Prompt Tuning for Parameter-efficient Medical Image Segmentation

Marc Fischer[a,*], Alexander Bartler[a], Bin Yang[a]

[a]*Institute of Signal Processing and System Theory, University of Stuttgart, 70550 Stuttgart, Germany*

Abstract

Neural networks pre-trained on a self-supervision scheme have become the standard when operating in data rich environments with scarce annotations. As such, fine-tuning a model to a downstream task in a parameter-efficient but effective way, e.g. for a new set of classes in the case of semantic segmentation, is of increasing importance. In this work, we propose and investigate several contributions to achieve a parameter-efficient but effective adaptation for semantic segmentation on two medical imaging datasets. Relying on the recently popularized prompt tuning approach, we provide a prompt-able UNet (PUNet) architecture, that is frozen after pre-training, but adaptable throughout the network by class-dependent learnable prompt tokens. We pre-train this architecture with a dedicated dense self-supervision scheme based on assignments to online generated prototypes (contrastive prototype assignment, CPA) of a student teacher combination alongside a concurrent segmentation loss on a subset of classes. We demonstrate that the resulting neural network model is able to attenuate the gap between fully fine-tuned and parameter-efficiently adapted models on CT imaging datasets. As such, the difference between fully fine-tuned and prompt-tuned variants amounts to only 3.83 pp for the TCIA/BTCV dataset and 2.67 pp for the CT-ORG dataset in the mean Dice Similarity Coefficient (DSC, in %) while only prompt tokens, corresponding to 0.85% of the pre-trained backbone model with 6.8M frozen parameters, are adjusted. The code for this work is available on `https://github.com/marcdcfischer/PUNet`.

Keywords:
Self-Supervision, Semi-Supervision, Prompt Tuning, Semantic Segmentation, Transformer, Self-Attention

1. Introduction

With an ever increasing amount of radiological images to analyze, computer aided diagnosis (CAD) has become an integral part of medical studies. A basic processing block of many analyses is the semantic segmentation of medical imaging data. A plethora of approaches and neural network architectures has been proposed. Hereby, deep learning (DL) not only provides fast, but also robust and reproducible results, given sufficient quality of and annotations for the imaging data.

Commonly known, medical images are hard to segment due to its prevalence for inhomogeneities and variabilities. As such, many architectural advancements have been introduced for convolutional neural networks (CNN, Khan et al. (2020)) that progressively increased the robustness and performance of these networks. Recently, architectures that rely predominantly on transformer blocks (Vaswani et al., 2017) instead of traditional convolutions have shown a more broad applicability and superior performance. Medical variants have been proposed (Tang et al., 2022; Cao et al., 2021), which follow the popular UNet encoder-decoder structure (Ronneberger et al., 2015) and allow for excellent semantic segmentation.

In addition, the generalization capabilities of semantic segmentation approaches to unseen medical imaging data was long constrained by the scarcity and quality of annotations. Numerous schemes have been explored to alleviate this circumstance (Tajbakhsh et al., 2020). Recently, self-supervision by contrastive learning (Oord et al., 2018; Chen et al., 2020) has become incremental to leverage the abundance of (imaging) data, while keeping the requirement for the amount of annotated data low. Neural network weights are identified by a dedicated self-supervised pre-training scheme and serve as a basis for further downstream adaptation. A simple way to perform such an adaptation to the task at hand is to fully fine-tune the model by adjusting all its parameters. However, such a procedure incurs high memory and training time costs.

Despite these foundational contributions, self-supervised semantic segmentation models are often fine-tuned by adding an entirely new and task dependent decoder for every new encountered task (e.g. in the case of a new set of unseen classes). As such, the resulting end-to-end models are trained once for a specific combination of training data and available annotations. Inspired by recent successes in the natural language processing (NLP) community (Liu et al., 2021a; Li and Liang, 2021; Lester et al., 2021), we instead consider adapting neural network architectures by inserting additional tokens, called prompts, besides the input data. Tuning these prompts allows for the conditioning of pre-trained and subsequently frozen (language) models to perform specific downstream tasks. Hereby, learned continuous prompt tokens (vectors) are prepended to the (embedded)

---

arXiv:2211.09233v1 [cs.CV] 16 Nov 2022

input. The prompt tokens are updated by gradient decent in the backward pass like regular trainable neural network weights. However, they remain part of the input space and can be replaced or relearned for a new task while leaving the (frozen) backbone network task agnostic. This allows for a high degree of parameter sharing between tasks and thus low resulting memory costs when a model is extended to a new tasks.

From a practical standpoint, training only one architecture and adapting it in a parameter efficient way, opens up new possibilities on how to use the models in the medical context. We envision, only one (general purpose) architecture to be trained, which can be stored in a suitable (remote) location. On-premise, it can be adapted to the task of interest. As such, only minimal amounts of parameters have be stored, which would even allow for e.g. subject specific model tuning, which could aid in longitudinal studies. In addition, the private patient data remains on the location where the model was adapted on. In addition, task capabilities of a model could be extended and shared by publishing new prompt tokens.

To enable such a prompt tuning scheme for semantic segmentation on medical imaging data, we revisit aforementioned advancements and adapt them with the goal to facilitate task adaptation of a pre-trained but frozen model with minimal parameter adjustments. To this end, we

- introduce a deeply prompt-able encoder-decoder architecture (prompt-able UNet, PUNet) that can incorporate additional class-dependent prompt tokens to achieve dense binary and multi-class segmentation,

- contribute architectural components comprising prompt-able shifted window (PSWin) blocks, a heterogeneous bias score generation within the attention scheme, and a weighted similarity aggregation to enable token-dependent class predictions,

- propose a contrastive pre-training scheme specifically designed for dense self-supervision by soft assignments to online generated prototypes to establish anatomical representations while circumventing a hard separation of the contrastive attraction and repulsion,

- show that "prompting" of the pre-trained and frozen architecture by non-frozen (learned) prompt tokens is sufficient for adaptation to a segmentation downstream task on medical imaging data,

- leverage our assignement-based self-supervision scheme to enable the concurrent application of a prompt-dependent segmentation supervision in the pre-training phase, further reducing the performance gap between fully fine-tuned and efficiently adapted variants.

2. Related work

The work builds upon multiple recent developments. Notable mentions include the reliance on large models, the possibility for prompt tuning as downstream task adaptation, as well as recent advancements in using transformer blocks, but also self-supervised learning, and alternative pre-training strategies known from meta learning. Most notable works in these fields with respect to natural and medical images will be briefly covered.

2.1. Neural Network Architecture

Architectural design underwent a change from convolutional neural networks, especially suited for image data, to general-purpose transformer-based architectures (Vaswani et al., 2017) relying on attention layers. Originating in the NLP community, this design was subsequently applied to natural images by means of the Vision Transformer (ViT) (Dosovitskiy et al., 2020). Like its language model pendant, the vision variant can be scaled to billions of parameters (Zhai et al., 2022). In the following, transformer-based architectures have gained significant interest for image classification (Liu et al., 2021b), but also semantic segmentation (Strudel et al., 2021; Zheng et al., 2021), panoptic segmentation (Cheng et al., 2021), and have been applied successfully to medical images (Chen et al., 2021a; Xie et al., 2021; Hatamizadeh et al., 2022; Tang et al., 2022). Strudel et al. (2021) showed that a transformer named Segmenter, consisting of an encoder together with a joint transformer decoder for encoded content and output class tokens, can achieve superior results to popular convolutional architectures, such as DeepLabv3+ (Chen et al., 2018). Later, shifted window (SWin) blocks (Liu et al., 2021b) were introduced that greatly reduced the memory costs by limiting the self-attention to local non-overlapping windows that are subsequently shifted. Thus, a linear complexity is maintained in comparison to the quadratic complexity induced by self-attention layers. As such, the application of the attention scheme becomes viable throughout the encoder and decoder on medical data (Cao et al., 2021) while keeping memory requirements low. Still, for medical data, the encoder-decoder form with skip connections, popularized by the UNet (Ronneberger et al., 2015), remained prevalent. As such, Tang et al. (2022) proposed a SWin UNet Transformer (SWinUNETR) with SWin transformer blocks in the encoder and convolutional layers in the decoder. Cao et al. (2021) used a Shifted Window (SWin) UNet (Swin-UNet) with SWin transformer blocks in the encoder as well as the decoder.

2.2. Fine-Tuning

In recent years, model sizes have quickly grown from single digit millions (He et al., 2016), to hundreds of millions (Devlin et al., 2018), and more recently billions of parameters (Chowdhery et al., 2022). For very large architectures which are pre-trained on curated data at scale, one also speaks of foundation models (Bommasani et al., 2021). These models allow for unprecedented generalization and few-shot adaptation capabilities. However, with the increase of model parameters, effective and parameter-efficient adaptation methods of the transfer learning field become mandatory for model tuning to respective downstream tasks. As such, over-fitting on small scale downstream datasets and expansive storage costs can be alleviated. Such fine-tuning approaches adjust or inject a small trainable subset into a larger pre-trained model and optimize those

parameters on specific downstream tasks. Straightforward approaches include the replacement or addition of several output layers Mahajan et al. (2018), ranging from a classification head for classification to entire decoders for dense predictions. It is also possible to continue optimizing the bias terms of all neural network layers, while keeping the remaining bulk of parameters fixed (Zaken et al., 2021). Another parameter-efficient way is to include additional residual layers (Houlsby et al., 2019), called adapters, within existing blocks of layers. More sophisticated schemes achieve similar results by means of pruning based on sparse difference vectors (Guo et al., 2020) or small additive side networks (Zhang et al., 2020). He et al. (2021) proposed a unified view that generalizes some of the aforementioned concepts. For natural images (Jia et al., 2022) evaluated a subset of these approaches for image classification and segmentation.

Recently, a new paradigm called prompt tuning Liu et al. (2021a), achieved great successes in the NLP community (Li and Liang, 2021; Lester et al., 2021). Motivated by this success, first approaches explored this adaptation scheme for natural images by directly adapting the pixel space (Bahng et al., 2022) or via Visual Prompt Tuning (VPT) by Jia et al. (2022), where prompts can be injected deeply into the blocks of a dedicated frozen vision backbone (Dosovitskiy et al., 2020). Naturally, sophisticated generation schemes for prompts can be envisioned. For example, He et al. (2022) explore a HyperPrompt framework by generating layer specific prompt tokens based on global prompts and small networks.

2.3. Few-Shot Segmentation

Few-shot segmentation has been investigated to provide a robust generalization performance in the presence of few annotated samples. Hereby, prototypical networks (Snell et al., 2017; Wang et al., 2019), an instance of meta learning, have been explored. These networks learn to predict an embedding in which prototypes of classes can be established as robust and discriminative representatives. Another avenue relies on supervised pre-training on an available dataset followed by a subsequent transfer learning. Yet, with respect to medical images, Raghu et al. (2019) highlight issues for transfer learning of supervised models on natural data to medical imaging. However, for transfer learning between different medical datasets studies like Chen et al. (2019b) have established the effectiveness of supervised pre-training schemes. Nonetheless, their widespread adoption remains limited due to the inherent cost associated with providing expert annotations for 3D medical volumes at scale. We note that the main training process in VPT (Jia et al., 2022) also follows a supervised pre-training performed on another dataset.

2.4. Self-supervised Representation Learning

In the medical field, self-supervision (Taleb et al., 2020; Azizi et al., 2021; Ghesu et al., 2022) has become an integral part for the pre-training of models on the abundance of available (raw) data. Early approaches designed pre-text tasks, such as recombining jigsaw puzzles (Noroozi and Favaro, 2016), solving a rubik's cube (Zhuang et al., 2019), predicting rotations (Gidaris et al., 2018) or learning to reconstruct masked input content based on context by inpainting on natural (Pathak et al., 2016) and medical (Chen et al., 2019a; Haghighi et al., 2021; Zhou et al., 2021) images. The consistent anatomy allows for learning of anatomical regions, which can be leveraged for, e.g. landmark localization (Yan et al., 2022) or segmentation (Chaitanya et al., 2020). Hereby, the aforementioned vision transformers have been identified as excellent candidates for pre-training on natural images (Bao et al., 2021; Chen et al., 2021b; Caron et al., 2021) as well as on medical images (Liu et al., 2021b). Combinations of the schemes are also possible. For example, Tang et al. (2022) used a combination of inpainting, contrastive learning and rotation prediction to pre-train a SWin transformer encoder.

Several sophisticated approaches proposed to use contrastive learning in combination with prototypes to achieve robust embeddings. Again, these prototypes are codes which are used as surrogate targets. Following the seminal work of Oord et al. (2018) and their contrastive InfoNCE formulation, the prediction of direct similarities as popularized in (Chen et al., 2020) can be replaced by assignments to a surrogate (Li et al., 2020). In many cases, a momentum-updated teacher student combination allows for a variety of applications and augmentations (Baevski et al., 2022). Robust prototype codes can be either learned (Caron et al., 2020), drawn from support samples (Assran et al., 2021), aggregated into a (compressed or quantized) queue Dwibedi et al. (2021), generated by clustering Yue et al. (2021)) or be derived by predicted embeddings from a target such as a quantized vector out of a non-augmented source image Gidaris et al. (2021). Recently, Hénaff et al. (2022) relied on assignments to online generated dense masks by a k-means algorithm. These masks are in turn aggregated to object level prototypes and mapped to students as targets. An alternative approach has been considered by Ouyang et al. (2020), who proposed the use of offline generated clusters of superpixels for local and global surrogate targets. In general, assignments to prototypes provide a method for generalization unmatched by straightforward reconstruction schemes.

3. Methods

The methodical contributions can be separated into three aspects: incorporated architectural components, a dense self-supervision scheme and the dedicated training scheme. In order to achieve an architecture capable of the aforementioned goals, we first introduce an UNet-like architecture in Section 3.1 that can be prompted throughout the network. Hereby, prompt-able SWin blocks are presented in 3.1.1 which make use of recent advances of self-attention (transformer) layers. These are integrated in an encoder-decoder architecture suitable for medical images. Further adjustments include relative positional encodings as attention bias scores for the image content as well as learned encodings for the calculation of attention bias scores between prompts and image content. Together, they constitute the heterogeneous attention bias score introduced in Section 3.1.2. To keep the architecture flexible from start to end,
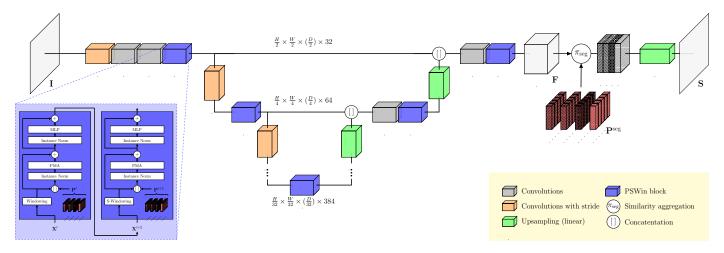
Figure 1: Schematic illustration of the proposed prompt-able UNet (PUNet). The network consists of an encoder with down-convolutions and a decoder with linear upsampling layers. A depth of 5 levels is chosen with 32, 64, 128, 256, 384 hidden channels C for each respective level. Throughout the network prompt-able shifted window (PSWin) blocks are placed. These blocks incorporate prompt-able multi-head attention (PMA) layers. This enable the injection of prompt tokens $\mathbf{P}$ that can be learned in a downstream adaptation task. The decoder embedding $\mathbf{F}$ is further processed by a similarity aggregation π_{seg} together with prompt tokens $\mathbf{P}^{\text{seg}}$ for the prediction of class probabilities in the segmentation map $\mathbf{S}$.

a dedicated output layer is employed by a weighted similarity aggregation across tokens, in order to predict desired class probabilities (Section 3.1.3).

We propose a dense self-supervision scheme applied during pre-training by a contrastive assignment-based loss in Section 3.2. In comparison to most common pre-training schemes, the whole backbone model is optimized due to the densely penalized output. It relies on a student teacher combination with two students, an exponential moving average (EMA)-updated teacher, and online generated prototype targets. In spite of altered variations of similar field of views (FOV), students are enforced to output similar assignments and thereby predicted representations as the teacher model (Section 3.2.1. The online clustering based on soft assignments weighted by relative spatial distances to prototype seeds is established in 3.2.2. The initial prototype seeds are drawn from the teacher predictions itself. The scheme and its formulation are adequate for extension to a concurrent class-conditional supervision by prompt insertion. We propose to apply the prompts per batch element to enhance the adaptation of the learned embedding and thereby its manipulation by instructions during the subsequent downstream adaptation. Resulting variations of the training scheme are explored in 3.3.

3.1. Prompt-able UNet (PUNet)

Instead of using large pre-trained models, as done in the NLP community, we limit ourselves to the investigation of easily applicable small models that can be readily applied to medical data, such as CT and MRT images. In the following, we introduce the dedicated architecture alongside task-dependent prompt tokens. Our prompt tokens can be considered a set of learnable instructions. The tokens aggregate all task-dependent information to achieve a parameter efficient fine-tuning. A new set of prompt tokens is provided for each task with subsets of the tokens representing respective classes for the binary as well as the multi-class case. The frozen backbone model remains task agnostic and thereby class agnostic. Accordingly, the model needs to learn to adapt the embedding predictions and its resulting segmentation masks in dependence of the given prompts. Once learned, the prompt sets can be swapped in accordance to the desired classes (labels) to be predicted. The use of prompts for different training variations is further described in Section 3.3.

We inject the tokens deeply into the network. This allows for an intermediate adaptation of encoded image content throughout the network. Hereby, attention layers provide a structured way to combine and process the heterogeneous encoded image and prompt information. We include memory efficient shifted window (SWin) attention blocks (Liu et al., 2021b) in our architecture, which quickly became popular in the medical field (Cao et al., 2021; Tang et al., 2022). Prior works, such as the UNETR (Hatamizadeh et al., 2022) and SWin-UNETR (Cao et al., 2021), considered a transformer architecture that integrates SWin blocks alongside patch merging and down-projection (linear) layers for dimension reduction. Dense predictions are subsequently retrieved by classical convolutional decoders. The Swin-UNet Cao et al. (2021) extended the model by integrating SWin blocks throughout an encoder-decoder with intermediate patch merging as well as patch expansion layers. However, we take a step back and employ conventional downstream convolutions (with a stride of 2) in the encoder and rely on linear upsampling layers in the decoder. For effective deep adaptation to a downstream task, we introduce prompting on all resolution levels of this prompt-able UNet (PUNet) architecture. The PUNet architecture is illustrated in Figure 1. We note that in this work there is no class token, a common constitutent of ViT approaches (Dosovitskiy et al., 2020; Jia et al., 2022), since we perform only segmentation and no classification. To keep a moderate parameter number, a single prompt-able block is applied after each down convolution and before each upsampling layer, replacing most con-
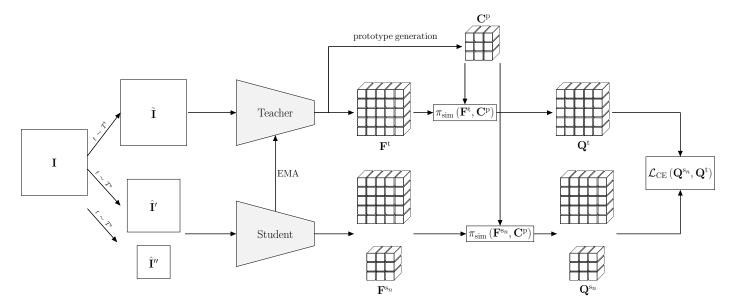
Figure 2: Input image slices **I** are augmented for a student teacher combination. Differently sized views $\hat{\mathbf{I}}'$, $\hat{\mathbf{I}}''$, with varying augmentations including partial masking, are passed to a student as well as a less augmented variant $\tilde{\mathbf{I}}$ to a teacher. The teacher network weights are kept up to date by an exponential moving average (EMA). Teacher output embeddings $\mathbf{F}^{\mathrm{t}}$ are further processed by an iterative clustering with spatially weighted assignments, which results in online generated prototypes $\mathbf{C}^{\mathrm{p}}$. Similarity assignments are calculated by means of a softmaxed cosine similarity π_{sim} with respect to predicted embeddings $\mathbf{F}^{\mathrm{t}}$ and $\mathbf{F}^{\mathrm{s}_n}$ for the nth student, as well as the prototype codes $\mathbf{C}^{\mathrm{p}}$. Derived assignments $\mathbf{Q}^{\mathrm{t}}$ and $\mathbf{Q}^{\mathrm{s}_n}$ are enforced to have similar contrastive prototype assignments (CPA) by a cross-entropy (CE) loss.

volutional layers of a traditional UNet. The attention layers operate on patches of 2×2 for the highest resolution to further save on memory. As such, a convolutional layer with a stride of 2 is being used as first layer. Accordingly, a final upsampling layer is applied to output predictions to achieve the original image (input) resolution. Thus, we are able to provide an architecture that fits the imaging data and allows for the placement of prompt-able blocks throughout the network. A weighted similarity aggregation π_{seg} to a set of prompt tokens precedes the final upsampling to generate class probabilities for the segmentation masks $\mathbf{S} \in \mathbb{R}^{W \times H \times M}$ for M classes (see Section 3.1.3).

3.1.1. Prompt-able SWin (PSWin) blocks

For this work, we propose to use prompt-able shifted window (PSWin) blocks containing attention layers. Like in previous work, a non-promptable variant of such a block consists of two known transformer blocks, with the content being windowed for the first and spatially shifted and subsequently windowed for the second block. A window size of 8×8 is used for both transformer blocks and a shift of 4×4 is applied to the content prior to windowing in the second block. To follow the windowed attention scheme with regular convolutional and upsampling layers in between, the content is rearranged in the respective windowed and shifted windowed content for each promptable block and subsequently reordered in its original embedding shape. The prompt-able block is depicted as part of Figure 1.

The windowed content within a transformer block can be jointly processed alongside prompt tokens. As such, the integrated prompt tokens provide learned information about the target objective within each prompt-able block. This means, for the attention layer within a transformer block the prompt

tokens are broadcasted and concatenated to each windowed content and processed in a prompt-able multi-head attention (PMA) scheme. We note, that the prompt tokens can be inserted objective-dependent for each batch element. The tokens are passed alongside image encodings to the attention layers which themselves are kept frozen after the pre-training phase. As such, a windowed content $\mathbf{X}^{\mathrm{w}} \in \mathbb{R}^{N_{\mathrm{w}} \times C}$ with N_{w} elements and C channels is passed to a prompt-able block together with a set of concatenated prompt tokens $\mathbf{P} = [\mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_M] \in \mathbb{R}^{N_{\mathrm{p}} \times C}$ with N_{p} total tokens and M subsets of tokens. The concatenation $[\cdot, \cdot]$ is performed along the first dimension. A subset of tokens $\mathbf{P}_m \in \mathbb{R}^{T \times C}$ is comprised of T tokens that represent a specific class m. Similarly, T tokens are used per class m in the weighted aggregation scheme introduced in Section 3.1.3. We rely on multiple tokens for each output class to ease learning variable representations for each respective class. For M active classes, this leads to $N_{\mathrm{p}} = M \cdot T$ total tokens that get processed alongside the windowed content in a PMA layer with

$$\mathbf{X}^{\mathrm{w},l+1} = \mathrm{PMA}\left(\left[\mathbf{X}^{\mathrm{w},l}, \mathbf{P}^l\right]\right). \quad (1)$$

for concatenated inputs $\mathbf{X}^{\mathrm{w},l}$ and $\mathbf{P}^l$ at layer l.

Similar forms of this joint processing by an attention layer have been shown to be effective in (Lester et al., 2021; Jia et al., 2022). They represent a simplified form of (Li and Liang, 2021) since prompts are concatenated to the respective layer input, instead of using two dedicated prompt sets for queries and keys respectively. Different from the stated related work, our prompt tokens are passed to windowed content. The similarity score within a head h of a PMA layer is calculated as $\mathrm{score}(\mathbf{Q}_h, \mathbf{K}_h) = \mathbf{Q}_h \mathbf{K}_h^{\mathrm{T}} / \sqrt{C_{\mathrm{head}}} \in \mathbb{R}^{N_{\mathrm{w}} \times (N_{\mathrm{w}} + N_{\mathrm{p}})}$ with C_{head} channels per head, queries $\mathbf{Q}_h = \mathbf{W}_h^{\mathrm{q}} \mathbf{X}^{\mathrm{w},l^{\mathrm{T}}} \in \mathbb{R}^{N_{\mathrm{w}} \times C_{\mathrm{head}}}$ contain-

ing only the windowed content (and no instructions), and keys $\mathbf{K}_h = \mathbf{W}_h^k \left[\mathbf{X}^{w,l}, \mathbf{P}^l \right]^T \in \mathbb{R}^{(N_w + N_p) \times C_{\text{head}}}$ containing both encoded content as well as prompt tokens. This ensues an attention formulation with head attention att_h^w and output $\mathbf{X}^{w,l+1}$ of the next layer $l + 1$ with

$$
\begin{aligned}
\text{att}_h^w &= \text{softmax}(\text{score}(\mathbf{Q}_h, \mathbf{K}_h))\mathbf{V}_h \in \mathbb{R}^{N_w \times C_{\text{head}}} \\
\mathbf{X}^{w,l+1} &= [\text{att}_1^w, ..., \text{att}_{\tilde{H}}^w]\mathbf{W}^{oT} \in \mathbb{R}^{N_w \times C}
\end{aligned}
\tag{2}
$$

for values $\mathbf{V}_h = \mathbf{W}_h^v \left[\mathbf{X}^{w,l}, \mathbf{P}^l \right]^T \in \mathbb{R}^{(N_w + N_p) \times C_{\text{head}}}$, a projection matrix $\mathbf{W}^o \in \mathbb{R}^{h \cdot C_{\text{head}} \times C}$ and $\tilde{H}$ heads. The resulting full transformer block consists of four sub-layers with residual connections. Starting from windowed rearranged encoded content $\mathbf{X}^{w,l} \in \mathbb{R}^{H' \cdot W' \cdot D' \times C}$ from an encoding $\mathbf{X}^l \in \mathbb{R}^{H \times W \times D \times C}$ at layer l, we apply the four steps formulated in equation 3 to each window. Hereby, an PMA layer l receives a separate set of prompts $\mathbf{P}^{l+n} \in \mathbb{R}^{N_p \times C}$ for the nth step. This set is provided to each windowed content $\mathbf{X}^{w,l+n}$ at the respective sub-step.

$$
\begin{aligned}
\mathbf{X}^{w,l+1} &= \text{W} - \text{PMA}(\text{IN}([\mathbf{X}^{w,l}, \mathbf{P}^l])) + \mathbf{X}^{w,l} \\
\mathbf{X}^{w,l+2} &= \text{Linear}(\text{IN}(\mathbf{X}^{w,l+1})) + \mathbf{X}^{w,l+1} \\
\mathbf{X}^{w,l+3} &= \text{SW} - \text{PMA}(\text{IN}([\mathbf{X}^{w,l+2}, \mathbf{P}^{l+2}])) + \mathbf{X}^{w,l+2} \\
\mathbf{X}^{w,l+4} &= \text{Linear}(\text{IN}(\mathbf{X}^{w,l+3})) + \mathbf{X}^{w,l+3}
\end{aligned}
\tag{3}
$$

with W-PMA and SW-PMA representing windowed and shifted windowed PMA blocks as well as instance norms (IN) Ulyanov et al. (2016) in between the PMA and linear layers. Note, that the typical MLP blocks at the end of an attention block Vaswani et al. (2017) are reduced to a singular linear layer to keep an overall similar parameter budget to the SwinUNETR.

3.1.2. Heterogeneous Bias Scores

We introduce attention bias scores to the combination of prompt tokens and windowed embedded image content within the joint attention scheme. As these are two heterogeneous inputs processed by the same PMA layer, we account for it by dedicated bias score generation schemes for relevant entries. For the encoded image content, an additive positional bias $\mathbf{B}^{\text{content}} \in \mathbb{R}^{N_w \times N_w}$ is incorporated in the attention scheme with scores based on relative spatial distances (Ramachandran et al., 2019; Cordonnier et al., 2019). This provides spatial locality to the attention layers of the architecture. In comparison, convolutional kernels provide this locality implicitly. In addition, a learnable bias score $\mathbf{B}^{\text{prompt}} \in \mathbb{R}^{N_w \times N_p}$ manipulating the attention score between prompt tokens $\mathbf{P}$ and encoded windowed content $\mathbf{X}_w$ is proposed. This enables us to also optimize the degree to which encoded content at each attention layer is influenced by its prompts. Further bias scores are not required, since the prompts are not processed beyond the attention operation. As such, we generate a head dependent bias matrix $\mathbf{B}_h = [\mathbf{B}_h^{\text{content}}, \mathbf{B}_h^{\text{prompt}}] \in \mathbb{R}^{N_w \times (N_w + N_p)}$ for each head h that can be added to the head dependent attention score values $\text{score}(\mathbf{Q}_h, \mathbf{K}_h)$ of windowed content and prompts. Overall, the resulting biased attention score yields $\text{score}(\mathbf{Q}_h, \mathbf{K}_h) = \mathbf{Q}_h \mathbf{K}_h^T / \sqrt{C_{\text{head}}} + \mathbf{B}_h / \sqrt{C_{\text{bias}}}$ with C_{bias} bias score channels.

First, we rely on learned bias scores $\mathbf{B}_h^{\text{content}}$ that are added to the attention scores of the windowed content based on relative spatial distances. These scores are dependent on the relative x and y positional pixel distances $d^{\text{row}}(\text{pos}_i, \text{pos}_j)$, $d^{\text{col}}(\text{pos}_i, \text{pos}_j)$ for each element i and j within a respective $N_w \times N_w$ window. Biases are calculated for each dimension based on the row and column differences. More specifically their resulting scalar scores are

$$
\begin{aligned}
b_h^{\text{row}}(d^{\text{row}}) &= \mathbf{w}_h^{\text{row}} \cdot \mathbf{E}^{\text{row}}[d^{\text{row}}] \in \mathbb{R} \\
b_h^{\text{col}}(d^{\text{col}}) &= \mathbf{w}_h^{\text{col}} \cdot \mathbf{E}^{\text{col}}[d^{\text{col}}] \in \mathbb{R}
\end{aligned}
\tag{4}
$$

for the dot product $\cdot$, a learned weight vector $\mathbf{w}^h \in \mathbb{R}^{C_{\text{bias}}}$ of a head h, and a learned embedding for the relative distance d^{row} in $\mathbf{E}^{\text{row}} \in \mathbb{R}^{N_d \times C_{\text{bias}}}$ as well as the relative distance d^{col} in $\mathbf{E}^{\text{col}} \in \mathbb{R}^{N_d \times C_{\text{bias}}}$ with N_d containing all possible distances. Scalar scores are aggregated in $\mathbf{B}_h^{\text{content}}$ according to the paired distances and averaged across both dimensions with $\mathbf{B}_h^{\text{content}} = (\mathbf{B}_h^{\text{row}} + \mathbf{B}_h^{\text{col}})/2$ before being integrated into the overall bias matrix $\mathbf{B}_h$.

Secondly, we integrate bias scores for attention scores between prompts and windowed encodings. For each prompt token t we calculate a bias entry

$$
b_h^{\text{prompt}}(t) = \mathbf{w}_h^{\text{prompt}} \cdot \mathbf{E}^{\text{prompt}}[t] \in \mathbb{R}
\tag{5}
$$

with a head-dependent weight vector $\mathbf{w}_h^{\text{prompt}} \in \mathbb{R}^{C_{\text{bias}}}$, and a learned embedding matrix $\mathbf{E}^{\text{prompt}} \in \mathbb{R}^{N_p \times C_{\text{bias}}}$. The entries of the prompt bias scores $\mathbf{B}_h^{\text{prompt}}$ are unique for each token t of a prompt set $\mathbf{P}$, yet, the same scores are broadcasted to the whole windowed content $\mathbf{X}^w$ (across all windows).

3.1.3. Cosine Similarity Aggregation

In addition to the prompt-able blocks, we employ a cosine similarity aggregation in conjunction with learnable prompt tokens to perform token-dependent class predictions. Unlike typical fine-tuning and prior prompting approaches, as explored by Jia et al. (2022), there is no requirement for a task-specific linear output layer or entire decoder to project representations into the right amount of output neurons. This allows for greater flexibility, since different segmentation classes as well as different amounts thereof can be predicted in each batch element depending on the selected prompt tokens. Segmentation predictions are generated by means of a cosine similarity calculation between the decoder output $\mathbf{F} \in \mathbb{R}^{W \times H \times C}$ and a final learnable set of prompt tokens $\mathbf{P}^{\text{seg}} \in \mathbb{R}^{N_p \times C}$ (see Figure 1). The result of this operation are the desired class probabilities $\mathbf{S} \in \mathbb{R}^{W \times H \times M}$ for each available class. This aggregation scheme can be considered an instance of prototypical networks (Snell et al., 2017) in which clustered prototypes are replaced by learned prompt tokens. A similar approach has been explored by (Strudel et al., 2021) on natural images. However, different from unique class tokens that are processed alongside regular encoded content in their decoder, we employ several learned prompt tokens directly representing respective downstream classes.

A cosine similarity measure is employed between a prompt token $\mathbf{P}_{m,t_m}^{\text{seg}} \in \mathbb{R}^C$ with token index t_m of class m together with

an embedding vector $\mathbf{F}_{i,j} \in \mathbb{R}^C$ for an entry with indices i, j.

$$\tilde{\mathbf{S}}_{i,j,m,t_m} = \text{sim}\left(\mathbf{F}_{i,j}, \mathbf{P}^{\text{seg}}_{m,t_m}\right) = \frac{\mathbf{F}_{i,j} \cdot \mathbf{P}^{\text{seg}}_{m,t_m}}{||\mathbf{F}_{i,j}|| \, ||\mathbf{P}^{\text{seg}}_{m,t_m}||} \in \mathbb{R} \qquad (6)$$

Instead of an average of all tokens T for each class m, a weighted similarity score is used to retrieve class probabilities $\mathbf{S}$ with entries

$$\mathbf{S}_{i,j,m} = \sum_{t_m=1}^{T} \text{softmax}(\tilde{\mathbf{S}}_{i,j,m}/\tau_{\text{agg}})_{t_m} \odot \tilde{\mathbf{S}}_{i,j,m,t_m} \in \mathbb{R} \qquad (7)$$

for the element-wise product $\odot$ and a temperature parameter τ_{agg}. The softmaxed weight does not influence the backward pass during training, since it is excluded from gradient updates. This way not every prompt token has to align to all relevant image content representations belonging to the same class.

3.2. Dense Self-Supervision

To establish robust anatomical representations suitable for further downstream tasks on medical imaging data, we employ a combination of data augmentation and a dedicated self-supervision scheme of aligning online generated prototypes between a teacher network and student networks. The whole scheme is depicted in Figure 2. It generates embeddings where anatomically similar regions are represented close to each other. It incorporates a momentum model with an EMA updated teacher and students, prominently used in (Grill et al., 2020). We incorporate two students, one processing a smaller input $\hat{\mathbf{I}}' \in \mathbb{R}^{W^{s_1} \times H^{s_1}}$ than the teacher input $\tilde{\mathbf{I}} \in \mathbb{R}^{W^t \times H^t}$ and the second one $\hat{\mathbf{I}}'' \in \mathbb{R}^{W^{s_2} \times H^{s_2}}$ more severely cropped to enforce robust embeddings $\mathbf{F}$ with focus on global as well as more local context alike. The smaller student FOVs are cropped out of the teacher FOV. Both students share the same underlying network weights. This multi-crop strategy has been proven beneficial in approaches proposed like (Chen et al., 2020; Caron et al., 2020). Furthermore, the proposed student teacher combination naturally allows for the integration of partially masked content in each student view as part of the augmentation pipeline. Thus, embeddings have to be robustly learned to ensure predicting an assignment similar to the one provided by a non-masked teacher, further enforcing the estimation of output representations based on context. Different from most approaches mentioned in 2.4, we are solely interested in predicting a learned embedding, despite adverse effects being present in the input views, instead of reconstructing the original input itself.

3.2.1. Contrastive Prototype Assignments (CPA)

To facilitate the learning of an effective embedding $\mathbf{F}$, the two students are penalized to predict the same contrastive prototype assignments (CPA) as its teacher. For each student teacher combination a cross entropy loss $\mathcal{L}_{\text{CE}}$ is applied.

$$\mathcal{L}_{\text{CPA}} = \frac{1}{2} \sum_{n=1}^{2} \mathcal{L}_{\text{CE}}\left(\mathbf{\Phi}^{s_n}, \mathbf{\Phi}^t\right) \qquad (8)$$

The teacher provides a guiding assignment $\mathbf{\Phi}^t \in \mathbb{R}^{W^t \times H \times C}$ which has to be replicated by the student by its assignment $\mathbf{\Phi}^{s_n} \in \mathbb{R}^{W^{s_n} \times H \times C}$. Thus, we encourage the prediction of similar output embeddings $\mathbf{F}^t$ of the teacher and $\mathbf{F}^s$ of a student by enforcing identical assignments to online generated prototypes $\mathbf{C}^p \in \mathbb{R}^{N_k \times C}$ for N_k prototypes. We calculate similarity assignment entries via

$$\mathbf{\Phi}_{i,j,k} = \pi_{\text{sim}}(\mathbf{F}, \mathbf{C}^p)_{i,j,k} = \text{softmax}(\text{sim}(\mathbf{F}, \mathbf{C}^p)/\tau_{\text{assign}})_{i,j,k}$$
$$= \frac{\exp\left(\text{sim}(\mathbf{F}_{i,j}, \mathbf{c}_k/\tau_{\text{assign}})\right)}{\sum_{\bar{k}=1}^{N_k} \exp\left(\text{sim}(\mathbf{F}_{i,j}, \mathbf{c}^p_{\bar{k}}/\tau_{\text{assign}})\right)} \qquad (9)$$

for a pixel with indices i, j and a prototype k. We rely on the cosine similarity operator sim defined in equation 6 and a temperature τ_{assign}. For the teacher assignment a smaller temperature value is applied in the softmax than is used in the student, encouraging progressively confident predictions to the generated clusters (Assran et al., 2021). To get the proper target for a student pixel at $\text{pos}^s_{i,j}$, the spatially closest teacher assignment $\mathbf{\Phi}^t_{u,v}$ at position $\text{pos}^t_{u,v}$ is queried. This is possible, since the position $\text{pos}_{i,j}$ of an output pixel with indices i, j is known by a common underlying coordinate grid of the original 3D volume, which is shifted in accordance to spatial image augmentations. For computational efficiency $\mathbf{F}^t$ and $\mathbf{F}^s$ are sampled with a factor of 2 for the loss calculation. Hereby, a random spatial jitter (shift) is applied before sampling $\mathbf{F}^s$ to prevent potential gridding artefacts.

We note, that this CPA loss $\mathcal{L}_{\text{CPA}}$ follows the contrastive formulation known from (Oord et al., 2018). However, since the loss target $\mathbf{\Phi}^t$ is an online generated soft assignment, our self-supervision scheme does not enforce a strict division into positives (that should be attracted) and negatives (that should be repelled). This allows for the concurrent application of a segmentation supervision (see Section 3.3), where the predicted embedding has to adhere to multiple objectives at the same time. The implicit separation also differentiates our method from approaches, such as the self-supervised anatomical embedding of Chaitanya et al. (2020); Yan et al. (2022), where positives and negatives are sampled based on their proximity to the anchor location for multiple (global and local) embeddings. The included pre-defined separation heuristics may be in violation to a second (segmentation) loss target.

3.2.2. Online Prototype Generation

For our case, bootstrapping codes Caron et al. (2020) or maintaining a relevant queue Gidaris et al. (2021) is non-trivial given the partial FOV of the input data and the textural and shape variabilities as well as inhomogeneities encountered in medical images. For similar reasons, we also do not rely on codes represented by superpixels, such as by aggregated online generated (Hénaff et al., 2022). For medical images, superpixels require a deliberate offline scheme (Ouyang et al., 2020) suitable for the underlying data. Instead, our approach falls in the category of derived prototypes. We rely on an iterative clustering scheme based on calculated assignments to update prototype clusters. Hereby, our generation process can
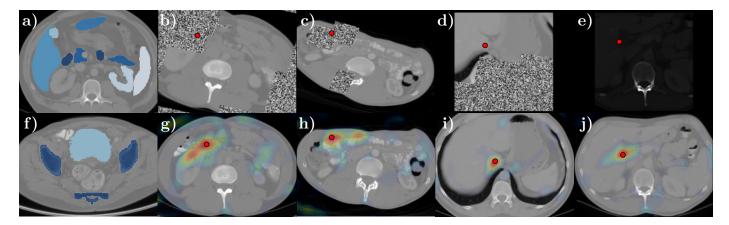
7

Figure 3: a/f) Exemplary slices of the TCIA/BTCV and CT-ORG dataset, with annotated masks shown in shades of blue, b-e) augmented student views with masked regions or strong contrast adjustments, g-j) respective teacher views with overlays of the cosine similarity of the predicted teacher embedding $\mathbf{F}^t$ and the student embedding $\mathbf{F}^s_{i,j}$ at an arbitrary selected point of interest (indicated by a red dot) with indices i, j in the corresponding student view. Highly similar regions appear red in the teacher view. The approach learns a robust embedding that enforces context learning, and is thus able to generate proper similarities, despite the origin region being severely affected. Note, that teacher augmentations have been disabled for better visual clarity in this illustration.

be considered as a k-means clustering with spatially weighted soft assignments. The online generated prototypes $\mathbf{C}^p$ represent clustered representatives of the teacher embedding $\mathbf{F}^t$. For simplicity, we reuse the similarity operator π_{sim} to assign features to prototypes. As such, we calculate prototype assignments $\mathbf{\Phi}^p \in \mathbb{R}^{W^t \times H \times C} = \pi_{sim}(\mathbf{F}^t, \mathbf{C}^p)$ similar to equation 9. Initial prototypes are drawn from interpolated entries of the predicted teacher embedding $\mathbf{F}^t$ at selected seed points. The seed points are defined by a grid overlayed on the teacher embeddings. The grid size and as such the amount of clusters is being determined by a reduction factor with respect to the output size of $\mathbf{F}^t$.

To further guide the clustering process, the assignments of each entry to the current prototypes are weighted by a positional bias with $\tilde{\mathbf{\Phi}}^p = \mathbf{W}^p \odot \mathbf{\Phi}^p$. This promotes a spatial proximity to clusters with centroids in a close neighborhood being preferred. The weight vector $\mathbf{W}^p$ is derived from the relative spatial distances between the pixel with indices i, j and the position of the prototype k. The weight entries are defined by the relative Gaussian distance

$$\mathbf{W}^p_{i,j,k} = \exp\left(-\frac{\|pos^t_{i,j} - pos^p_k\|^2}{2 \cdot \sigma^2}\right) \qquad (10)$$

with σ^2 being determined based on a pre-defined full width at half maximum (FWHM) parameter with the width given in pixels. For each iteration a new resulting prototype centroid $\mathbf{C}^p_k$ and its new position pos_k are derived, based on the degree to which the teacher embedding $\mathbf{F}^t$ belongs to the prior iteration of prototype centroids, represented by the $\tilde{\mathbf{\Phi}}^p$.

$$\mathbf{C}^p_k = \frac{\sum_{i,j} \tilde{\mathbf{\Phi}}^p_{i,j,k} \mathbf{F}^t_{i,j}}{\sum_{i,j} \tilde{\mathbf{\Phi}}^p_{i,j,k}} , \quad pos^p_k = \frac{\sum_{i,j} \tilde{\mathbf{\Phi}}^p_{i,j,k} pos^t_{i,j}}{\sum_{i,j} \tilde{\mathbf{\Phi}}^p_{i,j,k}} \qquad (11)$$

In essence, we sample features of a momentum-updated teacher and calculate weighted assignment to generate spatially guided online prototype centroids. In turn, these assignments to the newly generated prototypes of the predicted teacher and student

embeddings are enforced to be similar by the CPA loss $\mathcal{L}_{CPA}$. Working with online generated prototypes, we also avoid the need for further losses, such as prevalent in mutual information maximization schemes (Peng et al., 2021), to enforce well distributed assignments. In addition, relying on the online prototype generation brings the benefit of avoiding a complex queue all together without sacrificing convergence to similar representations for similar regions. Contrary, using a queue would require to establish and update a diverse and representative set that represents meaningful targets. Overall, no further guidance such as pre-defined heuristics (Chaitanya et al., 2020) or auxiliary labels (Ouyang et al., 2020) are required. The scheme shares some similarity with the work of Hénaff et al. (2022), which relies on masks generated by a k-means algorithm which are mapped on to the student FOVs and used as assignment targets. Contrary to (Hénaff et al., 2022), we do not use aggregated object representations, but simply use the generated prototypes. Instead, our soft assignment allows for the assignment of pixels to multiple online generated prototypes which may share similar characteristics.

3.3. Training Procedure

With the prompt-able architecture and the dedicated self-supervision scheme in place, we can train the given neural network and subsequently adapt the model in a parameter-efficient manner to an unseen downstream task. For clarity, we introduce two distinct training stages: phase 1 (P1) where a fully learnable model is (pre-)trained and phase 2 (P2) where the pre-trained model is adapted by selective parameter training to the downstream tasks. We investigate the performance of the proposed approach by considering several experiments, where the model is frozen in P2. In some cases, we are also interested in full fine-tuning of the whole model and consider this as non-frozen adaptation. When a model is denoted as frozen, the bulk of the architecture is non-trainable, but prompts $\mathbf{P}$ throughout the network, including $\mathbf{P}^{seg}$, as well as attention bias scores b^{prompt}_h remain trainable.
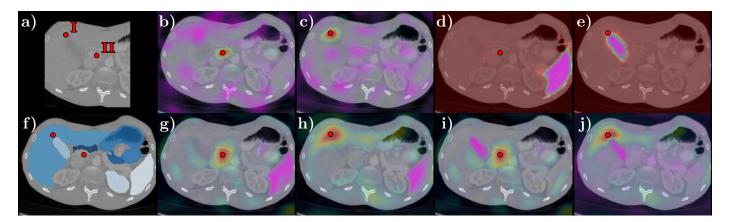
8

Figure 4: Visualization of cosine similarities between predicted teacher embeddings $\mathbf{F}^t$ and a student embedding $\mathbf{F}^s_{i,j}$ at arbitrary selected points of interest (red dots with labels I and II) with respective indices i, j for the student view shown in (a). f) The respective teacher slice with points of interest located closely above the gall bladder (I) and below the pancreas (II). b-c) Teacher views with cosine similarities for the self-supervised pre-trained model for respective points of interest. The similarity is densely concentrated around the queried point. d-e) Cosine similarities of the segmentation supervised pre-trained model with prompt tokens active for the spleen (d) and the gallbladder (e). Pink regions indicate highly dissimilar regions (inverse of the similarity map) and serve as indication of the generated segmentation masks. There is no difference with respect to location outside of the active target region. g-j) Similarity maps for the combined self- and segmentation supervised pre-trained model. Densely concentrated similarities are visible alongside highly dissimilar active target regions. In (j) it can be seen, that the prompt tokens can successfully alter regions that showed high similarity in (h) but belong now to the active target region (by the change from red (in h) to pink (in j) in the periphery of the point of interest.

In phase 1, the self-supervision loss $\mathcal{L}_{\text{CPA}}$ is applied. If this remains the sole loss, the prompt-able blocks process only encoded image content, with the prompts and its bias scores not being inserted in the PMA layer. However, the prompts are considered if a concurrent segmentation loss is applied, making use of the prompts and thereby enabling the network to learn to prompt during pre-training. For this case, softmaxed predictions of each pixel $\hat{y}_i$ are penalized by a class-weighted focal loss Lin et al. (2017)

$$\mathcal{L}_{\text{focal}} = -\frac{1}{W \cdot H} \sum_{i=1}^{W \cdot H} \sum_{l=1}^{M} \alpha_m (1 - \hat{y}_i)^\gamma \log{(\hat{y}_i)} \, y_{i,l} \quad (12)$$

with a focusing parameter γ, one-hot encoded true mask values $y_{i,m}$ and weighting factors α_m for M label categories (classes). α_m is calculated for each class based on a heuristic which builds on the average foreground to background ratios of a training set.

Since the architecture allows for batch element dependent prompt insertion, classes declared to be used for training can be randomly drawn and applied in a joint batch. As such, respective (learned) prompt tokens $\mathbf{P}$ are stored for each combination of classes and drawn in accordance to the selected task. At the same time, the shared neural network weights get trained with respect to all available tasks and their classes. This allows for efficient pre-training not only of a suitable representation induced by the self-supervision, but also class-dependent predictions which aid in enforcing good generalizations to further unseen classes during adaptation. Thus, we keep gradual differences for anatomical regions while learning to adapt the predicted embedding to follow clear separations between known prompt-dependent regions. Hence, in its most advanced form, we train the model on a subset of all available classes and perform the adaptation on a disjoint set. As such, our scheme can be seen as a combination of supervised and self-supervised

learning to provide more robust classification or segmentation results in the case of scarce annotations. In phase 2, we rely solely on the aforementioned focal loss $\mathcal{L}_{\text{focal}}$ to penalize the correct prediction of classes based on aggregated class probabilities.

Several variations of the prompting scheme can be considered independent of the training phases. The architecture can be used to provide binary predictions, with each class having their own background and foreground tokens, or with a dedicated set of background and foreground prompt tokens for each class. The latter one describes the typical multi-class case. It is also possible to provide multiple binary segmentations and recombine them in a multi-class segmentation in a post-processing step. Following the typical fine-tuning scheme, one can ignore the instructions and use the proposed network architecture with a fixed linear layer to project embeddings to the desired class probabilities. In that case, a simple linear (projection) layer provides a fixed number of classes instead of using the proposed similarity comparison with prompt tokens.

4. Results

4.1. Experimental Setup

We conduct our experiments on two publicly available medical CT datasets. The first is a joint set, the TCIA/BTCV dataset Gibson et al. (2018), comprised of 89 subjects with densely annotated 3D volumes of the The Cancer Imaging Archive (TCIA) Pancreas-CT dataset Roth et al. (2016) and the Beyond the Cranial Vault (BTCV) abdomen dataset Landman et al. (2015). We include eight organ masks, that are part of both sub-groups, namely the spleen, left kidney, gallbladder, liver, esophagus, stomach, pancreas and duodenum. The second, the CT-ORG dataset Rister et al. (2019) includes a diverse set of five organs for 139 subjects. Here the liver, the bladder, both kidneys, but

Method	Parameters	Amount	TCIA/BTCV		CT-ORG	
			P1	P2	P1	P2
nnUNet	30.0M		**83.94 ± 4.64 (84.70)**	–	**91.44 ± 6.15 (93.15)**	–
UNet	5.0M		82.45 ± 4.48 (82.97)	82.93 ± 4.11 (84.39)	89.01 ± 6.95 (90.53)	89.90 ± 5.75 (91.56)
UNETR	87.3M		78.31 ± 5.64 (79.44)	79.50 ± 4.80 (81.14)	87.96 ± 6.64 (90.24)	88.83 ± 6.06 (90.35)
SwinUNETR	6.3M	100%	82.15 ± 4.41 (83.37)	82.18 ± 3.96 (82.21)	89.67 ± 6.26 (91.75)	**90.18 ± 5.98 (92.05)**
PUNet (binary)	6.8M		82.07 ± 4.38 (81.85)	**83.45 ± 3.64 (84.02)**	88.50 ± 6.48 (90.13)	89.74 ± 6.13 (91.39)
PUNet (multi c.)	7.3M		79.92 ± 5.52 (79.66)	80.98 ± 5,17 (81.03)	90.64 ± 6.01 (92.45)	89.75 ± 6.74 (91.28)
PUNet (fixed)	6.8M		82.50 ± 3.88 (82.15)	82.45 ± 3.70 (82.85)	90.25 ± 6.09 (91.95)	89.70 ± 6.12 (91.02)
nnUNet	30.0M		66.64 ± 10.92 (67.64)	–	**88.06 ± 10.48 (91.50)**	–
UNet	5.0M		68.87 ± 6.85 (68.63)	72.13 ± 6.11 (72.66)	86.05 ± 10.79 (89.51)	85.90 ± 12.04 (90.10)
UNETR	87.3M		65.85 ± 7.88 (67.14)	70.34 ± 5.42 (70.61)	84.68 ± 11.25 (88.68)	85.82 ± 9.43 (88.22)
SwinUNETR	6.3M	10%	67.76 ± 7.20 (66.99)	**72.71 ± 5.74 (72.42)**	86.00 ± 11.03 (89.80)	86.49 ± 11.06 (89.52)
PUNet (binary)	6.8M		70.78 ± 7.04 (72.41)	72.03 ± 7.14 (72.60)	86.39 ± 13.10 (90.71)	**87.97 ± 9.10 (91.01)**
PUNet (multi c.)	7.3M		67.59 ± 7.30 (68.76)	69.91 ± 7.07 (70.50)	86.66 ± 9.64 (89.68)	86.72 ± 10.97 (90.82)
PUNet (fixed)	6.8M		**71.00 ± 7.80 (72.58)**	71.43 ± 7.35 (72.32)	86.83 ± 10.53 (90.18)	87.30 ± 8.82 (89.84)

Table 1: Mean ± standard deviation (median) of the segmentation performance expressed by the Dice Similarity Coefficient (DSC, in %) for the two investigated TCIA/BTCV and CT-ORG datasets. The nnUNet is trained within its framework. All other UNet baseline variants (UNet, UNETR, SwinUNETR) are integrated by means of the MONAI framework and trained in identical fashion to the prompt-able UNet (PUNet) variants. The variants differ by using prompt tokens for binary predictions (binary), concurrent prompt tokens for each class (multi class) and a fixed output layer (fixed) in which case no prompt tokens are passed to the prompt-able blocks. Architectures are investigated for a single phase 1 (P1) training where only a segmentation loss is applied for the present case and a two phase performance (P2) where a pre-training is applied prior to a full fine-tuning on the respective classes. In addition, runs are performed for 100% and 10% of all available training data.

also visually distinct structures such as the lungs and the general bone structure are annotated. This dataset includes contrast and non-contrast enhanced CT scans. We use a fixed random split for all datasets, corresponding to a 70% training, 10% validation, and 20% test split. This results in splits of 60/10/19 subjects for TCIA/BTCV and 97/14/28 subjects for CT-ORG. Exemplary axial slices with overlayed segmentation masks are shown in Figure 3 (a/f). For the joint application of losses, we split the classes of the datasets into two disjoint groups. For TCIA/BTCV, we have the spleen, left kidney, gallbladder, and liver in the first group and the esophagus, stomach, pancreas and duodenum in the second group. For CT-Org the liver, bladder, and kidneys are grouped together with the lungs and bones comprising the second group.

In a pre-processing step, all volumes are re-scaled to a target resolution of 1.25 mm × 1.25 mm × 2.5 mm and cropped to an in-plane matrix size of 280 × 280 pixels. We set input FOVs to 256 × 256 pixels for the teacher, and 224 × 224 as well as 160×160 pixels for the two students. Facilitated by the MONAI framework (MONAI Consortium, 2020), the data augmentation includes a clipped intensity re-scaling, random spatial crops in accordance with the input sizes of the teacher and both students, random bias field, contrast adjustments, histogram and intensity shift and scales as well as affine transformations including rotations, scale and shear changes. In addition, random regions of student inputs are locally masked with dropouts and shuffling of the respective content.

The training procedure differs between the two phases P1 and P2. Phase 1 follows a training process with an Adam optimizer with decoupled weight decay (AdamW) (Loshchilov and Hutter, 2017) with a learning rate of 10^{-4} for neural network parameters, 10^{-3} for prompting parameters, and a weight decay of 10^{-2}. P1 is applied for 400 epochs of 5000 samples drawn out of the training set per epoch. In Phase 2 an Adam optimizer is used with a learning rate of 5×10^{-4} for eligible network parameters (relevant for ablation architectures) and 5×10^{-3} for prompt parameters. This phase is shortened to 100 epochs (for computational reasons) and provided with a one cycle learning rate scheduler (Smith and Topin, 2019). Further, hyperparameters are set empirically. Loss weights of 1.0 for the supervised loss $\mathcal{L}_{\text{focal}}$ and 10^{-2} for the self-supervised loss $\mathcal{L}_{\text{CPA}}$ are used. A γ of 4.0 is selected for the focal loss. For the self-supervision, a FWHM of 128 (pixels), a prototype cluster reduction factor of 8, softmax temperatures τ_{assign} of 0.033 for the teacher and 0.066 for students are applied. The softmax temperature τ_{agg} of the similarity aggregation is set to 0.1.

Throughout the architecture, batch normalization for convolutions and instance normalization within prompt blocks are used alongside leaky ReLU activations. A depth of 5 levels with 32, 64, 128, 256, 384 hidden channels C for each level is chosen. $\tilde{H} = 8$ attention heads are used for each PMA layer together. Within a PMA layer, we use $C_{\text{head}} = C/\tilde{H}$ channels per head and $C_{\text{bias}} = 32$ channels for instruction bias scores. In addition, $T = 16$ prompt token vectors are used to represent a class.

Predictions are evaluated by means of the Dice similarity coefficient (DSC), as a measure of the overlap quality between ground truth annotation and prediction, and the average symmetric surface distance (ASSD), to assess surface distances between ground truths and predictions. Metrics are calculated from whole 3D volumes. Metrics are reported as mean ± standard deviation across all test subjects of the average value of all foreground classes if not stated otherwise. In addition median values (in brackets) are mentioned. For hypothesis testing, p-values of a two-sided paired t-test are reported to assess differences in mean populations. A significance threshold of 0.05 is set for comparisons with respect to our proposed method.

		Variant	P1		TCIA/BTCV		CT-ORG	
			Seg.	Self	DSC	ASSD	DSC	ASSD
P2	Non-frozen	Joint	✓	✓	83.13 ± 3.86 (83.96)	1.92 ± 0.92 (1.73)	87.18 ± 6.53 (89.91)	3.64 ± 2.83 (2.97)
		Seg.	✓	-	83.10 ± 3.79 (83.47)	2.02 ± 1.09 (1.62)	**89.90 ± 6.02 (91.45)**	**3.17 ± 3.20 (1.89)**
		Self	-	✓	**83.45 ± 3.64 (84.02)**	**1.83 ± 0.92 (1.55)**	89.74 ± 6.13 (91.39)	3.29 ± 3.44 (2.10)
		Random	-	-	82.21 ± 3.63 (81.69)	2.05 ± 1.16 (1.69)	85.02 ± 7.28 (87.80)	6.80 ± 7.01 (4.47)
	Frozen	Joint	✓	✓	**79.62 ± 3.81 (79.55)**	**2.30 ± 1.14 (1.97)**	**87.23 ± 6.87 (88.42)**	6.59 ± 12.93 (2.68)
		Seg.	✓	-	75.73 ± 5.15 (75.49)	3.31 ± 1.76 (2.61)	82,13 ± 7.55 (83.83)	8.92 ± 8.50 (4.77)
		Self	-	✓	73.88 ± 4.69 (74.82)	3.01 ± 1.85 (2.52)	84.14 ± 6.88 (86.51)	**4.68 ± 4.09 (3.76)**
		Random	-	-	60.96 ± 7.02 (62.63)	5.35 ± 2.36 (4.52)	73.40 ± 8.30 (73.95)	10.11 ± 6.94 (8.50)

Table 2: Mean ± standard deviation (median) of the downstream segmentation performance depicted by the Dice Similarity Coefficient (DSC, in %) and Average Symmetric Surface Distance (ASSD, in mm) for different training scheme variations on the TCIA/BTCV and CT-ORG datasets. We included the following combinations of self- and segmentation supervision in the pre-training phase 1 (P1): self- and segmentation (joint), segmentation (seg.), self-supervision (self) and a random initialization (random) of the weights without any applied losses in P1. For the downstream phase 2 (P2), we differentiate between a trainable backbone architecture (non-frozen) and a non-trainable architecture (frozen) except for the prompt tokens (and its dedicated attention bias scores). For variants including a segmentation loss, two models are trained with subsets of the classes seen during P1 and a disjoint set seen during P2.

4.2. Experiments

In the following, we provide six experiments alongside qualitative examples to show the effectiveness of the aspects proposed in this work. We cover comparisons with state-of-the-art architectures, the impact of the self-supervision scheme, different pre-training combinations, various downstream adaptation approaches besides learning prompt tokens, the insertion positions of prompt tokens, and the behaviour in the annotation scarce case.

Qualitative results of the proposed self-supervised training scheme are portrayed in Figure 3 and 4. Hereby, we calculate cosine similarities between a predicted teacher embedding $\mathbf{F}^t$ and exemplary points of interest (indicated by a red dot) of the student embedding $\mathbf{F}^s_{i,j}$ for indices i, j. The overlay of the resulting similarity map on the teacher FOV depicts highly similar regions in red. For Figure 3, the similarity of arbitrary selected points in several augmented student views (b-e) is visualized for their corresponding unmasked teacher views (g-j). Despite severe augmentations, including extensive masking, the resulting cosine similarity values, overlayed in the teacher view, enable the identification of the original region. In Figure 4 the cosine similarity for two arbitrary points of interest of a student view (a) and the respective cosine similarity to predicted teacher embeddings are overlayed for the different training scheme variations. In dependence of the application of the losses in the pre-training phase 1, the network is able to identify similar regions for the self-supervision (*self*, b-c), segment annotated regions for the segmentation supervision (*seg.*, d-e) and localize regions while distinctively separating semantic regions for the application of both losses (*joint*, g-j).

We compare our introduced PUNet architecture in conjunction with and without the proposed self-supervision scheme. We provide several popular architectures as reference, namely the UNet, UNETR, and SWinUNETR. We train them with the same losses and augmentations as the PUNet. In addition, we consider the established nnUNet framework (Isensee et al., 2021), which provides a robust set of extensive augmentations, a well-tested architecture and its own combination of Dice and focal losses. We calculate values for phase 1 (P1) and phase 2 (P2) where possible. For this experiment,

P1 indicates a sole segmentation loss and P2 represents a self-supervised pre-training during P1 followed by full fine-tuning (non-frozen) in P2. Furthermore, we include the performance for 10% of the original annotated training data. Results for all cases on the TCIA/BTCV and CT-ORG datasets are depicted in Table 1. The reported DSC values show that all variants of the PUNet, the prompted binary and multi class variants, as well as a non-prompted variant with a fixed linear output layer, achieve a comparable performance to the related architectures. The PUNet provides a similar performance as the SwinUNETR with a comparable parameter count. Interestingly, inserting prompts into the prompt-able blocks and relying on the aggregation scheme does not lead to a performance decrease, compared to the fixed architecture variant. We also see, that the nnUNet is superior in P1. This gap is less prevalent if its performance is compared to DSC values of P2 with e.g. a difference of 0.94 percentage points (pp) between the nnUNet and the PUNet (binary). As expected, the benefit of the self-supervision scheme is more prominent when using only 10% of the available training data, especially for TCIA/BTCV where an increase in DSC is seen regardless of the used architecture. This effect is less pronounced for the relatively larger CT-ORG dataset.

In the second experiment, we investigate variations of the training schemes in P1 introduced in Section 3.3. We evaluate the impact on metric values in P2. Note, that in this case all architectures in P2 can be either non-frozen for full fine-tuning or frozen for selective training. We consider four different training schemes for P1. As a baseline we introduce a random weight initialization with no actual pre-training performed. The other three variants are the proposed self-supervision scheme (*self*) where no prompts are inserted into the prompt blocks in P1, pre-training by a sub-group of available annotations under use of class-conditional prompts in P1 with downstream adaptation on unseen classes in P2 (*seg.*), and a combination of the class-conditional segmentation and self-supervision losses (*joint*). For the latter case, pre-training and the respective adaptation is performed on the disjoint groups respectively. Mean DSC values for all cases are considered in Table 2. Naturally, the non-frozen models which allow for full parameter adapta-

Method	Parameters	Trainable	TCIA/BTCV		CT-ORG	
			DSC	p-value	DSC	p-value
Fixed	130	0.00%	24.52 ± 5.07 (23.04)	< 0.05	32.38 ± 4.58 (32.98)	< 0.05
Bias	15k	0.23%	69.00 ± 11.13 (70.69)	< 0.05	78.87 ± 7.03 (80.25)	< 0.05
Prompting - w/o prompt bias scores	44k	0.64%	77.73 ± 4.14 (77.50)	< 0.05	85.30 ± 6.87 (87.74)	< 0.05
Prompting	57k	0.85%	79.62 ± 3.81 (79.55)	1.00	87.23 ± 6.87 (88.42(	1.00
Bias + prompting	73k	1.08%	79.94 ± 4.10 (80.15)	0.23	88.14 ± 6.31 (90.01)	< 0.05
Adapter	325k	4.81%	81.07 ± 4.69 (82.13)	< 0.05	88.92 ± 6.33 (90.52)	< 0.05
Decoder	2948k	43.63%	**82.49** ± **4.50** (**83.20**)	< 0.05	**90.51** ± **6.05** (**92.25**)	< 0.05
Fine-tuning (non-frozen)	6816k	100.85%	83.13 ± 3.86 (83.96)	< 0.05	**87.18** ± **6.53** (**89.91**)	< 0.05
Fine-tuning (non-frozen) - fixed	6758k	100.00%	**83.53** ± **3.86** (**83.99**)	< 0.05	84.00 ± 7.20 (85.32)	< 0.05

Table 3: Mean ± standard deviation (median) of the downstream segmentation performance of different adaptation schemes depicted by the Dice Similarity Coefficient (DSC, in %) on the TCIA/BTCV and CT-ORG datasets. Statistical differences (p-values) are calculated between alternatives and our prompting variant. All shown approaches relied on the same jointly pre-trained backbone architecture, which is considered frozen during the downstream adaptation. The number of additional parameters is indicated in absolute values alongside the percentage with respect to the backbone architecture.

tion provide an upper baseline for the frozen counterparts. In the non-frozen case, there is little difference between the *self*, *seg.*, and *joint* schemes. There is a performance drop-off of 1.24 pp for TCIA/BTCV and 4.88 pp for CT-ORG when comparing the random baseline to the best variants. For a frozen model however, a good initialization becomes mandatory, with the self-supervision and segmentation pre-training both providing substantial increases in DSC values above the random initialization. The combination of both pre-training schemes is even more beneficial and is 3.83 pp and 2.67 pp below the non-frozen models. The ASSD values, indicating deviations from the predicted segmentation mask boundary to the ground truth, shows a similar behavior to the DSC values, with lowest values achieved for the *joint* scheme in the frozen case. Yet, they differ for the CT-ORG data. This is only the case due to a skewed distribution, since the median value 2.68 is still well below the value of the self variant of 3.76. In all cases, we see, that even the random initialization is adaptable to new classes by the inserted prompt tokens, albeit with a larger performance decrease.

As seen in the previous experiments, we establish the combination of pre-training and prompting as a valid scheme for a downstream adaptation to unseen classes. Next, we want to identify how effective it is in operating on a frozen model compared to established approaches. As such, we consider a series of common adaptation approaches and report results alongside the amount of trainable parameters in Table 3. All ablations use the same PUNet backbone architecture, with the self-supervision and segmentation pre-training applied in P1. We consider the following ablations: a trainable linear output layer (fixed), trainable bias and normalization parameters (bias) and the combination of bias and prompting (bias + prompting), prompting without additional attention bias scores for prompt tokens, an additional linear layer (including normalization and activation) at the end of each prompt-able block (adapter), a non-frozen (fully trainable) decoder (decoder), as well as full fine-tuning for the prompted and non-prompted architecture. Results are reported with respect to DSC values and a p-value indicating statistical differences between the respective variant and our proposed prompting scheme. A fixed layer incurs the least parameter adjustments (130), followed by bias adjustments (15k), prompting (57k) and the adapter variant (325k). All require less than 5% of parameters to be adjusted com-

pared to a full fine-tuning. The decoder already requires a large amount of parameter adjustments of 43.63%. For the performance, we see that full fine-tuning as well as the decoder variant are superior to parameter efficient schemes. With decreasing amounts of parameters the performance gradually continues to worsens with substantial drop-offs for the bias and fixed variants. This trend is present for TCIA/BTCV and CT-ORG alike. The single fixed layer, is not sufficient to provide adequate adaptation. Yet, our prompting scheme is only 2.87 pp and 3.28 pp in DSC below the decoder adaptation while requiring only 1.93% of its amount of parameter. It is also close in performance to the adapter variant with a difference of 1.45 and 1.69 pp in DSC with 17.5% of the amount of its parameters. All results are statistically significant, except for the prompting and bias + prompting variant on TCIA/BTCV, where the differences are minor.

To further deepen our understanding of the efficacy of the adaptation schemes, we consider the case of limited annotated training data during the downstream task in P2. This does not affect P1, where the whole training data and respective masks (of seen classes) are still available for the self- and segmentation supervision. However, in P2 only few annotated subjects for the new unseen classes are made available. We consider a subset of the variants introduced in the previous experiment. In addition, we include a prompting variant with only self-supervision and a variant with only segmentation supervision. We vary the number of available annotations from 2 subjects to 4, 8, and the whole TCIA/BTCV dataset. To relate the amounts to the first experiment, 2 subjects correspond to 3.3%, 4 subjects to 6.7% and 8 subjects to 13.3% of the whole training dataset. The boxplots are depicted in Figure 5. The overall order of the performance of different schemes seen in Table 3 remains mostly intact for lower amounts of annotated data. The combination of self supervision and segmentation pre-training is superior to each of the losses alone as well as the bias adaptation for all amounts. This variant is also slightly better than the adapter approach for 4 and 8 subjects. As before, there is little benefit in applying bias adaptations in combination with the prompt tuning. Full fine-tuning is superior in most cases, however, for 2 and 4 subjects, decoder only training lead to the best adaptation. To a lesser degree, this aspect is also present for the comparison of our joint scheme and the full fine-tuning, with
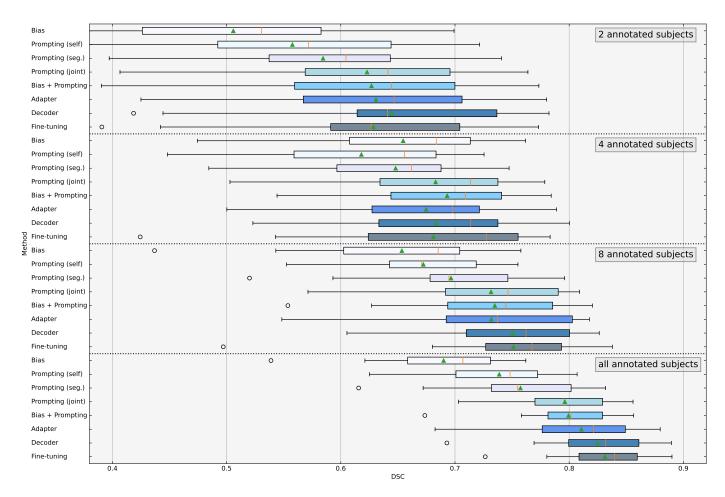
Figure 5: Boxplots depicting the downstream segmentation performance of adaptation variations by the Dice Similarity Coefficient (DSC, in %) on the TCIA/BTCV dataset for varying amounts of available annotated subjects during the adaptation phase. The amount of annotated subjects (whole 3D volumes) includes 2, 4, 8 as well as the whole training set (60). Besides the differently pre-trained prompting variants (seg., self, joint), the alternatives make use of the jointly pre-trained network. Fine-tuning refers to the fully prompted and non-frozen variant. The boxplots include a mean (green diamond) and a median (orange line).

both performing similarly for 2 and 4 subjects.

For architectural ablations, we considered variants of the final similarity aggregation scheme. We differentiate between the weighted aggregation, as proposed in Section 3.1.3, a top-k selection with $k = 3$, indicating that the three most similar prompt token vectors are considered eligible, and a simple mean aggregation, where the similarity of all token vectors of a class is simply averaged to identify its overall similarity score. In addition, we investigated the impact of amounts of tokens used to represent a class. Hereby, we vary the amount from a single token to 32 tokens. Note, that the amount of tokens also influences the amount of parameters, that can be adjusted. Investigations are performed on the TCIA/BTCV dataset and depicted in Table 4. The proposed scheme is slightly ahead, followed by top-k and the mean aggregation. However, the differences are statistically insignificant as shown by a high p-value (> 0.05). With each token the performance drastically increased with respect to DSC as well as ASSD. This trend diminishes for higher amounts of tokens. For 16 and 32 tokens, the difference became insignificant and the DSC and ASSD values stagnated.

Further, we can vary the selection of PSWin blocks at which prompts are inserted into the network. It is possible, to use deep prompting throughout the architecture (ours, full), to only rely on the final similarity aggregation (sim. agg.), which is similar to earlier prototype networks, to consider only the first prompt block (and in this work the final aggregation), to make it more akin to large language model prompting (start), or to rely only on the final aggregation in combination with prompt-able blocks adjusted solely in the encoder or decoder. The impact of the placement is reported by DSC and p-values in Table 5. As expected, deep prompting performs best, with the decoder only variant following with a decrease of 2.8 pp in DSC. This is more important, than early prompting in the architecture. The ablation also shows, that merely adding prompt tokens to the first prompt-able block in the encoder and the final aggregation is not sufficient to achieve adaptation to an unseen class.

5. Discussion

The proposed architecture enables the insertion of prompt tokens throughout the whole segmentation network. This is not possible with classical networks in the medical field which relied predominantly on convolutions, which are inherently limited in their ability to process heterogeneous content in an effi-

13

Method	DSC	ASSD	p-value
Weighted agg. (std.)	**79.62 ± 3.81**	**2.30 ± 1.14**	1.00
top-k agg.	79.34 ± 3.72	2.31 ± 1.10	0.11
Mean agg.	79.23 ± 4.62	2.38 ± 1.57	0.34
$T = 1$	52.19 ± 5.11	7.47 ± 3.14	< 0.05
$T = 2$	70.22 ± 4.05	3.85 ± 1.76	< 0.05
$T = 4$	74.82 ± 4.15	3.01 ± 1.66	< 0.05
$T = 8$	76.85 ± 4.29	2.71 ± 1.83	< 0.05
$T = 16$ (std.)	**79.62 ± 3.81**	**2.30 ± 1.14**	1.00
$T = 32$	79.51 ± 4.16	2.47 ± 1.72	0.71

Table 4: Mean ± standard deviation (median) of the Dice Similarity Coefficient (DSC, in %) and Average Symmetric Surface Distance (ASSD, in mm) and p-values with respect to the standard (std.) variant for different ablations during the downstream adaptation on the TCIA/BTCV dataset. We compared the proposed weighted similarity aggregation (weighted agg.), a top-k selection among available class tokens with $k = 3$ (top-k), and a mean aggregation (mean agg.) where similarities are averaged across all tokens of a class. In addition, the amount of tokens T per class within every prompt-able block and within the final similarity aggregation is varied between 1 and 32.

Method	Parameters	DSC	ASSD
Full	57k	**79.62 ± 3.81**	**2.30 ± 1.14**
Sim. agg.	675	1.58 ± 0.85	53.08 ± 26.36
Start (+ sim. agg.).	2k	5.58 ± 1.11	20.39 ± 9.17
Encoder (+ sim. agg.).	37k	70.37 ± 5.23	3.34 ± 1.35
Decoder (+ sim. agg.).	21k	76.82 ± 4.59	2.85 ± 1.86

Table 5: Mean ± standard deviation (median) of the downstream segmentation performance of different prompt insertion variants depicted by the Dice Similarity Coefficient (DSC, in %) and the Average Symmetric Surface Distance (ASSD, in mm) on the dataset alongside the amounts of prompt token (and attention bias score) parameter required for the adaptation.

cient manner. Despite the comparatively small number of available model and even fewer prompt parameters, our contribution shows great efficacy in the adaptation to unseen classes on two CT datasets with varying organs and structures. The underlying architecture delivers comparable performances to the recent SwinUNETR. This is the case for binary, multi-class and fixed predictions alike, despite the PUNet operating on half the resolution and thus relying on a simple final upsampling layer. No architectural alterations have to be performed when switching the prompts. Thereby, several segmentation targets can be trained on concurrently in a single batch (on frozen and non-frozen models). We show that prompting is effective at adapting a frozen pre-trained model in a downstream fine-tuning adaptation, significantly closing the performance gap between full fine-tuning and prompt tuning. The heterogeneous bias score further aids in adjusting the network to unseen classes. We find that a certain number of prompt tokens for each class suffice with further tokens leading to diminishing returns. We present a token dependent aggregation scheme that can replace a fixed output layer.

The self-supervision pre-training scheme with its online generated prototypes leads to a robust embedding, where anatomical regions can be distinguished from each other. This is the case in spite of heavy masking or the presence of a concurrent prompting objective. As can be seen, in Figure 3 and 4 the pre-training scheme not only allows for enhanced performance in the downstream task, but would also be usable for landmark localization and could be further strengthened for this

task by adjustment of the FWHM parameter in the online clustering. It also shows a beneficial increase in DSC values across different architectures. The benefit of including segmentation masks directly in the pre-training is shown by significantly increased DSC and reduced ASSD values. For TCIA/BTCV the segmentation pre-training is even more valuable than the self-supervision. With a segmentation pre-training, the network is able to learn to incorporate prompts in a more efficient way and is also acquainted with delineating borders for sharp mask predictions as can be seen by lower ASSD values. The introduction of our self-supervision as well as the segmentation supervision is most beneficial in the joint application. The pre-training strategies are as such complementary. The effect of applying a pre-training scheme becomes necessary, when working with frozen models in P2. As such, the gap between full fine-tuning and frozen model adaptation is significantly lessened.

The adaptation ablation experiments comparing prompting to common adaptation schemes establish a trend where more trainable parameters correlate with a higher DSC value. This trend however, achieves diminishing returns for increasing amounts of parameters. Under this aspect, the prompting scheme proofs especially flexible, since the prompts are not part of the backbone network architecture. The ablation of the prompt insertion indicates, that adjustments throughout the decoder are most important to incur meaningful changes in the output embedding, so that the aggregation can identify pixels of a certain class properly. This is in line with the recent Segmenter architecture (Strudel et al., 2021), which relied on a joint processing of class tokens and encoded image content in its decoder.

The scope of this work remains limited. We focus on the impact of the introduced training schemes with respect to their efficacy for different downstream adaptation strategies. We do not evaluate the pre-training strategies against state-of-the-art approaches of each respective sub-field and do not claim superiority. We do not compare our method against the vast number of alternatives schemes in the label scarce case known from the semi-supervised literature Yang et al. (2021). In addition, the label scarcity could be extended to multiple seeds for better statistical power. The proposed approach is a mere first step with a multitude of potential extensions. The architecture could operate on full resolution inputs at the cost of higher memory consumption and training time. As shown for language models by Brown et al. (2020), we expect that greatly increasing the neural network model parameters can have strong beneficial effects on the label scarce performance and may also positively influence the requirements on the amount of prompt token parameters and additive bias scores. In addition, neighborhood attention (Hassani et al., 2022; Hassani and Shi, 2022) has been proposed recently to avoid the use of potentially restrictive shifted windows entirely. More sophisticated hierarchical clustering could be employed for the online prototype generation, e.g. on multiple resolution levels. Likewise, a more sophisticated prompt token generation could be introduced, to further reduce the amount of stored parameters, as done by He et al. (2022). We note that when relying on the binary case, an implicit extension to incremental class learning can be achieved

circumventing catastrophic forgetting entirely. Further investigations are to be performed for the combination of prompts from different known classes or the introduction of a new class in the multi-class case. Hereby, the model needs to be able to adjust prompt tokens to the new circumstances while not forgetting to predict known classes.

6. Conclusion

In this work, we propose advancements in the efficacy and applicability of using pre-trained frozen models for downstream segmentation tasks based on a prompt tuning scheme. We establish a scheme and architecture which provides natural insertion points for task-dependent tuning, while leaving the original pre-trained model intact. Introducing several pre-requisites, we make a step towards closing the performance between non-frozen and frozen models, allowing for readily re-usable models in the field. In light of the results, further investigations, e.g. in the composability of prompt tokens of different classes seem promising and interesting avenues for future endeavours.

References

Assran, M., Caron, M., Misra, I., Bojanowski, P., Joulin, A., Ballas, N., Rabbat, M., 2021. Semi-supervised learning of visual features by non-parametrically predicting view assignments with support samples, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8443–8452.

Azizi, S., Mustafa, B., Ryan, F., Beaver, Z., von Freyberg, J., Deaton, J., Loh, A., Karthikesalingam, A., Kornblith, S., Chen, T., Natarajan, V., Norouzi, M., 2021. Big self-supervised models advance medical image classification, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3478–3488.

Baevski, A., Hsu, W.N., Xu, Q., Babu, A., Gu, J., Auli, M., 2022. Data2vec: A general framework for self-supervised learning in speech, vision and language. arXiv preprint arXiv:2202.03555 .

Bahng, H., Jahanian, A., Sankaranarayanan, S., Isola, P., 2022. Visual prompting: Modifying pixel space to adapt pre-trained models. arXiv preprint arXiv:2203.17274 .

Bao, H., Dong, L., Wei, F., 2021. Beit: Bert pre-training of image transformers. arXiv preprint arXiv:2106.08254 .

Bommasani, R., Hudson, D.A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M.S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N.S., Chen, A.S., Creel, K.A., Davis, J., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L.E., Goel, K., Goodman, N.D., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D.E., Hong, J., Hsu, K., Huang, J., Icard, T.F., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P.W., Krass, M.S., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X.L., Li, X., Ma, T., Malik, A., Manning, C.D., Mirchandani, S.P., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J.C., Nilforoshan, H., Nyarko, J.F., Ogut, G., Orr, L.J., Papadimitriou, I., Park, J.S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y.H., Ruiz, C., Ryan, J., R'e, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K.P., Tamkin, A., Taori, R., Thomas, A.W., Tramèr, F., Wang, R.E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S.M., Yasunaga, M., You, J., Zaharia, M.A., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., Liang, P., 2021. On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258 .

Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T.J., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess,

B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D., 2020. Language models are few-shot learners. Advances in neural information processing systems 33, 1877–1901.

Cao, H., Wang, Y., Chen, J., Jiang, D., Zhang, X., Tian, Q., Wang, M., 2021. Swin-unet: Unet-like pure transformer for medical image segmentation. arXiv preprint arXiv:2105.05537 .

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., Joulin, A., 2020. Unsupervised learning of visual features by contrasting cluster assignments. Advances in Neural Information Processing Systems 33, 9912–9924.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A., 2021. Emerging properties in self-supervised vision transformers, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9650–9660.

Chaitanya, K., Erdil, E., Karani, N., Konukoglu, E., 2020. Contrastive learning of global and local features for medical image segmentation with limited annotations. Advances in Neural Information Processing Systems 33, 12546–12558.

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y., 2021a. Transunet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306 .

Chen, L., Bentley, P., Mori, K., Misawa, K., Fujiwara, M., Rueckert, D., 2019a. Self-supervised learning for medical image analysis using image context restoration. Medical image analysis 58, 101539.

Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European conference on computer vision (ECCV), pp. 801–818.

Chen, S., Ma, K., Zheng, Y., 2019b. Med3d: Transfer learning for 3d medical image analysis. arXiv preprint arXiv:1904.00625 .

Chen, T., Kornblith, S., Norouzi, M., Hinton, G., 2020. A simple framework for contrastive learning of visual representations, in: International conference on machine learning, PMLR. pp. 1597–1607.

Chen, X., Xie, S., He, K., 2021b. An empirical study of training self-supervised vision transformers, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9640–9649.

Cheng, B., Schwing, A., Kirillov, A., 2021. Per-pixel classification is not all you need for semantic segmentation. Advances in Neural Information Processing Systems 34, 17864–17875.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A.B., Barnes, P., Tay, Y., Shazeer, N.M., Prabhakaran, V., Reif, E., Du, N., Hutchinson, B.C., Pope, R., Bradbury, J., Austin, J., Isard, M., Gur-Ari, G., Yin, P., Duke, T., Levskaya, A., Ghemawat, S., Dev, S., Michalewski, H., García, X., Misra, V., Robinson, K., Fedus, L., Zhou, D., Ippolito, D., Luan, D., Lim, H., Zoph, B., Spiridonov, A., Sepassi, R., Dohan, D., Agrawal, S., Omernick, M., Dai, A.M., Pillai, T.S., Pellat, M., Lewkowycz, A., Moreira, E., Child, R., Polozov, O., Lee, K., Zhou, Z., Wang, X., Saeta, B., Díaz, M., Firat, O., Catasta, M., Wei, J., Meier-Hellstern, K.S., Eck, D., Dean, J., Petrov, S., Fiedel, N., 2022. Palm: Scaling language modeling with pathways. arXiv preprint arXiv:2204.02311 .

Cordonnier, J.B., Loukas, A., Jaggi, M., 2019. On the relationship between self-attention and convolutional layers. arXiv preprint arXiv:1911.03584 .

Devlin, J., Chang, M.W., Lee, K., Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 .

Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., Zisserman, A., 2021. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9588–9597.

Ghesu, F.C., Georgescu, B., Mansoor, A., Yoo, Y., Neumann, D., Patel, P.B., Vishwanath, R.S., Balter, J.M., Cao, Y., Grbic, S., Comaniciu, D., 2022. Self-supervised learning from 100 million medical images. arXiv preprint arXiv:2201.01283 .

Gibson, E., Giganti, F., Hu, Y., Bonmati, E., Bandula, S., Gurusamy, K., Davidson, B., Pereira, S.P., Clarkson, M.J., Barratt, D.C., 2018. Automatic multi-organ segmentation on abdominal ct with dense v-networks. IEEE transac-

tions on medical imaging 37, 1822–1834.

Gidaris, S., Bursuc, A., Puy, G., Komodakis, N., Cord, M., Perez, P., 2021. Obow: Online bag-of-visual-words generation for self-supervised learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6830–6840.

Gidaris, S., Singh, P., Komodakis, N., 2018. Unsupervised representation learning by predicting image rotations. arXiv preprint arXiv:1803.07728 .

Grill, J.B., Strub, F., Altch'e, F., Tallec, C., Richemond, P.H., Buchatskaya, E., Doersch, C., Pires, B.Á., Guo, Z.D., Azar, M.G., Piot, B., Kavukcuoglu, K., Munos, R., Valko, M., 2020. Bootstrap your own latent-a new approach to self-supervised learning. Advances in neural information processing systems 33, 21271–21284.

Guo, D., Rush, A.M., Kim, Y., 2020. Parameter-efficient transfer learning with diff pruning. arXiv preprint arXiv:2012.07463 .

Haghighi, F., Taher, M.R.H., Zhou, Z., Gotway, M.B., Liang, J., 2021. Transferable visual words: Exploiting the semantics of anatomical patterns for self-supervised learning. IEEE transactions on medical imaging 40, 2857–2868.

Hassani, A., Shi, H., 2022. Dilated neighborhood attention transformer. arXiv preprint arXiv:2209.15001 .

Hassani, A., Walton, S., Li, J., Li, S., Shi, H., 2022. Neighborhood attention transformer. arXiv preprint arXiv:2204.07143 .

Hatamizadeh, A., Tang, Y., Nath, V., Yang, D., Myronenko, A., Landman, B., Roth, H.R., Xu, D., 2022. Unetr: Transformers for 3d medical image segmentation, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 574–584.

He, J., Zhou, C., Ma, X., Berg-Kirkpatrick, T., Neubig, G., 2021. Towards a unified view of parameter-efficient transfer learning. arXiv preprint arXiv:2110.04366 .

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

He, Y., Zheng, H., Tay, Y., Gupta, J., Du, Y., Aribandi, V., Zhao, Z., Li, Y., Chen, Z., Metzler, D., Cheng, H.T., Chi, E.H., 2022. Hyperprompt: Prompt-based task-conditioning of transformers, in: International Conference on Machine Learning, PMLR. pp. 8678–8690.

Hénaff, O.J., Koppula, S., Shelhamer, E., Zoran, D., Jaegle, A., Zisserman, A., Carreira, J., Arandjelović, R., 2022. Object discovery and representation networks. arXiv preprint arXiv:2203.08777 .

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S., 2019. Parameter-efficient transfer learning for nlp, in: International Conference on Machine Learning, PMLR. pp. 2790–2799.

Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H., 2021. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. Nature methods 18, 203–211.

Jia, M., Tang, L., Chen, B.C., Cardie, C., Belongie, S., Hariharan, B., Lim, S.N., 2022. Visual prompt tuning. arXiv preprint arXiv:2203.12119 .

Khan, A., Sohail, A., Zahoora, U., Qureshi, A.S., 2020. A survey of the recent architectures of deep convolutional neural networks. Artificial intelligence review 53, 5455–5516.

Landman, B., Xu, Z., Igelsias, J., Styner, M., Langerak, T., Klein, A., 2015. Miccai multi-atlas labeling beyond the cranial vault - workshop and challenge, in: Proc. MICCAI Multi-Atlas Labeling Beyond Cranial Vault—Workshop Challenge, p. 12. doi:10.7303/syn3193805.

Lester, B., Al-Rfou, R., Constant, N., 2021. The power of scale for parameter-efficient prompt tuning. arXiv preprint arXiv:2104.08691 .

Li, J., Zhou, P., Xiong, C., Hoi, S.C., 2020. Prototypical contrastive learning of unsupervised representations. arXiv preprint arXiv:2005.04966 .

Li, X.L., Liang, P., 2021. Prefix-tuning: Optimizing continuous prompts for generation. arXiv preprint arXiv:2101.00190 .

Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., Neubig, G., 2021a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. arXiv preprint arXiv:2107.13586 .

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021b. Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022.

Loshchilov, I., Hutter, F., 2017. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 .

Mahajan, D., Girshick, R., Ramanathan, V., He, K., Paluri, M., Li, Y., Bharambe, A., Van Der Maaten, L., 2018. Exploring the limits of weakly supervised pretraining, in: Proceedings of the European conference on computer vision (ECCV), pp. 181–196.

MONAI Consortium, 2020. Monai: Medical open network for ai. URL: https://monai.io, doi:10.5281/zenodo.4323058.

Noroozi, M., Favaro, P., 2016. Unsupervised learning of visual representations by solving jigsaw puzzles, in: European conference on computer vision, Springer. pp. 69–84.

Oord, A.v.d., Li, Y., Vinyals, O., 2018. Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 .

Ouyang, C., Biffi, C., Chen, C., Kart, T., Qiu, H., Rueckert, D., 2020. Self-supervision with superpixels: Training few-shot medical image segmentation without annotation, in: European Conference on Computer Vision, Springer. pp. 762–780.

Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A., 2016. Context encoders: Feature learning by inpainting, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2536–2544.

Peng, J., Pedersoli, M., Desrosiers, C., 2021. Boosting semi-supervised image segmentation with global and local mutual information regularization. Machine Learning for Biomedical Imaging 1, 1–10.

Raghu, M., Zhang, C., Kleinberg, J., Bengio, S., 2019. Transfusion: Understanding transfer learning for medical imaging. Advances in neural information processing systems 32.

Ramachandran, P., Parmar, N., Vaswani, A., Bello, I., Levskaya, A., Shlens, J., 2019. Stand-alone self-attention in vision models. Advances in Neural Information Processing Systems 32.

Rister, B., Shivakumar, K., Nobashi, T., Rubin, D.L., 2019. Ct-org: Ct volumes with multiple organ segmentations. The Cancer Imaging Archive doi:10.7937/tcia.2019.tt7f4v7o.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer. pp. 234–241.

Roth, H.R., Farag, A., Turkbey, E., Lu, L., Liu, J., Summers, R.M., 2016. Data from pancreas-ct. the cancer imaging archive. IEEE Transactions on Image Processing doi:10.7937/K9/TCIA.2016.tNB1kqBU.

Smith, L.N., Topin, N., 2019. Super-convergence: Very fast training of neural networks using large learning rates, in: Artificial intelligence and machine learning for multi-domain operations applications, SPIE. pp. 369–386.

Snell, J., Swersky, K., Zemel, R., 2017. Prototypical networks for few-shot learning. Advances in neural information processing systems 30.

Strudel, R., Garcia, R., Laptev, I., Schmid, C., 2021. Segmenter: Transformer for semantic segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 7262–7272.

Tajbakhsh, N., Jeyaseelan, L., Li, Q., Chiang, J.N., Wu, Z., Ding, X., 2020. Embracing imperfect datasets: A review of deep learning solutions for medical image segmentation. Medical Image Analysis 63, 101693.

Taleb, A., Loetzsch, W., Danz, N., Severin, J., Gaertner, T., Bergner, B., Lippert, C., 2020. 3d self-supervised methods for medical imaging. Advances in Neural Information Processing Systems 33, 18158–18172.

Tang, Y., Yang, D., Li, W., Roth, H.R., Landman, B., Xu, D., Nath, V., Hatamizadeh, A., 2022. Self-supervised pre-training of swin transformers for 3d medical image analysis, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20730–20740.

Ulyanov, D., Vedaldi, A., Lempitsky, V., 2016. Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022 .

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems 30.

Wang, K., Liew, J.H., Zou, Y., Zhou, D., Feng, J., 2019. Panet: Few-shot image semantic segmentation with prototype alignment, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9197–9206.

Xie, Y., Zhang, J., Shen, C., Xia, Y., 2021. Cotr: Efficiently bridging cnn and transformer for 3d medical image segmentation, in: International conference on medical image computing and computer-assisted intervention, Springer. pp. 171–180.

Yan, K., Cai, J., Jin, D., Miao, S., Guo, D., Harrison, A.P., Tang, Y., Xiao, J., Lu, J., Lu, L., 2022. Sam: Self-supervised learning of pixel-wise anatomical

embeddings in radiological images. IEEE Transactions on Medical Imaging .

Yang, X., Song, Z., King, I., Xu, Z., 2021. A survey on deep semi-supervised learning. arXiv preprint arXiv:2103.00550 .

Yue, X., Zheng, Z., Zhang, S., Gao, Y., Darrell, T., Keutzer, K., Vincentelli, A.S., 2021. Prototypical cross-domain self-supervised learning for few-shot unsupervised domain adaptation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13834–13844.

Zaken, E.B., Ravfogel, S., Goldberg, Y., 2021. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. arXiv preprint arXiv:2106.10199 .

Zhai, X., Kolesnikov, A., Houlsby, N., Beyer, L., 2022. Scaling vision transformers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12104–12113.

Zhang, J.O., Sax, A., Zamir, A., Guibas, L., Malik, J., 2020. Side-tuning: a baseline for network adaptation via additive side networks, in: European Conference on Computer Vision, Springer. pp. 698–714.

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H.S., Zhang, L., 2021. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 6881–6890.

Zhou, Z., Sodha, V., Pang, J., Gotway, M.B., Liang, J., 2021. Models genesis. Medical image analysis 67, 101840.

Zhuang, X., Li, Y., Hu, Y., Ma, K., Yang, Y., Zheng, Y., 2019. Self-supervised feature learning for 3d medical images by playing a rubik's cube, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer. pp. 420–428.