HHS Public Access

Author manuscript

Med Phys. Author manuscript; available in PMC 2026 March 01.

Published in final edited form as:

Med Phys. 2025 March ; 52(3): 1573–1588. doi:10.1002/mp.17541.

Self-supervised learning improves robustness of deep learning lung tumor segmentation to CT imaging differences

Jue Jiang,

Aneesh Rangnekar,

Harini Veeraraghavan*

Department of Medical Physics, Memorial Sloan Kettering Cancer Center, NY, New York, USA.

Abstract

Background: Self-supervised learning (SSL) is an approach to extract useful feature representations from unlabeled data, and enable fine-tuning on downstream tasks with limited labeled examples. Self-pretraining is a SSL approach that uses curated downstream task dataset for both pretraining and fine-tuning. Availability of large, diverse, and uncurated public medical image sets presents the opportunity to potentially create foundation models by applying SSL in the ”wild” that are robust to imaging variations. However, the benefit of wild- vs self-pretraining has not been studied for medical image analysis.

Purpose: Compare robustness of wild versus self-pretrained models created using convolutional neural network (CNN) and transformer (vision transformer [ViT] and hierarchical shifted window [Swin]) models for non-small cell lung cancer (NSCLC) segmentation from 3D computed tomography (CT) scans.

Methods: CNN, ViT, and Swin models were wild-pretrained using unlabeled 10,412 3D CTs sourced from the cancer imaging archive and internal datasets. Self-pretraining was applied to same networks using a curated public downstream task dataset (n = 377) of patients with NSCLC. Pretext tasks introduced in self-distilled masked image transformer were used for both pretraining approaches. All models were fine-tuned to segment NSCLC (n = 377 training dataset) and tested on two separate datasets containing early (public N = 156) and advanced stage (internal N = 196) NSCLC. Models were evaluated in terms of: (a) accuracy, (b) robustness to image differences from contrast, slice thickness, and reconstruction kernels, and (c) impact of pretext tasks for pretraining. Feature reuse was evaluated using centered kernel alignment.

Results: Wild-pretrained Swin models resulted in higher feature reuse at earlier level layers and increased feature differentiation close to output. Wild-pretrained Swin out-performed self-pretrained models for analyzed imaging acquisitions. Neither ViT nor CNN showed a clear benefit of wild-pretraining compared to self-pretraining. Masked image prediction pretext task that forces networks to learn the local structure resulted in higher accuracy compared to contrastive task that models global image information.

* jiangj1@mskcc.org .

Conclusion: Wild-pretrained Swin networks were more robust to analyzed CT imaging differences for lung tumor segmentation than self-pretrained methods. ViT and CNN models did not show a clear benefit for wild-pretraining over self-pretraining.

I. Introduction

Self-supervised learning (SSL) has emerged as a promising approach to train large deep learning (DL) foundation models in both computer vision and medical image analysis. SSL methods use unlabeled datasets for pretraining. SSL is thus more practical to implement compared to methods that require labeled datasets for pretraining[1,2], especially when labeled data is scarce. SSL pretraining with large numbers of diverse and unlabeled image sets, called wild-pretraining[3], forms the basis of training high-capacity foundation models[4,5]. Pretrained models can be fine-tuned with relatively few labeled downstream task examples[6–8]. SSL pretrained models have demonstrated effectiveness for medical image segmentation[8–12].

SSL pretraining can be performed with: (a) either large-scale, diverse, and unlabeled upstream datasets that are often unrelated to task-specific downstream datasets, called ‘wild’ pretraining, (b) or use curated task-specific downstream data without exposing labels, and called ‘self’ pretraining. Self-pretraining is data and time-efficient due to limited size of downstream task data, uses data that is highly relevant to the task, and is thus commonly used in medical applications.

The key intuition for using wild-pretraining is that models trained with large numbers of examples with substantial imaging variations would extract universally applicable and reusable feature representations, and provide robustly accurate performance for clinical use. Positive impact of larger training dataset on downstream accuracy has typically been studied in the context of supervised pretraining followed by transfer learning often applied to image classification[2,13]. SSL converts learning from unlabeled datasets to a supervised problem by creating pretext tasks that uses images as ground truth. Hence, efficacy of SSL to extract useful features for robust generalization is likely linked to the pretext tasks. Examples of pretext tasks include masked image prediction[8,14], masked image reconstruction[15], jigsaw puzzles[6,16], as well as combination of contrastive and image reconstruction losses[17,18].

Importantly, whether and why wild-pretrained models using SSL strategy would yield robust performance has not been studied to our best knowledge. Understanding the relative merits of the wild-pretrained versus self-pretrained neural network architectures such as convolutional neural network (CNN), vision transformer (ViT), and hierarchical shifted window transformer (Swin), and their capability to provide robustly accurate segmentations despite imaging variations is an essential first step before the models can be further refined for clinical use or assessed for clinical impact.

Relatedly, the network architecture can also impact the extracted features and their effectiveness to capture the underlying spatial and contextual information in images. ViT[19], which uses a sequence of image tokens obtained by tokenization of non-overlapping input image patches, has the lowest inductive bias as it eliminates all relative spatial and local

relationships. Swin[20] adds back some of the inductive bias by employing cross-window attention and hierarchical scaling. CNN has the highest inductive bias as it maintains the spatial locality of the images along the different feature layers.

Hence, the objective and key novelty of this work was to systematically study the relative benefits of SSL pretraining strategy (wild- versus self-pretraining) together with the underlying networks (CNN versus ViT versus Swin) and pretext tasks in terms of robust performance under common imaging and data variations. Wild-pretraining was implemented on a relatively large and uncurated 3D medical datasets (> 10,000 CTs) for all three architectures. All models following fine-tuning will be made available for reproducible research upon manuscript acceptance.

Finally, we studied why accuracy differences occur following fine-tuning by analyzing feature reuse performed using centered kernel alignment (CKA)[21] from pretraining in models subjected to self- and wild-pretraining and multiple pretext tasks. CKA provides insights into what extent the features in the different layers were modified as result of fine-tuning. Higher feature reuse in the earlier layers, which extracts low-level features like edges has shown to improve downstream task accuracy on computer vision[3] and medical image classification tasks[2]. We also analyzed the pattern of feature reuse for tumor segmentation. To our best knowledge, this is the first study to systematically evaluate the robustness of SSL pretrained models to imaging and data variations and analyze under what conditions (pretraining approach, network, pretext tasks) robust generalization occurs. Systematic analysis as we performed could inform strategies to select and refine models for routine clinical use. Our work impacts clinical development and evaluation of models for clinical deployment.

Our contributions are: (a) comparative analysis of different SSL-based wild-pretraining and self-pretraining applied to three common architectures, a ViT, Swin, and a U-Net-based CNN for lung tumor segmentation, (b) systematic analysis of robustness of the various methods to CT acquisition differences (distribution shift) as well as disease differences (early versus late stage cancers resulting in a concept shift); we evaluated the performance of the various models on an open-source CT lung tumor phantom dataset to assess their capability to segment images that are different from real patient scans. Finally, (c) we studied whether and why wild-pretraining increased robustness by analyzing feature reuse.

II. Methods

II.A. Background terminology

- **Self-supervised learning (SSL):** A machine learning approach to use unlabeled datasets and extract useful features for downstream tasks by creating pretext tasks that use the unlabeled data as the ”ground truth”.
- **Self-pretraining:** SSL pretraining performed using the same curated task-specific labeled data without exposing labels, which is also used for finetuning.
- **Wild-pretraining:** SSL pretraining performed with large, unlabeled, uncurated and relatively uncurated datasets that do not share the same characteristics

(disease, disease site, imaging characteristics) as the curated task-specific downstream dataset.

- **Two-stage pretraining:** SSL pretraining approach where wild-pretraining is followed by self-pretraining as used in this study for ablation analysis.
- **Upstream versus downstream dataset:** Upstream dataset is used to pretrain network with pretext tasks. A downstream dataset is a labeled dataset used for fine-tuning the network for performing specific tasks.
- **Pretext tasks:** Pretext tasks enable a network to 'learn' the underlying feature representations by providing supervisory signals that use images themselves as 'ground truth', thus removing the need for external user-annotated labels.
- **Fine tuning:** Fine tuning is performed with downstream labeled dataset using supervised learning (combination of Dice and cross entropy losses).
- **Scratch training:** Supervised learning performed with randomly initialized network weights using the downstream task data. No pretraining is required for scratch training.

II.B. Datasets

Details of upstream datasets used in wild-pretraining, downstream task datasets used in self-pretraining as well fine-tuning are shown with respective imaging acquisitions and disease details in Table 1. Retrospective analysis of institutional datasets was approved with the local institutional review board, with a waiver for written informed consent, and was compliant with the Health and Insurance Portability and Accountability Act. Due to the wide variation in the convolution kernels used in the testing datasets, we categorized them as Recon 1 kernels: GE "standard" and "bone", Siemens with < B40; Recon 2: GE "Bone Plus", Siemens with ≥ B40 and < B50; Recon 3: GE "Lung", Siemens ≥ B50.

II.B.1. Upstream datasets used for wild-pretraining—

A total of uncurated 10,412 3D CT scans encompassing diseases from head and neck to the pelvis sourced from datasets provided publicly for variety of tasks including lesion detection[22], classification[24], and multi-organ and abdominal tumor segmentations were used without additional curation for pretraining (Table 1(A)). Out of 10,412 volumetric scans, N= 2,656 were sourced from public archive and remaining N = 7,756 were used from anonymized and uncurated institutional datasets. Anonymized institutional datasets were used without any curation and consisted of CT scans from patients treated for lung, esophageal (Internal 1) and head and neck (Internal 2) cancers treated with radiotherapy (RT).

II.B.2. Downstream or task datsets—

Entirely different cohorts of curated patient datasets were used for training (fine-tuning) and evaluation or testing as shown in Table 1 (B) and (C). The TCIA NSCLC dataset is used for multiple purposes across different settings including scratch-training, finetuning in wild-pretraining and self-pretraining.

Fine-tuning and self-pretraining dataset consisted of public domain dataset of 377 3D CTs of patients diagnosed with stage I-IV locally advanced non-small cell lung cancer

(LA-NSCLC) who underwent RT in a single institution[27]. Tumor contours are provided with the dataset. Patients had contrast or non-contrast CTs and were typically reconstructed with convolution kernels (≤ B30) and 3 mm slice thickness acquired using either Siemens or CMS manufacturers. Tumor sizes ranged with a median of 33.68 cc, and interquartile range (IQR) of 8.29 cc to 90.31 cc. Of the 377, 350 were used for model training/fine-tuning and the remaining 27 were set aside for validation and hyperparameter tuning.

Testing: Three held-out testing datasets were analyzed: (1) a public dataset (LRad) containing early stage (stage I-II) lung cancer, all of whom underwent surgery[28], (2) a dataset with primary and metastatic stage III-IV lung lesions (LC). Tumors in LRad dataset were smaller in size than those used in training and ranged in size with a median of 7.91cc, IQR of 3.60 cc to 28.23 cc. The tumors in LC dataset were of similar size (median of 19.54 cc, IQR of 6.64 cc to 66.97 cc) and stage as those used in training. A subset of 20 cases in LC had patients reconstructed with 2.5mm, 5mm slice thickness as well as with GE's lung kernel and GE's standard kernel and were evaluated to assess pairwise accuracy differences. (3) Finally, a single lung CT phantom dataset with 8 different lesions of varying shapes and sizes provided through the TCIA repository[29] was also evaluated. The lung phantom image scan consisted of 8 lesions of different shape and sizes, scanned at Columbia University Medical Center with GE scanner at 120 kVp. The images were reconstructed using a lung kernel with 1.25 mm slice thickness. The lung phantom scans were evaluated to assess robustness to images that did not correspond to a real patient.

II.C. Network architectures

Three representative published 3D network architectures consisting of a CNN-based PRCLv2[18], transformer based ViT[19] and Swin[20] networks often used in medical image segmentation[9,15,30] were analyzed. The schematics of the three networks are shown in Figure 1.

The CNN-based PRCLv2 model with 17,111,499 parameters (Figure 1 (A)) used a published non-skip U-Net[18]. The encoder and decoder layers were composed of convolutional blocks with pooling (for encoder) or unpooling (for decoder) layers, and activation functions.

The 3D ViT (Figure 1 (B)) and 3D Swin networks (Figure 1 (C)) used transformer blocks in the encoder and convolutional blocks with skip connections in the decoder. The ViT encoder consisted of 12 transformer blocks, 8 multi-head self-attention at each stage, and a feature embedding size of 768. The ViT network operated on a patch size of $8 \times 8 \times 8$. The 3D Swin network used the Swin encoder backbone, with a depth of [2,2,8,2] and corresponding [4,4,8,16] multi-head attention for the four stages of hierarchical transformer blocks. It operated on a patch size of $2 \times 2 \times 2$ pixels, a window size of $4 \times 4 \times 4$ pixels, and the feature embedding size of 384. An additional convolutional block was added to the decoder to process the extracted encoder feature in both ViT and Swin architectures. The total number of parameters for ViT and Swin-based segmentation models were 46,405,874 and 64,698,114, respectively. All networks used an image size of $128 \times 128 \times 128$ pixels.

II.D. Pretext tasks for unsupervised pretraining

Wild and self-pretraining were performed on the models using same pretext tasks, with large-scale upstream data for former and task-specific downstream data for the latter, respectively.

Pretext tasks conformed to published works developed for specific architectures. Transformer models including the Swin and ViT used tasks designed to optimize extraction of feature tokens. CNN-based PRCLv2[18] used global image-based contrastive and multi-scale image reconstruction tasks developed for the same model. PRCLv2 processes images as a whole and hence could not be trained with token distillation and token masking based tasks, which require a sequence of non-overlapping image patch tokens.

Both transformer methods used a combination of pretext tasks consisting of input image token masking based masked image token prediction (MIP), masked patch token distillation (MPD) and image token distillation (ITD). These tasks were developed for the self-distillation masked image transformer (SMIT) model for medical image organ segmentation[8]. Self-distillation learning was performed by creating an exponentially moving average teacher model with identical architecture as the student model; the latter model is retained for fine-tuning and testing. The teacher network is provided with uncorrupted sequence of tokens created from image patches while the student network is provided with a corrupted sequence of tokens randomly ‘masked’. Self-distillation tasks force the student network to extract the same set of feature tokens as the teacher. The goal of self-distillation is to force the student network to learn a stable and robust feature embedding from a noisy input. This approach has shown to yield superior performance on natural image classification[31,32] and medical image segmentation[8] tasks.

Masked image prediction or MIP involves a dense pixel regression of the pixel intensities within the masked image patches and is optimized by minimizing the L1 norm between the predicted image patches and the corresponding unmasked patches.

Image token distillation or ITD is a self-distillation task, whereby the student network learns to extract the same global feature token embedding as the teacher by minimizing the cross entropy between the two feature embedding. The student is provided with a randomly masked token sequence and the teacher with an uncorrupted token sequence, respectively.

Masked patch token distilation or MPD forces the student to extract the features underlying the masked input patch tokens. This is accomplished by matching the tokens extracted by the student for the masked patches with the corresponding tokens extracted by the teacher that is provided with unmasked sequence of tokens. Cross entropy loss with temperature scaling to adjust the sharpness of the token distribution was used.

Contrastive[17] loss was used as an ablation pretext task to assess whether learning a global image representation was beneficial for segmentation. This task minimized the L2-norm distance between features extracted from cropped image views of the same images (positive samples) while maximizing the distance between cropped image views arising from different images (or negative samples).

III. Experiments and ablation studies

Robustness to CT acquisition differences was measured by comparing the segmentation accuracies for Recon 1, Recon 2 and Recon 3 kernels, contrast and non-contrast CTs, and different slice reconstruction thicknesses. In all experiments, the statistical comparisons were done between self- and wild-pretrained model instances of the same architectures using paired, two-sided, Wilcoxon signed rank tests at 95% significance level. In addition, robustness of each model with respect to reconstruction kernels was performed using paired comparison of results produced on Recon 1 on the test set, which corresponds to the convolutional kernel used in the training set, with respect to Recon 2 and Recon 3, respectively. Phantom datasets that did not correspond to real patient scans were analyzed to evaluate robustness of the various models.

Impact of using a homogeneous dataset for fine-tuning the networks was studied by using a subset of 279 non-contrast CT scans acquired using the Siemens scanner and reconstructed at 3 mm slice thickness and convolution kernel range of B19 to B30 (or Recon 1).

Ablation analysis evaluated the impact of pretext tasks (MIP, ITD, MPD, SMIT, as well as contrastive pretraining) on downstream task accuracy, two-stage pretraining, and feature reuse for both self- and wild-pretraining configurations for the Swin network.

Two-stage pretraining employed sequential pretraining using wild- followed by self-pretraining to refine the encoder weights using the upstream or wild-pretraining dataset (Table 1 (A)) followed by downstream or task-specific dataset without labels (Table 1 (B)) using the SMIT pretext task. Fine-tuning was performed using supervised losses by providing tumor segmentation as labels in the fine-tuning dataset (Table 1 (B)).

III.A. Implementation details

Pretraining (self and wild) was performed by generating two augmented views, one masked view provided to student using a default masking ratio of 0.75 and the second unmasked view provided to the teacher, respectively. All images were clipped in the HU value of [−500,500], normalized to [0,1] and resampled to a uniform voxel spacing of 2mm × 2mm × 2mm and then randomly cropped to 128 × 128 × 128 voxels to generate the 3D views. The networks were optimized using ADAMw[33] with a cosine learning rate scheduler[34] trained for 500 epochs with an initial learning rate of $8e^{-4}$ and warmup for 50 epochs. In order to account for fewer unlabeled examples available for self-pretraining, the self-pretraining was performed for 2,000 epochs with a warmup for 200 epochs. Online data augmentations were performed to increase data variations. A path drop rate of 0.1 was applied to the student model, and all experiments were conducted on 4 NVIDIA A100 GPUs (4 × 80GB memory) using an effective batch size of 8 for ViT and 32 for Swin transformers. Model hyperparameters for pretraining were selected from published studies[8,18].

Fine-tuning used only the student network and was performed on 4 NVIDIA A100 GPUs. All analyzed network configurations were trained with a learning rate of $2e^{-4}$ for 1,000 epochs. The Swin models were fine-tuned with a batch size of 24 while the ViT models used a batch size of 4 due to memory limitations. Fine-tuning and testing were performed

after resampling all the images to a uniform voxel spacing of 1.5 mm × 1.5 mm × 2.0 mm voxels. Early stopping selected the model with the best validation accuracy to prevent over-fitting. All model instances for ViT, Swin and CNN were fine-tuned on an identical downstream dataset of 350 training and 27 validation to generate volumetric segmentation of lung tumors. The models were optimized using a combination of cross-entropy and Dice overlap losses. The validation set was used for select the best model for testing.

III.B. Metrics

Tumor detection rate (DR)[35] was computed to measure the accuracy of detected tumors. Detected tumors (D_{tumor}) were counted as those with an overlap between the algorithm and manual segmentation of at least $\tau \geq 0.5$[35], computed as $\tau = \frac{TP}{N}$. TP is the true positives and N is the total number of pixels within manual delineation. DR was then computed as $DR = \frac{D_{tumor}}{N_{tumor}}$, where N_{tumor} is the total number of tumors.

Tumor segmentations were compared against manual delineations using Dice similarity coefficient (DSC) and Hausdroff distance at 95^{th} percentile (HD95). Coefficient of determination was computed to measure the dependence of segmentation accuracy with respect to tumor volume and presented using R^2 statistic and the confidence intervals.

Centered kernel alignment (CKA)[21] was used to measure feature reuse between the wild-pretrained and self-pretrained models before and after fine-tuning. CKA computes a normalized similarity of two feature representations $\boldsymbol{M}$ and $\boldsymbol{N}$ in terms of the Hilbert-Schmidt Independence Criterion (HSIC):

$$\mathrm{CKA}(\boldsymbol{X}, \boldsymbol{Y}) = \frac{\mathrm{HSIC}_0(\boldsymbol{X}, \boldsymbol{Y})}{\sqrt{\mathrm{HSIC}_0(\boldsymbol{X}, \boldsymbol{X})\mathrm{HSIC}_0(\boldsymbol{Y}, \boldsymbol{Y})}} \tag{1}$$

where $\boldsymbol{X}=\boldsymbol{M}\boldsymbol{M}^{\boldsymbol{T}}$ and $\boldsymbol{Y}=\boldsymbol{N}\boldsymbol{N}^{\boldsymbol{T}}$ are the Gram matrices of feature $\boldsymbol{M}$ and $\boldsymbol{N}$ respectively. CKA computation requires the feature activation of entire dataset to be stored in the memory, which is difficult to implement for transformers that have a large number of parameters. Minibatch CKA was computed by averaging HSIC scores over k minibatches[36] as:

$$\mathrm{CKA}_{\mathrm{minibatch}} = \frac{\frac{1}{\mathrm{k}}\sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{HSIC}_1\left(\mathbf{M}_\mathbf{i}\mathbf{M}_\mathbf{i}^{\mathbf{T}}, \mathbf{N}_\mathbf{i}\mathbf{N}_\mathbf{i}^{\mathbf{T}}\right)}{\sqrt{\frac{1}{\mathrm{k}}\sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{HSIC}_1\left(\mathbf{M}_\mathbf{i}\mathbf{M}_\mathbf{i}^{\mathbf{T}}, \mathbf{M}_\mathbf{i}\mathbf{M}_\mathbf{i}^{\mathbf{T}}\right)}\sqrt{\frac{1}{\mathrm{k}}\sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{HSIC}_1\left(\mathbf{N}_\mathbf{i}\mathbf{N}_\mathbf{i}^{\mathbf{T}}, \mathbf{N}_\mathbf{i}\mathbf{N}_\mathbf{i}^{\mathbf{T}}\right)}} \tag{2}$$

where M_i and N_i are matrices containing activation of the i^{th} minibatch. An unbiased estimator of HSIC[37] was computed to mitigate dependency of CKA values on the batch size:

$$\mathrm{HSIC}_1(\boldsymbol{X}, \boldsymbol{Y}) = \frac{1}{n(n-3)}\left(\mathrm{tr}(\tilde{X}\tilde{Y}) + \frac{\mathbf{1}^{\mathrm{T}}\tilde{X}\mathbf{1}\mathbf{1}^{\mathrm{T}}\tilde{Y}\mathbf{1}}{(n-1)(n-2)} - \frac{2}{(n-1)}\mathbf{1}^{\mathrm{T}}\tilde{X}\tilde{Y}\mathbf{1}\right)$$

(3)

where $\widetilde{X}$ and $\widetilde{Y}$ were obtained by setting the diagonal entries of the similarity matrices X and Y to zero.

IV. Results

IV.A. Tumor detection accuracy

Both Swin and ViT models produced a higher tumor DR when using wild-pretraining compared to self-pretraining as well as scratch training. Swin benefitted most from wild-pretraining with respect to DR on both datasets (Table 2). Wild-pretrained ViT produced higher DR than self-pretrained ViT for the LRad, but not for the LC dataset. The wild-pretrained CNN resulted in a slightly higher DR for LRad but lower DR for the LC dataset compared with its self-pretrained instance. The tumors detected by at least one model are used to report segmentation accuracy, yielding 139 for LRad and 179 tumors for the LC dataset, respectively.

IV.B. Segmentation accuracy

Tumor segmentation accuracies for all three analyzed networks are shown in Table 2. Both ViT and Swin models produced a higher accuracy with either pretraining approach compared to their scratch trained counterparts. Wild-pretrained Swin produced a significantly higher segmentation accuracy than self-pretrained Swin (DSC $p < 0.0001$, HD95 $p < 0.0001$) on the LRad dataset but not the LC dataset (DSC $p = 0.32$, HD95 $p = 0.063$). On the other hand, neither wild-pretrained ViT nor wild-pretrained CNN methods resulted in a significantly different accuracy compared to their self-pretrained counterparts (Table 2). Scratch and both pretrained CNN models were similarly accurate for the LC dataset.

IV.C. Dependence of segmentation accuracy on tumor volumes

Figure 2 shows the relation of DSC segmentation accuracy with respect to the tumor volume. The confidence intervals for the R^2 correlation plot are also shown. In general, all models showed wide confidence intervals with increasing tumor volumes for both datasets. In both datasets, ViT and Swin models showed a smaller dependence of DSC to the tumor volume compared to the CNN model. Furthermore, wild-pretraining slightly reduced the dependence of accuracy with respect to volume for the transformer models, indicating limited impact of tumor volume on DSC.

IV.D. Robustness to CT imaging variations

Table 3 shows the accuracy differences due to the presence and absence of CT contrast. All three models were significantly more accurate than their scratch-trained instances for contrast enhanced CTs. Both wild-pretrained Swin and ViT models were significantly more accurate than scratch trained models when evaluated on non-contrast CT scans. Wild-pretrained Swin was significantly more accurate than its self-pretrained instance for both contrast and non-contrast CT cases. On the other hand, wild-pretraining did not benefit accuracy gains over self-pretraining for ViT and CNN models, albeit wild-pretrained ViT

showed a trend towards significance for contrast scans. Majority of cases in the training set were non-contrast CTs. This result indicates that wild-pretraining improves accuracy over self-pretraining when applied to test set with different characteristics than the training set.

Figure 3 shows the segmentation accuracy with respect to CT reconstruction convolution kernels for the public LRad dataset (n=139 CTs). Wild-pretraining resulted in significantly higher accuracy for Swin over self-pretraining for Recon 1 ($p < 0.001$) and over scratch training for Recon 1 ($p < 0.001$) and Recon 3 kernels ($p = 0.006$). There was no significant difference between either wild- or self-pretrained and scratch trained ViT and CNN models for the various kernels. Wild-pretrained ViT ($p = 0.008$) and self-pretrained ViT were more accurate than scratch trained ViT ($p = 0.02$) for Recon 3 kernel.

Next, we measured the robustness of segmentation with different reconstruction kernels by comparing accuracy differences for the same model across the convolution kernels. Whereas, wild-pretraining resulted in similar accuracies across reconstruction kernels for Swin, self-pretrained Swin produced less accurate segmentations with Recon 3 kernel compared to other two kernels. This indicates that wild-pretraining increased robustness for Swin with respect to reconstruction kernels. ViT showed similar mean accuracy for the three reconstruction kernels with self- and wild-pretraining. Scratch training showed a larger variation in the accuracy across the kernel reconstructions for both Swin and ViT architectures. The CNN network showed a wide variation in accuracy for all three kernels and training strategies.

Subset analysis of an additional set of 20 patients with paired reconstructions showed that wild-pretrained Swin produced significantly different accuracies between the two kernels when using 5 mm slices ($p = 0.036$) but not with 2.5 mm slices (Table 4). On the other hand, self-pretrained Swin model showed similar but lower accuracy than the wild-pretrained Swin model for both kernel reconstructions and slice thicknesses. Wild-pretrained ViT and CNN models were robust to convolutional kernel differences with both slice thicknesses, as was the self-pretrained CNN model. However, both model architectures were less accurate than wild-pretrained Swin. The self-pretrained ViT showed significant accuracy difference between the convolutional kernels for the 2.5 mm slices but not the 5mm slices. These results indicate that slice thickness may impact the accuracy and robustness of the transformer models with respect to the convolutional kernels.

Figure 4 shows the segmentations produced by the three networks when subjected to self-and wild-pretraining on two representative patients with contrast CT scan using (A) GE's lung reconstruction or Recon 3 kernel, (B) GE's standard reconstruction or Recon 1 kernel and (C) non-contrast CT scan. In the case of both ViT and Swin models, wild-pretrained instances better approximate the manual delineations irrespective of image contrast compared to the self-pretrained instances. The results are similar for both self- and wild-pretrained instances when applied to the CNN model.

Table 5 shows the detection rate (DR) and the segmentation accuracy for the detected tumors on the lung phantom dataset. The wild-pretrained Swin achieved the highest DR across all models. Although the wild-pretrained Swin produced a similar DSC and a

slightly higher HD95 than the self-pretrained Swin, it had a much higher DR, indicating that wild-pretrained model was better at also detecting the small lesions. Similar scores were observed for wild-pretrained ViTs, where the detection rate was higher and the HD95 lower, suggesting a potential trade-off in finer detection abilities with the self-pretrained models. The CNN model on the other hand showed very low tumor detection capability compared to the transformer models. Figure 5 shows the segmentations produced by self- and wild-pretrained CNN, ViT and Swin models on the phantom dataset.

IV.E. Ablation analysis

IV.E.1. Impact of different pretext tasks and pretraining strategy—Table 6 shows the detection rate and segmentation accuracy for the detected tumors for the network subjected to wild- and self-pretraining using multiple pretext tasks. As shown, wild-pretraining with the SMIT pretext tasks that combined MIP, ITD, and MPD losses resulted in the highest tumor segmentation accuracy. Also, wild-pretraining with SMIT task was more accurate than self-pretraining using the same pretext task. Two-stage training that sequentially performed wild- followed by self-pretraining did not result in a higher accuracy compared to the wild-pretrained SMIT model. As shown in Table 6, two-stage pretraining deteriorated detection and segmentation accuracy on both LRad and LC datasets.

On the other hand, masked image prediction or MIP resulted in higher accuracy for self-pretraining compared to wild-pretraining, indicating the influence of pretext task on downstream accuracy. It is notable that self-pretrained Swin using MIP also resulted in the higher DR compared to all evaluated configurations. All other pretext tasks resulted in similar accuracies for the self- and wild-pretrained versions using those same tasks.

When comparing the efficacy of pretext tasks, SMIT resulted in the highest accuracy with wild-pretraining compared to all other pretext tasks using wild- or self-pretraining, followed by MIP pretext task used in self-pretraining. Contrastive and image token distillation (ITD) both of which focus on learning the global image features resulted in worse accuracy compared to other pretext tasks applied with wild-pretraining. These results indicate that losses that encourages the network to learn features to distinguish images as a whole is less beneficial for segmenting individual structures such as tumors.

IV.E.2. Impact of fine-tuning with homogeneous dataset—We evaluated the impact of pretraining strategies when the networks were fine-tuned using a homogenous dataset (279 non-contrast CTs, Siemens, 3mm thickness, Recon 1) and evaluated on testing dataset from LRad with varied CT acquisitions. As shown in Table 7, wild-pretrained Swin models resulted in significantly higher accuracy for contrast but not the non-contrast scans, indicating that wild-pretraining benefits when the test dataset differs from training dataset. On the other hand, wild-pretraining only resulted in a trend toward significance for ViT with contrast scans, indicating that network architecture combined with pretraining strategy is important for achieving robustness.

IV.F. Feature reuse analysis

IV.F.1. Pretraining strategy impacts feature reuse with different CT acquisitions—Figure 6 shows the feature reuse between wild-pretrained and fine-tuned as well as self-pretrained and fine-tuned models computed using CKA for Recon 1 and Recon 3 kernels and for contrast and non-contrast CT from the public LRad dataset. The aforementioned CT variations were examined as they were more frequent in the analyzed dataset.

Self-pretrained models resulted in higher feature reuse in all layers for contrast and non-contrast CTs (Figure 6(A, B)). Whereas some feature reuse especially in the low level layers is beneficial, feature reuse even in high level features close to output indicates limited adaptation of the network for the segmentation task. Furthermore, feature reuse in the off-diagonals (layers 4 to 14) indicates redundancy of features, indicating lack of different features extracted across the layers. On the other hand, wild-pretrained model resulted in resulted in larger deviations of features close to the later encoder layers (13 and 14) for contrast compared to non-contrast CTs (Figure 6 A and B). Analysis of Recon 1 and Recon 3 showed higher feature reuse with wild-pretrained models for the earlier layers compared to later layer (13 and 14) as shown in Figure 6 C and D. Larger deviation of feature reuse in the off-diagonals (layers 5 to 14) with respect to low-level layers (1 to 4) indicates higher feature differentiation across layers, which is important to improve segmentation accuracy. Self-pretraining showed a similar pattern of feature reuse as the wild-pretraining for lower-level layers. However, feature differentiation at high level layers was less marked with self-pretraining compared to wild-pretraining for both Recon 1 and Recon 3.

Feature reuse for two-stage training: We performed CKA on the LC dataset, which consisted of 196 patient scans. CKA analysis was performed with a minibatch of 4 as this was the maximum number that would fit into the A100 GPUs with 80GB memory to understand why accuracy reduction occured with two-stage pretraining compared to wild-pretraining. As shown in Figure 7(A,B), feature reuse with self-pretraining and wild-pretraining were consistent with the observations on the LRad dataset (Figure 6), showing high reuse across all layers for the former and larger feature deviations close to the output layers (13 and 14) for the latter. On the other hand, two-stage pretraining resulted in reduced feature reuse in layers 5 to 14. Very low feature reuse indicates that features from pretraining were discarded during fine-tuning, thus eliminating the benefit of two-stage pretraining. Further analysis comparing two-stage pretraining with respect to self-pretraining and wild-pretraining showed differences in features across multiple layers including 5 to 14 (Figure 7(D,E), albeit to a lesser degree compared to fine-tuning (Figure 7(C)). Furthermore, features extracted from wild-pretraining and self-pretraining are different across multiple layers (Figure. 7(F), which clearly shows that features are altered going from wild- to self-pretraining thus reducing feature reuse fine-tuning.

Impact of pretext tasks on feature reuse: Figure 8 shows the feature reuse analysis for the Swin model subjected to self- and wild-pretraining using various pretext tasks and evaluated on the LRad dataset. There was a considerable variation in the reuse of the features for the same network architecture (Swin) when wild-pretrained with different pretext tasks as

shown in Figure 8 (A) to (E). Concretely, contrastive task resulted in the highest feature reuse even across different feature layers (off-diagnoal entries in the CKA matrix) for both self- and wild-pretrained models. ITD, a global image feature matching loss, resulted in the lowest feature reuse, followed by ITD+MPD, MIP, and SMIT for both self- and wild-pretrained models(Figure 8). SMIT, which uses a combination of ITD, MIP and MPD, the latter two are spatial locality context losses, resulted in higher feature reuse in the early (1 to 4) and middle level features (5 to 9) compared to late level (10 to 14) layer features. In addition, the features across different layers (off-diagonal entries of the CKA matrix) were different between wild-pretrained and fine-tuned features for SMIT, MPD, and ITD tasks but not for the contrastive learning task. Self-pretraining (Figure 8 E) with SMIT resulted in more differentiation of off-diagonal features (layers 5 to 14 compared to earlier level features 1 to 4) but such differentiation was to a lesser degree than wild-pretrained SMIT. In general, wild-pretraining resulted in feature changes especially close to the later stage encoder layers (13 and 14) for all the pretext tasks when compared to self-pretraining.

For both self-pretrained and wild-pretrained models, a high similarity in the earlier level (layer 1–4) is present regardless of the pretext tasks, except for the ITD task for the self-pretrained model. On the other hand, across all tasks, feature dissimilarity increases close to output layer as the task varies between wild-pretraining and fine-tuning. This separation also leads to improved accuracy of the wild-pretrained Swin network.

V. Discussion

This study performed a comprehensive analysis of robustness to CT imaging variations due to scratch versus self- versus wild-pretraining of two transformer and one convolutional neural network architectures. Contrary to the intuition that pretraining should always benefit accuracy irrespective of the network, our results showed that the transformer methods benefited more from wild-pretraining compared to CNN. Specifically, both transformer models showed a higher benefit from wild-pretraining compared to scratch training of those same models, indicating that pretraining provides useful features for downstream tasks. Importantly, Swin network showed a significant accuracy improvement with wild-pretraining compared to self-pretraining for contrast and non-contrast CTs as well as for Recon 1 and Recon 3 convolution kernels. On the other hand, wild-pretrained ViT showed variable benefit compared to self-pretrained ViT. Wild-pretraining increased accuracy especially when the test dataset differed from the fine-tuning dataset (contained larger and later stage tumors and predominantly acquired with non-contrast) as seen for LRad (smaller tumors and varied acquisitions) as well as in the analysis of contrast CTs when the models were trained on the homogeneous dataset containing only non-contrast CTs. CNN-based PRCLv2, on the other hand, did not show a clear benefit with either wild- or self-pretraining.

Pretraining possibly benefited transformers more than CNNs, because the former are data hungry methods as they destroy local image context by representing images as a sequence of tokens. On the other hand, CNN retains and in fact leverages the spatial dependencies inherent within images to extract features, making them less reliant on large datasets compared to transformers. As a result, the benefit of pretraining was not evident for CNN.

In order to understand why performance differences occurred with pretraining, we analyzed feature reuse between the features extracted following wild- or self-pretraining and following fine-tuning. Our analysis showed a distinct pattern of feature reuse differences between wild- and self-pretrained models. Specifically, self-pretraining resulted in high feature reuse across all layers for contrast and non-contrast CTs, even in the off-diagonal layers, indicating high redundancy of features and no modification of features to adapt to the downstream task. On the other hand, wild-pretraining resulted in higher feature reuse in the early (extracting edges) and mid-layers (extracting textures), which typically are reusable across tasks. However, deviations in features occurred only close to output layers (13 to 14), where one expects specialization of extracted features to conform to the task. Such a feature reuse pattern has been attributed to higher accuracy in natural and medical image classification tasks in other prior works[2,3].

Furthermore, two-stage pretraining applied to Swin did not increase downstream accuracy and in fact resulted in a deterioration of accuracy. Two-stage pretraining greatly diminished feature reuse except for the earliest layers (1 to 4). Very low feature reuse resulted from large changes in features from wild- to self-pretraining, which indicates that features from wild-pretraining were discarded following self-pretraining, thus eliminating the benefit of multi-stage pretraining for the downstream tasks. It is possible that the task dataset which contained larger tumors and more non-contrast CTs may have over-specialized features to similar examples, that led to lowering of accuracy for smaller tumors and cases with contrast-CTs. In depth analysis of performing two-stage pretraining by appropriately freezing network layers, number of self-pretraining epochs, pretext tasks are beyond scope of this work. Our results are partially consistent with prior work involving natural image analysis that showed fine-tuning following two-stage pretraining was less accurate on some tasks compared to fine-tuning following single-stage pretraining[38].

Another important aspect of our pretraining approach involves the use of unlabeled datasets. Different from prior works that used supervised pretraining with large and labeled datasets[1,2], SSL pretraining is reliant on the pretext tasks created to convert the unsupervised learning problem into a supervised learning problem. Hence, efficacy is tied to the ability of the model to extract robust, useful, and reusable features from the pretext tasks. Our analysis showed that a combination of masked image token prediction (MIP) and self-distillation based global image token (ITD) and masked patch token distillation (MPD) losses as used in the SMIT[8] was most effective for increasing accuracy with wild-pretraining. On the other hand, MIP task resulted in a higher accuracy with self-pretraining for the Swin model. Tasks that focused on learning global image features such as contrastive loss and ITD did not result in any difference between self- and wild-pretraining.

Feature reuse analysis also revealed that contrastive pretraining resulted in very high feature reuse across the layers that indicated less adaptation and diversity of features for the downstream task. On the other hand, ITD, which also focuses on learning global image representation resulted in very high feature diversity across all layers especially for self-pretrained models, indicating no benefit with pretraining. Somewhat similar to the SMIT pretext task, MIP when applied with wild-pretraining resulted in higher feature reuse in earlier layers but larger diversity in later layers that leads to higher accuracy. One important

consequence of this analysis is the possibility that fine-tuning needs to be performed only for the later layers while freezing the early layers, that can lead to more efficient fine-tuning, reduced demands on labeled data, and compute time, all of which will be analyzed in future.

VI. Conclusions

Systematic analysis of the SSL pretraining performed with unlabeled image sets in combination with three broad deep network architectures and multiple pretext tasks showed that benefit of wild-pretraining is architecture and pretext task dependent. Concretely, Swin models benefited most from wild-pretraining. Our results also showed that wild-pretraining improved accuracy over scratch and self-pretraining especially when the testing data deviated from the training/fine-tuning dataset. These results indicate that careful evaluation of architectures in terms of accuracy and robustness to common data and image variations should be performed that extend beyond just assessing overall accuracy on the testing sets to inform improvements and further evaluations for clinical use.

Acknowledgement

This research was partially supported by the NCI R01CA258821 and the Memorial Sloan Kettering Cancer Center Support Grant/Core Grant NCI P30CA008748.

References

1. Ma J, He Y, Li F, Han L, You C, and Wang B, “Segment anything in medical images,” Nature Comm, vol. 15, no. 654, 2024.
2. Matsoukas C, Haslum JF, Sorkhei M, Söderberg M, and Smith K, “What makes transfer learning work for medical images: Feature reuse & other factors,” in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 9215–9224.
3. Goyal P, Caron M, Lefaudeux B, Xu M, Wang P, Pai V, Singh M, Liptchinsky V, Misra I, Joulin A, and Bojanowski P, “Self-supervised pretraining of visual features in the wild,” CoRR, vol. abs/2103.01988, 2021. [Online]. Available: https://arxiv.org/abs/2103.01988
4. Cheng J, Ye J, Deng Z, Chen J, Li T, Wang H, Su Y, Huang Z, Chen J, Jiang L et al. , “Sam-med2d,” arXiv preprint arXiv:2308.16184, 2023.
5. Liu J, Zhang Y, Chen J-N, Xiao J, Lu Y, A Landman B, Yuan Y, Yuille A, Tang Y, and Zhou Z, “Clip-driven universal model for organ segmentation and tumor detection,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 21152–21164.
6. Zhou Z, Sodha V, Pang J, Gotway MB, and Liang J, “Models genesis,” Medical Image Analysis, vol. 67, p. 101840, 2021.
7. Umapathy L, Brown T, Mushtaq R, Greenhill M, Lu J, Martin D, Altbach M, and Bilgin A, “Reducing annotation burden in MR: A novel MR-contrast guided contrastive learning approach for image segmentation,” Med Phys, 2023.
8. Jiang J, Tyagi N, Tringale K, Crane C, and Veeraraghavan H, “Self-supervised 3d anatomy segmentation using self-distilled masked image transformer (smit),” in Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part IV. Springer, 2022, pp. 556–566.
9. Willemink M, Roth R, and Sandfort V, “Toward foundational deep learning models for medical imaging in the new era of transformer networks,” Radiol Artif Intell, vol. 4, no. 6, 2022.
10. Nguyen DMH, Nguyen H, Mai TTN, Cao T, Nguyen BT, Ho N, Swoboda P, Albarqouni S, Xie P, and Sonntag D, “Joint self-supervised image-volume representation learning with intra-inter contrastive clustering,” in Proc. 37th AAAI AAAI Press, 2023.

11. Yan X, Naushad J, Sun S, Han K, Tang H, Kong D, Ma H, You C, and Xie X, “Representation recovering for self-supervised pre-training on medical images,” in 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2023, pp. 2684–2694.
12. Qayyum A, Razzak I, Mazher M, Khan T, Ding W, and Niederer S, “Two-stage self-supervised contrastive learning aided transformer for real-time medical image segmentation,” IEEE Journal of Biomedical and Health Informatics, pp. 1–10, 2023.
13. Yu S, Liu L, Wang Z, Dai G, and Xie Y, “Transferring deep neural networks for the differentiation of mammographic breast lesions,” Science China Technological Sciences, vol. 62, pp. 441–447, 2019.
14. He K, Chen X, Xie S, Li Y, Doll’ar P, and Girshick RB, “Masked autoencoders are scalable vision learners. 2022 ieee,” in CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 15979–15988.
15. Tang Y, Yang D, Li W, Roth HR, Landman B, Xu D, Nath V, and Hatamizadeh A, “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, 2022, pp. 20730–20740.
16. Zhu J, Li Y, Hu Y, Ma K, Zhou SK, and Zheng Y, “Rubik’s cube+: A self-supervised feature learning framework for 3D medical image analysis,” Medical Image Analysis, vol. 64, p. 101746, 2020.
17. Taleb A, Loetzsch W, Danz N, Severin J, Gaertner T, Bergner B, and Lippert C, “3D self-supervised methods for medical imaging,” Advances in Neural Information Processing Systems, vol. 33, pp. 18158–18172, 2020.
18. Zhou H-Y, Lu C, Chen C, Yang S, and Yu Y, “A unified visual information preservation framework for self-supervised pre-training in medical image analysis,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 7, pp. 8020–8035, 2023. [PubMed: 37018263]
19. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, and Houlsby N, “An image is worth 16×16 words: Transformers for image recognition at scale,” in International Conference on Learning Representations (ICLR), 2021.
20. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, and Guo B, “Swin transformer: Hierarchical vision transformer using shifted windows,” in Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 10012–10022.
21. Cortes C, Mohri M, and Rostamizadeh A, “Algorithms for learning kernels based on centered alignment,” The Journal of Machine Learning Research, vol. 13, pp. 795–828, 2012.
22. Xiao Y, Yang G, and Song S, Lesion Segmentation in Surgical and Diagnostic Applications: MICCAI 2022 Challenges, CuRIOUS 2022, KiPA 2022 and MELA 2022, Held in Conjunction with MICCAI 2022, Singapore, September 18–22, 2022, Proceedings. Springer Nature, 2023, vol. 13648.
23. Ji Y, Bai H, Yang J, Ge C, Zhu Y, Zhang R, Li Z, Zhang L, Ma W, Wan X et al. , “Amos: A large-scale abdominal multi-organ benchmark for versatile medical image segmentation,” arXiv preprint arXiv:2206.08023, 2022.
24. Harmon SA, Sanford TH, Xu S, Turkbey EB, Roth H, Xu Z, Yang D, Myronenko A, Anderson V, Amalou A et al. , “Artificial intelligence for the detection of covid-19 pneumonia on chest ct using multinational datasets,” Nature communications, vol. 11, no. 1, p. 4080, 2020.
25. Heller N, Sathianathen N, Kalapara A, Walczak E, Moore K, Kaluzniak H, Rosenberg J, Blake P, Rengel Z, Oestreich M, Dean J, Tradewell M, Shah A, Tejpaul R, Edgerton Z, Peterson M, Raza S, Regmi S, Papanikolopoulos N, and Weight C, “C4kc-kits,” The Cancer Imaging Archive., 2019.
26. Roth HR, Lu L, Farag A, Shin H-C, Liu J, Turkbey EB, and Summers RM, “Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation,” in Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part I 18. Springer, 2015, pp. 556–564.
27. Aerts H, E. RV, Leijenaar RT, Parmar C, Grossmann P, Carvalho S, and Lambin P, “Data from NSCLC-radiomics. The Cancer Imaging Archive,” 2015.

28. Bakr S, Gevaert O, Echegaray S, Ayers K, Zhou M, Shafiq M, Zheng H, Benson JA, Zhang W, Leung AN et al. , "A radiogenomic dataset of non-small cell lung cancer," Scientific data, vol. 5, no. 1, pp. 1–9, 2018. [PubMed: 30482902]
29. Zhao B, "Lung phantom (version 2) [data set]," 10.7937/k9/tcia.2015.08a1ixoo, 2015.
30. Chen Z, Agarwal D, Aggarwal K, Safta W, Balan MM, and Brown K, "Masked image modeling advances 3d medical image analysis," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2023, pp. 1970–1980.
31. Caron M, Touvron H, Misra I, Jégou H, Mairal J, Bojanowski P, and Joulin A, "Emerging properties in self-supervised vision transformers," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9650–9660.
32. Zhou J, Wei C, Wang H, Shen W, Xie C, Yuille A, and Kong T, "ibot: Image bert pre-training with online tokenizer," arXiv preprint arXiv:2111.07832, 2021.
33. Loshchilov I and Hutter F, "Decoupled weight decay regularization," in International Conference on Learning Representations, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:53592270
34. Ilya L and Frank H, "Sgdr: Stochastic gradient descent with warm restarts," arXiv preprint arXiv:1608.03983, 2016.
35. Jiang J, Hu YC, Liu CJ, Halpenny D, Hellmann MD, Deasy JO, Mageras G, and Veeraraghavan H, "Multiple resolution residually connected feature streams for automatic lung tumor segmentation from ct images," IEEE Transactions on Medical Imaging, vol. 38, no. 1, pp. 134–144, 2019. [PubMed: 30040632]
36. Nguyen T, Raghu M, and Kornblith S, "Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth," arXiv preprint arXiv:2010.15327, 2020.
37. Song L, Smola A, Gretton A, Bedo J, and Borgwardt K, "Feature selection via dependence maximization." Journal of Machine Learning Research, vol. 13, no. 5, 2012.
38. Lee S, Kang M, Lee J, Hwang SJ, and Kawaguchi K, "Self-distillation for further pre-training of transformers," arXiv preprint arXiv:2210.02871, 2022.

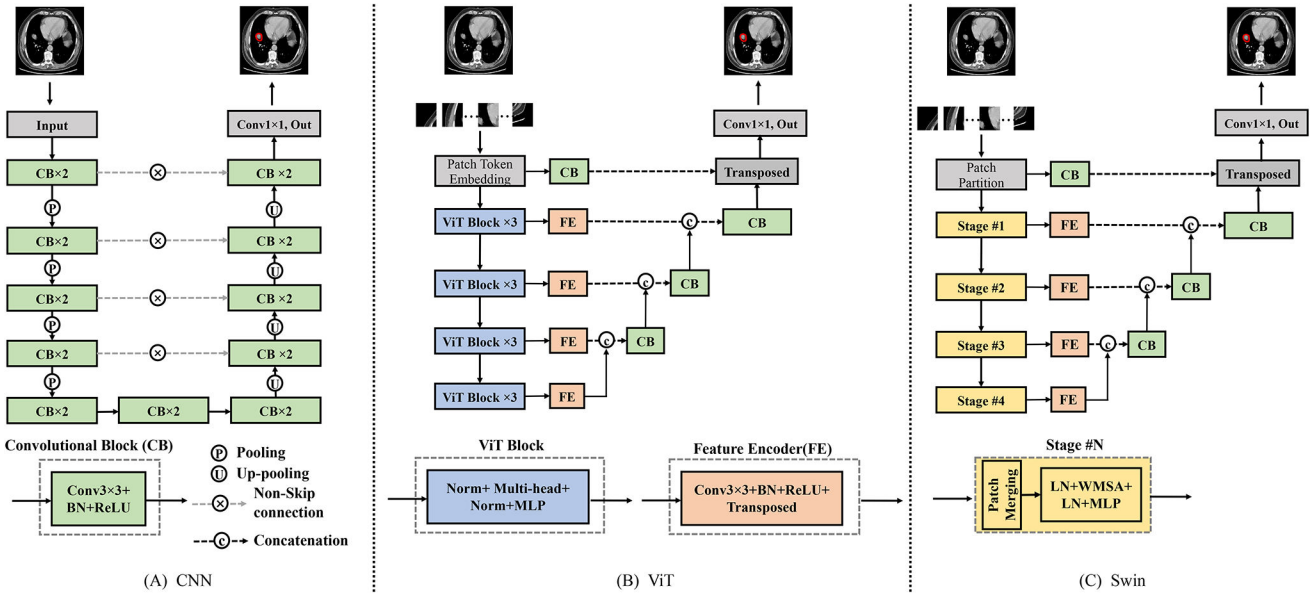

Figure 1:
Schematics of (A) CNN, (B) ViT, and (C) Swin networks analyzed in this study.

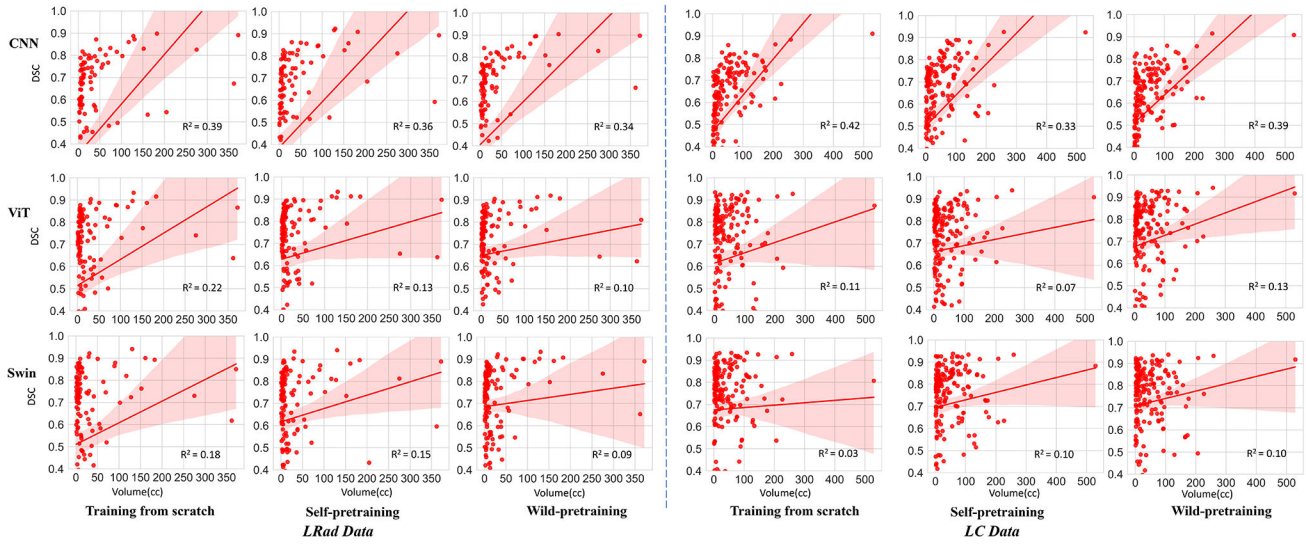

Figure 2:

The scatter plot of DSC versus tumor volume (cc) to assess dependency of accuracy on the tumor volume for the analyzed architectures subjected to scratch, self-pretraining and wild-pretraining followed by fine-tuning for the two testing datasets.

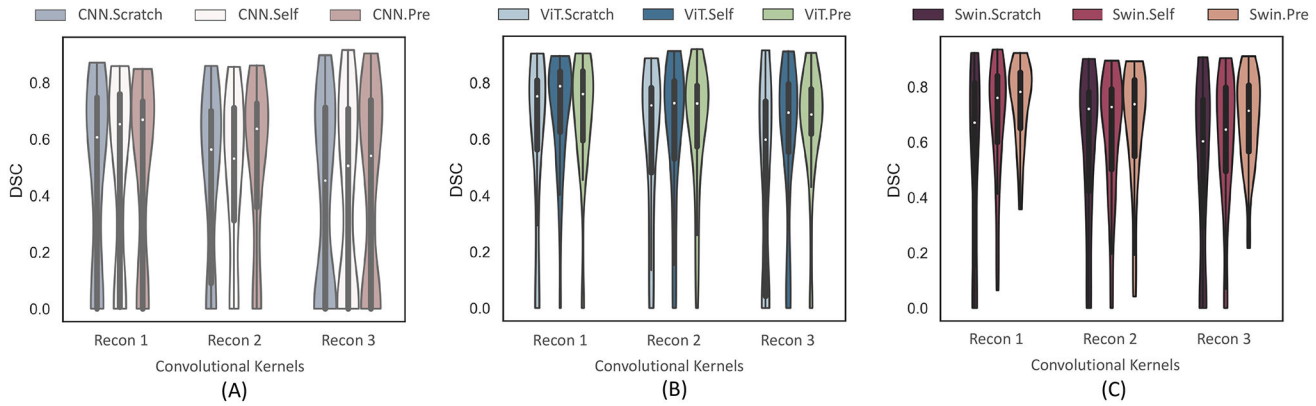

Figure 3:

Influence of CT reconstruction kernel on segmentation accuracy with (A) CNN backbone and (B) ViT backbone, (c) Swin backbone. The shaded regions correspond to the data distribution estimated using Kernel density estimation. The edges of the box plot correspond to the lower quartile (or 25th percentile) and upper quartile (or 75th percentile) ranges. The whisker bar extends to the smallest and largest data point within 1.5 times the interquartile range below the first or above the third quartile for the lower and upper ends of the whisker bars, respectively.

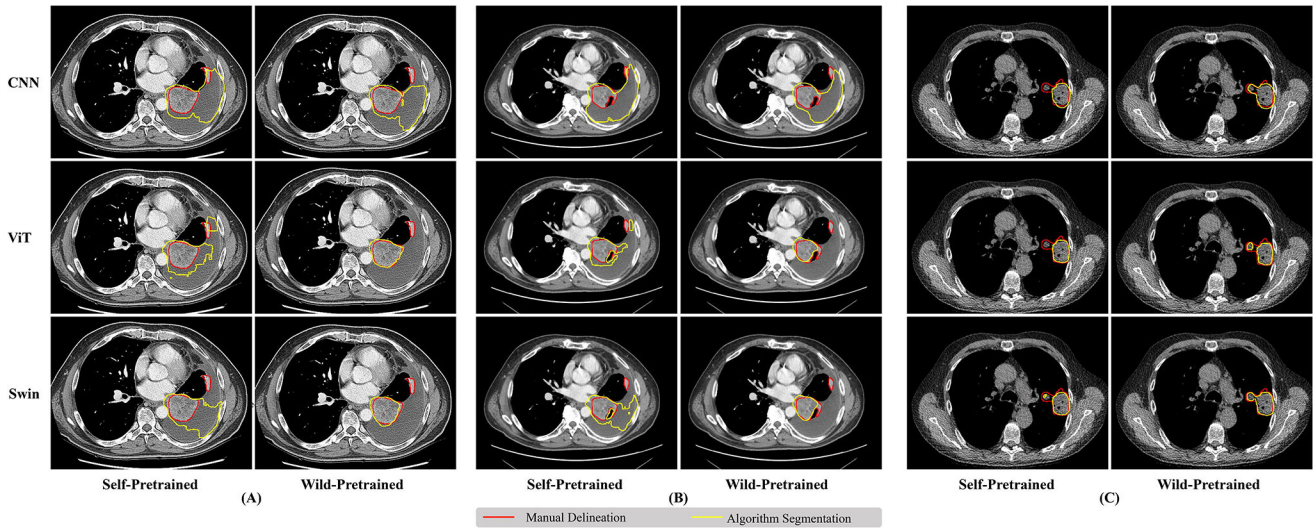

Figure 4:

Segmentation (yellow contour) produced using CNN, ViT and Swin-based SMIT model using wild-pretaining and self-pretraining. (A) Contrast CT scan using GE lung reconstruction (B)Contrast CT scan using GE standard reconstruction (C) Non-contrast CT scan using GE lung reconstruction. Manual delineation is shown in red and algorithm delineation is shown in yellow.

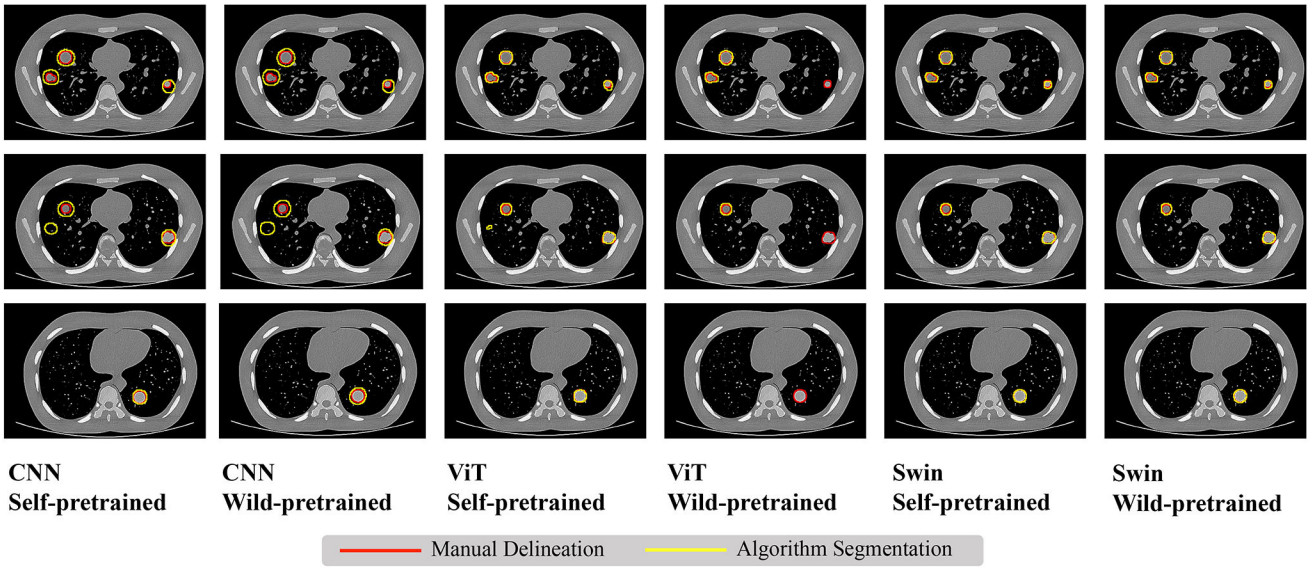

Figure 5:

Results on the phantom image scan (dataset). Yellow contour indicates the model segmentation and red contour indicates the phantom ground truth.

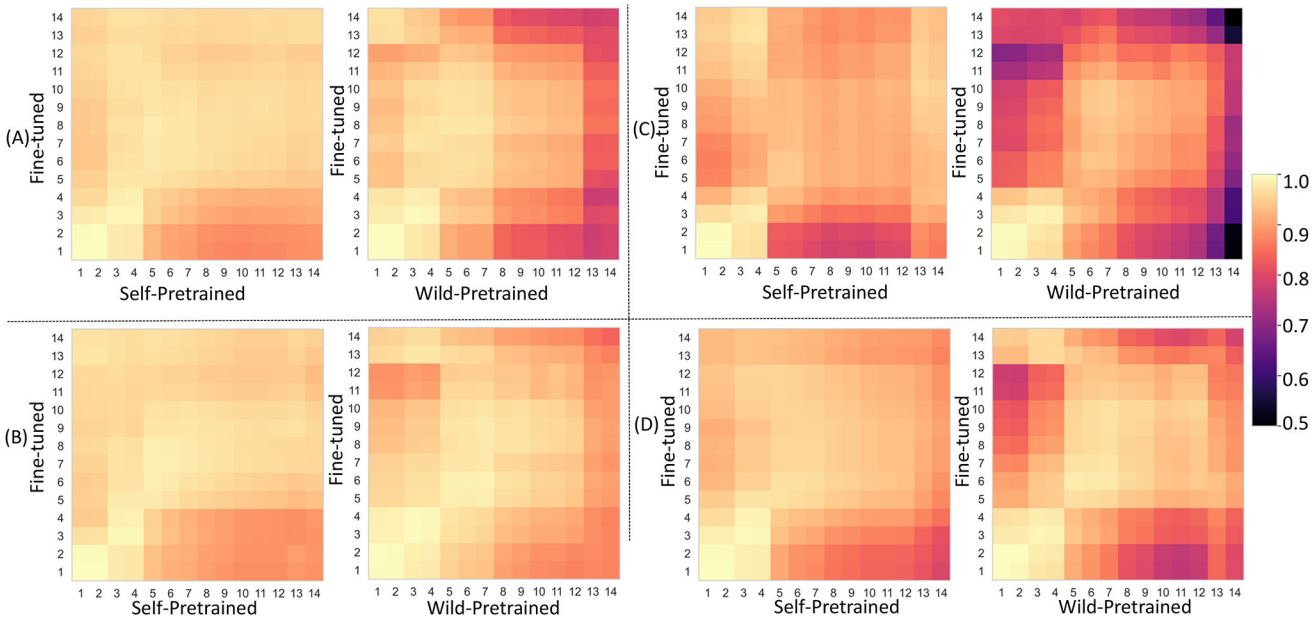

Figure 6:

CKA analysis performed on the Swin backbone on the public LRad data to measure feature reuse for (A) contrast and (B) non-contrast CT as well as CT images reconstructed using (C) Recon 1 and (D) Recon 3.

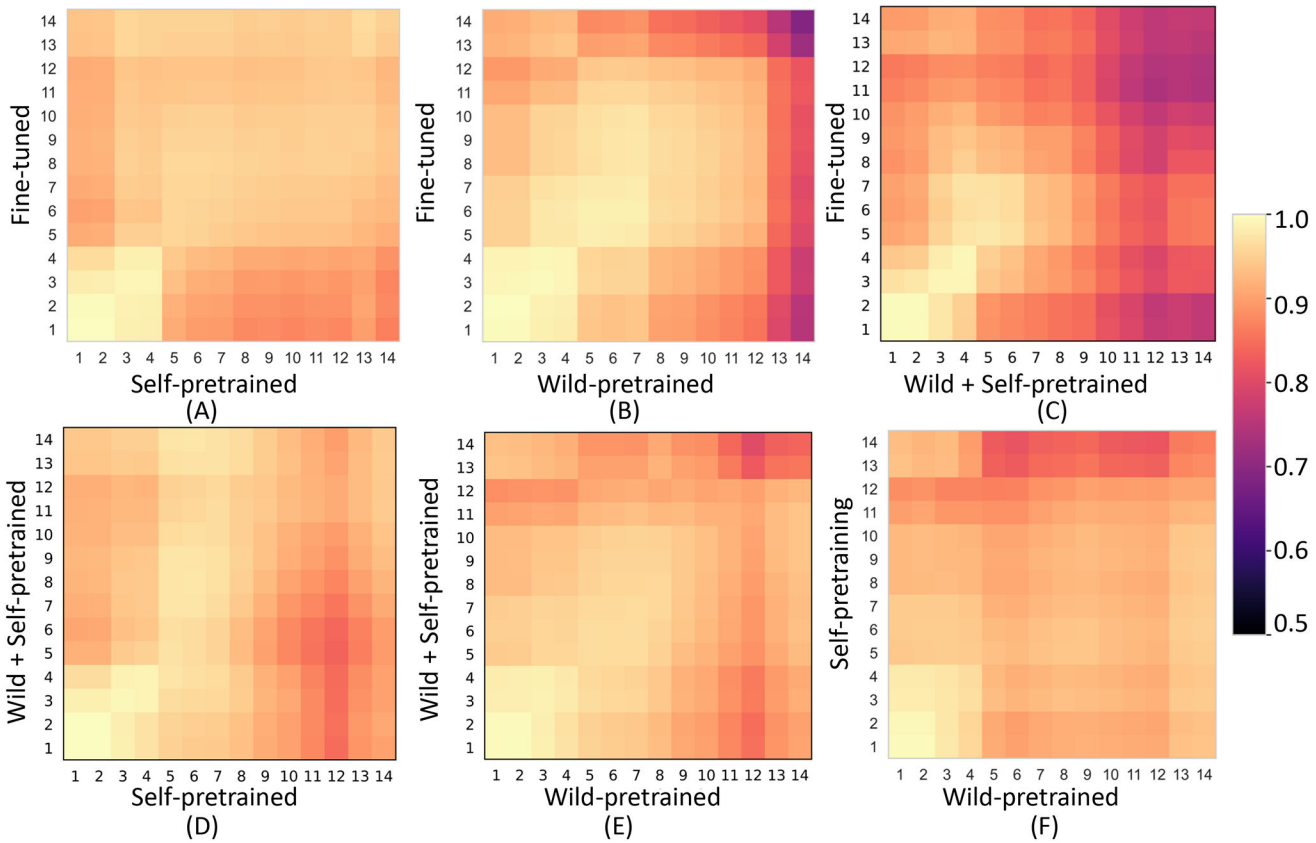

Figure 7:

CKA analysis performed on the Swin backbone on the 196 LC data for self-pretrained, wild-pretraind and two-stage (wild+self pretrained) models. We also show the cka analysis of self-pretrained versus wild+self-pretrained (D), wild-pretrained versus wild+self-pretrained (E) and wild-pretrained versus self-pretrained (F).

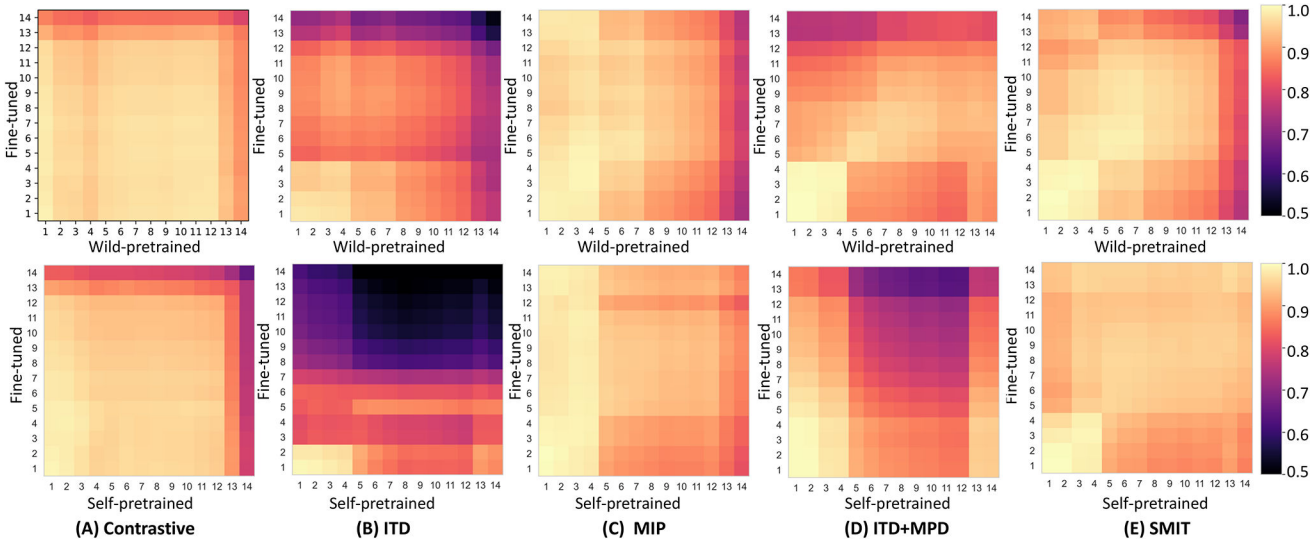

Figure 8:

CKA analysis performed on the Swin backbone on the 196 LC data using different pretext tasks including (A) Contrastive, (B)ITD, (C)MIP, (D)ITD+MPD and (E) SMIT with wild- and self-pretraining.

Table 1:

Summary of wild- pretraining, self-pretraining, fine-turning, and testing datasets.

Data	Location	Number	Manufacturer	Thickness	Kernels	Contrast Information
(A) Wild-pretraining (no labels):						
MELA 2022[22]	Chest	880	NA	1 mm	NA	contrast, non-contrast
AMOS 2022[23]	Chest-Abd-pelvis	360	NA	5 mm to 7.5 mm	NA	contrast, non-contrast
COVID-19[24]	Chest	609	NA	5 mm	NA	contrast, non-contrast
KITS[25]	Abdomen-pelvis	411	Siemens, Toshiba	3 mm	Recon 1	arterial, late, non-contrast
Pancreas CT[26]	Chest-Abdomen	80	NA	1 mm to 5mm	NA	contrast, non-contrast
Internal 1 Radiotherapy	Chest; Chest-abdomen	5,124	GE	3 mm to 5 mm	Recon 1, 3	contrast, non-contrast
Internal 2 Radiotherapy	Head and neck	2,632	GE	2.5 mm to 3 mm	Recon 1	contrast, non-contrast
(B) Fine-tuning† (with labels):						
TCIA NSCLC[27]	Chest-abdomen	377 (train 350, val 27)	Siemens, CMS	3 mm	Recon 1	contrast, non-contrast
(C) Testing:						
LRad[28]	Chest	156	Siemens, Toshiba, GE	0.9 mm to 5 mm	Recon 1, 2 Recon 3	contrast, non-contrast
LC	Chest-abdomen	196	GE	1.25 mm to 5mm	Recon 1, Recon 3	contrast, non-contrast
Lung Phantom[29]	Chest	1	GE	1.25 mm	Recon 1	contrast

†: used in self-pretraining. Recon 1 kernels: GE ”standard” and ”bone”, Siemens with < B40; Recon 2: GE ”Bone Plus”, Siemens with ≥ B40 and < B50; Recon 3: GE ”Lung”, Siemens ≥ B50. The TCIA NSCLC dataset is used for multiple purposes across different settings including scratch-training, finetuning in wild-pretraining and self-pretraining

Table 2:

Tumor detection and segmentation accuracy with pretraining methods and transformer architectures. DR: Detection rate. Significance tests measured DSC differences between wild-pretrained with scratch and self-pretrained instances of the same architecture.

Model	Training	Pretext Task	LRad DSC	p-value	HD_{95}	p-value	DR	LC DSC	p-value	HD_{95}	p-value	DR
CNN	Scratch	N/A	0.42±0.34	0.005	11.99±8.34	<0.0001	0.48	0.54±0.27	0.41	**12.36±10.37**	0.18	0.60
CNN	Self	PCRLv2	0.45±0.33	0.222	**10.07±8.32**	0.14	0.52	**0.56±0.23**	0.016	12.50±10.93	0.34	**0.67**
CNN	Wild	PCRLv2	**0.46±0.33**	-	10.94±8.21	-	0.54	0.54±0.24	-	12.81±10.97	-	0.62
ViT	Scratch	N/A	0.55±0.31	<0.0001	8.71±7.51	0.43	0.60	0.68±0.27	0.0026	12.76±20.51	0.19	0.74
ViT	Self	SMIT	0.64±0.25	0.095	8.59±7.53	0.29	0.65	0.71±0.22	0.19	**10.97±10.62**	0.22	**0.80**
ViT	Wild	SMIT	**0.66±0.22**	-	**8.52±7.39**	-	0.69	**0.72±0.21**	-	11.33±12.78	-	**0.80**
Swin	Scratch	N/A	0.54±0.31	<0.0001	11.73±11.46	0.0011	0.58	0.70 ± 0.22	0.0041	10.79±10.82	0.054	0.79
Swin	Self	SMIT	0.63±0.23	<0.0001	8.95±8.57	<0.0001	0.67	0.74 ± 0.18	0.096	10.71±11.96	0.069	0.80
Swin	Wild	SMIT	**0.69±0.18**	-	**7.58±7.12**	-	**0.76**	**0.75 ± 0.15**	-	**9.69±8.97**	-	**0.85**

Table 3:

Tumor segmentation accuracy differences due to CT contrast evaluated on LRad dataset. Significance testing measured DSC differences between scratch, self- and wild-pretrained instances of the same architecture.

Model	Training	Contrast (N=85)	p-value	Non-Contrast (N=54)	p-value
CNN	Scratch	0.47±0.33	0.014	0.35±0.33	0.28
CNN	Self	0.51±0.32	0.72	0.36±0.32	0.43
CNN	Wild	**0.52±0.32**	-	**0.38±0.33**	-
ViT	Scratch	0.60±0.30	0.00004	0.47±0.31	9.13e-6
ViT	Self	0.65±0.27	0.064	**0.65±0.19**	0.61
ViT	Wild	**0.67±0.23**	-	0.64±0.20	-
Swin	Scratch	0.58±0.30	3.26e-6	0.48±0.31	5.21e-7
Swin	Self	0.66±0.23	0.0041	0.60±0.23	6e-4
Swin	Wild	**0.70±0.19**	-	**0.68±0.16**	-

Table 4:

Robustness of tumor segmentation to different scan reconstructions. Significance tests compared wild-pretrained to self-and scratch trained instances of the same architecture.

Model	Training	Slice 2.5 mm Recon 3	Recon 1	p-value	Slice 5 mm Recon 3	Recon 1	p-value
CNN	Scratch	0.20±0.20	0.22±0.21	0.61	0.27±0.24	0.28±0.26	0.26
CNN	Self	0.21±0.20	0.22±0.20	0.57	0.27±0.19	0.31±0.27	0.16
CNN	Wild	0.23±0.21	0.24±0.23	0.68	0.30 ± 0.25	0.34±0.31	0.13
ViT	Scratch	0.47±0.34	0.54±0.32	0.08	0.49±0.34	0.50±0.30	0.64
ViT	Self	0.64±0.18	0.56±0.25	0.019	0.58±0.26	0.51±0.23	0.14
ViT	Wild	0.67±0.16	0.58±0.26	0.077	0.62±0.24	0.56±0.22	0.11
Swin	Scratch	0.52±0.32	0.36±0.36	0.37	0.57±0.48	0.47±0.34	0.12
Swin	Self	0.58±0.27	0.54±0.31	0.13	0.52±0.28	0.49±0.30	0.058
Swin	Wild	0.70±0.18	0.66±0.21	0.058	0.62±0.28	0.58±0.26	0.036

Table 5:

Models' accuracy on the phantom CT scan with 8 lesions.

Model	Training	DSC	HD_{95}	DR
CNN	Scratch	0.54 ± 0.10	4.75 ± 1.56	1.000
CNN	Self	0.59 ± 0.09	4.00 ± 1.13	1.000
CNN	Wild	0.53 ± 0.07	6.07 ± 0.88	1.000
ViT	Scratch	0.69 ± 0.37	3.02 ± 2.38	0.750
ViT	Self	0.71 ± 0.23	2.60 ± 0.49	0.750
ViT	Wild	0.61 ± 0.38	3.24 ± 2.01	0.750
Swin	Scratch	0.70 ± 0.29	2.37 ± 0.87	0.750
Swin	Self	0.86 ± 0.01	1.46 ± 0.06	1.000
Swin	Wild	0.86 ± 0.06	1.75 ± 0.74	0.875

Table 6:

Impact of pretext tasks on down-stream accuracy applied to Swin model.

Pretext Task	Pretraining	LRad DSC	HD_{95}	DR	LC DSC	HD_{95}	DR
MIP	Self	0.67 ± 0.19	8.71±8.26	**0.78**	0.72±0.18	11.81±15.52	**0.85**
MIP	Wild	0.64±0.25	8.58±8.77	0.72	0.73±0.17	11.01±11.92	0.84
ITD	Self	0.64±0.24	8.62 ± 8.35	0.72	0.71 ± 0.21	11.11±12.14	0.81
ITD	Wild	0.64±0.24	8.33±7.93	0.69	0.72±0.20	10.61±12.33	0.82
ITD & MPD	Self	0.66 ± 0.22	8.62±8.36	0.72	0.71±0.21	11.56± 14.24	0.83
ITD & MPD	Wild	0.66±0.21	8.62±7.22	0.72	0.72±0.20	10.46±9.68	0.82
Contrastive	Self	0.63±0.24	9.29 ± 9.23	0.69	0.71 ± 0.23	11.44± 12.93	0.79
Contrastive	Wild	0.64±0.24	9.08±9.68	0.73	0.71±0.21	11.34±12.05	0.79
SMIT	Self	0.63±0.23	8.95±8.57	0.67	0.74 ± 0.18	10.71±11.96	0.80
SMIT	Wild	**0.69±0.18**	**7.58±7.12**	0.76	**0.75 ± 0.15**	**9.69±8.97**	0.85
SMIT	Wild + Self	0.65±0.21	8.52±7.72	0.72	0.73±0.17	10.65±11.41	0.82

Table 7:

Testing accuracy of models fine-tuned with with homogenous task data (Non-contrast enchanced CT, Siemens, Recon 1).

Model	Training Strategy	Contrast (N=85)	p-value	Non-Contrast (N=54)	p-value
ViT	Self-pretraining	0.61±0.27	0.0869	0.59±0.24	0.440
ViT	Wild-pretraining	0.63±0.25		0.56±0.25	
Swin	Self-pretraining	0.65±0.24	0.019	0.63±0.20	0.412
Swin	Wild-pretraining	0.69±0.20		0.65±0.19	