressive performance in various vision tasks, they require a large amount of computational resources and may struggle with fine-grained details due to their global attention mechanism.

The Swin Transformer, or Shifted Window Transformer, addresses some of the limitations of ViTs by introducing a hierarchical representation with shifted windows. Unlike standard transformers that operate globally on the entire image, Swin Transformers divide the image into non-overlapping local windows and apply self-attention within each window. To capture cross-window connections, these windows are shifted in subsequent layers. This approach not only reduces computational complexity but also preserves fine-grained spatial information and enhances the model's ability to capture both local and global contexts.

Swin U-Net incorporates these Swin Transformer blocks into the classic U-Net architecture, maintaining the U-shaped structure with an encoder (contracting path) and a decoder (expanding path). The encoder utilizes Swin Transformer blocks to extract multi-scale features with rich contextual information, while the decoder reconstructs the image resolution, integrating features from the encoder through skip connections. These skip connections ensure that detailed spatial information is retained, crucial for precise segmentation.

**nnU-Net** [15] is a self-adapting framework for medical image segmentation that automatically configures itself to a given dataset, eliminating the need for manual tuning. It leverages the U-Net architecture and introduces dynamic pipeline adjustments based on dataset properties, such as input image size, resolution, and label characteristics. nnU-Net operates in three configurations: 2D, 3D full resolution, and 3D low resolution, ensuring optimal performance across various medical imaging tasks.

## 2.2 Transformers for Semantic Segmentation

In recent years, transformers have gained significant attention in the field of semantic segmentation due to their ability to capture long-range dependencies and global context. Two notable models that leverage these capabilities are the Dual Attention Network (DANet) [10] and SegFormer [33].

**Dual Attention Network** (DANet) [10] is an architecture for semantic segmentation that utilizes both spatial and channel-wise attention mechanisms. It features two main components: the Position Attention Module (PAM) and the Channel Attention Module (CAM). PAM captures spatial dependencies by modeling relationships between any two positions in the feature map, allowing the network to use global spatial context for accurate segmentation while CAM focuses on interdependencies among different channels in the feature map, enhancing the model's ability to utilize global contextual information. By combining PAM and CAM, DANet effectively captures both spatial and channel-wise dependencies, resulting in more precise and context-aware segmentation, thus improving performance in various computer vision tasks.

**SegFormer** [33] is a transformer-based architecture designed for semantic segmentation. Unlike traditional convolutional neural network (CNN) methods, SegFormer leverages the strengths of transformers to capture long-range dependencies and global context, which are crucial for precise segmentation. The architecture features a hierarchical structure with multiple stages, each composed of a transformer encoder and a lightweight MLP decoder. This design allows SegFormer to effectively capture both high-level semantic information and fine-grained details. Moreover, SegFormer is highly efficient and scalable, achieving state-of-the-art performance on various benchmark datasets while maintaining reduced computational overhead compared to other transformer-based models. Its robustness and efficiency make SegFormer a promising approach for various segmentation tasks in computer vision.

## 2.3 Domain Generalization and Adaptation

Domain Generalization is a technique aimed at improving the performance of a model across multiple unseen target domains without requiring access to target domain data during training. This approach seeks to build models that generalize well to new domains by learning domain-invariant features, thereby enhancing the robustness and applicability of the models across varied conditions.

Domain adaptation is a machine learning technique aimed at improving the performance of a model on a target domain that differs from the source domain where the model was trained. This technique addresses the challenge of ”domain shift,” where differences in data distributions can degrade model performance when applied to new, unseen datasets. It involves methods that modify the model or its training process to minimize these differences, enabling the model to generalize better across varied data conditions.

Domain adaptation is particularly crucial in medical image segmentation due to the inherent variability across different medical imaging datasets. Factors such as differences in imaging equipment, patient populations, and annotation protocols can significantly affect the appearance and quality of medical images. This variability often leads to poor generalization of segmentation models when applied to new datasets without adaptation. Implementing domain adaptation strategies allows these models to be robust across different clinical settings, enhancing their reliability and utility in medical diagnostics, treatment planning, and other healthcare applications. Additionally, domain adaptation reduces the need for extensive labeled data in each new clinical environment, which is particularly beneficial given the high costs and effort required to annotate medical images.

To address domain adaptation with few or no labeled target data, several commonly used techniques are employed. Two of these techniques are entropy minimization [31] and self-training [1]. Additionally, there is a more advanced framework known as DaFormer [13], which further enhances the effectiveness of domain adaptation strategies.

**Entropy minimization** [31] is a technique used in domain adaptation to encourage the model to make confident predictions on the target domain data without requiring any labeled target data. By minimizing the entropy of the output probability distribution, the model is incentivized to produce predictions that are closer to 0 or 1, effectively reducing uncertainty. This approach helps align the feature representations of the source and target domains, improving generalization to the target domain.

**Self-training** [1] is a technique where a model initially trained on source domain data is used to generate pseudo-labels for the unlabeled target domain data. The model is then retrained using these pseudo-labeled target data, iteratively refining its predictions. Self-training can be employed both without any labeled target data or with a few labeled examples to generate the pseudo-labels, enhancing the model’s adaptation and performance in unsupervised or semi-supervised manner by allowing it to learn the target domain’s specific characteristics.

**DAFormer** [13] is a state-of-the-art framework designed to enhance domain adaptation in semantic segmentation tasks. It incorporates adversarial learning and a transformer-based architecture to align features between the source and target domains effectively. The framework utilizes a dual-branch network where one branch learns segmentation features from the source domain while the other focuses on the target domain. An adversarial discriminator aligns these features to ensure the model learns domain-invariant representations, enhancing generalization capabilities. DAFormer is versatile, supporting various encoders like ResNet [12] and decoders like the SegFormer decoder [33]. By leveraging transformers for capturing long-range dependencies and adversarial learning for domain alignment, DAFormer significantly improves segmentation model performance across different domains, making it highly effective for real-world applications requiring robust domain adaptation.

## 2.4 Self-Supervised Learning and Foundation Models

**Vision foundation models** are large-scale neural networks pre-trained on extensive and diverse datasets to learn general-purpose features that can be applied across a wide range of vision tasks. These models serve as a foundational layer upon which specialized models can be built and fine-tuned for specific applications. The primary advantage of vision foundation models is their ability to capture rich, hierarchical representations of visual data, making them highly effective in tasks such as image classification, object detection, segmentation, and more. By leveraging these pre-trained models, the amount of labeled data and computational resources required to train models from scratch can be reduced significantly, while achieving state-of-the-art performance. Self-supervised learning has played a pivotal role in advancing vision foundation models. This approach leverages large amounts of unlabeled data to pre-train models by solving pretext tasks, which are designed to generate supervisory signals from the data itself. By training on diverse and rich datasets without the need for manual annotation, self-supervised learning enables vision foundation models to learn robust and transferable features, further enhancing their utility across various tasks. Prominent examples of such models include **Masked Autoencoders (MAE)** [11], **DINOv2** [25], and the **Segment Anything Model (SAM)** [18]. MAE focuses on reconstructing missing parts of input images, encouraging the model to develop a deep understanding of image structure and semantics. DINOv2, building on the principles of knowledge distillation, trains models to produce consistent representations across different views of the same image, enhancing their ability to generalize. SAM consists of three main components: an image encoder, a prompt encoder, and a mask decoder, as described in [18]. With its unique architecture, SAM aims to achieve universal image segmentation by training on a vast and diverse dataset, enabling it to segment any object in any image without the need for task-specific training. These vision foundation models, underpinned by self-supervised learning, are setting new benchmarks in various computer vision tasks mostly involving natural images and are poised to significantly impact the development of future CV applications.

Recently, foundational models have seen several adaptations and refinements. Particularly, the Segment Anything Model (SAM) has served as the basis for works such as HQSAM [17], MedSAM [21], and HSAM [6]. Notably, the latter two are tailored for medical applications, aligning closely with our research focus.

**HQ-SAM** [17] (High-Quality Segment Anything Model) is an advanced adaptation of the Segment Anything Model (SAM) designed to enhance segmentation quality, particularly for objects with intricate structures. Despite SAM's powerful zero-shot capabilities and flexible prompting, it often struggles with fine details and complex shapes. HQ-SAM addresses these limitations while maintaining SAM's efficiency and generalizability. HQ-SAM introduces two key innovations: the High-Quality Output Token and Global-Local Feature Fusion.

The High-Quality Output Token is a learnable token injected into SAM's mask decoder, responsible for predicting high-quality masks. This token enhances the model's ability to accurately segment fine details and complex structures. Global-Local Feature Fusion is another key mechanism that enriches mask decoder features by combining global semantic context and local boundary details. This involves extracting features from the early and final stages of SAM's ViT encoder and fusing them to enhance the segmentation of complex structures.

HQSAM demonstrates superior performance across diverse segmentation benchmarks, showing a significant improvement in average Dice scores compared to existing prompt-free SAM variants. By integrating these innovations, HQ-SAM offers a robust solution for high-quality segmentation tasks, particularly in scenarios requiring precise delineation of complex shapes and structures.

**MedSAM** [21] investigates the application of the Segment Anything Model (SAM) in medical image segmentation, utilizing its zero-shot learning capabilities and prompt-based interface while training on a diverse array of medical datasets. The study illustrates that SAM, with appropriate modifications, can achieve competitive performance across various medical imaging tasks. Notably, "*MedSAM's performance was found to be on par with four human experts and even surpassed that of two experts, highlighting its potential as a robust tool for medical image segmentation in clinical practice.*" However, the prompt-based approach presents limitations, especially in medical imaging, where specific prompts may not consistently align well with the diverse and complex nature of medical data. This underscores the need for further refinement to fully exploit SAM's potential in the medical domain.

**H-SAM** [6] (Hierarchical Segment Anything Model) is a specialized adaptation of the Segment Anything Model (SAM), designed to achieve high-quality segmentation without the need for prompts. It addresses the challenges SAM faces in medical imaging, such as substantial training costs and the requirement for high-quality prompts for optimal performance.

H-SAM employs a two-stage hierarchical decoding process. In the initial stage, SAM's original decoder generates a prior probabilistic mask, providing a preliminary segmentation. In the second stage, this initial mask is refined using two key innovations: the Class-Balanced, Mask-Guided Self-Attention Mechanism, which enhances image embedding by addressing the unbalanced label distribution, ensuring more detailed and accurate segmentation, and the Learnable Mask Cross-Attention Mechanism, which modulates interactions among different image regions based on the prior mask, improving segmentation quality.

Additionally, H-SAM incorporates a hierarchical pixel decoder to capture fine-grained and localized details more effectively. The model demonstrates improvements in average Dice scores compared to existing prompt-free SAM variants in multi-organ segmentation tasks using only 10% of 2D slices. H-SAM also outperforms state-of-the-art semi-supervised models, even without using any unlabeled data, making it a significant advancement in achieving high-quality, prompt-free segmentation, particularly in medical imaging contexts.

## 2.5 Parameter Efficient Fine Tuning Techniques

Parameter-efficient fine-tuning techniques are critical for effectively adapting large vision foundation models (VFMs) to specific tasks without incurring high computational costs. These methods are especially useful for foundation models, as they leverage pre-trained knowledge while requiring minimal additional parameters.

**Ladder Fine-Tuning** [5] introduces a parallel complementary network that integrates with the existing model architecture. It reuses and preserves the pre-trained model weights of the foundation model and only introduces minimal additional parameters and computation. This technique involves summing features from the foundation model and the complementary network to enhance the model's performance while keeping the number of trinable parameters small.

**Rein** [32] is a robust fine-tuning approach that parameter-efficiently harnesses VFMs for Domain Generalized Semantic Segmentation (DGSS). Rein employs a set of learnable tokens linked to distinct instances, refining and forwarding feature maps block by block within the backbone. This method enables precise feature refinement for each instance in an image with fewer trainable parameters, significantly boosting segmentation performance. Rein-LoRA further optimizes this by incorporating low-rank adaptation [14] techniques to reduce parameter counts even more, making the fine-tuning process more efficient and scalable.

These techniques enhance the utility of VFMs by reducing the need for extensive retraining, making them more adaptable to varied tasks and domains while maintaining high performance.

## 2.6 Data Enhancement Techniques

The following techniques are used in tandem with the segmentation model to improve performance.

**Rare class sampling** is a technique employed in segmentation tasks to address the challenge of imbalanced class distributions. In many real-world datasets, certain classes or categories are significantly underrepresented compared to others. This imbalance can lead to a model that performs well on frequent classes but poorly on rare ones. To mitigate this, rare class sampling adjusts the training process by increasing the representation of rare classes. This can be achieved by duplicating instances of the rare classes or generating synthetic samples through various methods as demonstrated in [13]. By presenting the model with a more balanced dataset, rare class sampling ensures that the model learns to accurately segment all classes, including those that are less frequent, thereby improving overall segmentation performance.

**Data augmentation** is a technique that artificially expands the size and diversity of a dataset by applying various transformations to the existing data. Common transformations include rotations, flips, translations, scaling, cropping, elastic deformations and adjustments in brightness, contrast, and color. The purpose of data augmentation is to expose the model to a broader range of variations during training, helping it become more robust and better generalize to unseen data. In segmentation tasks, data augmentation can significantly enhance the model's ability to handle real-world variations and prevent overfitting by ensuring the model does not become overly reliant on the specific characteristics of the training data. By enriching the training dataset with augmented samples, this technique helps in developing more accurate and resilient segmentation models.

These techniques, when combined with segmentation models, play a crucial role in improving model performance and ensuring accurate segmentation across diverse and imbalanced datasets.

# Chapter 3

# Materials and Methods

**A Modular Segmentation Framework** is developed for training and testing medical segmentation models compatible with a range of common input data formats. The framework supports encoder-decoder architectures with various backbones and projection heads, or standalone models for predicting segmentation masks. Loss functions and optimizers are switchable, while metrics such as Dice Similarity Coefficient (DSC) and mean Intersection over Union (mIoU) are automatically computed per slice and over the volume. These metrics, along with sample segmentation masks from the validation set, are logged during training for quick assessment of performance.

The aforementioned sections are examined in more detail in the following corresponding sections.

## 3.1 Data Loaders

NIfTI and HDF5 are among the most commonly used formats for medical applications, enhancing the storage, management, and analysis of medical imaging data. NIfTI is more specialized for neuroimaging, providing simplicity and efficiency for brain imaging data. In contrast, HDF5 offers broader applications due to its flexibility and capacity for handling large and complex datasets, making it suitable for a wide range of medical imaging tasks.

For our application, given the 2D nature of the foundation models under consideration, we focus on 2D segmentation of the volumes slice by slice. Our dataloaders are adapted to support these formats, returning the 2D slice along with a boolean flag indicating if the slice corresponds to the last one of the volume. This flag triggers the computation of metrics over the volume typically for the validation and test sets, where no shuffling is performed.

Additionally, we provide support for the PNG format for datasets with a common volume depth among samples. This broadens the range of datasets we can tackle and experiment with.

Finally, our dataloaders include rare class sampling as an option to cope with imbalanced classes within the datasets. They also support random data augmentations such as elastic deformations, structural augmentations, and photometric distortions. These augmentations help increase the diversity of the training data, improve the model's robustness, and reduce the risk of overfitting. Both features are intended for the training splits.

## 3.2 Encoder/Decoder Models

### 3.2.1 Backbones

Encoders are employed as feature extractors for segmentation. We have implemented support for DINOv2 [25], SAM [18], MedSAM [21] (which only differs in weights, not architecture, from SAM), and MAE [11] foundation image encoders, as well as ResNet [12], a commonly used backbone for comparison across all available sizes. All backbones are initialized with their pre-trained weights provided by the checkpoint path. DINOv2 uses a patch size of 14x14, while SAM and MAE use a patch size of 16x16.

To ensure the backbone input size is independent, we have implemented bilinear interpolation for the positional encodings across all foundation models, identical to the approach used in DINOv2. This enhancement was necessary to render the backbones more flexible and efficient, as their expected image size of 1024x1024 was too large for medical applications and resulted in higher VRAM usage. Normalization is performed as a pre-processing step, adhering to each backbone's pre-training requirements, and inputs are resized to be compatible with the chosen backbone (e.g., interpolated to a multiple of patch size). Features from various depths of the backbone can be saved and concatenated for the output, a method demonstrated to be effective in works such as [26], [4], [15], [33], and [25] for segmentation using foundation models.

The ResNet backbone was adapted to support extracting outputs from different depths, and each output can be passed through an additional convolutional layer and interpolation to achieve uniformity among output shapes. Additionally, the last convolutional block can be skipped, as was done in [5], to further decrease the backbone size if necessary.

All backbones can be frozen to skip gradient computation. Moreover, various fine-tuning strategies are implemented for parameter-efficient fine-tuning of backbones, which will be detailed in the following section.

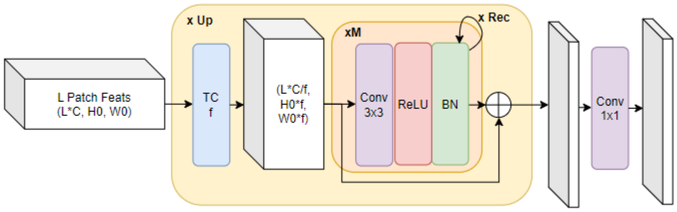
### 3.2.2 Decode Heads

Various projection heads, including standard ones such as linear projection, ResNet and U-Net [26] adapted ones, as well as transformer-based decoders like the Dual Attention (DA) decoder [10] and SegFormer head [33], are experimented with. Additionally, state-of-the-art methods including SAM's prompt encoder combined with its mask decoder, and adaptations from HSAM [6], HQSAM [17], and a custom fusion of the latter two that we name HQHSAM, are also explored which are further detailed below. Projection heads are always trained.

- **Linear Head**: Consists of a simple 2D batch norm and 1x1 convolution used to provide a simple and efficient way to map high-dimensional feature representations from the backbone network to the segmentation mask. This approach allows for quick computation and straightforward backpropagation, making it easier to integrate into various architectures. It was also utilized in [25] for segmentation tasks by concatenating the output of the backbone from the last four blocks.

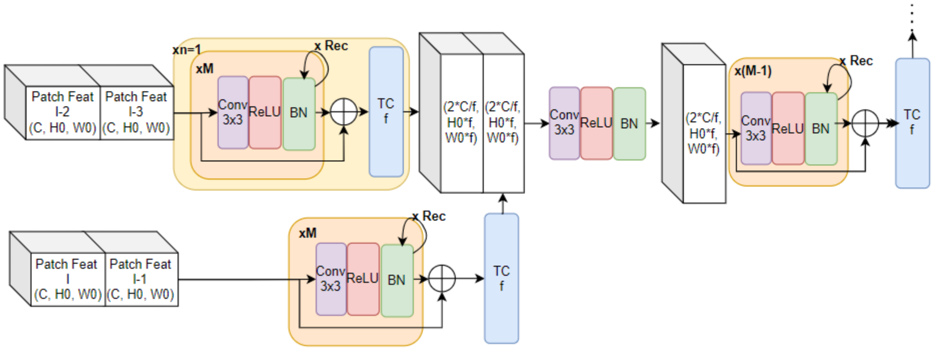
- **ResNet Head**: Following the motivation from ResNet, we have implemented a decoder that incorporates a chain of convolutional blocks with residual connections. Each block consists of the following layers in order: a transposed convolution reducing the dimensionality of the features by a factor of $f$ while increasing the size by the same amount, followed by 3x3 convolutions, non-linearity, and 2D batch norm layers, with the latter three repeated recursively. $M$ of these blocks are linked one after the other with skip connections. Above described list of ”Up Blocks” are repeated until the desired shape is achieved. The final scaling factor for spatial sizes and the feature dimensions can be calculated as $f_{tot} = f^{N_{up}}$. Finally, the features pass through a 1x1 convolution to extract the segmentation logits. $L$ backbone features from different depths can be concatenated and fed into this structure. Above described architecture is illustrated in Figure 3.1. To keep the number of parameters consistent among backbones of varying sizes, an initial convolutional layer is used to standardize the dimension for the residual blocks (this step is not visualized for readability).

Figure 3.1: ResNet Type Decoder



- **U-Net Head**: U-Net's expanding path is adopted using the aforementioned residual blocks. $L$ backbone features from different depths are combined at the input for each individual up block, where the dimensionality of the features is reduced and the size is increased by the same amount, followed by convolutional blocks with residual connections. Another set of features then goes through the residual blocks with transposed convolutions in between until the same dimension and size are achieved. These two features are concatenated and passed through a third residual block, which reduces the doubled dimensions due to the concatneation by 2 while maintaining the same shape. This forms the ”Up Blocks,” which are repeated N times for the decoder. Finally, a 1x1 convolution is performed, identical to the ResNet-type decoder, to obtain segmentation logits. To have N ”Up Blocks,” N+1 sets of backbone features (single, pairs, or triplets) are required. The first ”Up Block” for the UNet type decoder with L=2 is illustrated in Figure 3.2.

Figure 3.2: UNet Type Decoder



*First ”Up Block” is illustrated omitting the initial convolutions for dimensionality reduction on patch features for clarity*

- **DA Head**: Used for segmentation to simultaneously capture spatial and channel-wise dependencies. This mechanism enhances the model's ability to focus on important features and regions, leading to more accurate and context-aware segmentation results. MMSEG [7] implementation was used.

- **SegFormer Head**: Combines hierarchical feature representations from multiple network stages with lightweight MLP decoders, enabling efficient and accurate segmentation. This design allows the model to capture both fine details and global context, improving overall segmentation performance. The implementation from MMSEG was utilized.

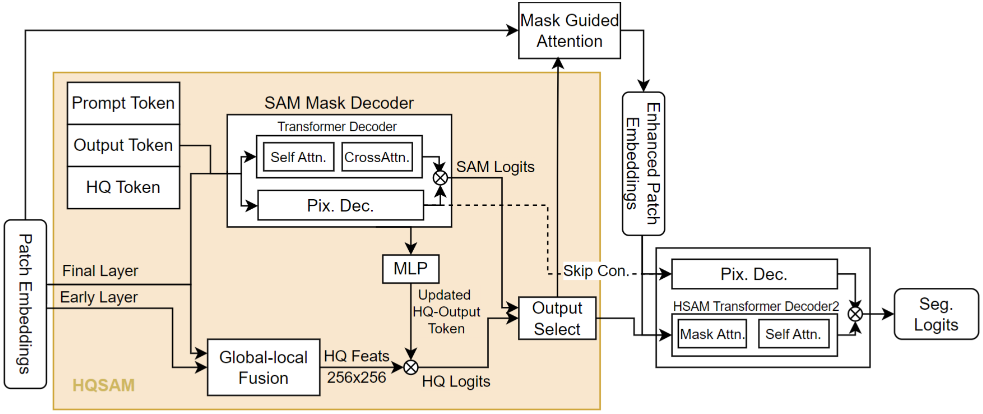
- **SAM Mask Decoder**: This powerful yet efficient transformer-based decoder design introduced in [18] is adapted to be compatible with any backbone within our framework. Both the prompt encoder, which uses the default prompts for prompt-free operation and can be trained or set to freeze, and the mask decoder can be initialized with pretrained weights from either SAM or MedSAM. The final neck which reduces the patch feature dimensions to 256 is only present for SAM and MedSam, thus for Dino and MAE, this neck is randomly initialized trained from scratch inside the decoder. The SAM Mask Decoder ensures robust performance and flexibility, making it a versatile component in various segmentation tasks. It serves as the basis for the two following decoder architectures.

- **HQSAM Head**: Enhances the original SAM decoder by introducing a High-Quality Output Token and a new mask prediction layer. This setup refines mask predictions by combining features from the first and last stages of the model to improve detail and accuracy in segmentation. During training, outputs originating from the HQ token embeddings are returned, while for validation and testing, the HQ output is summed with the SAM decoder embeddings.

- **HSAM Head**: Incorporates a two-stage hierarchical decoding process that enhances segmentation quality. It refines initial mask predictions with advanced mechanisms like mask-guided self-attention and learnable mask cross-attention, leading to more accurate and detailed segmentation results (class-balancing is omitted since rare class sampling is adopted). It's worth noting that the attention layers used for the second stage depend on the number of patches in the image, rendering the number of parameters correlated with the input image size (since patch size depends on the backbone with pre-trained weights). It is adapted to be compatible with all the backbones considered in our framework.

- **HQHSAM Head**: We have developed a fusion of two cutting-edge decoder designs by incorporating the HQ token and utilizing features from different depths of the backbone into the double-staged decoding architecture of the HSAM projection head. The output from the HQSAM block is fed into the secondary decoder block of HSAM. The resulting decoder diagram, shown in Figure 3.3, merges the designs of HQSAM [17] and HSAM [6] according to our implementation.

Figure 3.3: HQHSAM Decoder



*Yellow section indicates the HQSAM decoder part integrated into the double staged architecture of HSAM decoder*

### 3.2.3 Fine-Tuning Strategies

Given the size of the foundation models, especially the larger ones, full fine-tuning was not practical. It was even demonstrated in [32] to lag behind in performance compared to state-of-the-art fine-tuning techniques such as Rein [32]. Accordingly, we have implemented the following fine-tuning strategies to be used with the backbones considered for our framework.

- **Ladder Fine Tuning** [5]: This technique is implemented to accept pairs of supported backbones, typically involving a larger one with frozen parameters and a smaller one that is fine-tuned. The output of the smaller model is summed with the output of the larger model, enabling effective fine-tuning. Convolutional layers are used to equate the output dimension of the smaller backbone to the larger one, and the sizes are matched by means of interpolation.
- **Rein/Rein-LoRA** [32]: Techniques are implemented for efficient adaptation of vision foundation models. A set of learnable tokens are employed to refine and forward feature maps block by block within the backbone. Rein-LoRA optimizes this by incorporating low-rank adaptation techniques to reduce the number of parameters further, enhancing scalability and efficiency in the fine-tuning process, which can be enabled by setting a single flag in the code.

## 3.3 Benchmark Models

Two well-established segmentation models for medical imagery applications from the previous chapter, namely Vanilla U-Net [26] and Swin U-Net [4], are implemented in our framework. The motivation was to include purpose-built convolutional and transformer-based benchmarks to ensure a robust evaluation of performance. This approach maintains a common training, validation, and testing pipeline across all evaluated models, providing a fair and comprehensive comparison. Weights for these benchmarks are always randomly initialized, and the models are trained from scratch for each dataset since they are relatively easy to train with 31 and 27 million parameters, respectively. Moreover, the aforementioned encoder/decoder model types can be used with a pre-trained ResNet encoder and U-Net-type decoder for additional pre-trained comparison.

## 3.4 Domain Generalization and Adaptation

Domain generalization and adaptation are expected to perform very well using foundation models as backbones due to their pre-trained capabilities on large and diverse datasets, which equip them with robust feature extraction skills. These models can generalize across different domains effectively, leveraging their extensive pre-training to adapt to new, unseen data with minimal fine-tuning. Additionally, their self-supervised nature allows them to learn rich, transferable features without relying on labeled data, further enhancing their adaptability and performance in varied and complex scenarios.

Domain generalization performance is evaluated by training on one dataset and testing the model on all other available datasets for the same anatomy (with matching labels). This process is automated in our framework.

To test domain adaptation, we have incorporated two fundamental methods into our framework: entropy minimization [31] and self-training [1] which are unsupervised and semi-supervised techniques respectively. Only the projection head is tuned for target domain adaptation, with all the remaining parameters frozen.

- **Entropy Minimization**: The model is initially trained on the source domain dataset as usual. Entropy minimization is then adapted by loading these weights and training on the whole target domain dataset with the following self-supervised loss function, which only involves the predictions on the target domain, for a few epochs.

$$H(\hat{y}) = -\sum_{c} p(\hat{y}_c) \log p(\hat{y}_c) \tag{3.1}$$

- **Self-Training**: For self-training, similar to entropy minimization, source domain trained weights are loaded into the model. Self-training is implemented in a parameterized manner that allows choosing an arbitrary ratio of labeled data to use and setting the confidence threshold to mask pseudo-labels whose confidence is below the limit, which is computed by applying the softmax function on the predicted logits. Additionally, the updating frequency of the pseudo-labels is also parameterized. Other than these changes and freezing all backbone and fine-tuning parameters, training is carried out in the same manner as standard training for a fraction of the epochs.

## 3.5 Metrics

Complying with the literature, the Dice Similarity Coefficient (DSC) is computed over the volume for validation and testing. Additionally, the mean Intersection over Union (mIoU) can also be calculated in the same manner for further comparison. All metrics are computed for both the source and test domains, along with losses. Sample segmentation outputs and their corresponding input images are logged within our framework, enabling easy evaluation of the results both during training and post-evaluation across different runs.

## 3.6 Tools and Environment

All implementations were done in PyTorch, and the trainer is based on PyTorch-Lightning. Model checkpoints with the highest validation DSC over the volume, as well as those with the lowest loss, are saved. Logging is managed through Weights & Biases (WandB), enabling interactive access via the web interface.

# Chapter 4

# Experiments and Results

## 4.1 Datasets

Anatomies listed below, which include multiple datasets, are considered for our experiments. Each dataset is provided in either NIfTI, HDF5, or PNG format, all of which are supported by our framework.

- **Brain**: For brain segmentation, we utilize images from two publicly accessible datasets: the Human Connectome Project (HCP) [29] and the Autism Brain Imaging Data Exchange (ABIDE) [22]. The HCP dataset provides both T1-weighted (T1w) and T2-weighted (T2w) images for each participant, whereas the ABIDE dataset comprises T1-weighted (T1w) images collected from various imaging sites. Despite offering extensive imaging data, these datasets lack manual segmentation labels. Additionally, there are no publicly available large-scale brain MRI datasets with manual segmentations of multiple subcortical structures. To address this, we utilize the widely adopted FreeSurfer tool [9] to create pseudo ground truth segmentations for the HCP and ABIDE datasets closely following [16]. FreeSurfer is a reliable segmentation tool that performs well across various scanners and protocols. However, it is highly time-consuming, requiring up to 10 hours on a CPU to segment a single 3D MR image, and it is specifically designed for brain imaging. As a result, we obtain 15 labels for brain MRI in the HCP-T1w, HCP-T2w, ABIDE-Caltech-T1w, and ABIDE-Stanford-T2w datasets, namely: background, cerebellum gray matter, cerebellum white matter, cerebral gray matter, cerebral white matter, thalamus, hippocampus, amygdala, ventricles, caudate, putamen, pallidum, ventral DC, CSF, and brain stem. All datasets are saved in HDF5 format.

- **Prostate**: This study utilizes data from the National Cancer Institute (NCI) [3] and a private dataset acquired from the University Hospital of Zurich (USZ) [2]. Each slice in these datasets includes expert annotations for three labels: background, central gland (CG), and peripheral zone (PZ). The data is in HDF5 format.

- **Lumbar Spine**: For lumbar spine segmentation, we utilize images from the publicly accessible VerSe (Vertebral Segmentation) [27] and MrSegV (Magnetic Resonance Spine Segmentation) [24] datasets. The VerSe dataset consists of CT images, while the MrSegV dataset consists of MRI images. Both include labels for six classes. Pre-processed images available for our study are in PNG format, with a constant volume depth of 120 and 12 slices per volume for VerSe and MrSegV datasets respectively.

- **Brain Tumor**: For brain tumor segmentation, we utilize the BraTS dataset [23], which includes T1-weighted (T1) and Fluid-Attenuated Inversion Recovery (FLAIR) modalities. The dataset comprises 285 subjects with annotations for three labels: background, enhancing tumor, and tumor core. The images are available in NIfTI format, providing detailed volumetric data for analysis.

Dataset details for each anatomy, including the number of volumes, labels, the imaging modality, and data format, are summarized in Table 4.1. As some datasets (indicated in Table 4.1) are used during training for MedSam, they are excluded from some of the experiments when using the MedSam backbone.

Table 4.1: Dataset Details

| Anatomy | Dataset | Modality | Format | N_train | N_val | N_test | N_labels |
|---|---|---|---|---|---|---|---|
| Brain | HCP-T1 | MRI | HDF5 | 20 | 5 | 20 | 15 |
| | HCP-T2 | | | 20 | 5 | 20 | 15 |
| | ABIDE-C-T1 | | | 10 | 5 | 20 | 15 |
| | ABIDE-S-T1 | | | 10 | 5 | 20 | 15 |
| Prostate | NCI* | MRI | HDF5 | 10 | 5 | 15 | 3 |
| | USZ | | | 28 | 20 | 20 | 3 |
| Lumbar Spine | VerSe | CT | PNG | 116 | 1 | 15 | 6 |
| | MRSegV | MRI-T1 | | 162 | 20 | 20 | 6 |
| Brain Tumor | BraTS-T1* | MRI | NIfTI | 198 | 30 | 57 | 3 |
| | BraTS-FLAIR* | | | 198 | 30 | 57 | 3 |

*(*) Datasets used for MedSam[21] training*

## 4.2 Implementation Details

### 4.2.1 Pre-processing

On most part, our pre-processings are based on [16]. Datasets are processed according to the following pipeline: Initially, N4 bias field correction [28] is performed for all MRI datasets, as it reduces intensity inhomogeneity, thereby improving the reliability of segmentation across different datasets. Then, volumes are normalized to the 1st and 99th percentiles according to $x_{\text{normalized}} = (x - x_p^1)/(x_p^{99} - x_p^1)$ followed by clipping, except for BraTS dataset, for which Z-normalization $x_{\text{normalized}} = (x - \mu)/\sigma$ and min-max normalization $x_{\text{normalized}} = (x - x_{\text{min}})/(x_{\text{max}} - x_{\text{min}})$ are applied consecutively resulting intensities in the range [0, 1].

For the brain and brain tumor datasets, the next step involves skull stripping, which sets the intensities of all non-brain voxels to zero.

We train the models in 2D given the 2D structure of the foundation models. Accordingly, for brain, prostate, and lumbar spine datasets, slices are adjusted to a uniform pixel size in the in-plane dimensions, then cropped and/or padded with zeros to achieve a consistent image size for each anatomical category. The standard pixel sizes for the brain, prostate, and lumbar spine datasets are 0.7mm$^2$, 0.625mm$^2$, and 0.7mm$^2$ respectively, while the fixed image size is 256x256. Pre-processed datasets for the above listed anatomies are provided by ETH Computer Vision Laboratory. On the other hand, for BraTS brain tumor datasets, slices and labels are simply interpolated to the fixed size of 224x224 following order 1 and order 0 interpolation without preserving equal voxel spacing for different volumes. The difference in the last step for BraTS is due to the fact that it was quickly added towards the very end with time constraints, although maximum difference in the dataset among volumes is small, a few pixels at most.

Finally, during training, slices from all datasets are bilinearly interpolated to a size of 224x224, which is compatible with both the 14 and 16 patch dimensions used for Dino and the other foundation backbones, respectively. This is followed by normalization using the values specific to each pre-trained backbone. Predictions are then interpolated back to the original size of the dataset for loss computation and evaluation.

### 4.2.2 Data Augmentations

The following random data augmentations are used during training to improve model robustness and generalization to unseen data. These augmentations simulate various inconsistencies and variations in medical images, allowing the model to learn more robust features and perform well across different clinical settings and imaging conditions.

- **PhotoMetricDistortion**: This augmentation addresses variations in imaging conditions, equipment, and protocols by applying random adjustments to brightness, contrast, saturation, and hue. Each transformation is applied sequentially with a probability of 0.5.
- **Elastic Transformation**: This augmentation simulates anatomical variations and slight movements during image acquisition by introducing random, localized deformations to the images. Elastic transformations are applied with a probability of 0.25.
- **Structural Augmentation**: This augmentation accounts for structural variations in patient anatomy and imaging setups by applying rotations, translations, and scaling. Structural augmentations are applied with a probability of 0.25, introducing controlled alterations to the structure of the images.

### 4.2.3 Optimization Strategy

Models are trained with the AdamW optimizer [20], using a weight decay of 1e-5 and default parameters for everything else. A linearly decaying learning rate is used, with a linear warm-up for 5% of the total number of epochs. The initial learning rate is either 1e-5, 2e-5, or 5e-5, chosen depending on the backbone and dataset for stability. By default, a learning rate of 5e-5 is used, except when the Dino backbone is applied to the HCP and VerSe datasets, for which the learning rate is set to 1e-5. For the rest of the datasets, the learning rate is 2e-5 for the Dino backbone. Additionally, a learning rate of 2e-5 is used when Reins fine-tuning is applied to any of the backbones.

The BraTS and VerSe datasets are trained for 80 and 100 epochs, respectively, while the rest of the models are trained for 120 epochs. Cross-entropy loss is used for all experiments except those with entropy minimization runs for domain generalization. Training is performed using either a single NVIDIA TITAN X or NVIDIA A6000 GPU.

### 4.2.4 Evaluation Metric

Checkpoints that correspond to the best validation Dice scores (DSC) are saved and used for testing. DSCs, calculated on the test sets and averaged with equal weight for all classes excluding the background (label 0), are reported in all the following tables in this study. Each model configuration is trained for a single run.

## 4.3 Experiments

### 4.3.1 Decode Head Selection

The first experiment we conducted aimed to test different decoder heads with DinoV2 [25], SAM [18], MedSAM [21], and MAE [11] backbones of the base size on several datasets. For SAM and MedSAM, the convolutional neck used after the last SAM ViT block [8] to reduce the channel dimension of the final output is not used in order to maintain the same output shape for all tested decode heads and to provide the richest information to the decoders for all backbones for a fair comparison. Additionally, some decoders require all output shapes of the backbone to be equal, thus we decided not to use SAM's neck for this experiment.

We've evaluated 9 different decode head architectures and 12 total configurations.

- The following 4 decoders use 4 concatenated output features from the last blocks of the backbones.
    - Linear
    - Segformer (SegFor) [33]
    - Dual Attention (DA) [10]: 2 different dimensions for channel and positional attentions, regular with 384 and small (DA S.) with 120 respectively
    - ResNet (RN) [12]: Reduces the dimension of each output patch feature to 384 before feeding in to the residual blocks.
- The U-Net decoder [26] combines 10 output features from the backbone to upscale the size and downscale the dimension 4 times, as described in the previous chapter, resulting in a factor of 16. Similar to the Resnet decoder, the initial convolutional layer handles reducing the dimensionality of the patch features. Two variants are tested: a regular version with a size of 576 and a smaller variant with 240 dimensions (Unet S.).
- The rest of the projection heads are based on SAM's prompt encoder and mask decoder transformer architecture. The prompt encoder is always initialized with SAM's weights, as it is independent of patch feature dimensions. The neck for 256 size reduction is initialized with matching SAM weights if possible (for backbone sizes of base, large, and huge), and randomly initialized otherwise (for small and giant-sized backbones).
    - Sam Prompt Encoder and Mask Decoder [18]: Tested for both when the prompt encoder is trained (S-PE-MD) and is set to freeze (S-FPE-MD).
    - HSAM [6]
    - HQSAM [17]: The output of the last ViT block is combined with the output after the first global attention block for SAM, while the output of the first block is used for the rest of the backbones for global-local fusion.
    - HQHSAM

We evaluated both in-domain and domain generalization performance to determine the optimal decode head for the rest of the experiments. For this purpose, the pre-trained backbones were set to freeze, and only the parameters of the decode head were updated. Brain, prostate, and lumbar spine anatomies were used for this experiment. It's worth noting that generalization tested on the NCI dataset is not taken into account for MedSam as it was used during the training of the backbone. The ABIDE Caltech, USZ, and MrSegV datasets served as the source domain for training, and domain generalization was tested for each anatomy on all its remaining datasets. In-domain or domain-specific performance is abbreviated as DS, whereas DG stands for domain generalization. These abbreviations will be used throughout the rest of the sections. Additionally, Dice scores for all datasets from each anatomy are averaged and abbreviated as Overall (OV).

We used the average DS, DG, and OV scores across all backbones (Dino, SAM, MedSAM, and MAE) for each dataset, which are then averaged for each anatomy. Finally, all anatomies are averaged again for the projection head selection for a clear comparison of the decode heads.

The in-domain, domain generalization, and overall performance of the decode heads for the aforementioned anatomies used in this experiment are available in Table 4.2. Decode heads are ranked based on their overall performance, with their differences compared to the best for each category indicated in percentages. The top 3 values are marked in bold for the last 4 columns. Table 4.2 also indicates the number of trainable parameters for each decode head (the number of parameters is for the brain anatomy and might be slightly different for other anatomies depending on the number of labels and patches). Detailed tables showing every run of this experiment as well as complementary summary tables for each individual backbone can be found in Appendix A.1.

Table 4.2: Decode Head Comparisons

| Rank | Decoder | #Trainable Param* | OV Mean | OV diff % | DS Mean | DS diff % | DG Mean | DG diff % |
|---|---|---|---|---|---|---|---|---|
| 1 | HQHSAM | 17.32M | 65.28 | 0.00 | **84.98** | **0.00** | **52.63** | **0.00** |
| 2 | HQSAM | 7.61M | 64.95 | -0.50 | **84.88** | **-0.12** | **52.18** | **-0.87** |
| 3 | HSAM | 15.98M | 63.94 | -2.05 | **83.39** | **-1.87** | 51.41 | -2.32 |
| 4 | S-PE-MD | 5.60M | 63.38 | -2.90 | 81.93 | -3.60 | **51.45** | **-2.25** |
| 5 | S-FPE-MD | 5.61M | 63.13 | -3.28 | 82.06 | -3.44 | 51.00 | -3.11 |
| 6 | Unet | 39.10M | 60.87 | -6.75 | 79.10 | -6.92 | 49.20 | -6.52 |
| 7 | DA | 7.87M | 59.63 | -8.66 | 76.98 | -9.41 | 48.51 | -7.83 |
| 8 | Unet S. | 39.21M | 59.48 | -8.88 | 78.10 | -8.10 | 47.61 | -9.54 |
| 9 | DA S. | 6.94M | 59.19 | -9.32 | 77.68 | -8.59 | 47.26 | -10.20 |
| 10 | Resnet | 24.35M | 56.88 | -12.86 | 77.88 | -8.36 | 43.60 | -17.16 |
| 11 | SegFor | 4.74M | 52.12 | -20.15 | 68.55 | -19.33 | 41.65 | -20.88 |
| 12 | Linear | 52.24K | 45.87 | -29.73 | 61.09 | -28.11 | 36.10 | -31.42 |

*(*): Shown for brain datasets using a patch size of 16*

Decode heads based on SAM's prompt encoder and mask decoder architecture demonstrate clear superiority across all evaluated categories. Further performance improvements are observed with HSAM [6] and HQSAM [17]. On average, our fusion of these two, namely HQHSAM, outperforms all decoders in every category. Based on this evaluation, we have concluded to use **HQHSAM** as the projection head for foundation model backbones in the rest of the experiments.

### 4.3.2 Fine Tuning Selection for the Backbones

The following experiment was conducted to determine the optimal fine-tuning strategy for each backbone. Similar to the previous experiment, brain, prostate, and lumbar spine anatomies were considered. For this and the subsequent experiments, one dataset from each anatomy is designated as the source domain, while the remaining datasets serve as the target domains. This approach is applied to all combinations, covering all possible scenarios. Another point, also mentioned in the previous sections, is that since the NCI [3] dataset was used for training MedSam [21], accordingly results for prostate anatomy are not available for the MedSam backbone in the rest of the experiments.

The DinoV2 [25], SAM [18], MedSam [21], and MAE [11] backbones were trained using the base sizes. As in the previous experiment, the convolutional neck available for the SAM and MedSAM backbones, applied to the output of the last ViT block, is skipped in the backbone since it's applied inside HQHSAM decoder as necessary for each backbone size.

The exception being when Ladder fine-tuning is used, where the Dino small size is employed as the tunable branch for the parallel backbones, while the other branch remains frozen, abbreviated as LadderD. The aforementioned final neck available for SAM and MedSam is used for the final output of the backbone to reduce the size of the ladder fine tunings neck, necessary for summation between the tuning branch (Dino small with a 386 output dimension) and the tuned model (SAM or MedSam with 768 dimensions before the neck and 256 after). This approach keeps the pre-trained neck weights of SAM and MedSam frozen, respecting the original implementation of Ladder fine-tuning for SAM [5] while training it (initialized with Sam's weights) on the decoder side for Dino and MAE, for which no such neck was trained originally.

For all fine tuning techniques, the backbone weights are set to freeze, and only the fine-tuning and decoder parameters are trained. DS, DG, and OV Dice scores across datasets and anatomies are averaged for each fine tuning technique and are presented in Table 4.3.2 for each backbone. Detailed tables, including each run as well as additional summary tables examining each anatomy individually, are available in the Appendix A.2.

Table 4.3: Fine Tuning Comparison for Foundation Backbones

| FT | Dino | | | SAM | | | MedSam* | | | MAE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DS | DG | OV | DS | DG | OV | DS | DG | OV | DS | DG | OV |
| Freeze | 80.72 | 41.77 | 58.36 | 83.04 | 37.96 | 57.54 | 85.55 | 27.34 | 50.95 | 83.58 | **42.01** | **59.61** |
| Reins LoRA | 82.08 | **44.44** | **60.53** | 82.93 | 38.45 | 57.69 | 85.61 | **28.85** | 51.92 | 82.63 | 40.63 | 58.48 |
| Reins | 81.83 | 44.42 | 60.41 | 82.68 | **39.00** | **57.84** | 85.64 | 26.98 | 50.82 | 82.59 | 39.06 | 57.64 |
| LadderD | **84.95** | 33.53 | 55.82 | **84.48** | 32.70 | 55.12 | **87.07** | 28.82 | **52.71** | **83.92** | 36.12 | 56.67 |

*(*):Including only brain and lumbar spine datasets*

Considering the nature of foundation models and their proven generalization performance, we aim to achieve good generalization and domain adaptation with our foundation backbone-based models. With this goal in mind, the fine-tuning strategy that yields the best overall performance (OV) has been chosen for each foundation backbone individually. Specifically, Reins LoRA [14], Rein, LadderD and No fine-tuning (Freeze) have been selected for Dino, SAM, MedSam and MAE, respectively (see Table 4.4).

It's worth noting that LadderD fine-tuning consistently achieved the best in-domain average Dice scores among backbones.

Table 4.4: Selected Fine-Tuning Strategies for Foundation Backbones

| **Backbone** | **Fine Tune** |
|---|---|
| Dino | Reins LoRA |
| SAM | Reins |
| MedSam | LadderD |
| MAE | Freeze |

### 4.3.3 In-Domain and Domain Generalization Performance Comparisons

Generalization performance was assessed using the previously defined HQHSAM decoder and fine-tuning strategies as outlined in Table 4.4. This evaluation covered all backbone sizes available for Dino, SAM, MedSam, and MAE, applied to brain, prostate, lumbar spine, and brain tumor datasets. Notably, brain tumor (BraTS datasets) was not included in earlier experiments. Similar to the NCI prostate dataset, the BraTS dataset was used for training MedSam. Consequently, the MedSam backbone was only trained on brain and lumbar spine datasets in the subsequent experiments.

Additionally, three benchmarks were introduced for comparison: Resnet101 encoder (pre-trained) [12] with U-Net [26] type expanding path, Vanilla U-Net, and Swin U-Net models (last 2 are trained from scratch as they're small). Averaged results for DS, DG, and OV Dice scores for brain and lumbar spine datasets are illustrated in Table 4.5, while prostate and brain tumor datasets are depicted in Table 4.6. As before, top 3 results on each column are bold. Furthermore, average results of all anatomies are available in Table 4.7 excluding MedSam as not all datasets are available.

Table 4.5: Brain and Lumbar Spine Datasets In-Domain and Domain Generalization Comparisons

| Fine Tuning | Backbone | Brain Datasets Summary | | | | Lumbar Spine Datasets Summary | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | #Train Par. [M] | DS | DG | OV | #Train Par. [M] | DS | DG | OV |
| Reins LoRA | Dino-B | 19.50 | 85.05 | **52.37** | **60.54** | 15.80 | 85.27 | 34.51 | 59.89 |
| Reins | SAM-B | 19.80 | 85.45 | 49.50 | 58.49 | 16.08 | 85.49 | 29.21 | 57.35 |
| LadderD | MedSam-B | 38.98 | 86.04 | 44.22 | 54.68 | 35.30 | **88.10** | 13.41 | 50.75 |
| Freeze | MAE-B | 17.32 | 85.54 | 47.25 | 56.82 | 13.60 | 86.79 | 36.17 | 61.48 |
| Reins LoRA | Dino-L | 21.00 | 84.61 | **54.09** | **61.72** | 17.40 | 85.06 | 31.30 | 58.18 |
| Reins | SAM-L | 22.70 | 85.50 | 51.78 | 60.21 | 19.00 | 86.95 | 38.03 | 62.49 |
| Freeze | MAE-L | 17.60 | 85.69 | 47.20 | 56.82 | 14.00 | 85.97 | **48.45** | **67.21** |
| Reins LoRA | Dino-G | 25.10 | 85.13 | **54.68** | **62.29** | 21.40 | 84.14 | **40.90** | **62.52** |
| Reins | SAM-H | 25.90 | 85.36 | 51.70 | 60.11 | 22.20 | 86.96 | 33.12 | 60.04 |
| Freeze | MAE-H | 18.40 | **86.14** | 48.55 | 57.94 | 14.80 | 85.13 | **42.15** | **63.64** |
| Full Fine Tune | Resnet+Unet | 68.20 | 86.07 | 43.99 | 54.51 | 68.20 | **87.89** | 2.35 | 45.12 |
| _ | Vanilla Unet | 31.00 | **88.47** | 44.96 | 55.84 | 31.00 | **87.02** | 1.22 | 44.12 |
| _ | SwinUNet | 27.20 | **86.71** | 41.68 | 52.94 | 27.20 | 86.79 | 5.63 | 46.21 |

Table 4.6: Prostate and Brain Tumor Datasets In-Domain and Domain Generalization Comparisons

| Fine Tuning | Backbone | Prostate Datasets | | | | Brain Tumor Datasets | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | #Train Par. [M] | DS | DG | OV | #Train Par. [M] | DS | DG | OV |
| Reins LoRA | Dino-B | 14.60 | 75.91 | **46.44** | **61.17** | 14.60 | 86.60 | 44.29 | 65.44 |
| Reins | SAM-B | 14.90 | 77.09 | 38.29 | 57.69 | 14.90 | 86.76 | **44.94** | 65.85 |
| Freeze | MAE-B | 12.42 | **78.41** | 42.63 | **60.52** | 12.40 | **87.30** | 44.24 | 65.77 |
| Reins LoRA | Dino-L | 16.20 | 72.95 | 42.74 | 57.84 | 16.20 | 87.13 | **50.84** | **68.98** |
| Reins | SAM-L | 17.80 | 76.07 | 39.01 | 57.54 | 17.80 | 86.78 | 44.31 | 65.55 |
| Freeze | MAE-L | 12.70 | **77.82** | **42.77** | 60.29 | 12.70 | **87.60** | 43.71 | 65.65 |
| Reins LoRA | Dino-G | 20.20 | 74.62 | 40.37 | 57.49 | 20.20 | 87.15 | 41.83 | 64.49 |
| Reins | SAM-H | 21.00 | 74.78 | 38.80 | 56.79 | 21.00 | 87.24 | **47.26** | **67.25** |
| Freeze | MAE-H | 13.50 | **79.53** | **45.95** | **62.74** | 13.50 | **87.54** | 44.44 | **65.99** |
| Full Fine Tune | RN101+Unet | 68.20 | 69.06 | 31.46 | 50.26 | 68.20 | 85.71 | 31.48 | 58.60 |
| _ | Vanilla Unet | 31.00 | 69.10 | 13.41 | 41.26 | 31.00 | 86.17 | 37.32 | 61.74 |
| _ | SwinUNet | 27.20 | 65.51 | 29.90 | 47.70 | 27.20 | 85.66 | 40.69 | 63.17 |

Table 4.7: Average In-Domain and Domain Generalization Comparisons

| Backbone | Fine Tuning | BB Sz | DS | DG | OV |
|---|---|---|---|---|---|
| Dino | Reins LoRA | Base | 83.21 | 44.40 | **61.76** |
| | | Large | 82.44 | **44.74** | 61.68 |
| | | Giant | 82.76 | 44.44 | 61.70 |
| SAM | Reins | Base | 83.70 | 40.48 | 59.84 |
| | | Large | 83.82 | 43.28 | 61.45 |
| | | Huge | 83.58 | 42.72 | 61.05 |
| MAE | Freeze | Base | **84.51** | 42.57 | 61.15 |
| | | Large | **84.27** | **45.53** | **62.49** |
| | | Huge | **84.58** | **45.27** | **62.58** |
| Resnet+Unet | Full Fine Tune | 101 | 82.18 | 27.32 | 52.12 |
| Vanilla Unet | #N/A | #N/A | 82.69 | 24.23 | 50.74 |
| SwinUNet | #N/A | #N/A | 81.17 | 29.47 | 52.50 |

*MedSam was excluded from this analysis as complete results for this backbone are not available.*

Overall, our models utilizing foundation model backbones demonstrate significantly superior generalization across all datasets. In terms of in-domain performance, our models consistently surpass the benchmarks for prostate and brain tumor datasets. However, they fall slightly short compared to the top-performing benchmarks for brain datasets, trailing by approximately 2.5 Dice score points, and for lumbar spine datasets, with the exception of MedSam, which achieves the highest overall DSC on lumbar spine datasets.

Comparing the foundation model backbones of different sizes for Dino, SAM, and MAE, as MedSam is only available in the base size, it can be observed that, on average, DG Dice scores were generally the highest with the large size. On the other hand, not much of a difference was observed for DS Dice scores among sizes. Considering that moving to Huge/Giant sizes does not offer a considerable benefit, and in most cases worsens the performance except for MAE, we have chosen to focus on the base and large sizes for the remainder of the experiments.

Anatomy-specific summaries, as well as tables detailing every run individually, are available in the Appendix A.3.

### 4.3.4 Domain Adaptation

The second key aspect we aimed to test was the domain adaptation capability of the foundation backbone models. To achieve this, we first experimented with entropy minimization. In this approach, we took the pre-trained model on the source domain and updated the trainable weights of the decoder while keeping the backbone and fine-tuning frozen for a few epochs in an unsupervised fashion using the entropy minimization objective [31] on the whole target domain dataset. After conducting trials with various backbones, datasets, and benchmarks, we concluded that this technique was not effective for our models, as they quickly overfitted. As a result, we shifted our focus to self-training [1] for semi-supervised domain adaptation.

#### Self Training

Self-training was tested with the following settings: Three labeled volumes were used from each dataset, and during training, these were concatenated with pseudo labels obtained through inference. Using fewer than three labeled volumes resulted in overfitting in most cases, so we opted for three. Pseudo labels, updated at every iteration, with a confidence (computed with softmax) below 90% were filtered out and not considered for backpropagation. Models were initialized with pre-trained weights from the source domain, with the backbone and fine-tuning parameters frozen, while only the decoder parameters were updated, identical to the entropy minimization trials.

Results for brain and lumbar spine, as well as prostate and brain tumor, are shown with respective DS performances and loss percentages for self-training compared to DS in separate columns in Tables 4.8 and 4.9. The highest value in each column is indicated with bold numbers as before. Supplementary tables showing all runs for the experiment individually are available in Appendix A.4.

Table 4.8: Brain and Lumbar Spine Datasets Self-Training Domain Adaptation

| Fine Tuning | Backbone | Brain Datasets | | | | Lumbar Spine Datasets | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | #Train Par. [M] | DS | Domain Shift | Dom Shift Loss % | #Train Par. [M] | DS | Domain Shift | Dom Shift Loss % |
| Reins LoRA | Dino-B | 17.80 | 85.05 | 80.28 | -5.61 | 14.10 | 85.27 | 38.31 | -55.08 |
| Reins | SAM-B | 17.30 | 85.45 | 80.62 | -5.66 | 13.60 | 85.49 | 38.93 | -54.47 |
| LadderD | MedSam-B | 16.50 | 86.04 | 74.62 | -13.28 | 12.90 | **88.10** | 36.97 | -58.04 |
| Freeze | Mae-B | 17.30 | 85.54 | 80.67 | -5.69 | 13.60 | 86.79 | **40.20** | **-53.68** |
| Reins LoRA | Dino-L | 18.10 | 84.61 | 80.14 | **-5.29** | 14.40 | 85.06 | 37.01 | -56.49 |
| Reins | SAM-L | 17.60 | 85.50 | **80.93** | **-5.34** | 14.00 | 86.95 | **39.55** | -54.52 |
| Freeze | Mae-L | 17.60 | 85.69 | 80.84 | -5.66 | 14.00 | 85.97 | 39.19 | **-54.41** |
| Full Fine Tune | Resnet+Unet | 25.70 | **86.07** | 78.41 | -8.90 | 25.70 | **87.89** | 26.08 | -70.33 |
| - | Vanilla Unet | 31.00 | **88.47** | **85.93** | **-2.86** | 31.00 | **87.02** | 32.31 | -62.88 |
| - | SwinUNet | 27.20 | **86.71** | **81.53** | -5.97 | 27.20 | 86.79 | **41.90** | **-51.73** |

Table 4.9: Prostate and Brain Tumor Self-Training Domain Adaptation

| Fine Tuning | Backbone | Prostate Datasets | | | | Brain Tumor Datasets | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | #Train Par. [M] | DS | Domain Shift | Dom Shift Loss % | #Train Par. [M] | DS | Domain Shift | Dom Shift Loss % |
| Reins LoRA | Dino-B | 12.90 | 75.91 | **64.40** | **-15.16** | 12.90 | 86.60 | **69.25** | **-20.03** |
| Reins | SAM-B | 12.40 | **77.09** | 62.83 | -18.49 | 12.40 | 86.76 | 55.81 | -35.67 |
| Freeze | Mae-B | 12.40 | **78.41** | **65.38** | -16.62 | 12.40 | **87.30** | **62.14** | **-28.83** |
| Reins LoRA | Dino-L | 13.20 | 72.95 | 62.98 | **-13.67** | 13.20 | **87.13** | 57.48 | -34.03 |
| Reins | SAM-L | 12.70 | 76.07 | 62.78 | -17.47 | 12.70 | 86.78 | 57.33 | -33.94 |
| Freeze | Mae-L | 12.70 | **77.82** | 64.38 | -17.27 | 12.70 | **87.60** | **68.39** | **-21.93** |
| Full Fine Tune | RN101+Unet | 25.70 | 69.06 | 57.02 | -17.43 | 25.70 | 85.71 | 55.00 | -35.83 |
| - | Vanilla Unet | 31.00 | 69.10 | **65.22** | **-5.62** | 31.00 | 86.17 | 60.48 | -29.81 |
| - | SwinUNet | 27.20 | 65.51 | 45.05 | -31.23 | 27.20 | 85.66 | 59.88 | -30.10 |

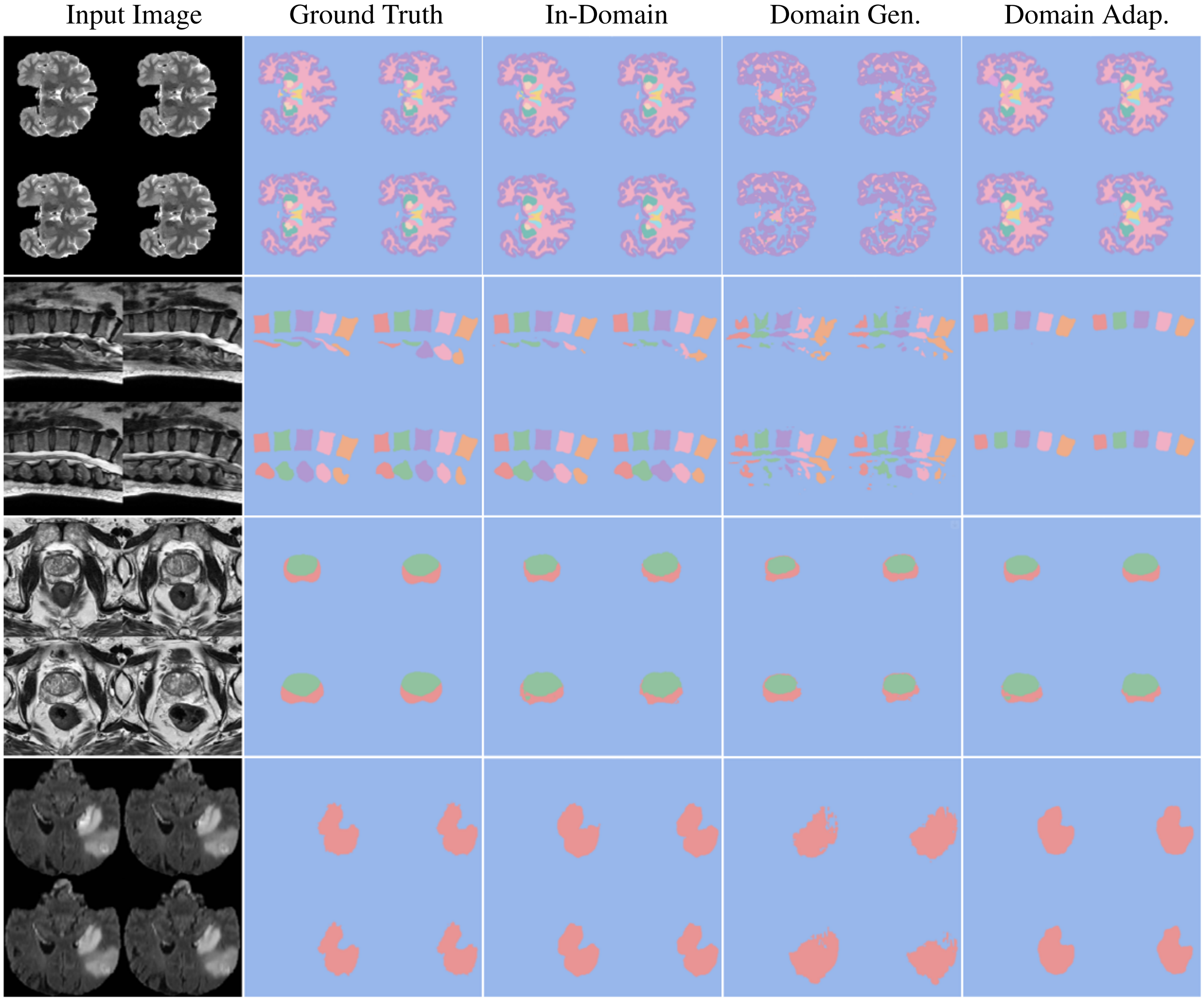
While our models offer better domain adaptation performance for brain tumor and prostate datasets, the best benchmark achieves better results for brain and lumbar spine datasets. The difference is marginal for lumbar spine and prostate datasets, although it is significant for brain and brain tumor datasets. Overall, while most foundation models offered comparable performance, Dino and MAE had a slight edge, with a significant difference observed for brain tumor datasets.

Backbone size does not seem to have a direct impact on performance for most cases, even worsening the results in some instances while improving them in others depending on the backbone and dataset. More sophisticated domain adaptation techniques, such as the DA Former framework [13], require further testing.

### 4.3.5 Visualization of Segmentations

Randomly chosen segmentation results for some of our models are depicted in Figure 4.1. One dataset from each anatomy is illustrated: HCP2, MRSegV, NCI, and BraTS-FLAIR for brain, lumbar spine, prostate, and brain tumor anatomies, respectively. Segmentation masks are obtained using MAE-base, SAM-large, MAE-base, and Dino-base models corresponding to each dataset with the chosen decode head and fine-tunings from previous experiments. Each row shows the input image, ground truth mask, DS mask, DG mask, and the mask obtained after self-training adaptation. The latter two masks, DG and self-training, are obtained from a model with the same architecture but trained on a different dataset for the same anatomy. The DG mask is obtained without any adaptation training, while the self-training mask is obtained after adaptation training. The HCP1 dataset is used as the source domain for the brain, while the remaining datasets are used for the other anatomies for the latter two masks.

Figure 4.1: Sample Segmentation Results For Each Anatomy



*Columns from left to right are Input Image, Ground Truth, DS, DG, and Self-Training segmentation masks in order. Rows from top to bottom correspond to the HCP2, MRSegV, NCI, and BraTS-FLAIR datasets.*

# Chapter 5

# Discussion

We have demonstrated that SAM's [18] prompt encoder and mask decoder-based projection head, with the best performance achieved by our custom fusion of two state-of-the-art techniques, namely HSAM [6] and HQSAM [17], which we have named HQHSAM, offers excellent performance for medical segmentation applications with vision foundation model (VFM) backbones.

Furthermore, we have shown the superior performance of VFMs for domain generalization across 10 datasets spanning four different anatomies compared to the tested benchmarks, including application-specific models such as Unet [26] and Swin Unet [4]. Additionally, our in-domain (DS) performance was better for all tested variants of VFMs on prostate and brain tumor datasets. While for the spine, although not all variants surpassed the best benchmark, most came very close, and the best score was achieved with one of our implementations. However, we were unable to surpass the best benchmarks for brain datasets.

Although semi-supervised domain adaptation using a relatively simple technique, self-training, offered significantly better performance on brain tumor datasets compared to the benchmarks, the effectiveness of our models was not significant for prostate datasets, and they lagged behind for lumbar spine and brain datasets by marginal and significant amounts, respectively. More sophisticated techniques, such as the DAFormer [13], could offer great potential.

Additionally, comparisons against parameter-optimized specialized models such as nnUnet [15] would be beneficial for a more detailed comparison and to better demonstrate the effectiveness of our models.

# Appendix A

# The First Appendix

## A.1 Decode Head Comparisons

Table A.1: Decoders Comparisons - All Runs

| DecHd | #Train Param. | Backb. | **MRS.*** | VerSe | **Ab.Ca*** | Ab.St | HCP1 | HCP2 | **USZ*** | NCI | D.S. | D.G. | Ov. | DS Avg | DG Avg | Ov. Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Linear | 52.24K | Dino | 76.38 | 30.46 | 70.88 | 59.87 | 62.29 | 36.90 | 40.40 | 15.70 | 62.55 | 41.04 | 49.11 | 61.09 | 36.10 | 45.87 |
| | | Sam | 77.10 | 21.72 | 69.62 | 50.48 | 54.04 | 39.42 | 41.83 | 34.89 | 62.85 | 40.11 | 48.64 | | | |
| | | M.Sam | 68.50 | 5.24 | 68.02 | 35.74 | 50.45 | 11.85 | 29.98 | – | 55.50 | 25.82 | 38.54 | | | |
| | | MAE | 75.38 | 24.86 | 66.85 | 49.77 | 51.45 | 27.26 | 48.17 | 33.70 | 63.47 | 37.41 | 47.18 | | | |
| SegFor | 4.74M | Dino | 82.80 | 38.50 | 75.63 | 64.51 | 68.22 | 37.31 | 45.32 | 14.59 | 67.92 | 44.63 | 53.36 | 68.55 | 41.65 | 52.12 |
| | | Sam | 83.71 | 28.57 | 76.16 | 63.33 | 62.84 | 33.99 | 52.98 | 29.85 | 70.95 | 43.72 | 53.93 | | | |
| | | M.Sam | 80.27 | 12.51 | 75.55 | 56.48 | 63.44 | 14.31 | 41.00 | – | 65.61 | 36.69 | 49.08 | | | |
| | | MAE | 82.34 | 40.47 | 73.71 | 60.78 | 65.42 | 21.06 | 53.16 | 20.05 | 69.74 | 41.56 | 52.12 | | | |
| DA | 24.35M | Dino | 85.05 | 45.02 | 78.77 | 70.57 | 72.58 | 46.44 | 64.95 | 45.21 | 76.26 | 55.96 | 63.57 | 76.98 | 48.51 | 59.63 |
| | | Sam | 84.63 | 32.79 | 77.98 | 69.47 | 68.93 | 44.06 | 72.96 | 23.80 | 78.52 | 47.81 | 59.33 | | | |
| | | M.Sam | 82.24 | 13.97 | 77.00 | 64.02 | 67.50 | 15.23 | 59.17 | – | 72.80 | 40.18 | 54.16 | | | |
| | | MAE | 83.51 | 38.56 | 75.37 | 65.70 | 68.40 | 29.14 | 82.16 | 48.71 | 80.35 | 50.10 | 61.44 | | | |
| DA S. | 6.94M | Dino | 84.79 | 44.06 | 77.31 | 68.49 | 71.36 | 39.17 | 58.78 | 33.43 | 73.63 | 51.30 | 59.67 | 77.68 | 47.26 | 59.19 |
| | | Sam | 84.26 | 21.75 | 77.24 | 67.34 | 67.91 | 44.53 | 80.95 | 41.20 | 80.82 | 48.55 | 60.65 | | | |
| | | M.Sam | 81.90 | 13.97 | 76.04 | 57.54 | 67.77 | 16.56 | 76.07 | – | 78.00 | 38.96 | 55.69 | | | |
| | | MAE | 83.19 | 36.52 | 73.93 | 62.81 | 66.20 | 31.92 | 77.74 | 53.80 | 78.29 | 50.25 | 60.76 | | | |
| S-FPE-MD | 5.60M | Dino | 84.64 | 44.58 | 78.12 | 60.26 | 72.04 | 46.48 | 77.88 | 45.69 | 80.21 | 53.81 | 63.71 | 82.06 | 51.00 | 63.13 |
| | | Sam | 86.99 | 31.07 | 85.05 | 58.13 | 77.03 | 62.85 | 76.38 | 47.81 | 82.81 | 55.38 | 65.66 | | | |
| | | M.Sam | 84.65 | 37.47 | 85.23 | 49.10 | 73.75 | 18.02 | 72.97 | – | 80.95 | 44.59 | 60.17 | | | |
| | | MAE | 87.10 | 35.72 | 84.19 | 68.94 | 77.99 | 17.40 | 81.52 | 51.03 | 84.27 | 50.22 | 62.99 | | | |
| S-PE-MD | 5.61M | Dino | 84.30 | 45.78 | 78.31 | 60.64 | 72.06 | 48.07 | 78.10 | 52.10 | 80.24 | 55.73 | 64.92 | 81.93 | 51.45 | 63.38 |
| | | Sam | 87.22 | 26.84 | 85.19 | 59.03 | 77.23 | 64.01 | 75.07 | 38.65 | 82.49 | 53.15 | 64.16 | | | |
| | | M.Sam | 84.65 | 34.89 | 85.17 | 44.68 | 74.79 | 21.81 | 75.94 | – | 81.92 | 44.04 | 60.28 | | | |
| | | MAE | 87.14 | 38.34 | 84.50 | 69.29 | 77.74 | 21.46 | 77.51 | 57.51 | 83.05 | 52.87 | 64.19 | | | |
| HQSAM | 7.61M | Dino | 86.02 | 36.12 | 84.68 | 69.30 | 77.00 | 27.37 | 76.74 | 56.52 | 82.48 | 53.26 | 64.22 | 84.88 | 52.18 | 64.95 |
| | | Sam | 87.58 | 34.68 | 85.66 | 73.86 | 78.59 | 29.79 | 85.73 | 53.17 | 86.32 | 54.02 | 66.13 | | | |
| | | M.Sam | 86.26 | 27.86 | 85.02 | 65.83 | 77.89 | 16.16 | 83.40 | – | 84.89 | 46.94 | 63.20 | | | |
| | | MAE | 87.52 | 35.34 | 85.02 | 72.10 | 77.77 | 25.56 | 84.91 | 61.72 | 85.82 | 54.50 | 66.24 | | | |
| HSAM | 15.98M | Dino | 83.84 | 42.30 | 77.06 | 58.48 | 71.01 | 42.40 | 77.12 | 58.80 | 79.34 | 54.60 | 63.88 | 83.39 | 51.41 | 63.94 |
| | | Sam | 87.19 | 38.09 | 84.64 | 70.60 | 78.00 | 30.17 | 85.06 | 50.45 | 85.63 | 53.46 | 65.53 | | | |
| | | M.Sam | 85.53 | 29.84 | 83.94 | 55.34 | 76.99 | 15.19 | 82.88 | – | 84.12 | 44.34 | 61.39 | | | |
| | | MAE | 87.20 | 37.01 | 84.53 | 69.63 | 77.80 | 25.01 | 81.72 | 56.84 | 84.48 | 53.26 | 64.97 | | | |
| HQHSAM | 17.32M | Dino | 86.51 | 37.42 | 85.06 | 70.42 | 77.64 | 29.99 | 78.21 | 56.66 | 83.26 | 54.43 | 65.24 | 84.98 | 52.63 | 65.28 |
| | | Sam | 87.50 | 36.20 | 86.20 | 75.23 | 79.18 | 34.65 | 84.67 | 51.34 | 86.12 | 55.32 | 66.87 | | | |
| | | M.Sam | 86.55 | 26.68 | 85.54 | 63.54 | 77.38 | 18.61 | 82.25 | – | 84.78 | 46.55 | 62.94 | | | |
| | | MAE | 87.25 | 39.79 | 86.20 | 71.35 | 79.37 | 24.71 | 83.84 | 55.98 | 85.76 | 54.24 | 66.06 | | | |
| Resnet | 39.21M | Dino | 87.35 | 25.17 | 85.76 | 78.64 | 78.47 | 32.83 | 57.14 | 11.00 | 76.75 | 45.22 | 57.05 | 77.88 | 43.60 | 56.88 |
| | | Sam | 86.96 | 28.53 | 87.42 | 79.21 | 76.31 | 20.52 | 65.22 | 23.23 | 79.87 | 45.56 | 58.43 | | | |
| | | M.Sam | 85.68 | 3.12 | 87.21 | 77.68 | 77.29 | 11.08 | 49.72 | – | 74.20 | 42.29 | 55.97 | | | |
| | | MAE | 87.68 | 12.55 | 88.40 | 80.38 | 80.38 | 10.21 | 65.98 | 23.18 | 80.69 | 41.34 | 56.10 | | | |
| Unet | 39.10M | Dino | 88.46 | 22.18 | 88.62 | 81.71 | 79.88 | 31.64 | 58.65 | 42.31 | 78.58 | 51.54 | 61.68 | 79.10 | 49.20 | 60.87 |
| | | Sam | 87.37 | 26.58 | 87.75 | 78.93 | 76.38 | 30.52 | 73.44 | 40.32 | 82.85 | 50.55 | 62.66 | | | |
| | | M.Sam | 86.27 | 3.04 | 87.63 | 77.93 | 76.77 | 12.85 | 56.15 | – | 76.68 | 42.65 | 57.23 | | | |
| | | MAE | 87.98 | 29.40 | 88.62 | 81.99 | 80.20 | 14.99 | 58.28 | 53.76 | 78.29 | 52.07 | 61.90 | | | |
| Unet S. | 7.87M | Dino | 87.77 | 37.28 | 87.52 | 79.89 | 79.35 | 38.18 | 61.18 | 29.16 | 78.82 | 52.77 | 62.54 | 78.10 | 47.61 | 59.48 |
| | | Sam | 87.47 | 21.77 | 86.57 | 77.65 | 75.93 | 32.47 | 67.81 | 32.31 | 80.62 | 48.03 | 60.25 | | | |
| | | M.Sam | 85.96 | 6.50 | 86.78 | 75.10 | 76.87 | 12.63 | 53.24 | – | 75.33 | 42.78 | 56.73 | | | |
| | | MAE | 87.26 | 30.19 | 87.76 | 79.46 | 80.56 | 17.74 | 57.86 | 26.45 | 77.63 | 46.88 | 58.41 | | | |

Table A.2: Decode Head Comparisons - Dino

| Rank | Decoder | #Trainable Param | OV Mean | OV diff % | DS Mean | DS diff % | DG Mean | DG diff % |
|---|---|---|---|---|---|---|---|---|
| 1 | HQHSAM | 17.32M | 65.24 | 0.00 | 83.26 | 0.00 | 54.43 | -2.75 |
| 2 | S-PE-MD | 5.61M | 64.92 | -0.49 | 80.24 | -3.63 | 55.73 | -0.42 |
| 3 | HQSAM | 7.61M | 64.22 | -1.56 | 82.48 | -0.94 | 53.26 | -4.83 |
| 4 | HSAM | 15.98M | 63.88 | -2.09 | 79.34 | -4.71 | 54.60 | -2.44 |
| 5 | S-FPE-MD | 5.60M | 63.71 | -2.34 | 80.21 | -3.66 | 53.81 | -3.85 |
| 6 | DA | 24.35M | 63.57 | -2.55 | 76.26 | -8.41 | 55.96 | 0.00 |
| 7 | Unet S. | 7.87M | 62.54 | -4.13 | 78.82 | -5.33 | 52.77 | -5.70 |
| 8 | Unet | 39.10M | 61.68 | -5.45 | 78.58 | -5.62 | 51.54 | -7.90 |
| 9 | DA S. | 6.94M | 59.67 | -8.53 | 73.63 | -11.57 | 51.30 | -8.33 |
| 10 | Resnet | 39.21M | 57.05 | -12.56 | 76.75 | -7.82 | 45.22 | -19.19 |
| 11 | SegFor | 4.74M | 53.36 | -18.21 | 67.92 | -18.43 | 44.63 | -20.26 |
| 12 | Linear | 52.24K | 49.11 | -24.72 | 62.55 | -24.87 | 41.04 | -26.66 |

Table A.3: Decode Head Comparisons - SAM

| Rank | Decoder | #Trainable Param | OV Mean | OV diff % | DS Mean | DS diff % | DG Mean | DG diff % |
|---|---|---|---|---|---|---|---|---|
| 1 | HQHSAM | 17.32M | 66.87 | 0.00 | 86.12 | -0.23 | 55.32 | -0.10 |
| 2 | HQSAM | 7.61M | 66.13 | -1.10 | 86.32 | 0.00 | 54.02 | -2.46 |
| 3 | S-FPE-MD | 5.60M | 65.66 | -1.81 | 82.81 | -4.07 | 55.38 | 0.00 |
| 4 | HSAM | 15.98M | 65.53 | -2.01 | 85.63 | -0.80 | 53.46 | -3.46 |
| 5 | S-PE-MD | 5.61M | 64.16 | -4.06 | 82.49 | -4.44 | 53.15 | -4.02 |
| 6 | Unet | 39.10M | 62.66 | -6.30 | 82.85 | -4.02 | 50.55 | -8.73 |
| 7 | DA S. | 6.94M | 60.65 | -9.31 | 80.82 | -6.38 | 48.55 | -12.34 |
| 8 | Unet S. | 7.87M | 60.25 | -9.91 | 80.62 | -6.61 | 48.03 | -13.28 |
| 9 | DA | 24.35M | 59.33 | -11.28 | 78.52 | -9.04 | 47.81 | -13.67 |
| 10 | Resnet | 39.21M | 58.43 | -12.63 | 79.87 | -7.48 | 45.56 | -17.73 |
| 11 | SegFor | 4.74M | 53.93 | -19.35 | 70.95 | -17.81 | 43.72 | -21.06 |
| 12 | Linear | 52.24K | 48.64 | -27.27 | 62.85 | -27.19 | 40.11 | -27.57 |

Table A.4: Decode Head Comparisons - MedSam

| Rank | Decoder | #Trainable Param | OV Mean | OV diff % | DS Mean | DS diff % | DG Mean | DG diff % |
|---|---|---|---|---|---|---|---|---|
| 1 | HQSAM | 7.61M | 63.20 | 0.00 | 84.89 | 0.00 | 46.94 | 0.00 |
| 2 | HQHSAM | 17.32M | 62.94 | -0.42 | 84.78 | -0.13 | 46.55 | -0.81 |
| 3 | HSAM | 15.98M | 61.39 | -2.87 | 84.12 | -0.91 | 44.34 | -5.53 |
| 4 | S-PE-MD | 5.61M | 60.28 | -4.63 | 81.92 | -3.50 | 44.04 | -6.16 |
| 5 | S-FPE-MD | 5.60M | 60.17 | -4.80 | 80.95 | -4.65 | 44.59 | -5.01 |
| 6 | Unet | 39.10M | 57.23 | -9.44 | 76.68 | -9.67 | 42.65 | -9.13 |
| 7 | Unet S. | 7.87M | 56.73 | -10.25 | 75.33 | -11.27 | 42.78 | -8.86 |
| 8 | Resnet | 39.21M | 55.97 | -11.45 | 74.20 | -12.59 | 42.29 | -9.89 |
| 9 | DA S. | 6.94M | 55.69 | -11.88 | 78.00 | -8.12 | 38.96 | -16.99 |
| 10 | DA | 24.35M | 54.16 | -14.31 | 72.80 | -14.24 | 40.18 | -14.39 |
| 11 | SegFor | 4.74M | 49.08 | -22.35 | 65.61 | -22.72 | 36.69 | -21.84 |
| 12 | Linear | 52.24K | 38.54 | -39.02 | 55.50 | -34.62 | 25.82 | -44.99 |

***Including only brain and spine datasets*

Table A.5: Decode Head Comparisons - MAE

| Rank | Decoder | #Trainable Param | OV Mean | OV diff % | DS Mean | DS diff % | DG Mean | DG diff % |
|---|---|---|---|---|---|---|---|---|
| 1 | HQSAM | 7.61M | 66.24 | 0.00 | 85.82 | 0.00 | 54.50 | 0.00 |
| 2 | HQHSAM | 17.32M | 66.06 | -0.27 | 85.76 | -0.06 | 54.24 | -0.47 |
| 3 | HSAM | 15.98M | 64.97 | -1.92 | 84.48 | -1.55 | 53.26 | -2.28 |
| 4 | S-PE-MD | 5.61M | 64.19 | -3.10 | 83.05 | -3.22 | 52.87 | -2.99 |
| 5 | S-FPE-MD | 5.60M | 62.99 | -4.92 | 84.27 | -1.80 | 50.22 | -7.86 |
| 6 | Unet | 39.10M | 61.90 | -6.55 | 78.29 | -8.77 | 52.07 | -4.46 |
| 7 | DA | 24.35M | 61.44 | -7.24 | 80.35 | -6.37 | 50.10 | -8.07 |
| 8 | DA S. | 6.94M | 60.76 | -8.27 | 78.29 | -8.77 | 50.25 | -7.79 |
| 9 | Unet S. | 7.87M | 58.41 | -11.82 | 77.63 | -9.54 | 46.88 | -13.98 |
| 10 | Resnet | 39.21M | 56.10 | -15.32 | 80.69 | -5.98 | 41.34 | -24.14 |
| 11 | SegFor | 4.74M | 52.12 | -21.31 | 69.74 | -18.74 | 41.56 | -23.75 |
| 12 | Linear | 52.24K | 47.18 | -28.78 | 63.47 | -26.04 | 37.41 | -31.36 |

## A.2 Fine-Tuning Comparisons

Table A.6: Fine-Tuning Comparisons - Brain Datasets - All Runs

| Fine Tuning | #Tr. Par. FT. [M] | #Tr. Par. Dec. [M] | #Train Par. [M] | Backbone | Train D.Set | HCP1 Dice | HCP2 Dice | Ab. C. Dice | Ab. S. Dice | DS Dice | DG Dice | OV Dice |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Freeze | 0.00 | 17.80 | 17.80 | Dino | HCP1 | 85.01 | 19.16 | 76.75 | 64.24 | 84.17 | 49.55 | 58.20 |
| | | | | | HCP2 | 34.22 | 84.45 | 37.23 | 23.45 | | | |
| | | | | | Ab. C. | 77.64 | 29.99 | 85.06 | 70.42 | | | |
| | | | | | Ab. S. | 68.68 | 15.82 | 76.96 | 82.16 | | | |
| | 0.00 | 17.32 | 17.32 | Sam | HCP1 | 86.42 | 22.71 | 79.27 | 73.09 | 85.43 | 49.88 | 58.77 |
| | | | | | HCP2 | 23.35 | 85.29 | 23.95 | 14.59 | | | |
| | | | | | Ab. C. | 79.18 | 34.65 | 86.20 | 75.23 | | | |
| | | | | | Ab. S. | 75.64 | 15.11 | 81.82 | 83.80 | | | |
| | 0.00 | 17.32 | 17.32 | MedSam | HCP1 | 86.00 | 8.98 | 77.53 | 64.90 | 84.76 | 40.79 | 51.78 |
| | | | | | HCP2 | 11.15 | 84.58 | 14.59 | 9.26 | | | |
| | | | | | Ab. C. | 77.38 | 18.61 | 85.54 | 63.54 | | | |
| | | | | | Ab. S. | 64.89 | 6.65 | 72.00 | 82.93 | | | |
| | 0.00 | 17.32 | 17.32 | MAE | HCP1 | 86.65 | 17.46 | 79.28 | 74.20 | 85.54 | 47.25 | 56.82 |
| | | | | | HCP2 | 17.01 | 85.32 | 19.18 | 12.62 | | | |
| | | | | | Ab. C. | 79.37 | 24.71 | 86.20 | 71.35 | | | |
| | | | | | Ab. S. | 74.83 | 15.11 | 81.82 | 84.00 | | | |
| Reins LoRA | 1.70 | 17.80 | 19.50 | Dino | HCP1 | 86.27 | 24.87 | 78.64 | 68.38 | 85.05 | 52.37 | 60.54 |
| | | | | | HCP2 | 34.89 | 84.91 | 37.76 | 26.75 | | | |
| | | | | | Ab. C. | 78.15 | 31.02 | 85.79 | 72.85 | | | |
| | | | | | Ab. S. | 72.68 | 21.63 | 80.87 | 83.23 | | | |
| | 1.78 | 17.32 | 19.10 | Sam | HCP1 | 86.53 | 22.52 | 79.13 | 74.51 | 85.42 | 49.41 | 58.41 |
| | | | | | HCP2 | 21.75 | 85.38 | 20.74 | 12.86 | | | |
| | | | | | Ab. C. | 79.64 | 37.42 | 86.11 | 74.78 | | | |
| | | | | | Ab. S. | 73.02 | 14.48 | 82.12 | 83.64 | | | |
| | 1.78 | 17.32 | 19.10 | MedSam | HCP1 | 86.31 | 10.60 | 79.23 | 69.88 | 84.97 | 42.48 | 53.11 |
| | | | | | HCP2 | 11.33 | 84.53 | 11.54 | 9.13 | | | |
| | | | | | Ab. C. | 77.72 | 19.84 | 85.63 | 68.07 | | | |
| | | | | | Ab. S. | 69.96 | 6.70 | 75.81 | 83.40 | | | |
| | 1.78 | 17.32 | 19.10 | MAE | HCP1 | 86.65 | 16.10 | 79.46 | 74.18 | 85.47 | 47.64 | 57.10 |
| | | | | | HCP2 | 19.25 | 85.41 | 22.00 | 13.09 | | | |
| | | | | | Ab. C. | 78.94 | 22.94 | 86.25 | 74.32 | | | |
| | | | | | Ab. S. | 73.72 | 17.08 | 80.64 | 83.57 | | | |
| Reins | 2.50 | 17.80 | 20.30 | Dino | HCP1 | 86.34 | 25.15 | 78.59 | 70.87 | 85.06 | 52.46 | 60.61 |
| | | | | | HCP2 | 35.30 | 84.83 | 38.45 | 23.79 | | | |
| | | | | | Ab. C. | 77.90 | 31.68 | 85.91 | 74.32 | | | |
| | | | | | Ab. S. | 72.66 | 20.32 | 80.44 | 83.14 | | | |
| | 2.48 | 17.32 | 19.80 | Sam | HCP1 | 86.50 | 21.06 | 79.26 | 72.16 | 85.45 | 49.50 | 58.49 |
| | | | | | HCP2 | 23.83 | 85.34 | 24.92 | 13.56 | | | |
| | | | | | Ab. C. | 79.64 | 36.32 | 86.13 | 74.73 | | | |
| | | | | | Ab. S. | 74.81 | 12.26 | 81.49 | 83.84 | | | |
| | 2.48 | 17.32 | 19.80 | MedSam | HCP1 | 86.38 | 8.69 | 77.70 | 63.37 | 84.94 | 41.04 | 52.01 |
| | | | | | HCP2 | 12.56 | 84.52 | 12.94 | 9.64 | | | |
| | | | | | Ab. C. | 77.54 | 15.59 | 85.44 | 65.06 | | | |
| | | | | | Ab. S. | 67.08 | 6.62 | 75.64 | 83.43 | | | |
| | 2.48 | 17.32 | 19.80 | MAE | HCP1 | 86.30 | 17.20 | 80.24 | 74.98 | 85.40 | 47.15 | 56.71 |
| | | | | | HCP2 | 15.13 | 85.42 | 17.45 | 12.17 | | | |
| | | | | | Ab. C. | 79.21 | 26.51 | 86.23 | 74.57 | | | |
| | | | | | Ab. S. | 74.09 | 13.70 | 80.52 | 83.66 | | | |
| LadderD | 22.60 | 17.80 | 40.40 | Dino | HCP1 | 87.11 | 10.52 | 79.21 | 76.62 | 86.62 | 45.67 | 55.91 |
| | | | | | HCP2 | 11.50 | 86.37 | 13.59 | 9.54 | | | |
| | | | | | Ab. C. | 78.42 | 16.75 | 87.54 | 78.33 | | | |
| | | | | | Ab. S. | 79.55 | 10.85 | 83.14 | 85.45 | | | |
| | 22.45 | 16.53 | 38.98 | Sam | HCP1 | 86.74 | 11.03 | 78.80 | 75.76 | 86.08 | 44.44 | 54.85 |
| | | | | | HCP2 | 9.28 | 85.85 | 9.20 | 7.76 | | | |
| | | | | | Ab. C. | 78.95 | 13.14 | 86.98 | 78.17 | | | |
| | | | | | Ab. S. | 77.31 | 12.01 | 81.92 | 84.74 | | | |
| | 22.45 | 16.53 | 38.98 | MedSam | HCP1 | 86.66 | 10.10 | 77.91 | 75.81 | 86.04 | 44.22 | 54.68 |
| | | | | | HCP2 | 10.67 | 85.66 | 10.83 | 9.13 | | | |
| | | | | | Ab. C. | 79.03 | 10.62 | 86.98 | 77.41 | | | |
| | | | | | Ab. S. | 77.60 | 9.46 | 82.07 | 84.87 | | | |
| | 22.64 | 17.32 | 39.96 | MAE | HCP1 | 86.54 | 11.01 | 79.09 | 75.67 | 85.85 | 45.61 | 55.67 |
| | | | | | HCP2 | 11.53 | 85.71 | 15.48 | 11.14 | | | |
| | | | | | Ab. C. | 78.65 | 17.83 | 86.73 | 77.83 | | | |
| | | | | | Ab. S. | 77.00 | 9.66 | 82.48 | 84.42 | | | |

Table A.7: Fine-Tuning Comparisons - Brain Datasets - Summary

| Rank | FT | OV | OV % | DS | DS% | DG | DG% |
|---|---|---|---|---|---|---|---|
| 1 | Reins LoRA | 57.29 | 0.00 | 85.23 | -1.07 | 47.98 | 0.00 |
| 2 | Reins | 56.96 | -0.59 | 85.21 | -1.08 | 47.54 | -0.92 |
| 3 | Freeze | 56.39 | -1.57 | 84.98 | -1.36 | 46.87 | -2.32 |
| 4 | LadderD | 55.28 | -3.52 | 86.15 | 0.00 | 44.99 | -6.24 |

Table A.8: Fine-Tuning Comparisons - Lumbar Spine Datasets - All Runs

| Fine Tuning | #Tr. Par. FT. | #Tr. Par. Dec. | #Train Par. | Backbone | Train D.Set | VerSe Dice | MR SegV Dice | DS Dice | DG Dice | OV Dice |
|---|---|---|---|---|---|---|---|---|---|---|
| Freeze | 0.00 | 14.10 | 14.10 | Dino | VerSe<br>MR SegV | 83.77<br>37.42 | 30.08<br>86.51 | 85.14 | 33.75 | 59.45 |
| | 0.00 | 13.60 | 13.60 | Sam | VerSe<br>MR SegV | 86.45<br>36.20 | 18.10<br>87.50 | 86.98 | 27.15 | 57.06 |
| | 0.00 | 13.60 | 13.60 | MedSam | VerSe<br>MR SegV | 86.14<br>26.68 | 1.09<br>86.55 | 86.35 | 13.89 | 50.12 |
| | 0.00 | 13.60 | 13.60 | MAE | VerSe<br>MR SegV | 86.33<br>39.79 | 32.54<br>87.25 | 86.79 | 36.17 | 61.48 |
| Reins LoRA | 1.70 | 14.10 | 15.80 | Dino | VerSe<br>MR SegV | 83.87<br>32.92 | 36.09<br>86.67 | 85.27 | 34.51 | 59.89 |
| | 1.78 | 13.60 | 15.38 | Sam | VerSe<br>MR SegV | 86.60<br>36.20 | 19.04<br>87.80 | 87.20 | 27.62 | 57.41 |
| | 1.78 | 13.60 | 15.38 | MedSam | VerSe<br>MR SegV | 86.35<br>29.02 | 1.43<br>86.16 | 86.26 | 15.23 | 50.74 |
| | 1.78 | 13.60 | 15.38 | MAE | VerSe<br>MR SegV | 86.27<br>39.56 | 25.94<br>87.61 | 86.94 | 32.75 | 59.85 |
| Reins | 2.50 | 14.10 | 16.60 | Dino | VerSe<br>MR SegV | 84.14<br>37.01 | 32.66<br>86.27 | 85.21 | 34.84 | 60.02 |
| | 2.48 | 13.60 | 16.08 | Sam | VerSe<br>MR SegV | 82.99<br>35.31 | 23.10<br>87.99 | 85.49 | 29.21 | 57.35 |
| | 2.48 | 13.60 | 16.08 | MedSam | VerSe<br>MR SegV | 86.30<br>23.45 | 2.38<br>86.38 | 86.34 | 12.92 | 49.63 |
| | 2.48 | 13.60 | 16.08 | MAE | VerSe<br>MR SegV | 83.54<br>38.28 | 15.70<br>87.55 | 85.55 | 26.99 | 56.27 |
| LadderD | 22.60 | 14.10 | 36.70 | Dino | VerSe<br>MR SegV | 87.82<br>22.19 | 3.11<br>87.73 | 87.78 | 12.65 | 50.21 |
| | 22.38 | 12.92 | 35.30 | Sam | VerSe<br>MR SegV | 88.20<br>9.87 | 12.22<br>87.91 | 88.06 | 11.05 | 49.55 |
| | 22.38 | 12.92 | 35.30 | MedSam | VerSe<br>MR SegV | 88.17<br>24.28 | 2.54<br>88.02 | 88.10 | 13.41 | 50.75 |
| | 22.68 | 13.60 | 36.28 | MAE | VerSe<br>MR SegV | 86.75<br>26.79 | 11.03<br>87.61 | 87.18 | 18.91 | 53.05 |

Table A.9: Fine-Tuning Comparisons - Lumbar Spine Datasets - Summary

| Rank | FT | OV | OV % | DS | DS% | DG | DG% |
|---|---|---|---|---|---|---|---|
| 1 | Freeze | 57.03 | 0.00 | 86.31 | -1.67 | 27.74 | 0.00 |
| 2 | Reins LoRA | 56.97 | -0.10 | 86.42 | -1.55 | 27.53 | -0.77 |
| 3 | Reins | 55.82 | -2.12 | 85.65 | -2.43 | 25.99 | -6.31 |
| 4 | LadderD | 50.89 | -10.76 | 87.78 | 0.00 | 14.00 | -49.51 |

Table A.10: Fine-Tuning Comparisons - Prostate Datasets - All Runs

| Fine Tuning | #Tr. Par. FT. | #Tr. Par. Dec. | #Train Par. | Backbone | Train D.Set | NCI Dice | USZ Dice | DS Dice | DG Dice | OV Dice |
|---|---|---|---|---|---|---|---|---|---|---|
| Freeze | 0.00 | 12.90 | 12.90 | Dino | NCI<br>USZ | 67.51<br>56.66 | 27.37<br>78.21 | 72.86 | 42.02 | 57.44 |
| | 0.00 | 12.42 | 12.42 | Sam | NCI<br>USZ | 68.78<br>51.34 | 22.36<br>84.67 | 76.73 | 36.85 | 56.79 |
| | 0.00 | 12.42 | 12.42 | MAE | NCI<br>USZ | 72.98<br>55.98 | 29.28<br>83.84 | 78.41 | 42.63 | 60.52 |
| Reins LoRA | 1.70 | 12.90 | 14.60 | Dino | NCI<br>USZ | 69.19<br>61.76 | 31.11<br>82.62 | 75.91 | 46.44 | 61.17 |
| | 1.78 | 12.42 | 14.20 | Sam | NCI<br>USZ | 66.63<br>54.77 | 21.89<br>85.71 | 76.17 | 38.33 | 57.25 |
| | 1.78 | 12.42 | 14.20 | MAE | NCI<br>USZ | 68.95<br>55.02 | 27.95<br>82.01 | 75.48 | 41.49 | 58.48 |
| Reins | 2.50 | 12.90 | 15.40 | Dino | NCI<br>USZ | 68.71<br>60.93 | 31.01<br>81.76 | 75.24 | 45.97 | 60.60 |
| | 2.48 | 12.42 | 14.90 | Sam | NCI<br>USZ | 68.77<br>53.08 | 23.49<br>85.40 | 77.09 | 38.29 | 57.69 |
| | 2.48 | 12.42 | 14.90 | MAE | NCI<br>USZ | 68.71<br>58.68 | 27.40<br>84.93 | 76.82 | 43.04 | 59.93 |
| LadderD | 22.60 | 12.90 | 35.50 | Dino | NCI<br>USZ | 76.86<br>59.23 | 25.29<br>84.03 | 80.45 | 42.26 | 61.35 |
| | 22.38 | 11.63 | 34.01 | Sam | NCI<br>USZ | 72.80<br>60.44 | 24.78<br>85.80 | 79.30 | 42.61 | 60.96 |
| | 22.68 | 12.42 | 35.10 | MAE | NCI<br>USZ | 72.04<br>61.49 | 26.21<br>85.43 | 78.74 | 43.85 | 61.29 |

Table A.11: Fine-Tuning Comparisons - Prostate Datasets - Summary

| Rank | FT | OV | OV % | DS | DS% | DG | DG% |
|---|---|---|---|---|---|---|---|
| 1.00 | LadderD | 61.20 | 0.00 | 79.49 | 0.00 | 42.91 | 0.00 |
| 2.00 | Reins | 59.41 | -2.93 | 76.38 | -3.92 | 42.43 | -1.11 |
| 3.00 | Reins LoRA | 58.97 | -3.65 | 75.85 | -4.58 | 42.08 | -1.92 |
| 4.00 | Freeze | 58.25 | -4.82 | 76.00 | -4.40 | 40.50 | -5.61 |

Table A.12: Fine-Tuning Comparisons - All Datasets Averaged - Summary

| Rank | FT | OV | OV % | DS | DS % | DG | DG % |
|---|---|---|---|---|---|---|---|
| 1 | Reins LoRA | 57.74 | 0.00 | 82.50 | -2.34 | 39.20 | 0.00 |
| 2 | Reins | 57.39 | -0.61 | 82.41 | -2.44 | 38.65 | -1.39 |
| 3 | Freeze | 57.22 | -0.90 | 82.43 | -2.42 | 38.37 | -2.11 |
| 4 | LadderD | 55.79 | -3.38 | 84.47 | 0.00 | 33.97 | -13.34 |

## A.3 Domain Generalization

Table A.13: DG/DS - Brain Datasets - All Runs

| Train D.Set | #Train Par. [M] | Fine Tuning | Backbone | BB Sz | HCP1 Dice | HCP2 Dice | Ab. C. Dice | Ab. S. Dice | Dom. Gen | Overall |
|---|---|---|---|---|---|---|---|---|---|---|
| HCP1 | 19.50 | Reins LoRA | Dino | Base | 86.27 | 24.87 | 78.64 | 68.38 | 57.30 | 64.54 |
| | 19.80 | Reins | Sam | Base | 86.50 | 21.06 | 79.26 | 72.16 | 57.49 | 64.75 |
| | 38.98 | LadderD | MedSam | Base | 86.66 | 10.10 | 77.91 | 75.81 | 54.61 | 62.62 |
| | 17.32 | Freeze | MAE | Base | 86.65 | 17.46 | 79.28 | 74.20 | 56.98 | 64.40 |
| | 21.00 | Reins LoRA | Dino | Large | 85.64 | 28.27 | 76.49 | 67.41 | 57.39 | 64.45 |
| | 22.70 | Reins | Sam | Large | 86.49 | 26.42 | 79.92 | 72.34 | 59.56 | 66.29 |
| | 17.60 | Freeze | MAE | Large | 86.73 | 14.71 | 79.92 | 74.72 | 56.45 | 64.02 |
| | 25.10 | Reins LoRA | Dino | Giant | 86.26 | 24.41 | 77.91 | 70.91 | 57.74 | 64.87 |
| | 25.90 | Reins | Sam | Huge | 86.35 | 25.58 | 80.33 | 72.72 | 59.54 | 66.25 |
| | 18.40 | Freeze | MAE | Huge | 86.97 | 14.63 | 79.72 | 75.67 | 56.67 | 64.25 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 86.92 | 8.93 | 80.79 | 75.88 | 55.20 | 63.13 |
| | 31.00 | #N/A | Vanilla Unet | #N/A | 88.94 | 10.85 | 80.20 | 73.77 | 54.94 | 63.44 |
| | 27.20 | #N/A | SwinUNet | #N/A | 87.86 | 8.92 | 79.38 | 65.94 | 51.41 | 60.53 |
| HCP2 | 19.50 | Reins LoRA | Dino | Base | 34.89 | 84.91 | 37.76 | 26.75 | 33.13 | 46.08 |
| | 19.80 | Reins | Sam | Base | 23.83 | 85.34 | 24.92 | 13.56 | 20.77 | 36.91 |
| | 38.98 | LadderD | MedSam | Base | 10.67 | 85.66 | 10.83 | 9.13 | 10.21 | 29.07 |
| | 17.32 | Freeze | MAE | Base | 17.01 | 85.32 | 19.18 | 12.62 | 16.27 | 33.53 |
| | 21.00 | Reins LoRA | Dino | Large | 41.19 | 84.19 | 40.43 | 28.80 | 36.81 | 48.65 |
| | 22.70 | Reins | Sam | Large | 27.62 | 85.44 | 27.32 | 16.20 | 23.71 | 39.15 |
| | 17.60 | Freeze | MAE | Large | 18.40 | 85.46 | 21.77 | 13.09 | 17.75 | 34.68 |
| | 25.10 | Reins LoRA | Dino | Giant | 42.08 | 85.03 | 41.16 | 27.58 | 36.94 | 48.96 |
| | 25.90 | Reins | Sam | Huge | 28.07 | 85.41 | 29.86 | 14.62 | 24.18 | 39.49 |
| | 18.40 | Freeze | MAE | Huge | 20.31 | 85.95 | 19.26 | 14.16 | 17.91 | 34.92 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 11.25 | 86.34 | 8.20 | 7.75 | 9.07 | 28.39 |
| | 31.00 | Vanilla Unet | #N/A | #N/A | 12.02 | 87.92 | 11.73 | 8.54 | 10.76 | 30.05 |
| | 27.20 | SwinUNet | #N/A | #N/A | 11.57 | 86.68 | 14.92 | 9.17 | 11.89 | 30.59 |
| Ab. C. | 19.50 | Reins LoRA | Dino | Base | 78.15 | 31.02 | 85.79 | 72.85 | 60.67 | 66.95 |
| | 19.80 | Reins | Sam | Base | 79.64 | 36.32 | 86.13 | 74.73 | 63.56 | 69.21 |
| | 38.98 | LadderD | MedSam | Base | 79.03 | 10.62 | 86.98 | 77.41 | 55.69 | 63.51 |
| | 17.32 | Freeze | MAE | Base | 79.37 | 24.71 | 86.20 | 71.35 | 58.48 | 65.41 |
| | 21.00 | Reins LoRA | Dino | Large | 78.46 | 34.84 | 85.50 | 74.35 | 62.55 | 68.29 |
| | 22.70 | Reins | Sam | Large | 79.54 | 37.57 | 86.20 | 75.06 | 64.06 | 69.59 |
| | 17.60 | Freeze | MAE | Large | 79.24 | 27.03 | 86.36 | 74.51 | 60.26 | 66.79 |
| | 25.10 | Reins LoRA | Dino | Giant | 78.98 | 37.68 | 86.12 | 75.84 | 64.17 | 69.66 |
| | 25.90 | Reins | Sam | Huge | 79.41 | 38.29 | 86.22 | 76.97 | 64.89 | 70.22 |
| | 18.40 | Freeze | MAE | Huge | 79.54 | 29.86 | 87.01 | 77.76 | 62.39 | 68.54 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 77.91 | 7.36 | 87.13 | 78.76 | 54.68 | 62.79 |
| | 31.00 | Vanilla Unet | #N/A | #N/A | 78.58 | 11.37 | 89.58 | 79.96 | 56.64 | 64.87 |
| | 27.20 | SwinUNet | #N/A | #N/A | 77.83 | 15.65 | 87.25 | 72.22 | 55.23 | 63.24 |
| Ab. S. | 19.50 | Reins LoRA | Dino | Base | 72.68 | 21.63 | 80.87 | 83.23 | 58.39 | 64.60 |
| | 19.80 | Reins | Sam | Base | 74.81 | 12.26 | 81.49 | 83.84 | 56.19 | 63.10 |
| | 38.98 | LadderD | MedSam | Base | 77.60 | 9.46 | 82.07 | 84.87 | 56.38 | 63.50 |
| | 17.32 | Freeze | MAE | Base | 74.83 | 15.11 | 81.82 | 84.00 | 57.25 | 63.94 |
| | 21.00 | Reins LoRA | Dino | Large | 73.99 | 25.78 | 79.11 | 83.12 | 59.63 | 65.50 |
| | 22.70 | Reins | Sam | Large | 75.27 | 22.47 | 81.68 | 83.86 | 59.81 | 65.82 |
| | 17.60 | Freeze | MAE | Large | 71.44 | 10.30 | 81.25 | 84.20 | 54.33 | 61.80 |
| | 25.10 | Reins LoRA | Dino | Giant | 73.86 | 26.06 | 79.72 | 83.09 | 59.88 | 65.68 |
| | 25.90 | Reins | Sam | Huge | 73.23 | 19.63 | 81.67 | 83.47 | 58.18 | 64.50 |
| | 18.40 | Freeze | MAE | Huge | 73.90 | 15.10 | 82.66 | 84.61 | 57.22 | 64.07 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 79.02 | 9.35 | 82.70 | 83.89 | 57.02 | 63.74 |
| | 31.00 | Vanilla Unet | #N/A | #N/A | 78.69 | 8.32 | 85.52 | 87.42 | 57.51 | 64.99 |
| | 27.20 | SwinUNet | #N/A | #N/A | 64.29 | 6.59 | 73.70 | 85.03 | 48.19 | 57.40 |

Table A.14: DG/DS - Brain Datasets - Summary

| #Train Par. [M] | Fine Tuning | Backbone | BB Sz | DS | DG | OV | DS size % | DG size % | OV size % |
|---|---|---|---|---|---|---|---|---|---|
| 19.50 | Reins LoRA | Dino | Base | 85.05 | **52.37** | **60.54** | -0.09 | -4.22 | -2.81 |
| 19.80 | Reins | Sam | | 85.45 | 49.50 | 58.49 | -0.05 | -4.40 | -2.86 |
| 38.98 | LadderD | MedSam | | 86.04 | 44.22 | 54.68 | | | |
| 17.32 | Freeze | MAE | | 85.54 | 47.25 | 56.82 | -0.69 | -2.68 | -1.94 |
| 21.00 | Reins LoRA | Dino | Large | 84.61 | **54.09** | **61.72** | -0.60 | -1.08 | -0.92 |
| 22.70 | Reins | Sam | | 85.50 | 51.78 | 60.21 | 0.00 | 0.00 | 0.00 |
| 17.60 | Freeze | MAE | | 85.69 | 47.20 | 56.82 | -0.52 | -2.78 | -1.94 |
| 25.10 | Reins LoRA | Dino | Giant | 85.13 | **54.68** | **62.29** | 0.00 | 0.00 | 0.00 |
| 25.90 | Reins | Sam | Huge | 85.36 | 51.70 | 60.11 | -0.16 | -0.17 | -0.16 |
| 18.40 | Freeze | MAE | Huge | **86.14** | 48.55 | 57.94 | 0.00 | 0.00 | 0.00 |
| 68.20 | Full Fine Tune | Resnet+Unet | 101 | 86.07 | 43.99 | 54.51 | | | |
| 31.00 | #N/A | Vanilla Unet | #N/A | **88.47** | 44.96 | 55.84 | | | |
| 27.20 | #N/A | SwinUNet | #N/A | **86.71** | 41.68 | 52.94 | | | |

Table A.15: DG/DS - Lumbar Spine Datasets - All Runs

| Train D.Set | #Train Par.[M] | Fine Tuning | Backbone | BB Sz | VerSe | MR SegV | Overall |
|---|---|---|---|---|---|---|---|
| VerSe | 15.80 | Reins LoRA | Dino | Base | 83.87 | 36.09 | 59.98 |
| | 16.08 | Reins | Sam | Base | 82.99 | 23.10 | 53.05 |
| | 35.30 | LadderD | MedSam | Base | 88.17 | 2.54 | 45.36 |
| | 13.60 | Freeze | MAE | Base | 86.33 | 32.54 | 59.44 |
| | 17.40 | Reins LoRA | Dino | Large | 83.61 | 15.82 | 49.72 |
| | 19.00 | Reins | Sam | Large | 86.24 | 35.94 | 61.09 |
| | 14.00 | Freeze | MAE | Large | 83.99 | 53.58 | 68.79 |
| | 21.40 | Reins LoRA | Dino | Giant | 81.28 | 41.20 | 61.24 |
| | 22.20 | Reins | Sam | Huge | 86.53 | 33.63 | 60.08 |
| | 14.80 | Freeze | MAE | Huge | 82.12 | 43.49 | 62.81 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 87.82 | 0.00 | 43.91 |
| | 31.00 | #N/A | Vanilla Unet | #N/A | 85.97 | 0.21 | 43.09 |
| | 27.20 | #N/A | SwinUNet | #N/A | 86.63 | 1.22 | 43.93 |
| MR SegV | 15.80 | Reins LoRA | Dino | Base | 32.92 | 86.67 | 59.80 |
| | 16.08 | Reins | Sam | Base | 35.31 | 87.99 | 61.65 |
| | 35.30 | LadderD | MedSam | Base | 24.28 | 88.02 | 56.15 |
| | 13.60 | Freeze | MAE | Base | 39.79 | 87.25 | 63.52 |
| | 17.40 | Reins LoRA | Dino | Large | 46.77 | 86.51 | 66.64 |
| | 19.00 | Reins | Sam | Large | 40.12 | 87.66 | 63.89 |
| | 14.00 | Freeze | MAE | Large | 43.31 | 87.94 | 65.63 |
| | 21.40 | Reins LoRA | Dino | Giant | 40.59 | 87.00 | 63.80 |
| | 22.20 | Reins | Sam | Huge | 32.60 | 87.39 | 60.00 |
| | 14.80 | Freeze | MAE | Huge | 40.80 | 88.13 | 64.47 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 4.70 | 87.96 | 46.33 |
| | 31.00 | #N/A | Vanilla Unet | #N/A | 2.22 | 88.07 | 45.15 |
| | 27.20 | #N/A | SwinUNet | #N/A | 10.03 | 86.95 | 48.49 |

Table A.16: DG/DS - Lumbar Spine Datasets - Summary

| #Train Par. [M] | Fine Tuning | Backbone | BB Sz | DS | DG | OV | DS size % | DG size % | OV size % |
|---|---|---|---|---|---|---|---|---|---|
| 15.80 | Reins LoRA | Dino | Base | 85.27 | 34.51 | 59.89 | 0.00 | -15.63 | -4.21 |
| 16.08 | Reins | Sam | Base | 85.49 | 29.21 | 57.35 | -1.69 | -23.21 | -8.23 |
| 35.30 | LadderD | MedSam | Base | 88.10 | 13.41 | 50.75 | | | |
| 13.60 | Freeze | MAE | Base | 86.79 | 36.17 | 61.48 | 0.00 | -25.35 | -8.52 |
| 17.40 | Reins LoRA | Dino | Large | 85.06 | 31.30 | 58.18 | -0.25 | -23.47 | -6.94 |
| 19.00 | Reins | Sam | Large | 86.95 | 38.03 | 62.49 | -0.01 | 0.00 | 0.00 |
| 14.00 | Freeze | MAE | Large | 85.97 | **48.45** | **67.21** | -0.95 | 0.00 | 0.00 |
| 21.40 | Reins LoRA | Dino | Giant | 84.14 | **40.90** | **62.52** | -1.33 | 0.00 | 0.00 |
| 22.20 | Reins | Sam | Huge | **86.96** | 33.12 | 60.04 | 0.00 | -12.92 | -3.92 |
| 14.80 | Freeze | MAE | Huge | 85.13 | **42.15** | **63.64** | -1.92 | -13.00 | -5.31 |
| 68.20 | Full Fine Tune | Resnet+Unet | 101 | **87.89** | 2.35 | 45.12 | | | |
| 31.00 | #N/A | Vanilla Unet | #N/A | **87.02** | 1.22 | 44.12 | | | |
| 27.20 | #N/A | SwinUNet | #N/A | 86.79 | 5.63 | 46.21 | | | |

Table A.17: DG/DS - Prostate Datasets - All Runs

| Train D.Set | #Train Par. [M] | Fine Tuning | Backbone | BB Sz | NCI Dice | USZ Dice | Overall |
|---|---|---|---|---|---|---|---|
| NCI | 14.60 | Reins LoRA | Dino | Base | 69.19 | 31.11 | 50.15 |
| | 14.90 | Reins | Sam | Base | 68.77 | 23.49 | 46.13 |
| | 12.42 | Freeze | MAE | Base | 72.98 | 29.28 | 51.13 |
| | 16.20 | Reins LoRA | Dino | Large | 68.72 | 28.04 | 48.38 |
| | 17.80 | Reins | Sam | Large | 67.58 | 25.13 | 46.36 |
| | 12.70 | Freeze | MAE | Large | 72.07 | 26.20 | 49.14 |
| | 20.20 | Reins LoRA | Dino | Giant | 70.63 | 27.93 | 49.28 |
| | 21.00 | Reins | Sam | Huge | 65.62 | 23.41 | 44.52 |
| | 13.50 | Freeze | MAE | Huge | 73.85 | 30.00 | 51.93 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 63.46 | 18.98 | 41.22 |
| | 31.00 | #N/A | Vanilla Unet | #N/A | 75.33 | 25.39 | 50.36 |
| | 27.20 | #N/A | SwinUNet | #N/A | 58.66 | 25.09 | 41.88 |
| USZ | 14.60 | Reins LoRA | Dino | Base | 61.76 | 82.62 | 72.19 |
| | 14.90 | Reins | Sam | Base | 53.08 | 85.40 | 69.24 |
| | 12.42 | Freeze | MAE | Base | 55.98 | 83.84 | 69.91 |
| | 16.20 | Reins LoRA | Dino | Large | 57.43 | 77.18 | 67.31 |
| | 17.80 | Reins | Sam | Large | 52.88 | 84.55 | 68.72 |
| | 12.70 | Freeze | MAE | Large | 59.33 | 83.57 | 71.45 |
| | 20.20 | Reins LoRA | Dino | Giant | 52.80 | 78.60 | 65.70 |
| | 21.00 | Reins | Sam | Huge | 54.18 | 83.94 | 69.06 |
| | 13.50 | Freeze | MAE | Huge | 61.90 | 85.21 | 73.56 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 43.94 | 74.66 | 59.30 |
| | 31.00 | #N/A | Vanilla Unet | #N/A | 1.43 | 62.87 | 32.15 |
| | 27.20 | #N/A | SwinUNet | #N/A | 34.70 | 72.35 | 53.53 |

Table A.18: DG/DS - Prostate Datasets - Summary

| #Train Par. [M] | Fine Tuning | Backbone | BB Sz | DS | DG | OV | DS size % | DG size % | OV size % |
|---|---|---|---|---|---|---|---|---|---|
| 14.60 | Reins LoRA | Dino | Base | 75.91 | **46.44** | **61.17** | 0.00 | 0.00 | 0.00 |
| 14.90 | Reins | Sam | | 77.09 | 38.29 | 57.69 | 0.00 | -1.85 | 0.00 |
| 12.42 | Freeze | MAE | | **78.41** | 42.63 | **60.52** | -1.41 | -7.23 | -3.54 |
| 16.20 | Reins LoRA | Dino | Large | 72.95 | 42.74 | 57.84 | -3.89 | -7.97 | -5.44 |
| 17.80 | Reins | Sam | | 76.07 | 39.01 | 57.54 | -1.32 | 0.00 | -0.26 |
| 12.70 | Freeze | MAE | | **77.82** | **42.77** | 60.29 | -2.15 | -6.93 | -3.90 |
| 20.20 | Reins LoRA | Dino | Giant | 74.62 | 40.37 | 57.49 | -1.70 | -13.07 | -6.02 |
| 21.00 | Reins | Sam | Huge | 74.78 | 38.80 | 56.79 | -2.99 | -0.54 | -1.56 |
| 13.50 | Freeze | MAE | Huge | **79.53** | **45.95** | **62.74** | 0.00 | 0.00 | 0.00 |
| 68.20 | Full Fine Tune | Resnet+Unet | 101 | 69.06 | 31.46 | 50.26 | | | |
| 31.00 | #N/A | Vanilla Unet | #N/A | 69.10 | 13.41 | 41.26 | | | |
| 27.20 | #N/A | SwinUNet | #N/A | 65.51 | 29.90 | 47.70 | | | |

Table A.19: DG/DS - Brain Tumor Datasets - All Runs

| Train D.Set | #Train Par. [M] | Fine Tuning | Backbone | BB Sz | T1 | FLAIR | Overall |
|---|---|---|---|---|---|---|---|
| T1 | 14.60 | Reins LoRA | Dino | Base | 82.14 | 60.96 | 71.55 |
| | 14.90 | Reins | Sam | Base | 82.34 | 55.06 | 68.70 |
| | 12.40 | Freeze | MAE | Base | 83.07 | 60.23 | 71.65 |
| | 16.20 | Reins LoRA | Dino | Large | 82.95 | 63.49 | 73.22 |
| | 17.80 | Reins | Sam | Large | 82.26 | 57.39 | 69.83 |
| | 12.70 | Freeze | MAE | Large | 83.30 | 57.30 | 70.30 |
| | 20.20 | Reins LoRA | Dino | Giant | 82.73 | 60.28 | 71.51 |
| | 21.00 | Reins | Sam | Huge | 82.95 | 58.60 | 70.78 |
| | 13.50 | Freeze | MAE | Huge | 83.37 | 60.89 | 72.13 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 81.22 | 53.82 | 67.52 |
| | 31.00 | #N/A | Vanilla Unet | #N/A | 81.86 | 54.84 | 68.35 |
| | 27.20 | #N/A | SwinUNet | #N/A | 79.43 | 52.25 | 65.84 |
| FLAIR | 14.60 | Reins LoRA | Dino | Base | 27.62 | 91.05 | 59.34 |
| | 14.90 | Reins | Sam | Base | 34.82 | 91.18 | 63.00 |
| | 12.40 | Freeze | MAE | Base | 28.24 | 91.53 | 59.89 |
| | 16.20 | Reins LoRA | Dino | Large | 38.19 | 91.30 | 64.75 |
| | 17.80 | Reins | Sam | Large | 31.23 | 91.30 | 61.27 |
| | 12.70 | Freeze | MAE | Large | 30.11 | 91.90 | 61.01 |
| | 20.20 | Reins LoRA | Dino | Giant | 23.37 | 91.57 | 57.47 |
| | 21.00 | Reins | Sam | Huge | 35.91 | 91.52 | 63.72 |
| | 13.50 | Freeze | MAE | Huge | 27.98 | 91.70 | 59.84 |
| | 68.20 | Full Fine Tune | Resnet+Unet | 101 | 9.14 | 90.20 | 49.67 |
| | 31.00 | #N/A | Vanilla Unet | #N/A | 19.79 | 90.48 | 55.14 |
| | 27.20 | #N/A | SwinUNet | #N/A | 29.12 | 91.89 | 60.51 |

Table A.20: DG/DS - Brain Tumor Datasets - Summary

| #Train Par. [M] | Fine Tuning | Backbone | BB Sz | DS | DG | OV | DS size % | DG size % | OV size % |
|---|---|---|---|---|---|---|---|---|---|
| 14.60 | Reins LoRA | Dino | Base | 86.60 | 44.29 | 65.44 | -0.64 | -12.88 | -5.13 |
| 14.90 | Reins | Sam | | 86.76 | **44.94** | 65.85 | -0.54 | -4.90 | -2.07 |
| 12.40 | Freeze | MAE | | **87.30** | 44.24 | 65.77 | -0.34 | -0.45 | -0.33 |
| 16.20 | Reins LoRA | Dino | Large | 87.13 | **50.84** | **68.98** | -0.03 | 0.00 | 0.00 |
| 17.80 | Reins | Sam | | 86.78 | 44.31 | 65.55 | -0.52 | -6.23 | -2.53 |
| 12.70 | Freeze | MAE | | **87.60** | 43.71 | 65.65 | 0.00 | -1.64 | -0.50 |
| 20.20 | Reins LoRA | Dino | Giant | 87.15 | 41.83 | 64.49 | 0.00 | -17.73 | -6.52 |
| 21.00 | Reins | Sam | Huge | 87.24 | **47.26** | **67.25** | 0.00 | 0.00 | 0.00 |
| 13.50 | Freeze | MAE | Huge | **87.54** | 44.44 | **65.99** | -0.07 | 0.00 | 0.00 |
| 68.20 | Full Fine Tune | Resnet+Unet | 101 | 85.71 | 31.48 | 58.60 | | | |
| 31.00 | #N/A | Vanilla Unet | #N/A | 86.17 | 37.32 | 61.74 | | | |
| 27.20 | #N/A | SwinUNet | #N/A | 85.66 | 40.69 | 63.17 | | | |

## A.4 Semi-Supervised Domain Adaptation - Self Training

Table A.21: Self Training - Brain Datasets - All Runs

| SD D.Set | #Train Par. [M] | Fine Tuning | Backbone | HCP1 | HCP2 | Ab. C. | Ab. S. | Domain Shift |
|---|---|---|---|---|---|---|---|---|
| HCP1 | 17.80 | Reins LoRA | DinoB | 86.27 | 77.48 | 84.09 | 81.76 | 81.11 |
| | 17.30 | Reins | SamB | 86.50 | 78.00 | 84.27 | 82.25 | 81.51 |
| | 16.50 | LadderD | MedSamB | 86.66 | 62.63 | 83.92 | 80.75 | 75.77 |
| | 17.30 | Freeze | MaeB | 86.65 | 78.90 | 84.29 | 82.51 | 81.90 |
| | 18.10 | Reins LoRA | DinoL | 85.64 | 76.28 | 83.32 | 80.46 | 80.02 |
| | 17.60 | Reins | SamL | 86.49 | 76.86 | 84.52 | 82.09 | 81.16 |
| | 17.60 | Freeze | MaeL | 86.73 | 78.59 | 84.72 | 82.71 | 82.01 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 86.92 | 70.66 | 85.18 | 82.25 | 79.36 |
| | 31.00 | _ | Vanilla Unet | 88.94 | 84.69 | 87.71 | 85.92 | 86.11 |
| | 27.20 | _ | SwinUNet | 87.86 | 79.75 | 85.45 | 83.42 | 82.87 |
| HCP2 | 17.80 | Reins LoRA | DinoB | 80.40 | 84.91 | 81.48 | 78.21 | 80.03 |
| | 17.30 | Reins | SamB | 80.73 | 85.34 | 82.11 | 78.38 | 80.41 |
| | 16.50 | LadderD | MedSamB | 69.75 | 85.66 | 71.33 | 62.15 | 67.74 |
| | 17.30 | Freeze | MaeB | 80.15 | 85.32 | 81.13 | 79.34 | 80.21 |
| | 18.10 | Reins LoRA | DinoL | 80.06 | 84.19 | 80.92 | 78.20 | 79.73 |
| | 17.60 | Reins | SamL | 81.95 | 85.44 | 81.89 | 79.01 | 80.95 |
| | 17.60 | Freeze | MaeL | 79.88 | 85.46 | 80.80 | 79.97 | 80.22 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 72.86 | 86.34 | 76.44 | 71.68 | 73.66 |
| | 31.00 | _ | Vanilla Unet | 86.09 | 87.92 | 86.22 | 85.37 | 85.89 |
| | 27.20 | _ | SwinUNet | 79.35 | 86.68 | 83.78 | 82.41 | 81.85 |
| Ab. C. | 17.80 | Reins LoRA | DinoB | 82.81 | 75.43 | 85.79 | 82.21 | 80.15 |
| | 17.30 | Reins | SamB | 83.20 | 73.94 | 86.13 | 82.65 | 79.93 |
| | 16.50 | LadderD | MedSamB | 83.86 | 69.69 | 86.98 | 82.97 | 78.84 |
| | 17.30 | Freeze | MaeB | 82.25 | 75.20 | 86.20 | 82.20 | 79.88 |
| | 18.10 | Reins LoRA | DinoL | 82.82 | 76.74 | 85.50 | 82.10 | 80.55 |
| | 17.60 | Reins | SamL | 83.09 | 75.67 | 86.20 | 82.61 | 80.46 |
| | 17.60 | Freeze | MaeL | 82.95 | 75.43 | 86.36 | 82.31 | 80.23 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 83.33 | 74.21 | 87.13 | 83.21 | 80.25 |
| | 31.00 | _ | Vanilla Unet | 86.14 | 84.29 | 89.58 | 86.80 | 85.74 |
| | 27.20 | _ | SwinUNet | 83.63 | 75.84 | 87.25 | 83.03 | 80.83 |
| Ab. S. | 17.80 | Reins LoRA | DinoB | 81.90 | 73.81 | 83.81 | 83.23 | 79.84 |
| | 17.30 | Reins | SamB | 81.96 | 75.70 | 84.20 | 83.84 | 80.62 |
| | 16.50 | LadderD | MedSamB | 83.41 | 60.46 | 84.51 | 84.87 | 76.13 |
| | 17.30 | Freeze | MaeB | 81.98 | 76.15 | 83.97 | 84.00 | 80.70 |
| | 18.10 | Reins LoRA | DinoL | 81.92 | 75.34 | 83.46 | 83.12 | 80.24 |
| | 17.60 | Reins | SamL | 82.73 | 76.44 | 84.30 | 83.86 | 81.16 |
| | 17.60 | Freeze | MaeL | 82.21 | 76.21 | 84.25 | 84.20 | 80.89 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 82.93 | 74.12 | 84.00 | 83.89 | 80.35 |
| | 31.00 | _ | Vanilla Unet | 86.09 | 84.30 | 87.57 | 87.42 | 85.99 |
| | 27.20 | _ | SwinUNet | 81.97 | 75.59 | 84.12 | 85.03 | 80.56 |

Table A.22: Self Training - Lumbar Spine Datasets - All Runs

| SD D.Set | #Train Par. [M] | Fine Tuning | Backbone | VerSe Dice | MR SegV Dice |
|---|---|---|---|---|---|
| VerSe | 14.10 | Reins LoRA | DinoB | 83.87 | 75.00 |
| | 13.60 | Reins | SamB | 82.99 | 77.25 |
| | 12.90 | LadderD | MedSamB | 88.17 | 69.85 |
| | 13.60 | Freeze | MaeB | 86.33 | 77.30 |
| | 14.40 | Reins LoRA | DinoL | 83.61 | 73.81 |
| | 14.00 | Reins | SamL | 86.24 | 78.51 |
| | 14.00 | Freeze | MaeL | 83.99 | 78.14 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 87.82 | 50.62 |
| | 31.00 | _ | Vanilla Unet | 85.97 | 62.55 |
| | 27.20 | _ | SwinUNet | 86.63 | 78.37 |
| MR SegV | 14.10 | Reins LoRA | DinoB | 1.61 | 86.67 |
| | 13.60 | Reins | SamB | 0.60 | 87.99 |
| | 12.90 | LadderD | MedSamB | 4.08 | 88.02 |
| | 13.60 | Freeze | MaeB | 3.10 | 87.25 |
| | 14.40 | Reins LoRA | DinoL | 0.21 | 86.51 |
| | 14.00 | Reins | SamL | 0.58 | 87.66 |
| | 14.00 | Freeze | MaeL | 0.24 | 87.94 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 1.54 | 87.96 |
| | 31.00 | _ | Vanilla Unet | 2.06 | 88.07 |
| | 27.20 | _ | SwinUNet | 5.42 | 86.95 |

Table A.23: Self Training - Prostate Datasets - All Runs

| SD D.Set | #Train Par. [M] | Fine Tuning | Backbone | NCI Dice | USZ Dice |
|---|---|---|---|---|---|
| NCI | 12.90 | Reins LoRA | DinoB | 69.19 | 58.39 |
| | 12.40 | Reins | SamB | 68.77 | 59.04 |
| | 12.40 | Freeze | MaeB | 72.98 | 60.32 |
| | 13.20 | Reins LoRA | DinoL | 68.72 | 59.78 |
| | 12.70 | Reins | SamL | 67.58 | 58.99 |
| | 12.70 | Freeze | MaeL | 72.07 | 60.23 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 63.46 | 52.53 |
| | 31.00 | _ | Vanilla Unet | 75.33 | 60.34 |
| | 27.20 | _ | SwinUNet | 58.66 | 45.08 |
| USZ | 12.90 | Reins LoRA | DinoB | 70.41 | 82.62 |
| | 12.40 | Reins | SamB | 66.62 | 85.40 |
| | 12.40 | Freeze | MaeB | 70.43 | 83.84 |
| | 13.20 | Reins LoRA | DinoL | 66.18 | 77.18 |
| | 12.70 | Reins | SamL | 66.56 | 84.55 |
| | 12.70 | Freeze | MaeL | 68.53 | 83.57 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 61.51 | 74.66 |
| | 31.00 | _ | Vanilla Unet | 70.09 | 62.87 |
| | 27.20 | _ | SwinUNet | 45.01 | 72.35 |

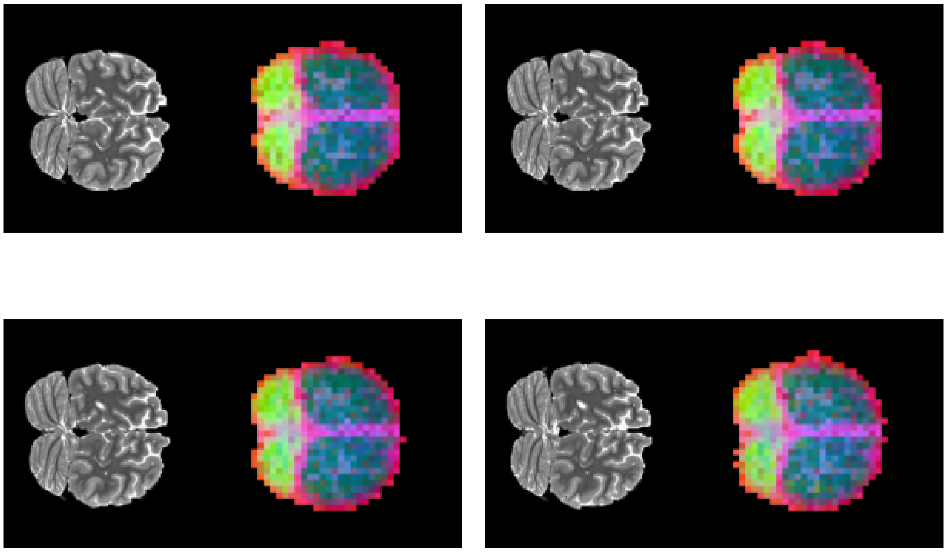
Table A.24: Self Training - Brain Tumor Datasets - All Runs

| SD D.Set | #Train Par. [M] | Fine Tuning | Backbone | T1 Dice | FLAIR Dice |
|---|---|---|---|---|---|
| T1 | 12.90 | Reins LoRA | DinoB | 82.14 | 80.64 |
| | 12.40 | Reins | SamB | 82.34 | 55.54 |
| | 12.40 | Freeze | MaeB | 83.07 | 65.53 |
| | 13.20 | Reins LoRA | DinoL | 82.95 | 58.70 |
| | 12.70 | Reins | SamL | 82.26 | 57.23 |
| | 12.70 | Freeze | MaeL | 83.30 | 79.52 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 81.22 | 54.22 |
| | 31.00 | _ | Vanilla Unet | 81.86 | 64.67 |
| | 27.20 | _ | SwinUNet | 79.43 | 64.48 |
| FLAIR | 12.90 | Reins LoRA | DinoB | 57.86 | 91.05 |
| | 12.40 | Reins | SamB | 56.08 | 91.18 |
| | 12.40 | Freeze | MaeB | 58.74 | 91.53 |
| | 13.20 | Reins LoRA | DinoL | 56.26 | 91.30 |
| | 12.70 | Reins | SamL | 57.43 | 91.30 |
| | 12.70 | Freeze | MaeL | 57.25 | 91.90 |
| | 25.70 | Full Fine Tune | Resnet+Unet | 55.78 | 90.20 |
| | 31.00 | _ | Vanilla Unet | 56.29 | 90.48 |
| | 27.20 | _ | SwinUNet | 55.27 | 91.89 |

## A.5 DinoV2 Self Attention Plots Without Training

Below are self-attention visualizations for some samples from the HCP1 [29] dataset. The visualization is based on PCA for three channels representing red, green, and blue, as was done in the Dino publication [25]. These were obtained at the very beginning of the project as a proof of concept.

Figure A.1: Dino Self-Attention PCA Visualization Without Training



# Bibliography

[1] Massih-Reza Amini, Vasilii Feofanov, Loic Pauletto, Lies Hadjadj, Emilie Devijver, and Yury Maximov. Self-training: A survey, 2024.

[2] Anton S Becker, Krishna Chaitanya, Khoschy Schawkat, Urs J Muehlematter, Andreas M Hötker, Ender Konukoglu, and Olivio F Donati. Variability of manual segmentation of the prostate in axial t2-weighted mri: a multi-reader study. *European journal of radiology*, 121:108716, 2019.

[3] N. Bloch, A. Madabhushi, and H. Huisman. Nci-isbi 2013 challenge: Automated segmentation of prostate structures, 2015.

[4] Hu Cao, Yueyue Wang, Joy Chen, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, and Manning Wang. Swin-unet: Unet-like pure transformer for medical image segmentation, 2021.

[5] Shurong Chai, Rahul Kumar Jain, Shiyu Teng, Jiaqing Liu, Yinhao Li, Tomoko Tateyama, and Yen wei Chen. Ladder fine-tuning approach for sam integrating complementary network, 2023.

[6] Zhiheng Cheng, Qingyue Wei, Hongru Zhu, Yan Wang, Liangqiong Qu, Wei Shao, and Yuyin Zhou. Unleashing the potential of sam for medical adaptation via hierarchical decoding, 2024.

[7] MMSegmentation Contributors. Openmmlab semantic segmentation toolbox and benchmark, 2020.

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[9] Bruce Fischl. Freesurfer. *Neuroimage*, 62(2):774–781, 2012.

[10] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation, 2019.

[11] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners, 2021.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[13] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation, 2022.

[14] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

[15] Fabian Isensee, Jens Petersen, Andre Klein, David Zimmerer, Paul F. Jaeger, Simon Kohl, Jakob Wasserthal, Gregor Koehler, Tobias Norajitra, Sebastian Wirkert, and Klaus H. Maier-Hein. nnu-net: Self-adapting framework for u-net-based medical image segmentation, 2018.

[16] Neerav Karani, Ertunc Erdil, Krishna Chaitanya, and Ender Konukoglu. Test-time adaptable neural networks for robust medical image segmentation. *Medical Image Analysis*, 68:101907, February 2021.

[17] Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Segment anything in high quality, 2023.

[18] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.

[19] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows, 2021.

[20] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

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

[22] Adriana Di Martino, Chao-Gan Yan, Qingyang Li, Erin Denio, Francisco X Castellanos, Kaat Alaerts, Jeffrey S Anderson, Michal Assaf, Susan Y Bookheimer, and et al. Mirella Dapretto. The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular Psychiatry*, 19(6):659–667, 2014.

[23] Bjoern H Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yvonne Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (brats), 2015.

[24] N/A. Mrspinesegv: Anonymized clinical lumbar spine mri dataset. Mendeley Data Repository, 2023.

[25] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024.

[26] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[27] Anjany Sekuboyina, Ali Bayat, Mohamed Elhabian Husseini, Markus Loffler, Markus Rempfler, Jan Kukačka, Jan Kirschke, and Bjoern H Menze. Verse: A vertebrae labelling and segmentation benchmark for multi-detector ct images, 2021.

[28] Nicholas J Tustison, Brian B Avants, Philip A Cook, Yuanjie Zheng, Alexander Egan, Paul A Yushkevich, and James C Gee. N4itk: improved n3 bias correction. *IEEE transactions on medical imaging*, 29(6):1310–1320, 2010.

[29] David C. Van Essen, Stephen M. Smith, Deanna M. Barch, Timothy E. Behrens, Essa Yacoub, and Kamil; WU-Minn HCP Consortium Ugurbil. The wu-minn human connectome project: an overview. *Neuroimage*, 80:62–79, Oct 2013.

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

[31] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization, 2021.

[32] Zhixiang Wei, Lin Chen, Yi Jin, Xiaoxiao Ma, Tianle Liu, Pengyang Ling, Ben Wang, Huaian Chen, and Jinjin Zheng. Stronger, fewer, & superior: Harnessing vision foundation models for domain generalized semantic segmentation, 2024.

[33] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers, 2021.