SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY — INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Large-scale Radiograph Pre-training

Niklas Bühler

SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY — INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Master's Thesis in Informatics

Large-scale Radiograph Pre-training

Author:	Niklas Bühler
Examiner:	Prof. Dr. Daniel Rückert
Supervisor:	Paul Hager, M.Sc.
Submission Date:	15.11.2024

I confirm that this master's thesis is my own work and I have documented all sources and material used.

Munich, 15.11.2024

Niklas Bühler

Abstract

In recent years, medical imaging datasets have expanded significantly, offering great potential for the development of machine learning solutions in the medical field. However, manual labeling of medical data is costly and poses a significant bottleneck to their utilization. Self-supervised learning (SSL) techniques offer a solution to this problem by extracting meaningful representations from the raw data itself, enabling label- and compute-efficient training of specialized models for downstream tasks. In this work, we demonstrate the effectiveness of SSL techniques, specifically the Masked Autoencoder (MAE) strategy, to generate such representations from a large-scale real-world clinical dataset comprising more than 600,000 radiograph images from various anatomical regions. We introduce a novel Dynamic Batch Binning technique that reduces necessary compute by 80% when training on datasets with high image resolution variability. Our results across several clinically relevant downstream tasks show that the generated representations substantially reduce the dependence on labeled data. This is evidenced by superior performance when comparing to supervised training from scratch. Furthermore, we demonstrate the efficacy of domain-specific pre-training by achieving improved performance on the specialized medical task of fracture detection, compared to broader ImageNet-21k pre-training, despite using only 5% of its training samples and 0.5% of its training iterations during pre-training. Our research thus demonstrates the potential of domain-specific MAE pre-training to significantly reduce the need for labeled training data in the medical domain, enabling a more effective utilization of large medical datasets with minimal labels.

Contents

1 Introduction

The recent increase in large-scale medical imaging datasets presents a significant opportunity for exploitation by machine learning methods. However, this opportunity is hindered by the labor-intensive and costly process of manual labeling by medical professionals, a necessary step for training supervised learning models. Self-supervised learning (SSL) emerges as a viable solution to this problem, particularly in the domain of medical imaging, where the amount of unlabeled data typically far exceeds that of labeled subsets. Pre-training, a common SSL technique, involves training a general foundation model in a self-supervised manner on vast amounts of unlabeled data, enabling it to extract meaningful representations from the inherent structure of the raw data itself. After such a pre-training, this foundation model can then be fine-tuned in a label- and compute-efficient way for specialized downstream tasks.

SSL techniques can be further partitioned into contrastive and autoassociative methods. Although contrastive learning frameworks like SimCLR [Che+20] surpassed previous SSL benchmarks on ImageNet [Den+09], we suggest that contrastive learning is not the optimal approach to extracting useful representations from medical images. This view is based on the core working principle of contrastive learning. The contrastive learning technique extracts representations by contrasting positive and negative pairs of instances, oftentimes leading the model to focus on global features instead of intricate details that are typically present in medical images and necessary to take into account for solving complex downstream tasks. In contrast, autoassociative learning techniques, like Masked Autoencoders (MAEs) [He+22], are trained to reconstruct their own input data. They can learn to reconstruct even complex details present in this data and are thus promising candidates for generating representations that capture the fine-grained features of medical images without explicit labels. Such versatile representations can subsequently serve as a strong foundation for various downstream tasks, reducing the dependence on large amounts of labeled data.

By leveraging data sourced from the Rechts der Isar Hospital (MRI), we were able to test this hypothesis in a large-scale, real-world clinical setting. Previously, we already tested the MAE SSL strategy in a smaller-scale pilot study on the publicly available IRMA dataset [Leh+03] consisting of 12,000 radiographs from various anatomical regions. This pilot study proved the feasibility of this approach and also showed preliminary evidence for the efficacy of the MAE SSL strategy in low-data scenarios: The pre-trained Vision Transformer Masked Autoencoder (ViT MAE) model outperformed a supervised Vision Transformer (ViT) model on an anatomical region classification (ARC) task in the low data (1%) regime by achieving a 84.58% vs. 79.08% top-1 accuracy, yielding a 7% relative improvement. In this work, we have extended the MAE pre-training to a much larger dataset sourced from the MRI, comprising more than 600,000 radiographs. Furthermore, we have evaluated the models on the more complex and clinically relevant downstream tasks of foreign material detection (FMD) and fracture detection (FRAC).

Scaling the pre-training to 600,000 real-world radiographs presented us with several challenges. On the one hand, the computational cost of pre-training increased significantly. This was not only due to the larger dataset size, but also due to training on high-resolution images, frequently reaching up to 3,072 × 3,072 pixels, instead of the standard resolution of 224 × 224 often used in the field of computer vision. This scaling of image resolution lead to a costly 188× increase in pixels per image[1], but medical conditions like hairline fractures might not be discernible with a resolution of only 224 × 224 pixels. Another challenge arising from the real-world nature of the scans in this dataset was handling the high variability in image resolutions, as there were 391,013 unique resolutions present in our dataset. Transformer-based architectures are in general capable of handling variable-sized inputs, but in order to enable efficient batch processing during training, we devised and implemented a novel batching strategy. Our strategy clusters images of similar resolution to minimize computational overhead caused by padding images in each batch to compatible sizes. Finally, optimizing data storage, caching and loading were crucial steps of our implementation, as the whole dataset comprises over 5.5 terabytes of storage.

This research explores the efficacy of MAEs for pre-training on a large-scale real-world radiograph dataset spanning various anatomical regions. Our contributions are the following:

- We demonstrate that the MAE SSL strategy effectively alleviates the labeling bottleneck in medical imaging by extracting high-quality, versatile embeddings from unlabeled data which significantly enhance performance on several medically relevant downstream tasks with minimal labels.
- We show that domain-specific pre-training can outperform even considerably larger general-purpose pre-training, as evidenced by improved performance on the specialized medical task of fracture detection.
- We propose practical solutions to the challenges posed by large-scale, real-world medical imaging datasets, paving the way for further exploration into the applicability of SSL techniques across medical imaging challenges.

[1] Due to the quadratic complexity of the Transformer self-attention mechanism, this would have resulted in an even more extreme 35,375× increase in operations per image, if not further addressed.

2 Background & Related Work

2.1 Transformer-based Models

Since the introduction of the Transformer in 2017, transformer-based architectures have taken over many subfields of machine learning, starting a revolution in natural language processing (NLP) and progressively also taking over the field of computer vision since their adoption to images in 2021. State-of-the-art architectures are increasingly shifting away from the convolutional paradigm and towards models based entirely on the attention mechanism or hybrid architectures.

Attention Is All You Need [Vas+17] introduced the Transformer architecture as a novel alternative to traditional recurrent neural networks (RNNs) or convolutional neural networks (CNNs) for sequence modeling and transduction tasks. It achieved state-of-the-art results in machine translation tasks and generalizes effectively to various applications. It relies solely on the attention mechanism, resulting in superior parallelizability and training efficiency compared to previous models. The Transformer model consists of an encoder and a decoder part. The encoder maps an input sequence of symbol representations to a sequence of continuous representations, also called embeddings. The decoder generates the output sequence one element at a time, in an auto-regressive fashion, i.e. consuming the previously generated symbols as additional input. Both the encoder and decoder utilize stacked self-attention and point-wise, fully connected layers.

Due to their superior results, parallelizability and flexibility regarding varying input sizes, transformer-based architectures serve as the base for all of our tested models.

An Image is Worth 16x16 Words [Dos+21] extended the Transformer architecture to computer vision by introducing the Vision Transformer (ViT). This approach reshapes 2D images into sequences of flattened 2D patches, which are fed into an architecture of alternating multi-headed self-attention and multi-layer perceptron blocks. Consequently, the outputs of the ViT encoder are representations of the image patches. A special classification token `[CLS]` is prepended to the sequence of patches. Its final representation serves as an aggregate representation of the entire image for downstream tasks, such as classification. The process of passing an image through the ViT architecture is visualized in Figure 2.1.

This work lays the base for extracting image representations using the Transformer architecture and is thus essential to our work. Furthermore, the authors highlight the possibility of pre-training the model on large datasets and subsequently fine-tuning it on more specific downstream tasks, by replacing the pre-trained prediction head with a zero-initialized feed-forward layer. They also mention the possibility of interpolating the pre-trained positional embeddings to perform fine-tuning on a different resolution than was used during pre-training, a method we heavily rely on in our work.

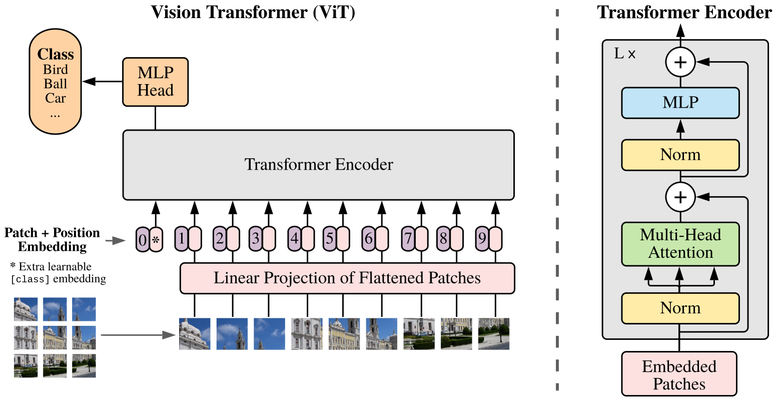

Figure 2.1: The architecture of the ViT model. An input image is reshaped into patches, combined with a positional embedding and fed into the Transformer encoder. The [CLS] token is prepended. Adapted from [Dos+21].

Masked Autoencoders are Scalable Vision Learners [He+22] proposed the Masked Autoencoder (MAE) for image reconstruction, demonstrating its effectiveness in achieving state-of-the-art accuracy in image classification and transfer learning tasks. The MAE training strategy consists of masking random patches of the input image and reconstructing the missing pixels. The asymmetric encoder-decoder architecture involves an encoder that operates only on unmasked patches and a lightweight decoder responsible for image reconstruction. Masking a high proportion (e.g. 75%) of the input yields a nontrivial and meaningful self-supervisory task. The lightweight decoder architecture, combined with the fact that the encoder only processes a small portion of the data, accelerates training significantly while also improving accuracy.

The MAE encoder consists of a ViT, which is applied only on the visible, unmasked patches of the input image. It embeds these patches by a linear projection with added positional embeddings and processes the resulting embeddings in a series of Transformer blocks. Because no mask tokens are fed into the encoder, it operates only on a small subset of visible tokens and thus only requires a fraction of compute and memory. The typically high masking ratio largely eliminates the inherent spatial redundancy of images, resulting in a task that cannot be solved easily by extrapolation.

The MAE decoder receives the full set of tokens as input, i.e. the encoded visible patches and the mask tokens (a single shared, learned vector that indicates the presence of a missing patch). Positional embeddings are added to all tokens before they are passed through another series of Transformer blocks. Notably, the decoder is only used during pre-training to perform the image reconstruction task and typically uses significantly less computation than the encoder (less than 10%). The reconstruction task consists of predicting the pixel values for each masked patch. As each element in the decoder's output is a vector of pixel values representing a patch, the loss can be computed by calculating the mean squared error between the reconstructed and the original image in the pixel space. The process of passing an image through the ViT MAE architecture is visualized in Figure 2.2.

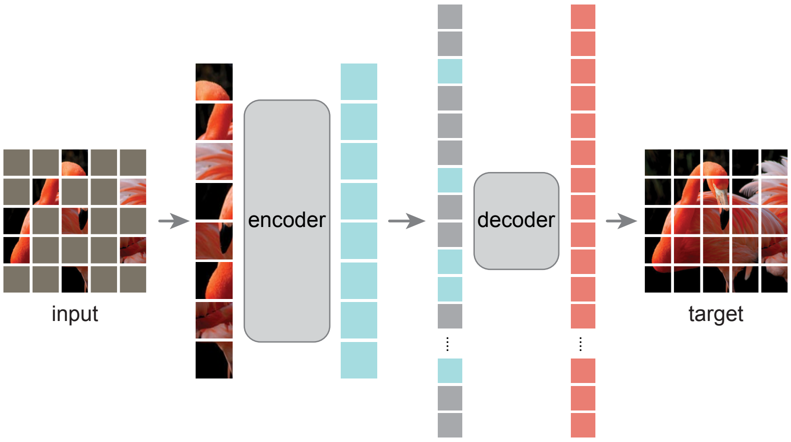

Figure 2.2: The architecture of the ViT MAE model. The input patches are masked according to the masking ratio. Only the visible patches are fed into the encoder and the encoder output plus the mask tokens are passed into the decoder. Finally, the decoder output is compared to the reconstruction target to determine the loss. Adapted from [He+22].

By choosing an appropriate patch size when transforming the input images into sequences of tokens, the MAE can learn to reconstruct even fine-grained but essential features of its input, like hairline fractures and other granular pathologies of interest when training on radiographs. Because pathologies will oftentimes be split across multiple patches, choosing an adequate masking ratio will force the model to learn to reconstruct pathologies that are only partly masked. As a consequence of learning to reconstruct these fine-grained features and pathologies, the model effectively learns to form meaningful internal representations of these characteristics as outputs of its encoder part. This is the vital property making the MAE encoder a promising candidate to serve as pre-trained foundation model for specialized downstream tasks. The patch size and masking ratio have to be chosen in a trade-off between modeling accuracy and efficiency / memory constraints.

2.2 Self-supervised Learning in Medical Imaging

Due to the prevalence of unlabeled data in medical imaging and the fact that generating high-quality labels for medical images is costly, self-supervised learning (SSL) plays an important role in the field. Self-supervised pre-training of models lessens their dependence on labeled data, thereby cutting costs, reducing training time and compute, and enabling researchers and medical personnel to build more specialized models in less time.

A Simple Framework for Contrastive Learning of Visual Representations [Che+20] introduced SimCLR, a contrastive learning approach for generating representations of images. SimCLR achieved competitive performance with fully supervised models on ImageNet classification and transfers well to other tasks. It relies heavily on data augmentations.

Despite its convincing performance, we believe the contrastive learning approach in general is not a fitting choice when trying to extract meaningful representations from medical images, as the model is not forced to focus on reconstructing fine-grained features of the medical data, but rather trained to contrast samples based on their global features.

Self Pre-training with Masked Autoencoders for Medical Image Classification and Segmentation [Zho+23] applied MAEs to medical image analysis. The authors argue that the MAE aggregates contextual information to infer masked image regions, enhancing the understanding of interdependencies among anatomical structures crucial in the medical imaging domain. The self pre-training method involves pre-training on the same dataset as used for the downstream task and fine-tuning with task-specific heads. Their experimental results demonstrate significant enhancements in medical image segmentation and classification performance compared to random initialization and traditional ImageNet pre-training methods. Notably, MAE self-pretraining showed promising performance even on small-scale medical datasets, surpassing existing approaches, including ImageNet transfer learning.

While this approach is in general similar to ours, the authors simply cropped out 224 × 224 pixel regions from the images in their dataset to conduct training on. This might be a viable strategy for their targeted task of classifying pathologies found in the Chest X-ray 14 dataset [Wan+17], but is certainly a limitation of their approach, since extensive cropping might remove pathologies and other clinically relevant details from the input data. Relying on other means of reducing the image resolution is also suboptimal, as evidenced by [Wol+23], where the authors showed that training with a resolution of 1,024 × 1,024 pixels on the same Chest X-ray 14 dataset improves classification performance, while training on lower resolutions, such as 256 × 256 pixels, is oftentimes insufficient for identifying small pathologies, forcing models to use spurious discriminating features.

Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture [Ass+23] focused on the partitioning of self-supervised learning from images into generative methods and invariance-based, or contrastive, methods. The authors claim that generative methods like masked pre-training require less prior knowledge than view-invariance approaches, but the resulting representations are typically of lower semantic level and require a more involved adaptation, e.g. end-to-end fine-tuning. Invariance-based methods on the other hand optimize the encoder to produce similar embeddings for two views of the same image. The authors argue that this approach can produce representations of high semantic level, but might also introduce strong biases which might be detrimental for certain downstream tasks. As a solution, they proposed I-JEPA, a method to learn strong representations without relying on hand-crafted view augmentations. I-JEPA works by predicting missing information in an abstract representation space. Representations of target blocks (parts of the image) have to be predicted from a single context block. These target representations are computed by a learned target-encoder network. The main idea behind this architectural choice is to eliminate irrelevant pixel-level details, such that the model can focus entirely on semantic features. The authors showed that their method outperformed pixel-reconstruction methods like the MAE on ImageNet-1k, while also being 10× more efficient, due to predicting in the representation space.

2.3 Training on High-resolution and Variable-resolution Images

Most computer vision models perform training and inference on the de facto standard resolution of 224 × 224 pixels. Medical images like radiographs however tend to be high-resolution and often vary significantly in their exact resolutions due to different anatomical regions being visualized with identical pixel spacing. While both convolutional methods and Transformers are in general flexible and able to take images of varying resolutions as their input, training in batches requires at least intra-batch images to be of the same resolution.

Typical strategies to deal with images of varying resolutions include resizing, cropping, and padding. However, in the context of medical imaging, these strategies are suboptimal. While resizing oftentimes distorts the original aspect ratio of the image, cropping might remove relevant information altogether, and padding introduces a potentially large amount of unnecessary padding tokens, which can be masked during training, but still lead to an increase in overhead. These problems become increasingly important with large-scale and real-world datasets, as the variance in resolutions grows and computational overhead gets more expensive.

Swin Transformer: Hierarchical Vision Transformer using Shifted Windows [Liu+21] and **Swin Transformer V2: Scaling Up Capacity and Resolution** [Liu+22] introduced and extended the Shifted Windows (Swin) Transformer. The Swin Transformer is the author's proposed general-purpose computer vision backbone. They argue that there are certain challenges in adapting the Transformer from text to vision, namely scale variations and high image resolutions. Their hierarchical architecture addresses these challenges by efficiently limiting self-attention computation to non-overlapping local windows, while at the same time allowing for cross-window connections. They showed that the Swin Transformer is flexible for modeling at various scales, while retaining linear computational complexity. In their second paper, introducing the second version of the Swin Transformer, they trained on images with resolutions of up to 1,536 × 1,536 pixels.

Swin MAE: Masked Autoencoders for Small Datasets [Dai+23] built upon the Swin Transformer to address the limited availability of large and well-annotated datasets in the medical domain. While the authors acknowledge the advantages of unsupervised learning for medical image analysis due to its label-free nature, they criticize its dependence on large datasets. Their proposed solution to this problem is the Shifted Windows Masked Autoencoder (Swin MAE), a Masked Autoencoder with a Swin Transformer as its backbone. They showed that the Swin MAE works effectively even on small datasets, consisting of a few thousand medical images, and without pre-training. Their model was capable of learning semantic features solely from images, with a performance comparable to that of a supervised Swin Transformer, pre-trained on ImageNet and transferred to the downstream tasks.

Patch n' Pack: NaViT, a Vision Transformer for any Aspect Ratio and Resolution [Deh+24] leveraged the flexible sequence-based modeling of ViTs to tackle the ubiquitous but suboptimal choice of resizing images to a fixed resolution before processing them in computer vision models. The Native Resolution Vision Transformer (NaViT) uses sequence packing to handle arbitrary resolutions and aspect ratios, i.e. multiple patches from different images are packed into a single sequence and processed by the model simultaneously. During self-attention and pooling operations, the receptive field of each token is limited to tokens from the same source image.

FlexiViT: One Model for All Patch Sizes [Bey+23] addressed the trade-off between efficiency and accuracy that is faced when choosing a patch size for training a ViT. Smaller patches lead to higher accuracy, but also increase computational cost, and changing the patch size typically requires retraining the model. The authors proposed randomizing the patch size during training and showed that this simple drop-in change leads to a single set of weights that performs well across a wide range of patch sizes. The implementation of randomized patch sizes during training time involves resizing the patch embedding weights and positional embeddings using bilinear interpolation. This approach was already briefly proposed in the original ViT paper [Dos+21] to enable fine-tuning at a higher resolution than was used during pre-training. In their experiments, the authors used an image resolution of 240 × 240 pixels and patch sizes ranging from 8 × 8 to 48 × 48 pixels. This technique has also been employed in works such as [Juy+24; Var+24].

VariViT: A Vision Transformer for Variable Image Sizes [Var+24] introduced the Variable Image Size Vision Transformer (VariViT) model architecture, which is capable of handling variable image sizes while maintaining a consistent patch size. The authors developed a novel positional embedding resizing scheme for a variable number of patches, as well as a new batching strategy to reduce computational complexity and computation time by up to 30%. This batching strategy groups images of the same size into batches and uses gradient accumulation to perform weight updates on gradients over several mini-batches.

While this paper focuses on 3D image representation learning and also utilizes the consistent center alignment property of tumor crops—two aspects that were less relevant in our own research—their batching strategy served as inspiration for our own custom batching strategy, which enabled us to efficiently handle the extremely high variation of image resolutions present in our dataset. The authors also mentioned the exploration of their batching strategy and positional embedding technique in the context of extremely large datasets or high-resolution images as promising avenues for further research, both of which we covered in our work.

3 Method

3.1 Data and Preprocessing

There were several characteristics of the real-world clinical data we used that not only necessitated specific preprocessing steps, but also had a direct influence on our training procedures. The two most prominent characteristics were the unusually large image dimensions, frequently reaching 3,072 × 3,072 pixels or higher, and the extreme variability in resolutions and aspect ratios present in the data.

3.1.1 Rechts der Isar Hospital Radiograph Data

The data sourced from the Rechts der Isar Hospital (MRI) consists of a total of 647,636 radiograph images stemming from 169,251 different patients and spanning 14 different anatomical regions. The exact distribution of radiographs across different anatomical regions is visualized in Table 3.1 and Figure 3.1. Samples of the raw imaging data of each anatomical region are visualized in Figure 3.2. In total, the dataset consists of over 5.5 TB of radiograph data.

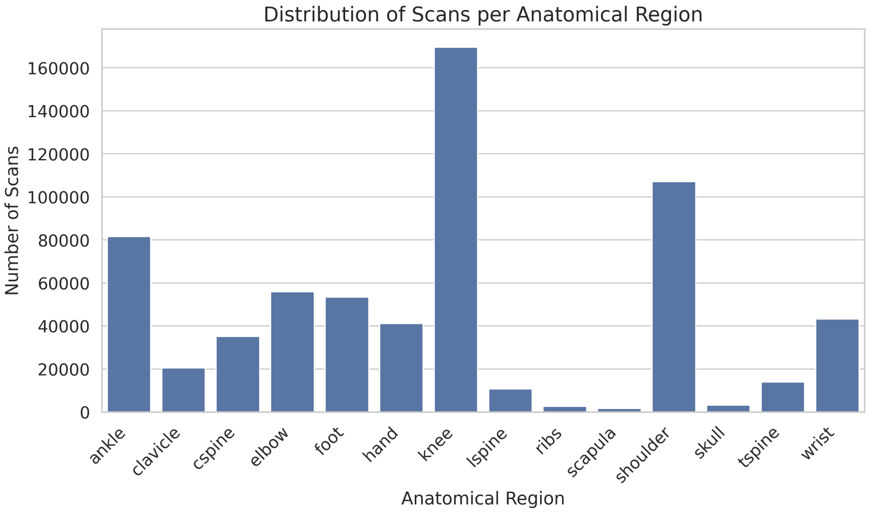

Figure 3.1: A histogram showing the distribution of scans across different anatomical regions in the MRI dataset.

The MRI dataset was compiled by extracting radiograph information from the Picture Archiving and Communication System (PACS), which serves as a central repository for

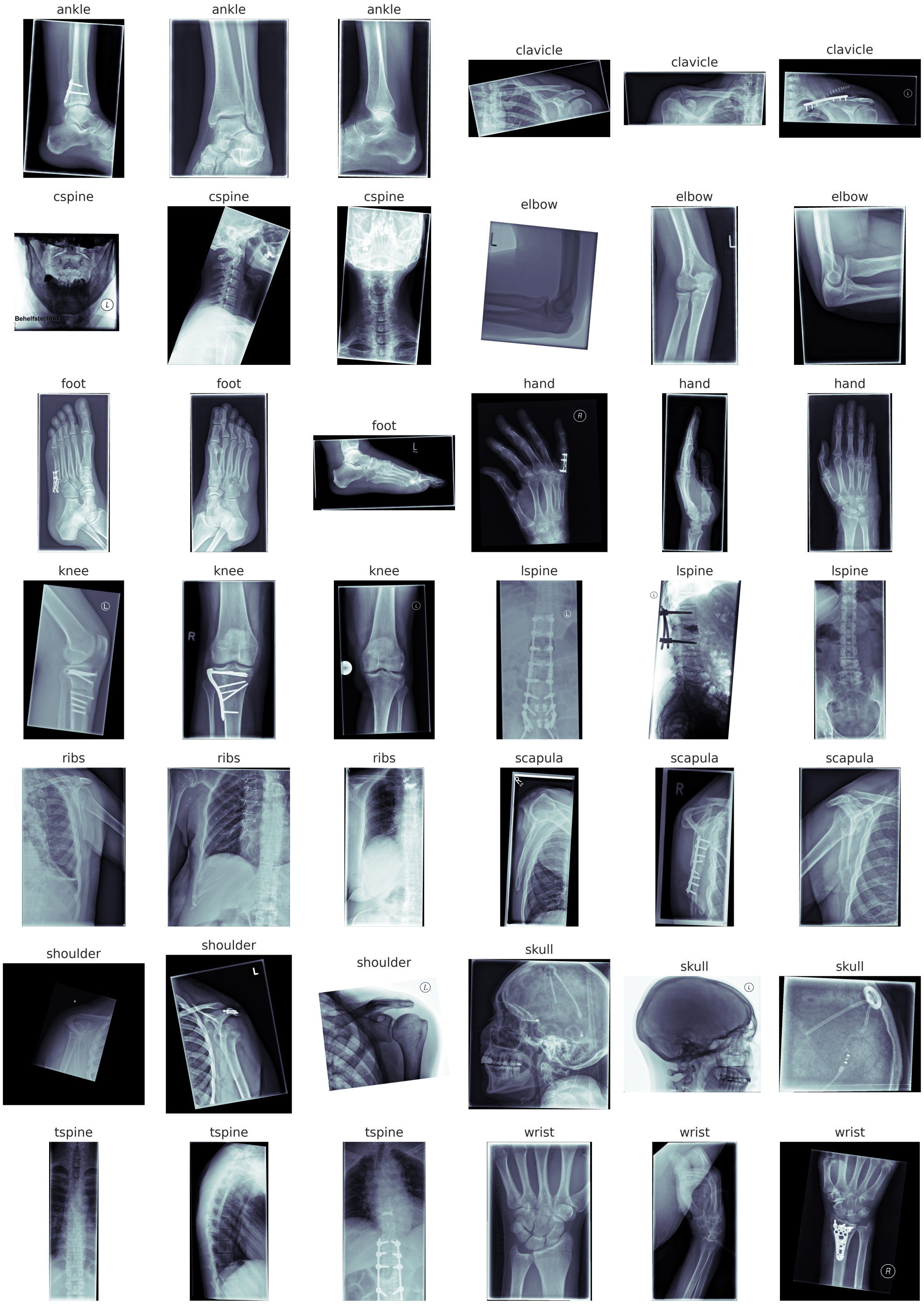

Figure 3.2: Raw imaging data of each anatomical region contained in the MRI dataset. Several regions overlap, some have scans taken from multiple different perspectives. Some scans are misclassified in the PACS. The visualized samples are still unprocessed, showing various falsely inverted scans. Many also show the black padding often contained in the raw radiograph data.

Table 3.1: An overview of the anatomical regions contained in the MRI dataset, after filtering and preprocessing.

Anatomical Region	# Patients	# Scans
Knee	39,970	169,595
Shoulder	25,314	107,070
Ankle	23,532	81,577
Elbow	12,610	55,914
Foot	16,226	53,385
Wrist	10,367	43,246
Hand	15,402	41,201
Cervical spine	9,838	35,150
Clavicle	3,844	20,506
Thoracic spine	5,834	13,995
Lumbar spine	3,272	10,756
Skull	1,122	3,164
Ribs	1,261	2,668
Scapula	659	1,650
Total	**169,251**	**639,877**

medical imaging data in the clinic. Each radiograph is stored in the Digital Imaging and Communications in Medicine (DICOM) format [Nat24], an international standard for transmitting, storing, processing, and displaying medical imaging information. DICOM files not only contain the raw image data, but also metadata like patient demographics and imaging modalities.

An important characteristic of this dataset is the extremely high variability in image resolutions and aspect ratios. In radiography, the term spatial resolution refers to the ability of an imaging modality to differentiate two adjacent structures as being distinct. It is typically measured in *line pairs per millimeter (lp/mm).* The DICOM standard [Nat24] contains a special `SpatialResolution` attribute for this, defined as *The inherent limiting resolution in mm of the acquisition equipment for high contrast objects for the data gathering and reconstruction technique chosen.* The `SpatialResolution` DICOM attribute in our dataset assumes values in the range 0.143–0.148, meaning the spatial resolution lies between 3.38 and 3.50 lp/mm[1]. Because an almost identical spatial resolution is obtained for scans across all anatomical regions, the resulting radiographs differ in size, just like the anatomical regions themselves differ in size. This fact, plus the fact that each radiograph is limited to the region of interest in order to minimize the radiation exposure of the patient, lead to almost every radiograph having a unique resolution and aspect ratio. Consequently, there are 391,013 unique resolutions present in the dataset. Histograms

[1] The human eye for example has the capability of differentiating a spatial resolution of 5 *lp/mm* at a viewing distance of 25 cm [HA15].

of the distribution of radiograph resolutions are shown in Figure 3.3.

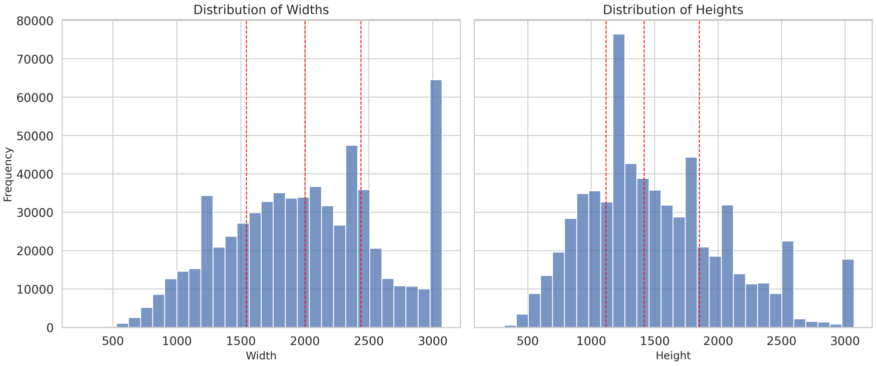

Figure 3.3: Histograms showing the distribution of radiograph resolutions present in the MRI dataset, after removing radiographs with a resolution higher than 3,072 × 3,072 pixels. The respective quartiles are marked by red lines.

3.1.2 Data Preprocessing

The imaging data in DICOM files is represented by a 2D pixel array of unsigned 16-bit integers. Additionally, the DICOM headers of each file contain information about patient demographics and imaging modalities.

By inspecting the maximum pixel value present in each pixel array, we filtered out five scans that were completely black and two more which were black with only patient information written on them. Inspecting the distribution of mean pixel values across all pixel arrays revealed the same black scans, as well as 31 additional scans that contained only a small portion of imaging data, surrounded by thick black borders. These were also removed from the dataset. The largest pixel arrays had a resolution of up to 8,000 × 5,000 pixels. In total, there were 7,721 images with a resolution larger than 3,072 × 3,072 pixels, stemming from scans of e.g. whole spines or whole legs. We determined this resolution to be a good cutoff point and dropped all larger scans from the dataset. The smallest pixel arrays contained scans of single fingers or toes, which we kept. The resulting distribution of widths and heights is shown in Figure 3.3. Out of the 647,636 valid images originally obtained from the MRI PACS, this left us with 639,877 valid, filtered radiographs.

There were 46,833 pixel arrays with a minimum pixel value greater than zero, many of them representing inverted scans. The raw imaging data visualized in Figure 3.2 contains several inverted scans. Occasionally, medical professionals invert radiographs on purpose to get a better view of the trabeculae. Nevertheless, after consultation with radiologists, we decided to correct for the inverting of radiographs as far as possible. We tested multiple approaches to find and correct inverted radiographs, including

- filtering by minimum pixel value > 0,

- filtering by minimum pixel value $> c$ with $c \in \mathbb{N}$,
- filtering by mean pixel value $> c$ with $c \in \mathbb{N}$,
- filtering by pixel values on the edges of images, and
- utilizing the DICOM headers
  - `ShowGrayscaleInverted`,
  - `PhotometricInterpretation` and
  - `PresentationLUTShape`.

However, we encountered false positives with all five approaches, and due to the large amount of scans in our dataset, there exists no practical way of making sure there are no false negatives as well. This also presents a challenge when comparing different strategies for identifying inverted radiographs. Ultimately, we inverted all images based on the `PhotometricInterpretation` DICOM header, as this seemed to be the most reliable way of identifying falsely inverted radiographs.

We manually inspected random samples of the data to identify common artifacts, some of which can be seen in Figure 3.2. The most common ones include markers to identify the laterality of the scanned body part, e.g. L/R, markers to identify the positioning of the patient during the scanning procedure, e.g. *Liegend/Stehend*, and x-ray reference spheres.

There are several other relevant points which do not require direct intervention during preprocessing, but are still important to keep in mind when working with this kind of data. For example, pathologies such as hairline fractures may only be detectable at a very fine-grained level of detail. Such information is almost certainly lost when resizing to a low resolution of 224 × 224 pixels, as is standard practice in computer vision. Furthermore, when considering multiple scans from the same patient, it is important to note that the patient could have been diagnosed elsewhere before, making their first scan in the dataset a follow-up scan in reality. Additionally, in some cases a fracture may not be visible in an initial scan but could appear in a follow-up scan, for instance through the process of demarcation.

3.2 Models

3.2.1 Vision Transformer Masked Autoencoder

The foundation for our pre-training is a Vision Transformer Masked Autoencoder Base (ViT MAE B) model, which is a MAE using a Vision Transformer Base (ViT-B) as its encoder. We modified the model to handle variable image sizes, by bilinearly interpolating the positional encoding, even during pre-training itself.

Choosing a suitable patch size is an important requirement for extracting meaningful representations, as the model best learns to reconstruct anomalies when they are partly masked during pre-training. The patch size also has direct influence on the memory requirements of training the model, so choosing it is a trade-off between accuracy and memory/compute capacity available. We tested possible patch sizes of 16, 32, 48 and 64 and determined a patch size of 48 × 48 pixels to be a valid choice. The original ViT-B expects an input image size of 224 × 224 pixels, resulting in 196 tokens per image, when using a patch size of 16 × 16 pixels. Our model supports a maximum resolution of 3,072 × 3,072 pixels, resulting in a maximum of 4,096 tokens per image, when using our adapted patch size of 48 × 48 pixels. The median image size in our data is roughly 2,000 × 1,500, so the median amount of tokens per image is roughly 1,302. This results in an average 7× increase in tokens input into the model. However, many of these tokens represent uninformative black padding, which is present in the raw data, even without padding during preprocessing. Our chosen patch size is visualized in Figure 3.4. We also tested different hidden and intermediate sizes for the MAE encoder, but decided to keep the default ViT-B configuration for these parameters. Because radiographs are monochrome, we only require one channel, instead of three, as used in the original ViT-B for RGB images. Apart from these changes, we kept the same configuration as the original ViT MAE B. The final configuration is listed in Table 3.2.

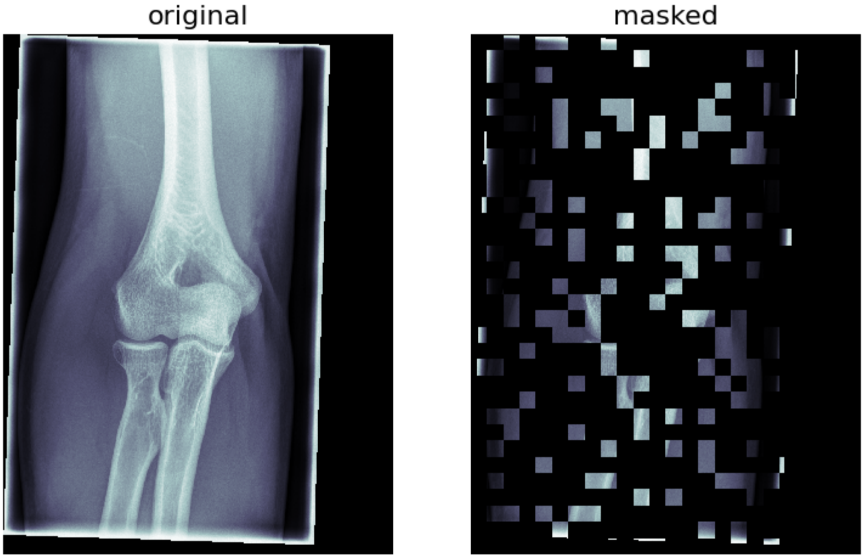

Figure 3.4: A visualization of the chosen patch size of 48 × 48 pixels and masking ratio of 75%. The image is padded to a multiple of the patch size. Note that most radiographs already contain a black border before preprocessing.

3.2.2 Vision Transformer

Because the encoder part of the MAE used during fine-tuning is essentially a Vision Transformer Base (ViT-B), we used an equivalent ViT-B model, once randomly initialized

Table 3.2: An overview of the ViT MAE B and ViT-B model configurations. Parameters that differ from their default definition are printed in bold with their original values in parentheses.

Parameter	ViT MAE B	ViT-B
Maximum image size	**3,072** (224)	**3,072** (224)
Number of channels	**1** (3)	3
Patch size	**48** (16)	16
Number of attention heads	12	12
Intermediate size	3,072	3,072
Number of hidden layers	12	12
Hidden size	768	768
Hidden activation	GELU	GELU
Decoder number of attention heads	16	/
Decoder intermediate size	2,048	/
Decoder number of hidden layers	8	/
Decoder hidden size	512	/

and once pre-trained on ImageNet data, as our baseline models to compare all fine-tuning experiments to. In order to use the ImageNet pre-trained weights for this model, we had to keep the default of three input channels, instead of using only one input channel as we did for the MAE model. To accommodate this, we replicated the single radiograph channel across all three input channels. After their respective initialization, these baseline models were fine-tuned in the same way and on the same data as the ViT MAE model pre-trained on radiograph data. An overview of the baseline model configuration is given in Table 3.2.

3.3 Training

Our approach encompasses two phases of training: pre-training and fine-tuning. The pre-training phase leverages large amounts of unlabeled data with the goal of extracting meaningful and versatile representations from the raw data itself. In this stage, the MAE model is trained to reconstruct masked portions of the input data, encouraging the model to capture underlying patterns and relationships in the data without requiring explicit labels. In the fine-tuning phase, this pre-trained foundation model is adapted to specific downstream tasks using small sets of labeled data spanning only a few hundred samples each. This transfer learning approach enables the model to converge faster and achieve a better performance than when starting with randomly initialized weights.

We stratify the train, validation and test set by patients, as to not contaminate the testing data with follow-up scans of patients already seen during training. All training runs were conducted using PyTorch [Pas+19] on an NVIDIA A40 GPU, with 48 GB of memory, and in a mixed precision setting.

3.3.1 Training on Variable-resolution Images

As outlined in Section 3.1, there are 391,013 unique resolutions present in our dataset. These different resolutions have to be handled appropriately during training in order to enable proper batch processing. Our transformer-based models are in general capable of handling variable-sized inputs, but all images contained in a batch still have to be of the same resolution and all image resolutions have to be multiples of the model's patch size. Furthermore, positional encoding requires special attention with differing input sizes. Rather than simply enabling variable resolution training, we designed our solution to this problem with an additional focus on computational efficiency. This is especially important due to the large resolutions and large-scale dataset we were training on.

Naive solutions to this problem include resizing, cropping, and padding. While resizing is a standard approach in the field of computer vision, it is suboptimal for medical imaging, as it destroys the original aspect ratio, as well as the uniform pixel spacing across images, and might introduce artifacts or lead to the loss of important fine-grained information. Cropping is also not an option in our situation, as it could remove crucial characteristics and anomalies from radiographs. This is especially problematic due to the large range of image resolutions present, as cropping large 3,072 × 3,072 scans to e.g. 1,000 × 1,000 pixels would result in the loss of about 90% of their original content. Padding all images to a fixed size does not run into these problems, but would unnecessarily increase computational complexity by introducing very large uninformative black borders, due to the wide range of resolutions present in the dataset.

Adjusting the patch sizes on a per-image basis, as proposed by [Bey+23], is suboptimal in the medical domain because the size of fractures and other pathologies is in general not directly correlated with the size of the investigated anatomical region. Although fractures are typically smaller in regions such as the hands compared to e.g. the femur, they can also present as finer structures in areas like the shoulder, skull, or ribs. Consequently, choosing a larger patch size for larger scans is not a straightforward solution. Utilizing a CNN or similar for embedding the images using a variable kernel size to obtain a fixed number of patches runs into the same problem, while also introducing a new embedding module which might not work as well as the default projection.

Keeping a fixed patch size and varying the masking ratio depending on the total amount of input tokens per image would not run into this same problem, but might introduce a bias against certain resolutions or even whole anatomical regions, as some images would not be masked at all, while others would be masked almost completely.

In general, removing tokens on a per-image basis might not necessarily speed up model training as memory allocation is typically slow, but using variable image sizes across batches can still offer improvements, as in this case only inter-batch memory usage changes. Thus, we decided to keep intra-batch image sizes constant, while minimizing introduced padding across batches.

Dynamic Batch Binning

Our solution to this problem was inspired by [Var+24], where the authors developed a custom batching strategy that groups images of the same sizes into batches, while allowing image size to vary from batch to batch. However, due to the high variability of resolutions in our dataset, this strict batch binning strategy, where each unique resolution forms its own bin, would lead to high fragmentation and many either incomplete or discarded batches. We thus propose the Dynamic Batch Binning (DBB) strategy. This strategy uses a fixed number of bins and corresponding resolutions, determined by the distribution of resolutions present in the dataset. Images are dynamically sorted into the bin representing the next higher resolution and padded to that resolution. This approach thus minimizes the additionally introduced padding while still forming large enough bins to avoid fragmentation.

The bins should be chosen with respect to the resolutions present in the dataset and the patch size to be used by the model. In theory, bin sizes can be picked completely dynamically, depending on the dataset at hand, but we picked them manually, by combining viable cutoff points that emerge from the distribution of resolutions with suitable multiples of the model's patch size. Considering the quartiles of the resolution distribution shown in Figure 3.3 and possible model patch sizes of 8 × 8, 16 × 16, 32 × 32 and 48 × 48 pixels, we decided to define the bins by specifying six common cutoff points for both width and height: 1,152, 1,536, 1,920, 2,304, 2,688, and 3,072. These cutoff points result in a total of $6 \cdot 6 = 36$ different bins. Thus, even in a validation setting of using only 5% of the whole dataset, each bin contains roughly 800 images to form batches from.

We employed this batching strategy during pre-training, where the savings in computational overhead were largest. Due to memory constraints, we additionally employed gradient accumulation across batches. During fine-tuning, we used exact image sizes, padded to the next multiple of the model's patch size, as our fine-tuning datasets were too small (several hundred images) to form meaningful bins. We also employed gradient accumulation to form virtual batches with an effective batch size larger than one during fine-tuning.

Positional Encoding

By design, and unlike CNNs, Transformers lack spatial awareness across tokens. The added positional encoding provides this spatial information to the model, enabling it to relate the relative positions of patches in an image. As the positional encoding scheme is fixed for a single model, handling images of varying resolutions is not trivial. The original ViT paper [Dos+21] proposed bilinear interpolation as a way to enable fine-tuning at a higher resolution than was used during pre-training. This is computationally efficient and a commonly used technique in the literature. We employed bilinear interpolation even throughout pre-training, to enable training on varying resolutions.

3.3.2 Pre-training Strategy

For pre-training, we split our data in a 80/5/15 train-validation-test split, stratified by patients. For enabling training on variable-sized images, we employed the previously defined DBB strategy. As recommended in the ViT MAE paper [He+22], we chose a fairly high masking ratio of 75% for the pre-training task of image reconstruction, which is visualized in Figure 3.4. We used a hardware batch size of 16[2] and accumulated gradients across every 64 batches, resulting in an effective batch size of 1,024. As an optimizer, we used AdamW [LH19] with a weight decay value of 0.05 and momentum values of $\beta_1 = 0.9, \beta_2 = 0.95$, as proposed in [He+22]. In order to determine a good base learning rate, we employed a grid search on the values $3 \cdot 10^{-3}$, $1 \cdot 10^{-3}$, $3 \cdot 10^{-4}$ and $1 \cdot 10^{-4}$, which we ran on a subset of 10.000 training samples for five epochs each. Out of these learning rates, $1 \cdot 10^{-4}$ performed best. For scheduling the learning rate, we used cosine annealing with a linear warm-up of 5 epochs and a minimum learning rate of $1 \cdot 10^{-5}$. Because pre-training on such a large dataset and images of such high resolutions is slow[3] and we could therefore only train for a small amount of epochs, we implemented a step-wise version of this scheduling policy, in order to get a better learning rate step resolution. The exact training configuration is listed in Table 3.3. We trained for a total of 10 epochs, which took 8 days and 7 hours.

Table 3.3: An overview of the pre-training configuration.

Parameter	Value
Maximum image size	3,072
Number of image channels	1
Validation set size	5%
Test set size	15%
Patch size	48
Mask Ratio	75%
Hardware batch size	16
Gradient accumulation	64
Effective batch size	1,024
Base learning rate	$1 \cdot 10^{-4}$
Loss function	Mean squared error
Optimizer	AdamW
Weight decay	0.05
Momentum	$\beta_1 = 0.9, \beta_2 = 0.95$
Number of epochs	10
Cosine annealing	5 + 5 warm-up

[2] This batch size was chosen primarily with respect to memory constraints.

[3] Mostly because of a slow network-attached storage (NAS) access speed, which hindered us from achieving a high GPU utilization.

3.3.3 Fine-tuning Strategy

For fine-tuning the pre-trained ViT MAE, we employed a linear probing strategy, by discarding the MAE decoder and adding a fully connected layer for classification on top of the MAE encoder's `[CLS]` token. This linear layer thus has an input size corresponding to the hidden size of the encoder and an output size corresponding to the amount of classes for the respective downstream task[4].

We evaluated our models on three different clinical downstream tasks of varying difficulty: anatomical region classification (ARC), foreign material detection (FMD) and fracture detection (FRAC). All labels for the ARC task were extracted directly from the PACS, while those for the other two tasks were created manually by radiologists. For the ARC task, we sampled a subset of 1,000 radiographs. For the other two tasks, we used all available 652 labeled data points. Examples of the different anatomical regions are given in Figure 3.2, examples of the presence and absence of foreign material can be seen in Figure 3.5 and examples of fractures are shown in Figure 3.6. We discarded all samples labeled as *Unsure* by the radiologists. An overview of the class distribution for each downstream task is given in Table 3.4.

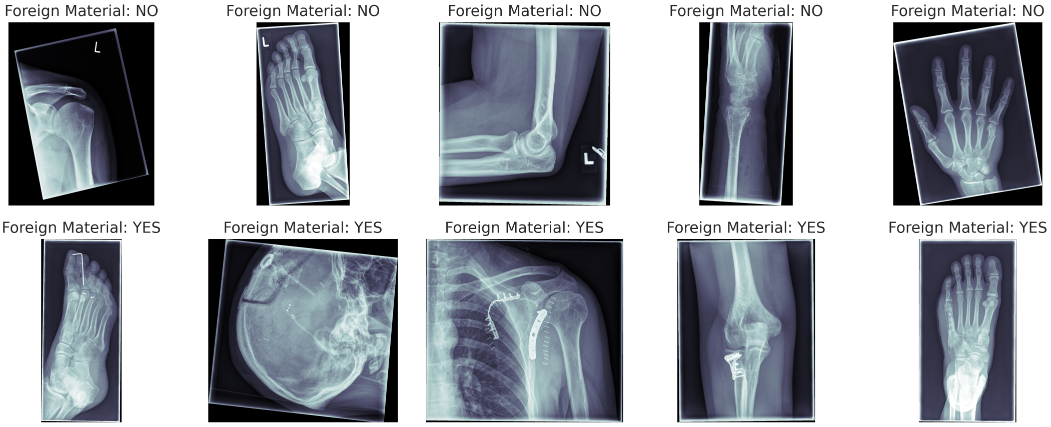

Figure 3.5: A visualization of radiographs with and without foreign material, labeled by radiologists. These samples were used in the FMD downstream task.

We tested different training strategies for fine-tuning; freezing the model, keeping it completely unfrozen, installing the final classification layer on top of a mean pooling of the encoder embeddings or adding it directly on top of the encoder's `[CLS]` token. Ultimately, we implemented the following fine-tuning strategy: For each downstream task, the ViT-B model was initialized with its respective weights—random, ImageNet pre-trained, or MRI pre-trained—and its classifier was replaced by a randomly initialized linear layer on top of the model's `[CLS]` token. During fine-tuning, all weights remained trainable.

[4] In our case either two for foreign material detection and fracture detection or 14 for anatomical region classification.

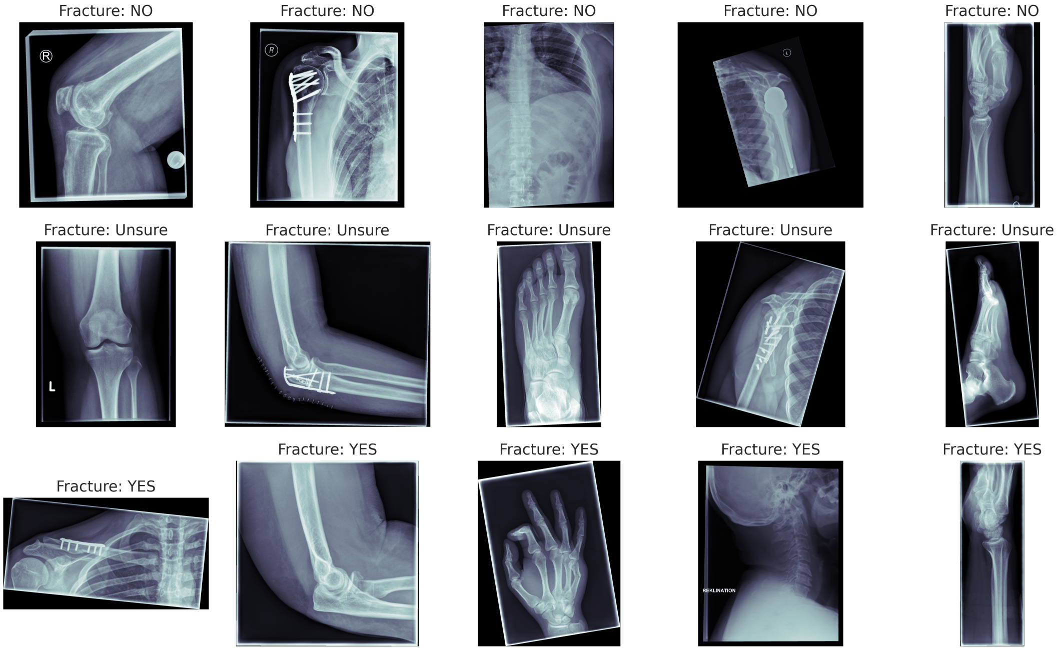

Figure 3.6: A visualization of radiographs with and without fractures, labeled by radiologists. These samples were used in the FRAC downstream task. All samples with the label *Unsure* were discarded.

Table 3.4: An overview of the samples per class for each downstream task.

Downstream Task	Class	# Samples	Percentage
Anatomical region classification (ARC)	Knee	170	17.0%
	Shoulder	115	11.5%
	Elbow	93	9.3%
	Foot	88	8.8%
	Cervical spine	80	8.0%
	Hand	79	7.9%
	Wrist	77	7.7%
	Clavicle	60	6.0%
	Thoracic spine	60	6.0%
	Ribs	48	4.8%
	Scapula	46	4.6%
	Ankle	39	3.9%
	Skull	38	3.8%
	Lumbar spine	7	7.0%
	Total	**1,000**	
Foreign material detection (FMD)	Present	221	33.9%
	Absent	431	66.1%
	Total	**652**	
Fracture detection (FRAC)	Present	212	32.5%
	Absent	440	67.5%
	Total	**652**	

We split the data in a 75/10/15 train-validation-test split, stratified by patients and respective labels for each downstream task. Due to the small dataset sizes, we did not employ our DBB strategy used during pre-training and instead trained using a batch size of one, simply padding each image to a multiple of the respective model's patch size: either 16 for the baselines, or 48 for the pre-trained model. We used gradient accumulation to achieve an effective batch size of 64. As an optimizer, we used AdamW with the same momentum values $\beta_1 = 0.9, \beta_2 = 0.95$ as during pre-training, but without weight decay, as proposed in [Dos+21].

In order to determine a good base learning rate, we performed a grid search on the ARC task, as this task was the easiest to train on and thus led to the biggest observable differences when testing different learning rates for a few epochs. We tested each learning rate for ten epochs on the whole training set of the ARC downstream task. For our pre-trained model, we tested the values $1 \cdot 10^{-3}$, $1 \cdot 10^{-4}$, $3 \cdot 10^{-5}$, $1 \cdot 10^{-5}$ and $1 \cdot 10^{-6}$ and found $3 \cdot 10^{-5}$ to perform best. For the baseline models, we tested the values $1 \cdot 10^{-3}$, $3 \cdot 10^{-4}$, $1 \cdot 10^{-4}$, $1 \cdot 10^{-5}$ and $1 \cdot 10^{-6}$ and found $1 \cdot 10^{-4}$ to perform best.

As proposed in [Dos+21], we used cosine annealing with a linear warm-up across ten epochs and a minimum learning rate of $1 \cdot 10^{-6}$ for scheduling the learning rate during fine-tuning. The annealing occurred across a maximum period of 100 epochs, which was never fully reached due to early stopping. In contrast to pre-training, we configured the scheduler to update the learning rate each epoch rather than each step, as the much smaller dataset sizes allowed for training for more than only a few epochs. We trained all models until convergence, determined by an early stopping policy, monitoring the balanced validation accuracy with a patience of ten epochs.

During fine-tuning, we employed random augmentations to prevent overfitting. The applied augmentations were a random composition of the following:

- Random affine transformation, scaling by $s_i \sim \mathcal{U}(0.9, 1.1)$ and rotating by $\theta_i \sim \mathcal{U}(-20°, 20°)$, applied with probability $p = 0.5$,
- Random horizontal flip, applied with probability $p = 0.5$,
- Random gamma correction by $\gamma = e^{\beta}$ with $\beta \sim \mathcal{U}(-0.5, 0.5)$, applied with probability $p = 0.5$,
- Random Gaussian noise with $\mu = 0$ and $\sigma \sim \mathcal{U}(0, 0.25)$, applied with probability $p = 0.3$,
- Random bias field artifacts [Sud+17] with a maximum magnitude of $n = 0.5$ and an order of 3, applied with probability $p = 0.3$,
- Random Gaussian blur with $\sigma_i \sim \mathcal{U}(0, 2)$, applied with probability $p = 0.3$,
- Random crop to $c \sim \mathcal{U}(0.9, 1.0)$ of the original size, always applied.

These individual augmentations and their random composition, as used during fine-tuning, are visualized in Figure 3.7.

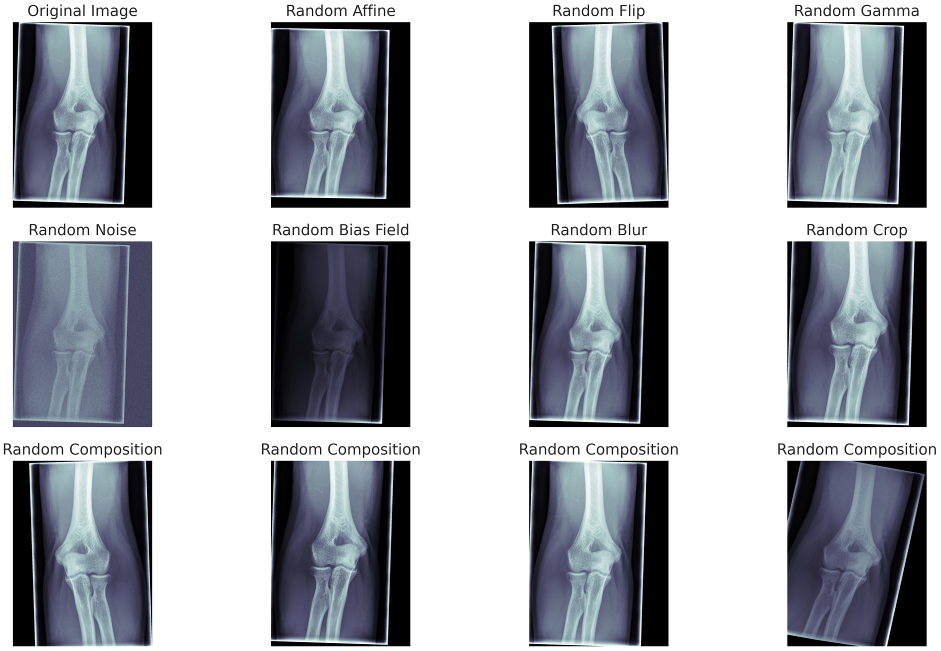

Figure 3.7: A visualization of the different augmentations employed during fine-tuning. The last row shows the random composition of augmentations, as used during fine-tuning.

Besides the weight initialization, the individually tuned learning rates, and the varying patch sizes and number of input image channels, all fine-tuning experiments were conducted under equal conditions. The exact training configurations for fine-tuning and training the baseline models are listed in Table 3.5.

Table 3.5: An overview of the training configurations used for fine-tuning and training the baseline models.

Parameter	Baseline	Fine-tuning
Maximum image size	3,072	3,072
Number of channels	3	1
Validation set size	10%	10%
Test set size	15%	15%
Patch size	16	48
Hardware batch size	1	1
Effective batch size	64	64
Base learning rate	$1 \cdot 10^{-4}$	$3 \cdot 10^{-5}$
Loss function	Cross-entropy	Cross-entropy
AdamW weight decay	0.0	0.0
AdamW momentum	$\beta_1 = 0.9, \beta_2 = 0.95$	$\beta_1 = 0.9, \beta_2 = 0.95$
Max. number of epochs	100	100
Early stopping patience	10	10
Cosine annealing	90 + 10 warm-up	90 + 10 warm-up

3.3.4 Training of Baseline Models

The ViT-B baseline model with pre-trained weights was originally pre-trained by the authors of [Dos+21]. They pre-trained for 90 epochs in a supervised manner on ImageNet-21k, comprising 14 million images at a resolution of 224 × 224 pixels. Afterwards, they fine-tuned on ImageNet-1k, comprising 1 million images, also at a resolution of 224 × 224. Finally, we fine-tuned again on each respective downstream task, as outlined in Subsection 3.3.3.

The ViT-B baseline model with randomly initialized weights was effectively trained from scratch during the fine-tuning phase.

4 Results

This chapter presents the results of our work, focusing on two main contributions: the gains in computational efficiency, achieved by introducing a novel batching strategy, as well as the performance gains on downstream tasks with minimal labels, attained by pre-training. The batching strategy is evaluated by comparing against other possible strategies, while our pre-trained model is evaluated by comparing against the previously defined baseline models on several clinical downstream tasks.

4.1 Impact of Dynamic Batch Binning on Computational Efficiency

As the limiting factor with regards to training duration in all of our experiments was the slow NAS access speed, comparing the number of operations performed, rather than the time spent training, provides a more meaningful evaluation of different batching strategies.

We thus base our comparison on the total amount of tokens fed into the model during pre-training, using either the minimum viable padding to the model's patch size (thereby enforcing a batch size of one), padding all images to a fixed resolution of 3,072 × 3,072 pixels, or our proposed Dynamic Batch Binning (DBB) strategy. As can be seen in Table 4.1, our solution introduced a padding token overhead of 19% when compared to the theoretical minimum, while padding to a fixed resolution would have introduced an overhead of 184%. When directly comparing our approach to the fixed resolution approach, savings in total processed tokens are 58%, i.e. we only processed 42% of the tokens that would have been processed when using a fixed image size instead. Of course, this ratio could be improved even further by defining a larger amount of smaller bins in a trade-off against efficient batch processing / bin fragmentation.

When translating this input token overhead into computational overhead, one has to consider the different operations performed during ViT MAE pre-training. First, each patch/token is linearly projected into a fixed-size embedding vector. The compute for this step scales linearly with the number of tokens. According to the fixed masking ratio, a proportion of the tokens is masked and the visible patches are processed by the Transformer encoder, which applies a series of self-attention and feed-forward operations. Due to the self-attention mechanism, the compute required for the Transformer encoder scales quadratically with the number of tokens. The compute for the decoder, i.e. for reconstructing the input image, also scales quadratically with the number of tokens. This means that as the total input token count increases, the compute required grows approximately quadratically, mostly due to the Transformer encoder. Thus, our approach saved roughly 82% of total compute when compared to the fixed resolution approach. The computational overhead introduced by different batching strategies is compared in more depth in Table 4.1.

Table 4.1: A comparison of input token overhead and computational overhead when employing different batching strategies during pre-training. All values are based on a single epoch on the full pre-training dataset of 639,877 images.

Batching Strategy	Total Input Tokens	Processed Tokens	Total Compute
Padded to patch size	$9.22 \cdot 10^8$	100%	100%
Fixed image size	$2.62 \cdot 10^9$	284%	807%
DBB	$1.10 \cdot 10^9$	119%	142%

4.2 Impact of Pre-training on Downstream Task Performance

In this section, we compare the downstream task performance of our pre-trained model against that of a ViT-B without pre-training, i.e. with randomly initialized weights, and that of a ViT-B which was pre-trained extensively on the ImageNet-21k dataset. For comparing the performances of the tested models, we measured each model's test set accuracy, as well as balanced test set accuracy, for each downstream task. The balanced accuracy is calculated as the mean of sensitivity and specificity and is a more informative metric than absolute accuracy when dealing with class imbalances.

Pre-training Results After pre-training for 10 epochs, the ViT MAE B achieved a final mean squared error (MSE) reconstruction loss of 0.0523 on the test set. Exemplary reconstructions of a masked radiograph from the validation set after different pre-training epochs are shown in Figure 4.1. However, assessing the pre-training performance solely based on the achieved reconstruction loss or the visual quality of the reconstructions might be misleading, as the ViT MAE decoder is deliberately designed in a lightweight fashion. The reason behind this architectural choice is that the image reconstruction task is not the final goal of pre-training and reconstruction has to remain a challenging task in order to train the encoder to generate embeddings in a maximally effective way[1].

Fine-tuning Results The utility of pre-trained models is best judged by evaluating their performance on downstream tasks. To this end, we defined three downstream tasks in Subsection 3.3.3, namely anatomical region classification (ARC), foreign material detection (FMD), and fracture detection (FRAC).

[1] This is also the reason why reconstructions in the literature are typically blurry, as can also be observed in [He+22; Xin+23].

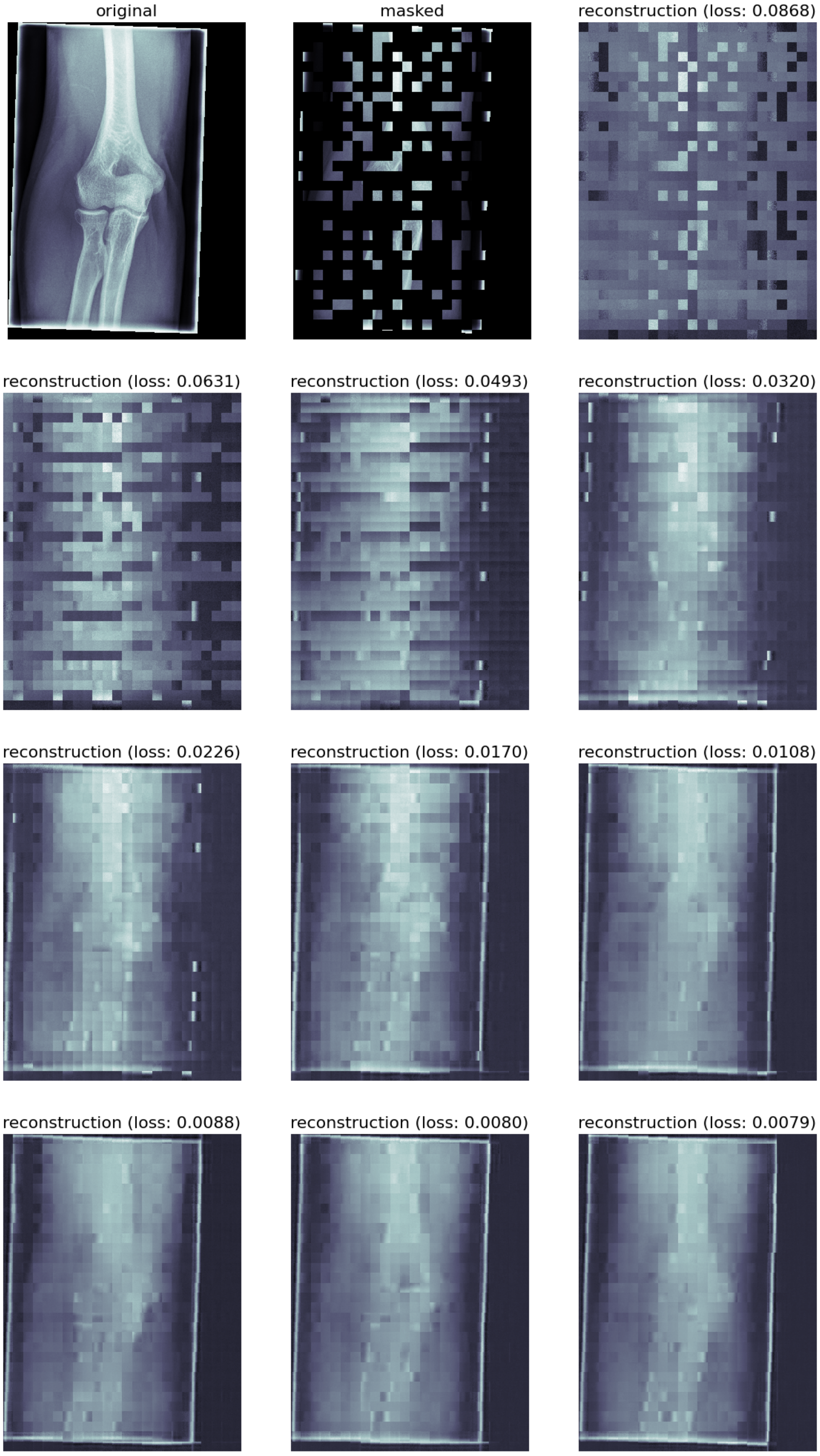
original
masked
reconstruction (loss: 0.0868)
reconstruction (loss: 0.0631)
reconstruction (loss: 0.0493)
reconstruction (loss: 0.0320)
reconstruction (loss: 0.0226)
reconstruction (loss: 0.0170)
reconstruction (loss: 0.0108)
reconstruction (loss: 0.0088)
reconstruction (loss: 0.0080)
reconstruction (loss: 0.0079)

Figure 4.1: Exemplary ViT MAE B reconstructions of a sample image from the validation set after 1-10 epochs of pre-training. The image was masked by 75%. The printed reconstruction loss is the MSE compared to the original input image.

On the ARC task, the ViT-B model pre-trained on ImageNet-21k achieved the highest balanced accuracy (92.21%) and absolute accuracy (93.57%). The ViT MAE B model pre-trained on radiographs outperformed the randomly initialized ViT-B (73.57% vs. 56.71%), but did not match the performance of the model pre-trained on the ImageNet-21k dataset. On the FMD task, the ImagetNet-21k pre-training similarly yielded the highest accuracy (92.88% balanced, 94.0% absolute accuracy). Again, the ViT MAE B model pre-trained on radiographs outperformed the randomly initialized ViT-B (57.99% vs. 46.88%), but also did not match the performance achieved through ImageNet-21k pre-training. On the FRAC task however, the ViT MAE B model pre-trained on radiographs achieved the best balanced accuracy (58.46%), outperforming the ImageNet-21k pre-training (56.86%), as well as the randomly initialized ViT-B (50.0%). A direct comparison of the performances on the downstream tasks is given in Table 4.2.

Table 4.2: Direct comparison of the downstream task performance of the randomly initialized ViT-B model without any pre-training, the ViT-B model pre-trained on the ImageNet-21k dataset, and our ViT MAE B model pre-trained on radiograph data. The reported values are balanced accuracies on the test set. Values in parentheses are the absolute test set accuracies, without balancing.

Downstream Task	ViT-B	ImageNet-21k ViT-B	ViT MAE B
ARC	56.89% (65.71%)	**92.21%** (93.57%)	73.57% (82.86%)
FMD	46.88% (60.00%)	**92.88%** (94.00%)	57.99% (68.00%)
FRAC	50.00% (65.35%)	56.86% (59.41%)	**58.46%** (62.38%)

5 Discussion & Conclusion

5.1 Discussion of Results

Our results demonstrate that pre-training in general significantly improves downstream task performance in settings with minimal labels. Moreover, domain-specific radiograph pre-training further improves performance upon vastly more extensive general pre-training in specialized and complex downstream tasks.

The strong performance of ImageNet-21k pre-training on the ARC and FMD tasks suggests that the ability to discern general visual features obtained during extensive ImageNet-21k pre-training benefit the model in such tasks, where discriminatory features are typically large and easy to identify. The increased performance of radiograph pre-training on the FRAC task, on the other hand, suggests that on tasks relying heavily on a deeper understanding of the domain and demanding a higher focus on fine-grained medical details, domain-specific pre-training is increasingly beneficial.

Comparing the amount of training data and training iterations used for both types of pre-training supports this hypothesis. For example, during ImageNet-21k pre-training, the model was trained for 90 epochs on 14 million images showing a wide range of subjects, resulting in $1.26 \cdot 10^9$ training iterations[1]. This scale and diversity of training data provided the model with a broad range of visual features to learn from, allowing it to generalize well to other tasks that can be solved by a good general visual understanding. In contrast, our model was trained for ten epochs on 600,000 radiographs, resulting in $6 \cdot 10^6$ training iterations, amounting to roughly 5% of total training samples and 0.5% of total training iterations compared to the ImageNet-21k pre-training. Due to this considerable disparity in pre-training volume, it might be worthwhile to explore how an even larger radiograph pre-training influences the obtained results.

5.2 Conclusion

The primary goal of this work was to facilitate the utilization of large-scale medical imaging datasets and tackle the labeling bottleneck in medical imaging. Our aim was to explore the potential of employing SSL techniques on this task by pre-training a ViT MAE on our large-scale real-world clinical dataset and evaluating the pre-trained model on several clinical downstream tasks.

Our results show that MAEs can effectively capture and extract the fine-grained features essential for solving complex medical imaging tasks. Specifically, we demonstrated

[1] Without considering the additional ImageNet-1k fine-tuning.

that our ViT MAE pre-trained on radiograph data achieved a superior performance compared to a baseline supervised ViT in low-data scenarios on all three tested downstream tasks, namely anatomical region classification (ARC), foreign material detection (FMD), and fracture detection (FRAC). Furthermore, the ViT MAE pre-trained on radiograph data yielded an increase in accuracy on the FRAC task compared to a more general pre-training on ImageNet-21k, despite being trained on only 5% of training samples and for only 0.5% of training iterations. This finding highlights the potential of domain-specific pre-training in the medical imaging domain.

Additionally, we provide several technical contributions to significantly improve training efficiency and lower computational costs for large-scale training on real-world medical imaging datasets. Our proposed Dynamic Batch Binning (DBB) strategy enables efficient training on image datasets with a high variability in image resolutions, saving roughly 80% of operations, compared to a naive fixed-size resolution approach.

6 Further Research

Our research generated numerous conceptual insights and potential directions for further experiments. We also faced several challenges and encountered certain limitations we did not yet fully resolve.

6.1 Engineering Challenges

We faced several engineering challenges, mostly concerning the pre-training task and mainly due to the large-scale, real-world nature of our data.

For example, because of the large amount and high resolutions of the images in our dataset, it was not possible to store the whole 5.5 TB of training data locally during pre-training. Therefore, we had to stream the data from a NAS during training, slowing down data loading by a factor of about 30-40×, which hindered us from achieving a high GPU utilization, making our optimizations with regards to computational efficiency and faster training speeds largely theoretical rather than practical. We tested various alternative formats for storing the data, comparing conversion time, space efficiency and opening speeds. Nevertheless, the true bottleneck of NAS access speed was not easily resolved by any of our attempts.

Furthermore, training on large and variable resolution images introduced several challenges at once. We successfully solved the challenge of training without using excessive padding by introducing the aforementioned DBB strategy and bilinearly interpolating the positional encodings. However, the changes in input sizes across batches led to PyTorch repeatedly recompiling the model during the first forward passes when training a compiled model. This was not easily resolved by passing an initial batch of maximum resolution images through the model when starting the training.

Training on extraordinarily high resolutions presented us with several trade-offs with regards to accuracy, efficiency and memory constraints, for example between possible patch sizes, intermediate and hidden sizes, batch sizes and masking ratios. Using gradient accumulation, we could alleviate the limitation of a maximum possible batch size.

6.2 Further Research

Investigation of Scaling Behavior with Increased Computational Resources. While our pre-trained ViT MAE model demonstrated strong performance on the FRAC task and showed good results on the other two tasks in the low data regime, an extended pre-training period could further enhance these results. Particularly the comparison with the superior results achieved by general ImageNet-21k pre-training on the more general downstream tasks of ARC and FMD, as well as the disparity of pre-training volume compared to ImageNet-21k pre-training, indicate the potential benefits of an even more extensive radiograph pre-training. Future research could explore training for additional epochs, potentially closing the gap to ImageNet pre-training on tasks depending mostly on a good general visual understanding and perhaps also further improving the lead on domain-specific tasks like FRAC.

Comprehensive Exploration of Hyperparameters and Associated Trade-offs. There are several pre-training hyperparameters we would like to test in more depth, for example comparing how downsampling the pre-training data affects downstream task performance of the pre-trained model. Another hyperparameter we would like to tune in more detail is the ViT MAE's masking ratio. Although we tested multiple patch sizes for the ViT MAE model, we would like to conduct more experiments on this hyperparameter as well, potentially even running the whole pre-training / fine-tuning pipeline on models of various patch sizes, to determine whether a higher resolution of patches helps the model improve downstream tasks performance.

All of these experiments focus on the trade-off of efficiency, compute and memory constraints with information passed through the model.

Label Scarcity in Medical Imaging. Another promising solution to the problem of label scarcity in medical imaging is to generate pseudo-labels. As there are radiologist reports accompanying a subset of the radiographs in our dataset, one approach to generate such pseudo-labels could be to use simple techniques like regular expressions for extracting high-probability ground truth labels from these reports. These high-probability ground truth labels could be checked more cheaply by radiologists than generating labels from scratch.

Building upon these high-probability ground truth labels, NLP techniques could be employed to extract further labels from the reports. There are several viable routes, like first translating the German reports to English using a (medical) translation model and subsequently using an English medical NLP model like e.g. RadBERT [Yan+22], fine-tuning a model like BERT [KT19], or even testing few-shot or zero-shot performance of a general large language model like e.g. Llama 3 [Dub+24]. The previously extracted high-probability ground truth labels could serve as a foundation for fine-tuning or evaluating the produced NLP pseudo-labels. Using or training a German end-to-end model is also possible, but we believe the outlined approach to be superior, as translation models have become highly effective and English models typically outperform German ones due to more extensive training data and greater research focus. Furthermore, medical root words are often similar in German and English.

Additional Research Directions. Given the high-resolution nature of the images in our dataset, exploring specialized variants of the ViT MAE, such as the Swin MAE architecture, or alternatives like I-JEPA, could be another promising direction for further research.

Another aspect of training on medical data like radiographs, which further research could focus on, is employing local-global training strategies, e.g. as was used in [Val+21], allowing Transformers to operate on the whole image for global features and focus on single patches for local features at the same time.

List of Figures

List of Tables

Bibliography

[Ass+23] M. Assran, Q. Duval, I. Misra, P. Bojanowski, P. Vincent, M. Rabbat, Y. LeCun, and N. Ballas. “Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture.” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 15619–15629.

[Bey+23] L. Beyer, P. Izmailov, A. Kolesnikov, M. Caron, S. Kornblith, X. Zhai, M. Minderer, M. Tschannen, I. Alabdulmohsin, and F. Pavetic. “FlexiViT: One Model for All Patch Sizes.” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 14496–14506.

[Che+20] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. “A Simple Framework for Contrastive Learning of Visual Representations.” In: *International Conference on Machine Learning*. PMLR. 2020, pp. 1597–1607.

[Dai+23] Y. Dai, F. Liu, W. Chen, Y. Liu, L. Shi, S. Liu, Y. Zhou, et al. “Swin MAE: Masked Autoencoders for Small Datasets.” In: *Computers in Biology and Medicine* 161 (2023), p. 107037.

[Deh+24] M. Dehghani, B. Mustafa, J. Djolonga, J. Heek, M. Minderer, M. Caron, A. Steiner, J. Puigcerver, R. Geirhos, I. M. Alabdulmohsin, et al. “Patch n’pack: NaViT, a Vision Transformer for any Aspect Ratio and Resolution.” In: *Advances in Neural Information Processing Systems* 36 (2024).

[Den+09] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. “Imagenet: A Large-scale Hierarchical Image Database.” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE. 2009, pp. 248–255.

[Dos+21] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.” In: *International Conference on Learning Representations*. ICLR. 2021.

[Dub+24] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, et al. “The Llama 3 Herd of Models.” In: *arXiv preprint arXiv:2407.21783* (2024).

[HA15] W. Huda and R. B. Abrahams. “X-Ray-Based Medical Imaging and Resolution.” In: *American Journal of Roentgenology* 204.4 (2015), W393–W397.

[He+22] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. “Masked Autoencoders are Scalable Vision Learners.” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE. 2022, pp. 16000–16009.

[Juy+24] D. Juyal, H. Padigela, C. Shah, D. Shenker, N. Harguindeguy, Y. Liu, B. Martin, Y. Zhang, M. Nercessian, M. Markey, et al. “PLUTO: Pathology-Universal Transformer.” In: *arXiv preprint arXiv:2405.07905* (2024).

[KT19] J. D. M.-W. C. Kenton and L. K. Toutanova. “Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding.” In: *Proceedings of NAACL-HLT*. Vol. 1. 2019, p. 2.

[Leh+03] T. M. Lehmann, H. Schubert, D. Keysers, M. Kohnen, and B. B. Wein. “The IRMA code for Unique Classification of Medical Images.” In: *Medical Imaging 2003: PACS and Integrated Medical Information Systems: Design and Evaluation*. Vol. 5033. SPIE. 2003, pp. 440–451.

[LH19] I. Loshchilov and F. Hutter. “Decoupled Weight Decay Regularization.” In: *7th International Conference on Learning Representations*. 2019.

[Liu+21] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. “Swin Transformer: Hierarchical Vision Transformer using Shifted Windows.” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 10012–10022.

[Liu+22] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, Y. Wei, J. Ning, Y. Cao, Z. Zhang, L. Dong, et al. “Swin Transformer V2: Scaling up Capacity and Resolution.” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 12009–12019.

[Nat24] National Electrical Manufacturers Association. *NEMA PS3 / ISO 12052, Digital Imaging and Communications in Medicine (DICOM) Standard*. `http://www.dicomstandard.org/`. 2024.

[Pas+19] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. “Pytorch: An Imperative Style, High-performance Deep Learning Library.” In: *Advances in Neural Information Processing Systems* 32 (2019).

[Sud+17] C. H. Sudre, M. J. Cardoso, S. Ourselin, A. D. N. Initiative, et al. “Longitudinal Segmentation of Age-related White Matter Hyperintensities.” In: *Medical Image Analysis* 38 (2017), pp. 50–64.

[Val+21] J. M. J. Valanarasu, P. Oza, I. Hacihaliloglu, and V. M. Patel. “Medical Transformer: Gated Axial-Attention for Medical Image Segmentation.” In: *Medical Image Computing and Computer Assisted Intervention*. 2021, pp. 36–46.

[Var+24] A. Varma, S. Shit, C. Prabhakar, D. Scholz, H. B. Li, D. Rueckert, B. Wiestler, et al. “VariViT: A Vision Transformer for Variable Image Sizes.” In: *Medical Imaging with Deep Learning*. 2024.

[Vas+17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention Is All You Need." In: *Proceedings of the International Conference on Neural Information Processing Systems*. NeurIPS. 2017, pp. 6000–6010.

[Wan+17] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers. "ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases." In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2097–2106.

[Wol+23] A. Wollek, S. Hyska, B. Sabel, M. Ingrisch, and T. Lasser. "Higher Chest X-ray Resolution Improves Classification Performance." In: *arXiv e-prints* (2023), arXiv–2306.

[Xin+23] X. Xing, G. Liang, C. Wang, N. Jacobs, and A.-L. Lin. "Self-supervised Learning Application on COVID-19 Chest X-ray Image Classification using Masked Autoencoder." In: *Bioengineering* 10.8 (2023), p. 901.

[Yan+22] A. Yan, J. McAuley, X. Lu, J. Du, E. Y. Chang, A. Gentili, and C.-N. Hsu. "RadBERT: Adapting Transformer-based Language Models to Radiology." In: *Radiology: Artificial Intelligence* 4.4 (2022), e210258.

[Zho+23] L. Zhou, H. Liu, J. Bae, J. He, D. Samaras, and P. Prasanna. "Self Pre-training with Masked Autoencoders for Medical Image Classification and Segmentation." In: *International Symposium on Biomedical Imaging*. IEEE. 2023, pp. 1–6.