bioengineering

Article

Improving Data-Efficiency and Robustness of Medical Imaging Segmentation Using Inpainting-Based Self-Supervised Learning

Jeffrey Dominic [1], Nandita Bhaskhar [2,*], Arjun D. Desai [1,2], Andrew Schmidt [1], Elka Rubin [1], Beliz Gunel [2], Garry E. Gold [1], Brian A. Hargreaves [1,2,3], Leon Lenchik [4], Robert Boutin [1] and Akshay S. Chaudhari [1,5,6]

[1] Department of Radiology, Stanford University, Stanford, CA 94305, USA
[2] Department of Electrical Engineering, Stanford University, Stanford, CA 94305, USA
[3] Department of Bioengineering, Stanford University, Stanford, CA 94305, USA
[4] Department of Radiology, Wake Forest University School of Medicine, Winston-Salem, NC 27101, USA
[5] Department of Biomedical Data Science, Stanford University, Stanford, CA 94305, USA
[6] Stanford Cardiovascular Institute, Stanford University, Stanford, CA 94305, USA
* Correspondence: nanbhas@stanford.edu

Abstract: We systematically evaluate the training methodology and efficacy of two inpainting-based pretext tasks of context prediction and context restoration for medical image segmentation using self-supervised learning (SSL). Multiple versions of self-supervised U-Net models were trained to segment MRI and CT datasets, each using a different combination of design choices and pretext tasks to determine the effect of these design choices on segmentation performance. The optimal design choices were used to train SSL models that were then compared with baseline supervised models for computing clinically-relevant metrics in label-limited scenarios. We observed that SSL pretraining with context restoration using 32 × 32 patches and Poission-disc sampling, transferring only the pretrained encoder weights, and fine-tuning immediately with an initial learning rate of 1×10^{-3} provided the most benefit over supervised learning for MRI and CT tissue segmentation accuracy ($p < 0.001$). For both datasets and most label-limited scenarios, scaling the size of unlabeled pretraining data resulted in improved segmentation performance. SSL models pretrained with this amount of data outperformed baseline supervised models in the computation of clinically-relevant metrics, especially when the performance of supervised learning was low. Our results demonstrate that SSL pretraining using inpainting-based pretext tasks can help increase the robustness of models in label-limited scenarios and reduce worst-case errors that occur with supervised learning.

Keywords: self-supervised learning; MRI; CT; segmentation; machine learning; deep learning

check for updates

Citation: Dominic, J.; Bhaskhar, N.; Desai, A.D.; Schmidt, A.; Rubin, E.; Gunel, B.; Gold, G.E.; Hargreaves, B.A.; Lenchik, L.; Boutin, R.; et al. Improving Data-Efficiency and Robustness of Medical Imaging Segmentation Using Inpainting-Based Self-Supervised Learning. *Bioengineering* **2023**, *10*, 207. https://doi.org/10.3390/bioengineering10020207

Academic Editors: Or Perlman, Efrat Shimron and Liang Luo

Received: 21 December 2022
Revised: 16 January 2023
Accepted: 23 January 2023
Published: 4 February 2023

1. Introduction

Segmentation is an essential task in medical imaging that is common across different imaging modalities and fields such as cardiac, abdominal, musculoskeletal, and lung imaging, amongst others [1–4]. Deep learning (DL) has enabled high performance on these challenges, but the power-law relationship between algorithmic performance and the amount of high-quality labeled training data fundamentally limits robustness and widespread use [5].

Recent advances in self-supervised learning (SSL) provide an opportunity to reduce the annotation burden for deep learning models [6]. In SSL, a model is first pretrained on a "pretext" task, during which unlabeled images are perturbed and the model is trained to predict or correct the perturbations. The model is then fine-tuned for downstream tasks. Previous works have shown that such models can achieve high performance even when fine-tuned on only a small labeled training set [7–9]. While most SSL models in computer vision have been used for the downstream task of image classification, segmentation comparatively remains an under-explored task [10].

In this work, we systematically evaluate the efficacy of SSL for medical image segmentation across two domains—MRI and CT. We investigate "context prediction" [7] and "context restoration" [8], two well-known and easy-to-implement archetypes of restoration-based pretext tasks that produce image-level representations during pretraining for eventual fine-tuning. Context prediction sets pixel values in random image patches to zero, while context restoration randomly swaps pairs of image patches within an image while maintaining the distribution of pixel values (Figure 1). For both tasks, the model needs to recover the original image given the corrupted image, a process we refer to as "inpainting". We consider these two tasks because they maintain same input-output sizes, akin to segmentation. We hypothesize that such pretext tasks allow construction of useful, image-level representations that are more suitable for downstream segmentation.

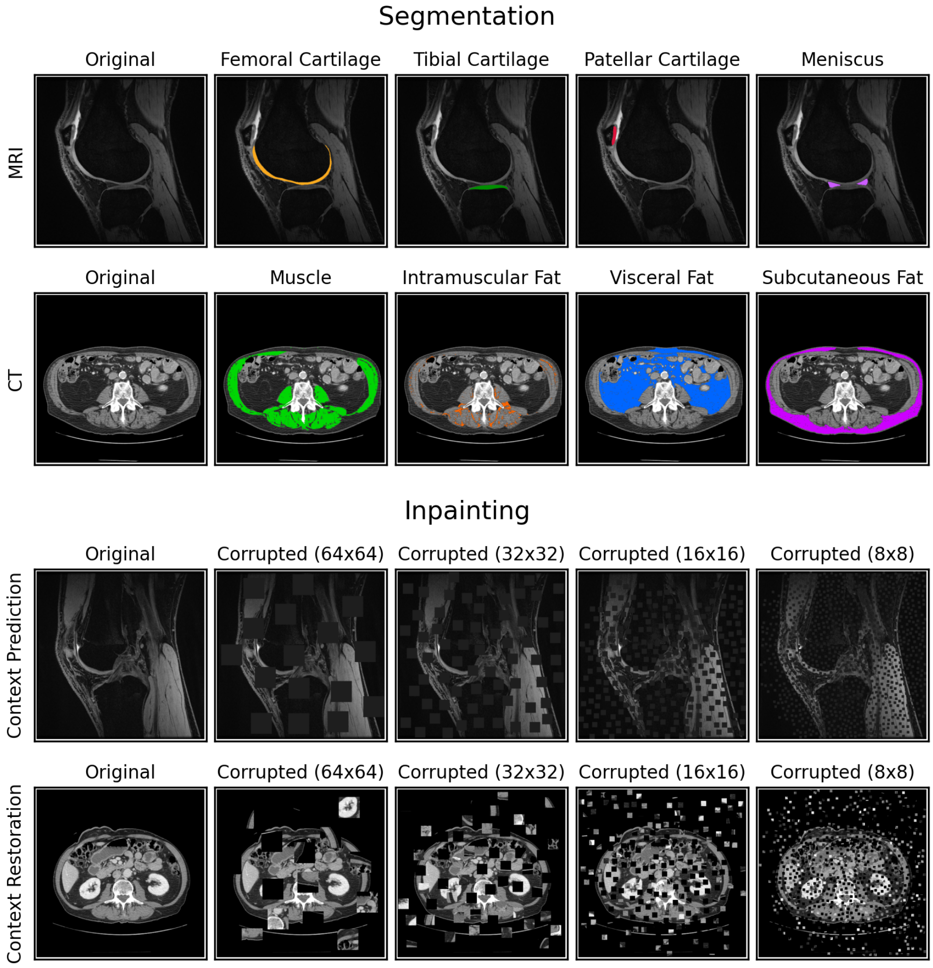


Figure 1. Example ground truth segmentations for the MRI and CT datasets (both with dimensions 512 × 512), and example image corruptions for context prediction (zero-ing image patches) and context restoration (swapping image patches). Since image corruption happens after normalization, the zero-ed out image patches for context prediction were actually replaced with the mean of the image. The "Inpainting" section depicts image corruptions with four different patch sizes: 64 × 64, 32 × 32, 16 × 16, and 8 × 8. The locations of these patches were determined using Poisson-disc sampling to prevent randomly overlapping patches.

While context prediction and context restoration have been proposed before, the effects of the large space of design choices for these two pretext tasks, such as patch sizes for image corruption and learning rates for transfer learning, are unexplored. In addition, prior works exploring SSL for medical image segmentation have primarily focused on the accuracy of segmentation using metrics such as Dice scores [8,11], but have not investigated if SSL can improve clinically-relevant metrics, such as T2 relaxation times for musculoskeletal MRI scans and mean Hounsfield Unit (HU) values for CT scans. These metrics can provide biomarkers of biochemical changes in tissue structure prior to the onset of gross morphological changes [12,13]. Furthermore, within the context of empirical data scaling laws in DL, past SSL works have rarely explored benefits of increasing the number of unlabeled images during pretraining [14]. Characterizing the efficiency of SSL methods with unlabeled data can lead to more informed decisions regarding data collection, an important practical consideration for medical image segmentation. In this work, we address the above gaps by (1) investigating how different design choices in SSL implementation affect the quality of the pretrained model, (2) calculating how varying unlabeled data extents affects SSL performance for downstream segmentation, (3) quantifying our results using clinically-relevant metrics to investigate if SSL can outperform supervised learning in label-limited scenarios, (4) evaluating where SSL can improve performance, across different extents of labeled training data availability, and (5) providing detailed analyses, recommendations, and open-sourcing our code to build optimal SSL models for medical image segmentation (code available at https://github.com/ad12/MedSegPy).

2. Materials and Methods

2.1. Datasets

2.1.1. MRI Dataset

We used 155 labeled knee 3D MRI volumes (around 160 slices per volume) from the SKM-TEA dataset [15] and 86 unlabeled volumes (around 160 to 180 slices per volume), each with slice dimensions of 512 × 512 (other scan parameters in [15]). All volumes were acquired using a 5-min 3D quantitative double-echo in steady-state (qDESS) sequence, which has been used for determining morphological and quantitative osteoarthritis biomarkers and for routine diagnostic knee MRI [16–19]. The labeled volumes included manual segmentations for the femoral, tibial, and patellar cartilages, and the meniscus. The labeled volumes were split into 86 volumes for training, 33 for validation, and 36 for testing, following the splits prescribed in [15]. The 86 training volumes were further split into additional subsets, consisting of 50% (43 volumes), 25% (22 volumes), 10% (9 volumes), and 5% (5 volumes) training data, to represent label-limited scenarios. All scans in smaller subsets were included in larger subsets.

2.1.2. CT Dataset

The 2D CT dataset consisted of 886 labeled and 7799 unlabeled abdominal CT slices at the L3 vertebral level. The unlabeled images were used in a prior study exploring the impact of body composition on cardiovascular outcomes [20]. The labeled slices included manual segmentations for subcutaneous, visceral, and intramuscular adipose tissue and muscle. These labeled slices were split into 709 slices for training, 133 for validation, and 44 for testing. The training set was split in a similar manner as the MRI volumes into 4 additional subsets of 50% (354 slices), 25% (177 slices), 10% (71 slices), and 5% (35 slices) training data. No metadata from the dataset were used in any models.

2.2. Data Preprocessing

All models segmented 2D slices for MRI and CT images. Each CT image was preprocessed at different windows and levels (*W*/*L*) of HU to emphasize different image contrasts, resulting in three-channel images: soft-tissue (*W*/*L* = 400/50), bone (*W*/*L* = 1800/40), and a custom setting (*W*/*L* = 500/50). All images were normalized to have zero mean and unit standard deviation, with MR images normalized by volume and CT images normalized per channel.

2.3. Model Architecture and Optimization

2D U-Net models [21] with Group Normalization [22], weight standardization [23], and He random weight initializations [24] were used for inpainting and segmentation

(Figure 2). Both inpainting and segmentation used identical U-Nets, except for the final convolutional layer, which we refer to as the "post-processing" layer. For inpainting, the post-processing layer produced an output image with the same number of channels as the input image, whereas for segmentation, it produced a 4-channel image for the four segmentation classes in each dataset.

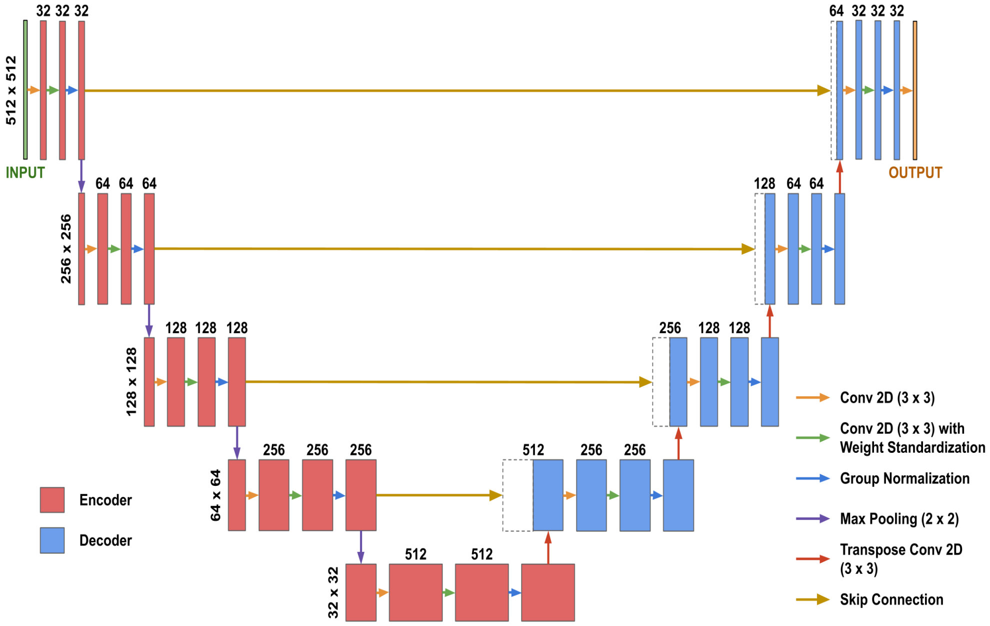


Figure 2. The U-Net architecture used for both inpainting and segmentation, which includes layers grouped into three categories: the "encoder" (in red), the "decoder" (in blue), and the "post-processing" layer (the final convolutional layer). Each dotted rectangular box represents a feature map from the encoder that was concatenated to the first feature map in the decoder at the same level.

We used L2 norm loss for inpainting and Dice loss, aggregated over mini-batches per segmentation class, for segmentation. All training was performed with early stopping and the ADAM optimizer [25] (β_1 = 0.99 and β_2 = 0.995) with a batch size of 9 on an NVIDIA 2080Ti GPU. Additional details are in Appendix A.1.

2.4. Image Corruption for Pretext Tasks

We incorporated random block selection to select the square image patches to corrupt during pretraining. To ensure the amount of corruption per image was fixed and did not affect later comparison, the patches for each image were iteratively selected and corrupted until 1/4 of the total image area was corrupted.

For context prediction, we selected and set random patches of dimensions $K \times K$ to zero in an iterative manner until the number of pixels set to zero equaled or exceeded 1/4 of the total image area. For context restoration, randomly selected pairs of non-overlapping $K \times K$ image patches were swapped in an iterative manner until the number of corrupted pixels equaled or exceeded 1/4 of the total image area. We refer to the result of both methods as "masks". The context prediction binary mask specified which pixels were zero and the context restoration mask was a list of patch pairs to be swapped. When pretraining with multi-channel CT images, the locations of the patch corruptions were identical across channels to avoid shortcut learning [26]. Example image corruptions are shown in Figure 1.

To train the model to inpaint any arbitrarily corrupted image region without memorization of image content, we sampled a random mask every iteration for all images. For computational efficiency, we precomputed 100 random masks before training. We further randomly rotated the masks by either 0, 90, 180, or 270° counter-clockwise to increase the effective number of masks used during training to 400.

2.5. Design Choices for SSL Implementation

Design choices for inpainting-based SSL segmentation revolving around pretraining task implementations [7,8] and transfer learning [27–29] have not been systematically compared. To overcome these shortcomings, we explored the following questions:

1. Which pretrained weights should be transferred for fine-tuning?
2. How should the transferred pretrained weights be fine-tuned?
3. What should be the initial learning rate when fine-tuning?
4. What patch size should be used when corrupting images for inpainting?
5. How should the locations of the patches be sampled when corrupting images for inpainting?

2.5.1. Design Choices for Transfer Learning (#1–3)

For design choice #1 (which pretrained weights to transfer), we compared transferring only the U-Net encoder weights [7] with transferring both the encoder and decoder weights [8].

For design choice #2, we compare (i) fine-tuning all pretrained weights immediately after transferring [27,28], and (ii) freezing pretrained weights after transferring and training until convergence, then subsequently unfreezing pretrained weights and training all weights until convergence [29,30].

For design choice #3, we selected four initial learning rates: 1×10^{-2}, 1×10^{-3}, 1×10^{-4}, and 1×10^{-5}, to evaluate whether pretrained features are distorted with larger learning rates.

To compare different combinations of these three design choices, we performed a grid search and defined the best combination to be the one with the best segmentation performance on the MRI test set when trained with the MRI training subset with 5% training data. More details are in Appendix B.1.

2.5.2. Design Choices for Pretraining (#4–5)

For design choice #4, we compare patch sizes of 64 × 64, 32 × 32, 16 × 16, and 8 × 8 (Figure 1). For design choice #5, we compare two sampling methods: (i) fully-random sampling where the location of each patch was selected at random and constrained to lie completely within the image [7,8], and (ii) Poisson-disc sampling that enforces the centers of all $K \times K$ patches to lie at least $K\sqrt{2}$ pixels away from each other to prevent overlapping patches [31]. To compare different combinations of design choices #4 and #5 and the two pretext tasks, we performed a grid search by training a model for each combination five times, each time using one of the five training data subsets, for both datasets. We also trained a fully-supervised model for each dataset and training data subset for a baseline comparison. All models were fine-tuned in an identical manner with the same random seed after pretraining, using the best combination of design choices #1–3. All inpainting models were compared by computing the L2 norm of the generated inpainted images. When computing the L2 norm value for each three-channel CT image, the L2 norm value was computed per channel and averaged across all channels. All segmentation models were compared by computing the Dice coefficient for each segmentation class in the test set, averaged across all available volumes/slices.

2.5.3. Optimal Pretraining Evaluation

We defined the optimal pretraining strategy as the strategy that provided the most benefit over supervised learning, across image modalities and training data extents, in the experiment described in Section 2.5.2.

For each baseline (fully-supervised model) and SSL model trained in the experiment using 50%, 25%, 10%, and 5% training data, we computed class-averaged Dice scores for every test volume/slice in the MRI and CT datasets. For each pretraining strategy and dataset, we compared whether the set of Dice scores of the corresponding SSL models were significantly higher than that of the respective fully-supervised models using one-sided

Wilcoxon signed-rank tests. As a heuristic, the pretraining strategies were sorted by their associated *p*-values and the pretraining strategy that appeared in the top three for both the MRI and CT datasets was selected as the optimal pretraining strategy. We defined the optimally trained model for each dataset as the SSL model that was pretrained with this optimal pretraining strategy and fine-tuned for segmentation using the best combination of design choices #1-3.

2.6. Impact of Extent of Unlabeled Data

To measure the effect of the number of pretraining images on downstream segmentation performance, the optimally trained model was pretrained with the standard training set as well as two supersets of the training set containing additional unlabeled imaging data. We refer to the standard training set as 100% pretraining data (86 volumes for MRI and 709 slices for CT). For the MRI dataset, the second and third sets consisted of 150% (129 volumes) and 200% (172 volumes) pretraining data, respectively. For the CT dataset, the second and third sets consisted of 650% (4608 slices) and 1200% (8508 slices) pretraining data, respectively. After pretraining, all the pretrained models were fine-tuned with the five subsets of labeled training data and a Dice score was computed for each fine-tuned model, averaged across all segmentation classes and all volumes/slices in the test set. To quantify the relationship between Dice score and the amount of pretraining data for each subset of labeled training data, a curve of best fit was found using non-linear least squares. The Residual Standard Error, defined as $\frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{n-2}$, was computed to quantify how well the curve of best fit fits the data.

For MRI and CT, the pretraining dataset that led to the best average Dice score across the extents of labeled training data was chosen for further experiments.

2.7. Comparing SSL and Fully-Supervised Learning

We compared baseline fully-supervised models and the optimally trained models pretrained with the chosen pretraining dataset from the experiment described in Section 2.6. For each training data subset, models were evaluated using two clinically-relevant metrics for determining cartilage, muscle, and adipose tissue health status. For MRI, we computed mean T2 relaxation time per tissue and tissue volume [32]. For CT, we computed cross-sectional area and mean HU value per tissue. We calculated their percentage errors by comparing them to values derived from using ground truth segmentations to compute the metrics.

To determine which images benefit maximally with SSL, we compared and visualized the percentage error in the clinically-relevant metrics between supervised learning and SSL. For both supervised learning and SSL, the percentage error for each test image was averaged over all classes and label-limited scenarios.

2.8. Statistical Analysis

All statistical comparisons were computed using one-sided Wilcoxon signed-rank tests. All statistical analyses were performed using the SciPy (v1.5.2) library [33], with Type-1 $\alpha = 0.05$.

3. Results

The subject demographics of all labeled and unlabeled volumes/slices are shown in Table 1.

Table 1. Demographics of the subjects included in this study. Age is shown as mean ± standard deviation. For the CT dataset, one subject did not have age information and four subjects did not have gender information.

MRI			
Split	**Gender**	**# Volumes (# Slices)**	**Age (Range)**
Train	Male	46 (7360)	44.7 ± 17.7 (17–75)
	Female	40 (6400)	42.9 ± 18.5 (16–87)
	Total	86 (13760)	43.9 ± 18.1 (16–87)
Validation	Male	18 (2880)	37.3 ± 16.8 (18–68)
	Female	15 (2400)	53.2 ± 14.9 (18–79)
	Total	33 (5280)	44.5 ± 17.8 (18–79)
Test	Male	26 (4156)	37.9 ± 14.9 (18–71)
	Female	10 (1584)	53.0 ± 11.9 (31–73)
	Total	36 (5740)	42.1 ± 15.6 (18–73)
Unlabeled	Male	37 (5446)	38.1 ± 16.9 (15–77)
	Female	49 (6686)	52.1 ± 18.5 (14–97)
	Total	86 (12132)	46.1 ± 19.1 (14–97)
CT			
Split	**Gender**	**# Slices**	**Age (Range)**
Train	Male	362	68.2 ± 11.4 (20–97)
	Female	343	71.1 ± 10.5 (18–95)
	Total	709	69.6 ± 11.1 (18–97)
Validation	Male	63	69.1 ± 9.5 (32–83)
	Female	69	71.0 ± 11.0 (32–89)
	Total	133	70.1 ± 10.4 (32–89)
Test	Male	18	70.6 ± 11.9 (47–92)
	Female	26	73.1 ± 11.7 (44–93)
	Total	44	72.1 ± 11.9 (44–93)
Unlabeled	Male	3167	51.5 ± 17.1 (18–101)
	Female	4632	51.6 ± 17.1 (18–100)
	Total	7799	51.6 ± 17.1 (18–101)

3.1. Design Choices for Transfer Learning

We observed that all pretrained model variants had high performance when first fine-tuned with an initial learning rate of 1×10^{-3} and then fine-tuned a second time with an initial learning rate of 1×10^{-4}. Transferring pretrained encoder weights only and fine-tuning once immediately with an initial learning rate of 1×10^{-3} achieved similar performance, with the added benefit of reduced training time. Consequently, we used these as the best combination of the three design choices for transfer learning. Additional details are in Appendix B.2.

3.2. Design Choices for Pretraining

The L2 norm consistently decreased as a function of patch size for all combinations of pretext tasks (context prediction and context restoration) and sampling methods (random and Poisson-disc) (Table 2). Furthermore, L2 norms for Poisson-disc sampling were significantly lower than those for random sampling ($p < 0.05$).

Dice scores for fully-supervised baselines ranged from 0.67–0.88 across subsets of training data for MR images. Downstream segmentation performance for the MRI dataset was similar for all combinations of pretext task, patch size, and sampling method (Figure 3). All SSL models matched (within 0.01) or outperformed the fully-supervised model in low-label regimes with 25% training data or less for the femoral cartilage, patellar cartilage, and meniscus, and had comparable performance for higher data extents. For the tibial cartilage, all SSL models outperformed the fully-supervised model when trained on 5% training data and had comparable performance for higher data extents. The difference in Dice score between each self-supervised model and the fully-supervised model generally increased as the amount of labeled training data decreased. SSL pretraining also enabled some models to outperform the fully-supervised model trained with 100% training data in patellar cartilage segmentation.

Dice scores for fully-supervised baselines were consistently higher for CT images than for MR images, with the exception of intramuscular adipose tissue. Unlike with the MRI dataset, downstream SSL segmentation for CT in low-label regimes depended on the pretext task and the patch size used during pretraining (Figure 4). Models pretrained with larger patch sizes (64 × 64; 32 × 32) often outperformed those pretrained with smaller patch sizes (16 × 16; 8 × 8) for muscle, visceral fat, and subcutaneous fat segmentation, when trained with either 5% or 10% labeled data. Furthermore, when 25% training data or less was used, models pretrained with 32 × 32 patches using context restoration almost always outperformed fully-supervised models for muscle, visceral fat, and subcutaneous fat segmentation, but rarely did so when pretrained using context prediction. For intramuscular fat, all SSL models had comparable performance with fully-supervised models in low-label regimes. For high-label regimes (over 25% labeled data), all SSL models had comparable performance with fully-supervised models for all four segmentation classes.

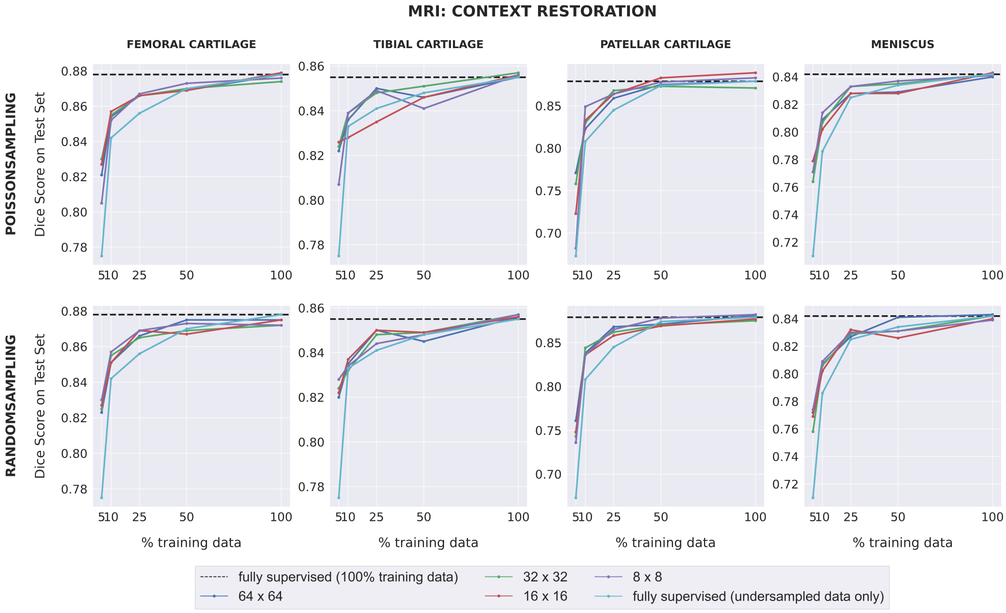


Figure 3. The downstream segmentation performance on the MRI dataset for the Context Restoration pretext task as measured by the Dice score for every combination of patch size and sampling method used during pretraining, evaluated in five different scenarios of training data availability. In each scenario, every model is trained for segmentation using one of the five different subsets of training data as described in Section 2.1.1. The black dotted line in each plot indicates the performance of a fully-supervised model trained using all available training images. The light blue curve indicates the performance of a fully-supervised model when trained using each of the five different subsets of training data. Similar plots for the Context Prediction pretext task are given in Appendix C.

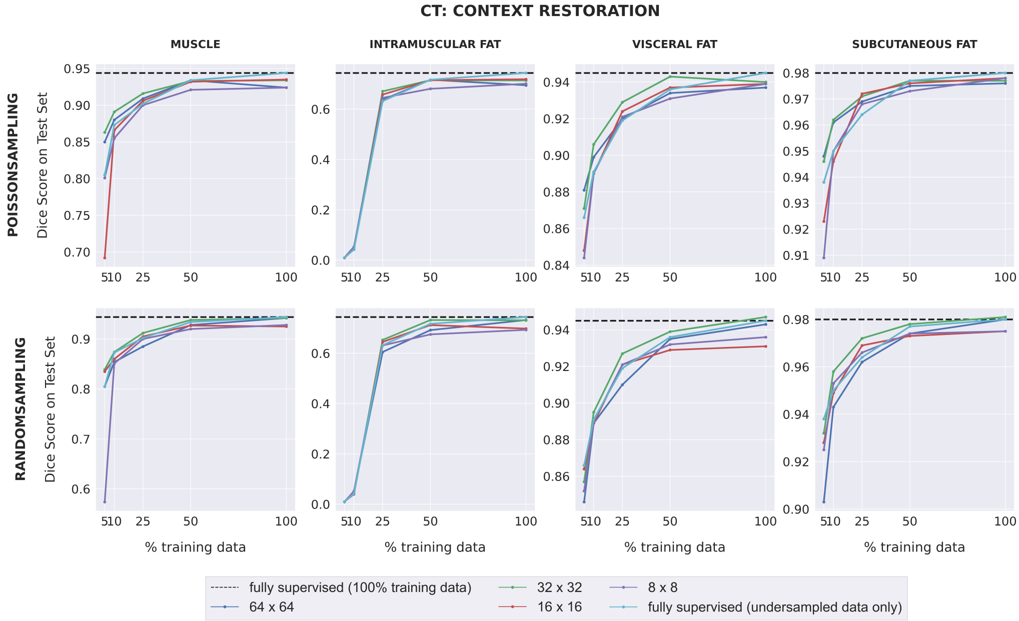


Figure 4. The downstream segmentation performance on the CT dataset for the Context Restoration pretext task as measured by the Dice score for every combination of patch size and sampling method used during pretraining, evaluated in five different scenarios of training data availability. In each scenario, every model is trained for segmentation using one of the five different subsets of training data as described in Section 2.1.2. The black dotted line in each plot indicates the performance of a fully-supervised model trained using all available training images. The light blue curve indicates the performance of a fully-supervised model when trained using each of the five different subsets of training data. Similar plots for the Context Prediction pretext task are given in Appendix C.

Table 2. Quantitative evaluation of inpainting for every combination of pretext task, patch size, and sampling method. All values are rounded to the nearest integer.

Pretext Task and Patch Size		**MRI**		**CT**	
		L2 Norm (Mean ± Std)		**L2 Norm (Mean ± Std)**	
Pretext Task	**Patch Size**	**Poisson-Disc**	**Random**	**Poisson-Disc**	**Random**
Context Prediction	64 × 64	94 ± 9	105 ± 9	123 ± 13	134 ± 18
	32 × 32	75 ± 8	81 ± 8	83 ± 9	112 ± 14
	16 × 16	61 ± 7	64 ± 7	66 ± 8	74 ± 10
	8 × 8	51 ± 6	52 ± 5	54 ± 7	57 ± 8
Context Restoration	64 × 64	96 ± 9	116 ± 11	142 ± 19	346 ± 158
	32 × 32	75 ± 8	84 ± 8	108 ± 12	127 ± 15
	16 × 16	62 ± 7	67 ± 7	86 ± 10	93 ± 12
	8 × 8	51 ± 5	56 ± 6	66 ± 8	80 ± 9

3.3. Optimal Pretraining Evaluation

The top 5 pretraining strategies for the MRI dataset and the top 3 pretraining strategies for the CT dataset led to significantly better segmentation performance compared to fully-supervised learning ($p < 0.001$) (Table 3).

For MRI, the top 5 strategies all consisted of pretraining with context restoration, with minimal differences in p-value based on the patch size and sampling method used. For CT, the top 5 strategies used a patch size of at least 32 × 32 during pretraining. The strategy of pretraining with context restoration, 32 × 32 patches, and Poisson-disc sampling was in the top 3 for both datasets, and was therefore selected as the optimal pretraining strategy.

Table 3. Summary of the top five combinations of pretext tasks, patch sizes, and sampling methods for each dataset with the corresponding *p*-value for each combination, and sorted by *p*-value in ascending order. The bolded pretext task, patch size, and sampling method were chosen as the best combination of the three design choices.

		MRI		
Rank	**Pretext Task**	**Patch Size**	**Sampling Method**	***p*-Value**
1	Context Restoration	64 × 64	Random	1.64×10^{-18}
2	Context Restoration	8 × 8	Random	1.89×10^{-18}
3	**Context Restoration**	**32 × 32**	**Poisson-Disc**	1.05×10^{-17}
4	Context Restoration	8 × 8	Poisson-Disc	4.03×10^{-17}
5	Context Restoration	32 × 32	Random	9.38×10^{-16}
		CT		
Rank	**Pretext Task**	**Patch Size**	**Sampling Method**	***p*-Value**
1	**Context Restoration**	**32 × 32**	**Poisson-Disc**	6.29×10^{-17}
2	Context Restoration	64 × 64	Poisson-Disc	1.88×10^{-9}
3	Context Restoration	32 × 32	Random	4.12×10^{-7}
4	Context Prediction	64 × 64	Poisson-Disc	0.1
5	Context Prediction	64 × 64	Random	0.66

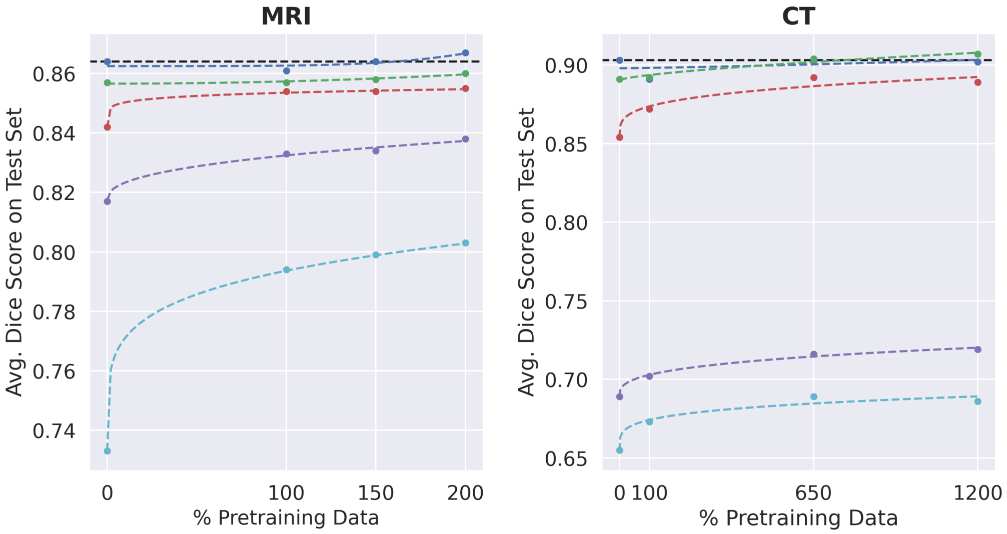
3.4. Impact of Extent of Unlabeled Data

For both datasets and for most subsets of labeled training data used during fine-tuning (except 25% and 10% labeled training data for MRI), the optimally trained model performed significantly better in downstream segmentation when pretrained on the maximum amount of data per dataset (200% pretraining data for MRI and 1200% pretraining data for CT) than when pretrained on only the training set ($p < 0.05$) as seen in Figure 5. When 25% or 10% labeled training data was used for MRI segmentation, the optimally trained model achieved a higher mean Dice score when pretrained on 200% pretraining data, but this was not statistically significant ($p = 0.3$ for 25% labeled training data and $p = 0.02$ for 10% labeled training data).

For MRI, Dice scores almost always improved as the amount of pretraining data increased. This improvement was greatest when only 5% of the labeled training data was used for training segmentation. Improvements in segmentation performance were slightly higher for CT. For all extents of labeled training data, segmentation performances improved when the amount of pretraining data increased from 100% to 650%. There was limited improvement when the amount of pretraining data increased from 650% to 1200%. For both datasets, when 25%, 10%, or 5% of the labeled training data was used, the change in dice score as a function of the amount of pretraining data followed a power-law relationship of the form $y = ax^k + c$ (residual standard errors ≤ 0.005), where the value of k was less than 0.5.

Pretraining on the maximum amount of data enabled the optimally trained models to surpass the performance of fully-supervised models for all extents of labeled training data, in both MRI and CT. For the MRI dataset, the highest improvement over supervised learning was observed when 5% labeled training data was used. For CT, considerable improvements over supervised learning were observed when 5%, 10%, or 25% labeled training data was used.

For both the MRI and CT datasets, the best average Dice score over all extents of labeled training data occurred when the maximum possible amount of pretraining data was used (200% pretraining data for MRI and 1200% pretraining data for CT).



MRI Power Law Parameters	100%	50%	25%	10%	5%
a	5.75 x 10^{-14}	6.03 x 10^{-8}	0.006	0.002	0.024
k	4.726	2.049	0.139	0.403	0.204
c	0.862	0.857	0.842	0.817	0.733
S	0.002	0.0	0.0	0.001	0.0

CT Power Law Parameters	100%	50%	25%	10%	5%
a	7.75 x 10^{-7}	1.48 x 10^{-4}	0.006	0.003	0.007
k	1.247	0.673	0.266	0.322	0.228
c	0.898	0.89	0.854	0.689	0.655
S	0.007	0.002	0.005	0.001	0.004

Figure 5. The downstream segmentation performance of the optimally trained model when pretrained with different amounts of pretraining data and fine-tuned using each of the five training data subsets. 100% pretraining data refers to the regular training set for each dataset. The data point for 0% pretraining data is the performance of a fully-supervised model. The black dotted line indicates the performance of a fully-supervised model trained on all available training data for the appropriate dataset. The other dotted lines are the best-fit curves for each of the training data subsets, modeled as a power-law relationship of the form $y = ax^k + c$. The values of a, k, c, and the Residual Standard Error (S) for the best-fit curves are displayed in the two tables.

3.5. Comparing SSL and Fully-Supervised Learning

For each dataset, optimally trained models were pretrained with the maximum amount of pretraining data from Section 3.4.

For all clinical metrics, using optimally trained models generally led to lower percent errors than using fully-supervised models in regimes of 10% and 5% labeled training data (Figure 6). These differences were especially pronounced for CT tissue cross-sectional area, MRI tissue volume, and MRI mean T2 relaxation time. With 5% labeled training data for MRI, segmentations from optimally trained models more than halved the percent error for both tissue volume and mean T2 relaxation time of patellar cartilage, compared to segmentations from fully-supervised models.

With 100% or 50% labeled training data, percent errors for all clinical metrics had lower improvement when optimally trained models were used. This was observed for CT tissue cross-sectional area, CT mean HU value, and MRI T2 relaxation time, where optimally trained models had similar or slightly worse performance than fully-supervised models when 100% or 50% labeled data was available. However, for MRI tissue volume, optimally trained models almost always outperformed the fully-supervised models, even in scenarios with large amounts of labeled training data.

For both datasets, clinical metrics improved the most for the most challenging classes to segment. This included intramuscular adipose tissue for CT, where percent error decreased from around 3940% to 3600% for tissue cross-sectional area when 10% labeled training data was used, and patellar cartilage for MRI, where percent error decreased from around 30% to 12% for tissue volume when 5% labeled training data was used.

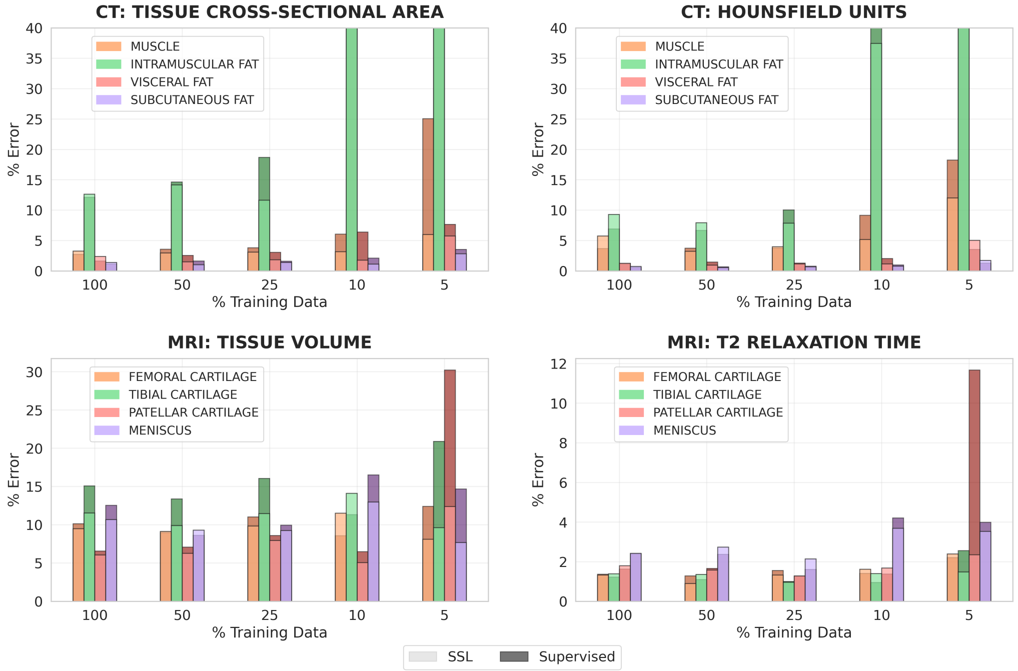


Figure 6. A comparison of the percent error in calculating clinical metrics for the MRI and CT datasets between when the tissue segmentations are generated by fully-supervised models and when the tissue segmentations are generated by optimally trained models, pretrained using 200% data for MRI and 1200% data for CT. Each bar represents the median percent error across the test set for a particular tissue, clinical metric, and label regime. The percent error in the calculation of tissue cross-sectional area and mean HU for intramuscular fat extends beyond the limits of the y-axis when 10% and 5% labeled training data for segmentation is used.

On a per-image basis, using SSL consistently matched or reduced the percent errors of supervised learning across both datasets and all clinical metrics (Figure 7). Furthermore, when using SSL, the percent error for all clinical metrics improved more for test images with larger percent errors when using supervised learning. For tissue cross-sectional area and mean HU value for CT, the improvement in SSL percent error gradually increased as the supervised percent error increased beyond 10%. The same pattern existed for MRI tissue volume as the supervised percent error increased beyond 20%. For MRI mean T2 relaxation time, the improvement in percent error when using SSL increased for most test images as the supervised percent error increased beyond 5%, but this was not as consistent as for the other clinical metrics. On average, when excluding intramuscular fat for CT, SSL decreased per-image percent errors for CT tissue cross-sectional area, CT mean HU value, MRI tissue volume, and MRI mean T2 relaxation time by 4.1, 1.9, 4.1, and 2.2%, respectively.

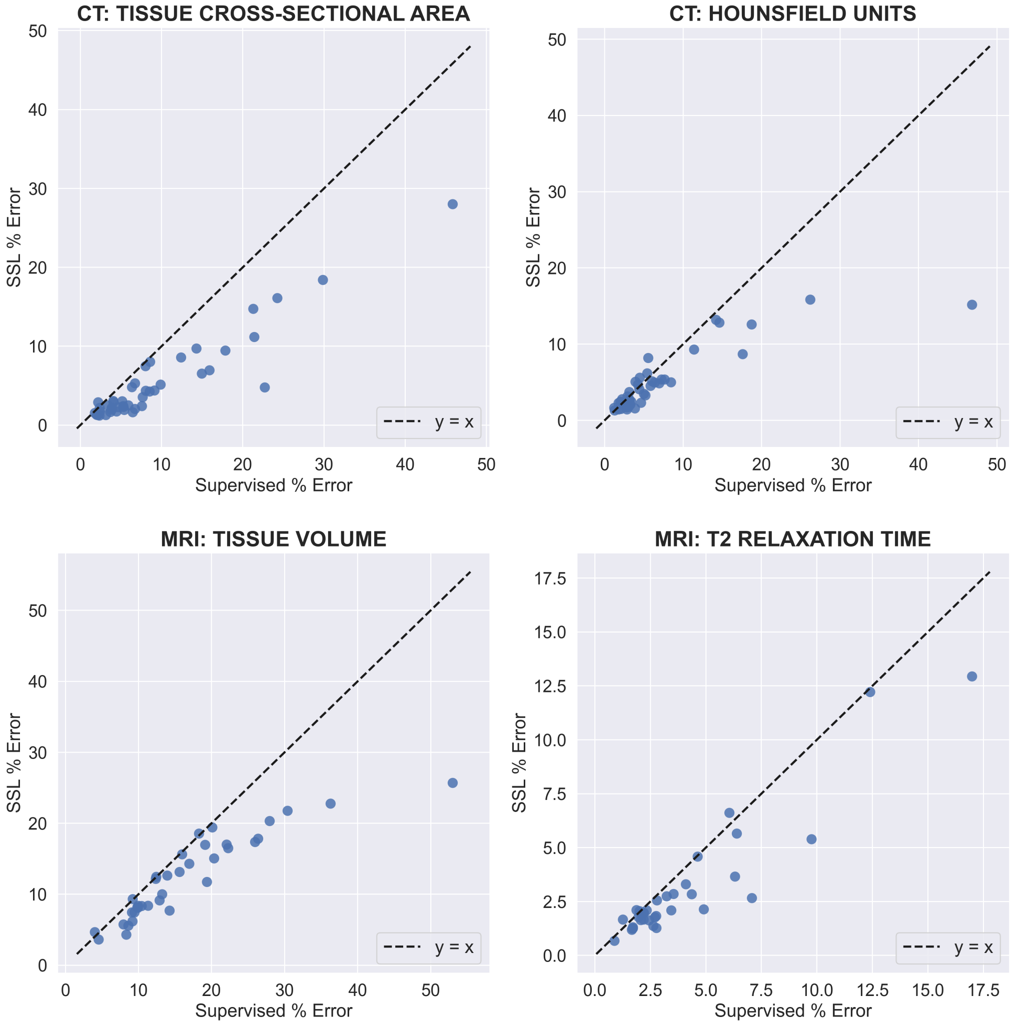


Figure 7. The relationship between the percent error when using supervised learning and the percent error when using SSL. Each blue point represents an image in the test set for the appropriate dataset. The percent error was averaged over all classes and label-limited scenarios. For CT, the intramuscular fat was excluded to prevent large percent error values. For MRI T2 relaxation time, one point with a high percent error for supervised learning was excluded to reduce the range of the x-axis.

4. Discussion

In this work, we investigated several key, yet under-explored design choices associated with pretraining and transfer learning in inpainting-based SSL for tissue segmentation. We examined the effect of inpainting-based SSL on the performance of tissue segmentation in various data and label regimes for MRI and CT scans, and compared it with fully-supervised learning. We quantified performance using standard Dice scores and four clinically-relevant metrics of imaging biomarkers.

We observed that the crosstalk between the initial and fine-tuning learning rate was a design choice that most affected model performance. All model variants achieved optimal performance with an initial learning rate of 1×10^{-3} and a fine-tuning learning rate of 1×10^{-4} (Figure A1). This suggests the need for not perturbing the pretrained representations from the pretext task with a large learning rate. Moreover, although freezing and then fine-tuning the transferred weights provided an improvement over fine-tuning immediately for this learning rate combination (Figure A1), the improvement was very small. This result matches the findings of Kumar et al. [30], where the performance of linear probing (freezing) and then fine-tuning only slightly improved the performance of fine-tuning immediately after transferring. Additional details are provided in Appendix B.3.

Here, we suggest some best practices for inpainting-based SSL for medical imaging segmentation tasks. We observed that downstream segmentation performance for MRI was similar for all combinations of pretext tasks, patch sizes, and sampling techniques. This observation remained consistent despite significant differences in the L2 norms of the inpainted images. While decreasing patch sizes and sampling patch locations via Poisson-disc sampling to ensure non-overlapping patches both resulted in significantly lower L2 norms, they did not improve downstream segmentation performance. These observations suggest a discordance between learning semantically meaningful representations and the accuracy of the pretext task metric. Thus, simply performing good enough pretraining may be more important than optimizing pretext task performance.

For both MRI and CT, segmentation performance usually increased in proportion to the amount of pretraining data. The highest improvements over supervised learning were observed in the context of very low labeled data regimes of 5–25% labeled data. These empirical observations across both MRI and CT demonstrate that pretraining with large enough datasets improves performance compared to only supervised training, especially when the amount of available training data is limited.

Similar to supervised learning, improvements in SSL Dice scores tended to follow a power-law relationship of the form $y = ax^k + c$ as the size of the unlabeled corpora increased [5]. The observations that the value of k was less than 0.5 when 25%, 10%, or 5% labeled data was used for either dataset and pretraining on 650% and 1200% CT pretraining data led to similar improvements over supervised learning suggest a limit exists where the learning capacity of a model saturates and additional unlabeled data may not improve downstream performance. A good practice for future segmentation studies may be to create Figure 5 to evaluate the trade-off between the challenges of annotating more images and acquiring more unlabeled images.

Compared to fully-supervised models, optimally trained models generally led to more accurate values for all clinical metrics in label-limited scenarios. We also observed that clinical metrics improved the most with SSL for tissue classes that had the highest percent error with fully-supervised learning—intramuscular adipose tissue in CT and patellar cartilage in MRI. This observation, combined with the Dice score improvement in low labeled data regimes, suggests that SSL may be most efficacious when the performance of the baseline fully-supervised model is low.

A similar pattern was observed on a per test image basis. For all clinical metrics, the improvement in percent error when using optimally trained models was greater for test images that performed poorly when using fully-supervised models. This suggests that SSL pretraining can reduce worst-case errors that occur with traditional supervised learning. Moreover, our observation that SSL percent errors consistently either matched or were lower than supervised percent errors indicates SSL pretraining also increases the robustness of models in label-limited scenarios.

However, we also observed that optimally trained models sometimes had similar or even worse performance than fully-supervised models for CT tissue cross-sectional area, CT mean HU value, and MRI T2 relaxation time in scenarios with 100% or 50% labeled data. This observation suggests that SSL does not have much benefit when the labeled dataset is large. In such cases, it may be more efficient to simply train a fully-supervised model, rather than spend additional time pretraining with unlabeled data.

When training with 5% labeled data for all MRI classes and muscle on CT, our optimal pretraining strategy improved Dice scores by over 0.05, compared to fully-supervised learning. In such cases, the Dice score for fully-supervised learning was 0.8 or lower, which suggests a critical performance threshold where inpainting-based SSL can improve segmentation performance over supervised learning. SSL may be beneficial in these cases because the models still have the capacity to learn more meaningful representations, compared to models with Dice scores over 0.8 that may already be saturated in their capacity to represent the underlying image.

Importantly, it should be noted that the improvement in segmentation performance with SSL pretraining in label-limited scenarios is on the similar order as prior advances that used complex DL architectures and training strategies [34–36]. Comparatively, our proposed SSL training paradigm offers an easy-to-use framework for improving model performance for both MRI and CT without requiring large and difficult to train DL models. Moreover, since we have already investigated different implementation design choices and experimentally determined the best ones, our proposed training paradigm will provide researchers with an implementation of inpainting-based SSL for their own work, without requiring them to spend resources/compute investigating these design choices again. This is especially important as we have shown that simply performing inpainting-based pretraining on the same data that is ordinarily only used for supervised learning improves segmentation accuracy compared to supervised learning only.

Study Limitations

There were a few limitations with this study. Although we investigated two different methods for selecting which pretrained weights to transfer, we did not conduct a systematic study across all possible choices due to computational constraints that made searching over the large search space too inefficient. We also leave other SSL strategies such as contrastive learning to future studies since it requires systematic evaluation of augmentations and sampling strategies. Furthermore, when we investigated the impact of unlabeled data extents on downstream segmentation performance, we did not pretrain our SSL models with equal extents of unlabeled MRI and CT data since we maximized the amount of available MRI data. In addition, our investigations in this work are limited to the U-Net architecture, though future work can explore other powerful segmentation architectures. Finally, we did not experiment with other optimizers potentially better than the ADAM optimizer. Recent studies [37] have shown that there may be value in optimizers such as Stochastic Gradient Descent for better generalization in natural image classification and that there is potential trade off while choosing different optimizers. We leave the systematic investigation of this issue on medical imaging data for future follow up work.

5. Conclusions

In this work, we investigated how inpainting-based SSL improves MRI and CT segmentation compared to fully-supervised learning, especially in label-limited regimes. We presented an optimized training strategy and open-source implementation for performing such pretraining. We describe the impact of pretraining task optimization and the relationship between the sizes of labeled and unlabeled training datasets. Our proposed approach for pretraining for improving segmentation performance that does not require additional manual annotation, complex model architectures, or model training techniques.

Author Contributions: Conceptualization, J.D. and A.S.C.; methodology, J.D., N.B., A.D.D., B.G. and A.S.C.; software, J.D., N.B. and A.D.D.; validation, J.D., N.B., A.S., E.R. and A.S.C.; formal analysis, J.D., N.B. and A.S.C.; investigation, J.D., N.B., A.D.D. and A.S.C.; resources, G.E.G., B.A.H., L.L., R.B. and A.S.C.; data curation, A.S., E.R., G.E.G., B.A.H., L.L., R.B. and A.S.C.; writing—original draft preparation, J.D., N.B., A.D.D. and A.S.C.; writing—review and editing, ALL; visualization, J.D. and N.B.; supervision, L.L., R.B. and A.S.C.; project administration, A.S.C.; funding acquisition, G.E.G., B.A.H. and A.S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by NIH grants R01 AR077604, R01 EB002524, R01 AR079431, K24 AR062068, and P41 EB027060; NIH contracts 75N92020C00008 and 75N92020C00021; the Precision Health and Integrated Diagnostics Seed Grant from Stanford University; National Science Foundation (DGE 1656518, CCF1763315, CCF1563078); DOD—National Science and Engineering Graduate Fellowship (ARO); GE Healthcare and Philips.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Institutional Review Board (or Ethics Committee) of Stanford University (protocol codes: 26840 and date of approval 14 October 2020, 58903 and date of approval 10 November 2020).

Informed Consent Statement: Informed consent was obtained for all subjects involved in the MRI analysis due to its prospective data acquisition. Informed consent was waived for all subjects in the CT analysis since it included retrospective analysis of already-acquired data.

Data Availability Statement: All MRI data can be accessed via the publicly-shared SKM-TEA data repository (https://github.com/StanfordMIMI/skm-tea), CT data can be made available via request to the authors in a manner compliant with IRB approvals and human subjects research.

Conflicts of Interest: The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Appendix A. Implementation Details

Appendix A.1. Model Architecture and Optimization

The loss function for inpainting was the L2 loss, implemented as in Equation (A1) for a model output (X) and ground truth (Y), where N, H, W, and C denote the batch size, height, width, and number of channels of the images, respectively.

$$L_{inpainting}(X,Y) = \frac{1}{C}\sum_{c=0}^{C-1}\frac{1}{N}\sum_{n=0}^{N-1}\sum_{h=0}^{H-1}\sum_{w=0}^{W-1}(X_{n,h,w,c} - Y_{n,h,w,c})^2 \tag{A1}$$

The loss function for segmentation was a variant of the Dice loss, implemented as in Equation (A2) for a model output (X) and ground truth (Y), where N, H, W, and C denote the batch size, height, width, and number of channels of the images, respectively. Due to the instability of pixel-wise losses for sparse classes, we used a batch-aggregate Dice loss, where the Dice loss was computed over the aggregate of a mini-batch per segmentation class and the final loss was the mean Dice loss across segmentation classes.

$$L_{segmentation}(X,Y) = \frac{1}{C}\sum_{c=0}^{C-1} 1 - \frac{2 * \sum_{n=0}^{N-1}\sum_{h=0}^{H-1}\sum_{w=0}^{W-1}(X_{n,h,w,c} * Y_{n,h,w,c}) + \epsilon}{\sum_{n=0}^{N-1}\sum_{h=0}^{H-1}\sum_{w=0}^{W-1}(X_{n,h,w,c} + Y_{n,h,w,c}) + \epsilon} \tag{A2}$$

For inpainting, the learning rate was set to 1×10^{-3} and decayed by a factor of 0.9 every 2 epochs. To prevent overfitting, early stopping [38] based on the validation L2 loss was used with a threshold of 50 and patience of 4 epochs. For a baseline fully-supervised segmentation, the initial learning rate was also set to 1×10^{-3} and the learning rate followed the same schedule as inpainting. For self-supervised segmentation, the fine-tuning initial learning rate was considered a design choice and is described in Section 2.5.1, but the learning rate schedule was the same as for inpainting and fully-supervised segmentation. For both fully-supervised and self-supervised segmentation, early stopping based on the validation Dice loss was used to prevent overfitting, with a threshold of 1×10^{-3} and a patience of 10 epochs. All inpainting and segmentation models were trained until the criteria for early stopping was achieved. The same random seed was used for all experiments.

All models were trained using the Keras (v2.1.6) software library [39], with Tensorflow (v1.15.0) [40] as the backend.

Appendix A.2. Design Choices for SSL Implementation

For all experiments described in Section 2.5, the self-supervised models were pretrained on all of the training data for the appropriate dataset.

Appendix B. Design Choices for Transfer Learning

Appendix B.1. Grid Search Implementation

As described in Section 2.5.1, we compared two methods for selecting which pretrained weights to transfer and two methods for fine-tuning the transferred pretrained weights. To compare the four combinations of these two design choices, we trained one model per combination. Since the first fine-tuning method, in which the pretrained weights are fine-tuned immediately, involves one training run, and the second fine-tuning method, in which the pretrained weights are first frozen, involves two training runs, we chose to train the two models in which the pretrained weights were fine-tuned immediately two times to ensure a fair comparison.

To investigate the impact of the initial learning rate during fine-tuning, we trained each of the four models four times during the first training run, each time with one of the four possible initial learning rates, and then trained each of the sixteen trained models again four times, each time with one of the four possible initial learning rates.

We selected the learning rates 1×10^{-2}, 1×10^{-3}, 1×10^{-4}, and 1×10^{-5} for specific reasons. 1×10^{-2} was selected as an example of a large learning rate, to determine if fine-tuning with a large learning rate will destroy pretrained features. 1×10^{-3} was selected as an example of a “common” learning rate, and was used as the initial learning rate for all our other experiments. Finally, 1×10^{-4} and 1×10^{-5} were selected arbitrarily as examples of small learning rates.

All pretrained weights were derived from an inpainting model that was trained with context prediction with 16 × 16 patches and Poisson-disc sampling, and all models were fine-tuned for segmentation using the MRI training subset with 5% data. The same random seed was used when training each model. All models were compared by computing the Dice coefficient for each volume in the MRI test set, averaged across the four segmentation classes.

Appendix B.2. Results

For the first training run following pretraining, higher initial learning rates of 1×10^{-2} and 1×10^{-3} produced better results. The FEF, FBF, and FFEF models had similar performance for all initial learning rates, and consistently performed better than the FFBF models (Figure A1).

When each type of model was trained for a second time with an initial learning rate of 1×10^{-2}, each model’s performance was similar to its performance when trained only once with an initial learning of 1×10^{-2}. This occurred regardless of the initial learning rate during the first training run. For example, the FEF, FBF, and FFEF models had relatively high performance when trained once with an initial learning rate of 1×10^{-3}, but when these models were trained for a second time with an initial learning rate of 1×10^{-2}, the performance of all three models dropped to the same level of performance as when each type of model was trained only once with an initial learning rate of 1×10^{-2}. Similarly, but in the opposite way, the FEF, FBF, and FFEF models had relatively low performance when trained once with an initial learning rate of 1×10^{-4} or 1×10^{-5}, but when these models were trained for a second time with an initial learning rate of 1×10^{-2}, the performance of all three models increased to the same level of performance as when each type of model was trained only once with an initial learning rate of 1×10^{-2}.

When each type of model was trained once with an initial learning rate of 1×10^{-2} and then trained a second time with a smaller learning rate, the FFBS models outperformed the FES and FBS models when the initial learning rate of the second training run was 1×10^{-3} or 1×10^{-4}, and the FFES models outperformed the FES and FBS models for all initial learning rates smaller than 1×10^{-2}.

When each type of model was trained once with an initial learning rate of 1×10^{-4} and then trained a second time with an initial learning rate of 1×10^{-3} or lower, the FES and FFES models had similar performance and always outperformed the FBS and FFBS models.

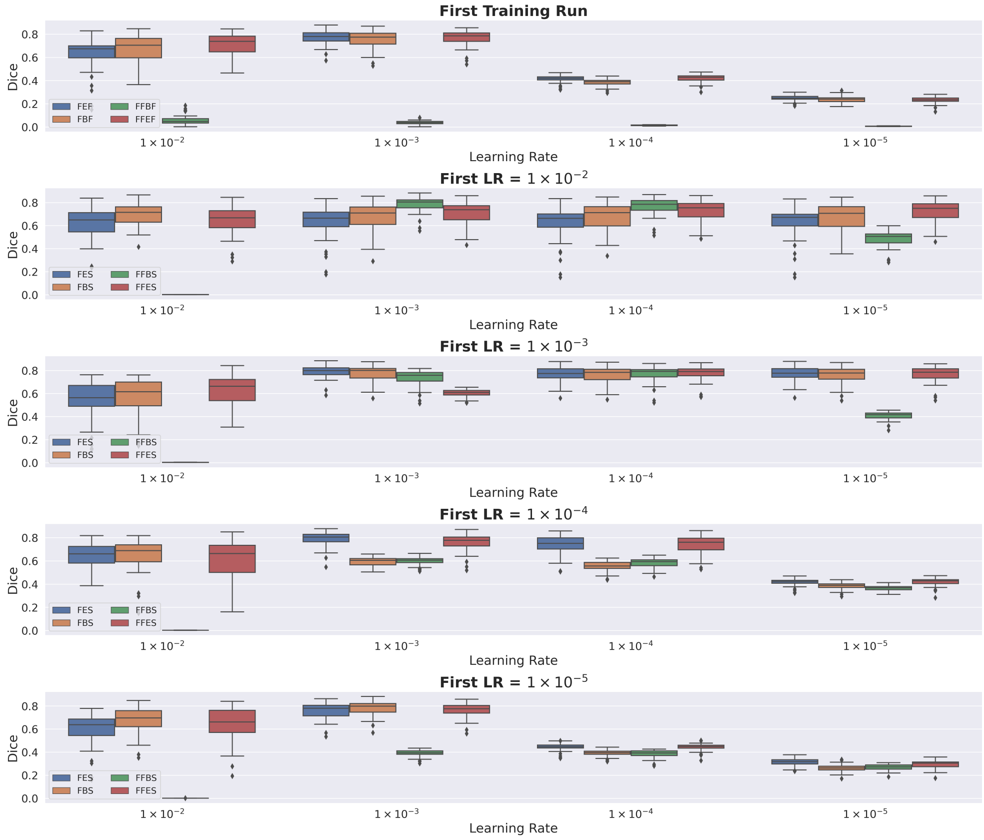


Figure A1. Box plots displaying the spread of Dice scores among the volumes in the MRI test set. The top row displays the spread of Dice scores after each type of model was trained once, with the initial learning rate set to the appropriate value on the x-axis. The remaining four rows display the spread of Dice scores after each model in the first row was trained again, with the initial learning rate set to the appropriate value on the x-axis. We used the following structure for acronyms to distinguish between the different types of models: **ABC**. If **A** is F, the pretrained weights were fine-tuned immediately, and if **A** is FF, the pretrained weights were first frozen and then fine-tuned. If **B** is E, only the pretrained encoder weights were transferred, and if **B** is B, both the pretrained encoder and decoder weights were transferred. If **C** is F, the model was trained only once (the first training run), and if **C** is S, the model was trained a second time (the second training run).

When each type of model was trained once with an initial learning rate of 1 × 10^{-3}, training each model again with an initial learning rate equal to or less than 1 × 10^{-3} did not improve or only slightly improved the model's performance. The exception was the FFBF model, for which the performance always increased by a large amount during the second training run, regardless of the initial learning rate used during the second training run. The FEF, FES, FBF, and FBS models had similar performance when the initial learning rates for the first and second training runs were set to 1 × 10^{-3}.

We concluded that FEF (fine-tuning immediately after transferring only the pretrained encoder weights), trained with an initial learning rate of 1 × 10^{-3}, was the best combination of design choices for transfer learning because this model achieved high segmentation performance with minimal training time.

Appendix B.3. Discussion

In this experiment, we determined the best combination of fine-tuning mechanism, weight loading strategy, and initial learning rate during fine-tuning. Overall, every model had high performance when first trained with an initial learning rate of 1 × 10^{-3} and then trained a second time with an initial learning rate of 1 × 10^{-4}, despite using different fine-tuning mechanisms and different methods for selecting which pretrained weights to transfer. This suggests choosing the initial learning rates for the first and second training runs is the design choice for transfer learning that most affects model performance.

If the initial learning rate of the first training run is too large, like 1×10^{-2}, the pretrained features are at risk of being destroyed. For example, the FEF and FBF models performed worse when trained with an initial learning rate of 1×10^{-2} than when trained with an initial learning rate of 1×10^{-3}. Furthermore, when the initial learning rate during the second training run is too large, a model has a risk of escaping out of an already found local minimum. For example, although the FBF model had high performance when trained once with an initial learning rate of 1×10^{-3}, its performance dropped when trained again with an initial learning rate of 1×10^{-2}.

On the other hand, if the initial learning rate is too small, a model may not be able to learn during fine-tuning. This was suggested by the low performance of all four types of models when they were trained once with an initial learning rate of 1×10^{-5} and then trained again with an initial learning rate of either 1×10^{-4} or 1×10^{-5}.

Although the design choice for transfer learning that most affects model performance is the initial learning rate during fine-tuning, the results of this experiment suggest that transferring only the pretrained encoder weights may lead to better performance gains than transferring both the pretrained encoder and decoder weights. For instance, in the first training run, the FFEF models performed similarly to the FEF models for all initial learning rates, suggesting the frozen encoder features in the FFEF models were as good as the fine-tuned encoder features in the FEF models. In addition, when the four types of models were first trained with an initial learning rate of 1×10^{-4} and then trained again with an initial learning rate of 1×10^{-3} or lower, the FES and FFES models always outperformed the FBS and FFBS models. These results suggest that transferring only the pretrained encoder weights provides a better initialization point for segmentation fine-tuning than transferring both the pretrained encoder and decoder weights.

Appendix C. Design Choices for Pretraining

Below, we provide additional figures that illustrate the effect of design choices on the context prediction task for MRI (Figure A2) and CT (Figure A3).

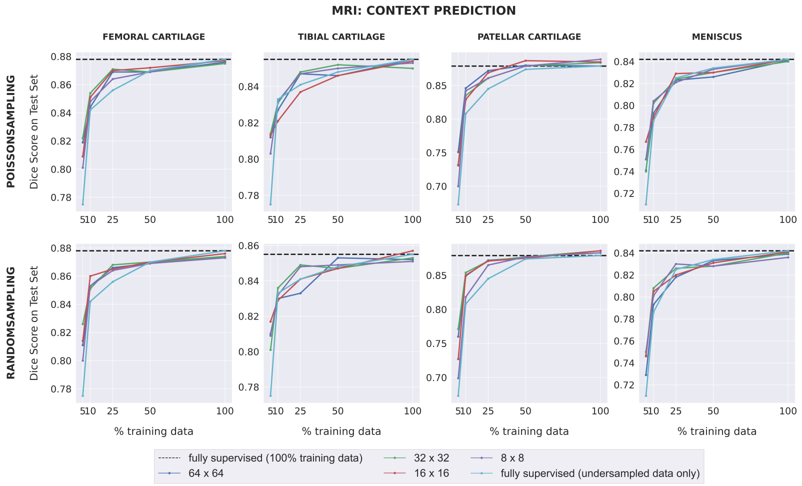


Figure A2. The downstream segmentation performance on the MRI dataset for the Context Prediction pretext task as measured by the Dice score for every combination of patch size and sampling method used during pretraining, evaluated in five different scenarios of training data availability. In each scenario, every model is trained for segmentation using one of the five different subsets of training data as described in Section 2.1.1. The black dotted line in each plot indicates the performance of a fully-supervised model trained using all available training images. The light blue curve indicates the performance of a fully-supervised model when trained using each of the five different subsets of training data.

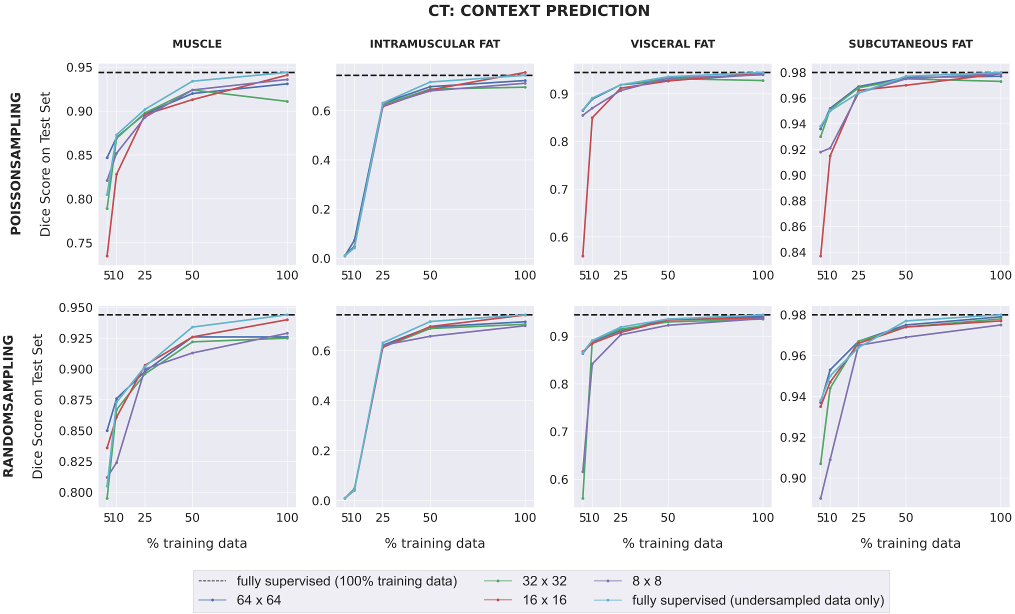


Figure A3. The downstream segmentation performance on the CT dataset for the Context Prediction pretext task as measured by the Dice score for every combination of patch size and sampling method used during pretraining, evaluated in five different scenarios of training data availability. In each scenario, every model is trained for segmentation using one of the five different subsets of training data as described in Section 2.1.2. The black dotted line in each plot indicates the performance of a fully-supervised model trained using all available training images. The light blue curve indicates the performance of a fully-supervised model when trained using each of the five different subsets of training data.

References

1. Campello, V.M.; Gkontra, P.; Izquierdo, C.; Martín-Isla, C.; Sojoudi, A.; Full, P.M.; Maier-Hein, K.; Zhang, Y.; He, Z.; Ma, J.; et al. Multi-Centre, Multi-Vendor and Multi-Disease Cardiac Segmentation: The M&Ms Challenge. *IEEE Trans. Med. Imaging* **2021**, *40*, 3543–3554 . [PubMed]
2. Kavur, A.E.; Gezer, N.S.; Barış, M.; Aslan, S.; Conze, P.H.; Groza, V.; Pham, D.D.; Chatterjee, S.; Ernst, P.; Özkan, S.; et al. CHAOS challenge-combined (CT-MR) healthy abdominal organ segmentation. *Med. Image Anal.* **2021**, *69*, 101950. [CrossRef] [PubMed]
3. Desai, A.D.; Caliva, F.; Iriondo, C.; Mortazi, A.; Jambawalikar, S.; Bagci, U.; Perslev, M.; Igel, C.; Dam, E.B.; Gaj, S.; et al. The international workshop on osteoarthritis imaging knee MRI segmentation challenge: A multi-institute evaluation and analysis framework on a standardized dataset. *Radiol. Artif. Intell.* **2021**, *3*, e200078. [CrossRef] [PubMed]
4. Fan, D.P.; Zhou, T.; Ji, G.P.; Zhou, Y.; Chen, G.; Fu, H.; Shen, J.; Shao, L. Inf-net: Automatic COVID-19 lung infection segmentation from ct images. *IEEE Trans. Med. Imaging* **2020**, *39*, 2626–2637. [CrossRef] [PubMed]
5. Desai, A.D.; Gold, G.E.; Hargreaves, B.A.; Chaudhari, A.S. Technical considerations for semantic segmentation in MRI using convolutional neural networks. *arXiv* **2019**, arXiv:1902.01977.
6. Fang, F.; Yao, Y.; Zhou, T.; Xie, G.; Lu, J. Self-supervised Multi-modal Hybrid Fusion Network for Brain Tumor Segmentation. *IEEE J. Biomed. Health Inform.* **2022**, *26*, 5310–5320. [CrossRef]
7. Pathak, D.; Krahenbuhl, P.; Donahue, J.; Darrell, T.; Efros, A.A. Context encoders: Feature learning by inpainting. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, June 2016; pp. 2536–2544.
8. Chen, L.; Bentley, P.; Mori, K.; Misawa, K.; Fujiwara, M.; Rueckert, D. Self-supervised learning for medical image analysis using image context restoration. *Med. Image Anal.* **2019**, *58*, 101539. [CrossRef]
9. Noroozi, M.; Favaro, P. Unsupervised learning of visual representations by solving jigsaw puzzles. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 69–84.
10. Jing, L.; Tian, Y. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 4037–4058. [CrossRef]
11. Chaitanya, K.; Erdil, E.; Karani, N.; Konukoglu, E. Contrastive learning of global and local features for medical image segmentation with limited annotations. *Adv. Neural Inf. Process. Syst.* **2020**, *1052*, 12546–12558.
12. Chaudhari, A.S.; Kogan, F.; Pedoia, V.; Majumdar, S.; Gold, G.E.; Hargreaves, B.A. Rapid Knee MRI Acquisition and Analysis Techniques for Imaging Osteoarthritis. *J. Magn. Reson. Imaging* **2020**, *52*, 1321–1339.

13. Boutin, R.D.; Houston, D.K.; Chaudhari, A.S.; Willis, M.H.; Fausett, C.L.; Lenchik, L. Imaging of Sarcopenia. *Radiol. Clin.* **2022**, *60*, 575–582. [CrossRef] [PubMed]
14. Goyal, P.; Mahajan, D.; Gupta, A.; Misra, I. Scaling and benchmarking self-supervised visual representation learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, October 2019; pp. 6391–6400.
15. Desai, A.D.; Schmidt, A.M.; Rubin, E.B.; Sandino, C.M.; Black, M.S.; Mazzoli, V.; Stevens, K.J.; Boutin, R.; Re, C.; Gold, G.E.; et al. Skm-tea: A dataset for accelerated mri reconstruction with dense image labels for quantitative clinical evaluation. In Proceedings of the Thirty-Fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2), Virtual, 6–14 December 2021.
16. Chaudhari, A.S.; Black, M.S.; Eijgenraam, S.; Wirth, W.; Maschek, S.; Sveinsson, B.; Eckstein, F.; Oei, E.H.; Gold, G.E.; Hargreaves, B.A. Five-minute knee MRI for simultaneous morphometry and T2 relaxometry of cartilage and meniscus and for semiquantitative radiological assessment using double-echo in steady-state at 3T. *J. Magn. Reson. Imaging* **2018**, *47*, 1328–1341. [CrossRef] [PubMed]
17. Chaudhari, A.S.; Stevens, K.J.; Sveinsson, B.; Wood, J.P.; Beaulieu, C.F.; Oei, E.H.; Rosenberg, J.K.; Kogan, F.; Alley, M.T.; Gold, G.E.; et al. Combined 5-minute double-echo in steady-state with separated echoes and 2-minute proton-density-weighted 2D FSE sequence for comprehensive whole-joint knee MRI assessment. *J. Magn. Reson. Imaging* **2018**, *49*, e183–e194. [CrossRef] [PubMed]
18. Eijgenraam, S.M.; Chaudhari, A.S.; Reijman, M.; Bierma-Zeinstra, S.M.; Hargreaves, B.A.; Runhaar, J.; Heijboer, F.W.; Gold, G.E.; Oei, E.H. Time-saving opportunities in knee osteoarthritis: T 2 mapping and structural imaging of the knee using a single 5-min MRI scan. *Eur. Radiol.* **2020**, *30*, 2231–2240. [CrossRef]
19. Chaudhari, A.S.; Grissom, M.J.; Fang, Z.; Sveinsson, B.; Lee, J.H.; Gold, G.E.; Hargreaves, B.A.; Stevens, K.J. Diagnostic accuracy of quantitative multicontrast 5-minute knee MRI using prospective artificial intelligence image quality enhancement. *Am. J. Roentgenol.* **2021**, *216*, 1614–1625. [CrossRef]
20. Chaves, J.M.Z.; Chaudhari, A.S.; Wentland, A.L.; Desai, A.D.; Banerjee, I.; Boutin, R.D.; Maron, D.J.; Rodriguez, F.; Sandhu, A.T.; Jeffrey, R.B.; et al. Opportunistic assessment of ischemic heart disease risk using abdominopelvic computed tomography and medical record data: A multimodal explainable artificial intelligence approach. *medRxiv* **2021**. [CrossRef]
21. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
22. Wu, Y.; He, K. Group normalization. In Proceedings of the European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
23. Qiao, S.; Wang, H.; Liu, C.; Shen, W.; Yuille, A. Micro-batch training with batch-channel normalization and weight standardization. *arXiv* **2019**, arXiv:1903.10520.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1026–1034.
25. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
26. Geirhos, R.; Jacobsen, J.H.; Michaelis, C.; Zemel, R.; Brendel, W.; Bethge, M.; Wichmann, F.A. Shortcut learning in deep neural networks. *Nat. Mach. Intell.* **2020**, *2*, 665–673. [CrossRef]
27. Kornblith, S.; Shlens, J.; Le, Q.V. Do better imagenet models transfer better? In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 2661–2671.
28. Newell, A.; Deng, J. How useful is self-supervised pretraining for visual tasks? In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 7345–7354.
29. Park, A.; Chute, C.; Rajpurkar, P.; Lou, J.; Ball, R.L.; Shpanskaya, K.; Jabarkheel, R.; Kim, L.H.; McKenna, E.; Tseng, J.; et al. Deep learning–assisted diagnosis of cerebral aneurysms using the HeadXNet model. *JAMA Netw. Open* **2019**, *2*, e195600. [CrossRef]
30. Kumar, A.; Raghunathan, A.; Jones, R.; Ma, T.; Liang, P. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv* **2022**, arXiv:2202.10054.
31. Bridson, R. Fast Poisson disk sampling in arbitrary dimensions. *SIGGRAPH Sketches* **2007**. [CrossRef]
32. Sveinsson, B.; Chaudhari, A.; Gold, G.; Hargreaves, B. A simple analytic method for estimating T2 in the knee from DESS. *Magn. Reson. Imaging* **2017**, *38*, 63–70. [CrossRef]
33. Virtanen, P.; Gommers, R.; Oliphant, T.E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nat. Methods* **2020**, *17*, 261–272. [CrossRef] [PubMed]
34. Dai, W.; Woo, B.; Liu, S.; Marques, M.; Tang, F.; Crozier, S.; Engstrom, C.; Chandra, S. Can3d: Fast 3d Knee Mri Segmentation Via Compact Context Aggregation. In Proceedings of the 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), Nice, France, 13–16 April 2021; pp. 1505–1508.
35. Perslev, M.; Pai, A.; Runhaar, J.; Igel, C.; Dam, E.B. Cross-Cohort Automatic Knee MRI Segmentation With Multi-Planar U-Nets. *J. Magn. Reson. Imaging* **2021**, *55*, 1650–1663. [CrossRef]
36. Panfilov, E.; Tiulpin, A.; Klein, S.; Nieminen, M.T.; Saarakkala, S. Improving robustness of deep learning based knee mri segmentation: Mixup and adversarial domain adaptation. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Republic of Korea, 27–28 October 2019; pp. 450–459.
37. Choi, D.; Shallue, C.J.; Nado, Z.; Lee, J.; Maddison, C.J.; Dahl, G.E. On empirical comparisons of optimizers for deep learning. *arXiv* **2019**, arXiv:1910.05446.

38. Prechelt, L. Early stopping-but when? In *Neural Networks: Tricks of the Trade*; Springer: Berlin/Heidelberg, Germany, 1998; pp. 55–69.
39. Chollet, F. Keras. 2015. Available online: https://keras.io (accessed on accessed date 21 December 2022).
40. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. Available online: https://www.tensorflow.org/ (accessed on accessed date 21 December 2022).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.