information

Article

Shape Matters: Detecting Vertebral Fractures Using Differentiable Point-Based Shape Decoding

Hellena Hempe [1], Alexander Bigalke [1,2] and Mattias Paul Heinrich [1,*]

[1] Institute of Medical Informatics, University of Lübeck, 23562 Lübeck, Germany; hellena.hempe@uni-luebeck.de (H.H.)
[2] Dräger, Drägerwerk AG & Co. KGaA, 23558 Lübeck, Germany
* Correspondence: mattias.heinrich@uni-luebeck.de

Abstract: Background: Degenerative spinal pathologies are highly prevalent among the elderly population. Timely diagnosis of osteoporotic fractures and other degenerative deformities enables proactive measures to mitigate the risk of severe back pain and disability. Methods: We explore the use of shape auto-encoders for vertebrae, advancing the state of the art through robust automatic segmentation models trained without fracture labels and recent geometric deep learning techniques. Our shape auto-encoders are pre-trained on a large set of vertebrae surface patches. This pre-training step addresses the label scarcity problem faced when learning the shape information of vertebrae for fracture detection from image intensities directly. We further propose a novel shape decoder architecture: the point-based shape decoder. Results: Employing segmentation masks that were generated using the TotalSegmentator, our proposed method achieves an AUC of 0.901 on the VerSe19 testset. This outperforms image-based and surface-based end-to-end trained models. Our results demonstrate that pre-training the models in an unsupervised manner enhances geometric methods like PointNet and DGCNN. Conclusion: Our findings emphasize the advantages of explicitly learning shape features for diagnosing osteoporotic vertebrae fractures. This approach improves the reliability of classification results and reduces the need for annotated labels.

Keywords: computer-aided diagnosis; deep learning; shape analysis; auto-encoder; spine; vertebral body fractures

Citation: Hempe, H.; Bigalke, A.; Heinrich, M.P. Shape Matters: Detecting Vertebral Fractures Using Differentiable Point-Based Shape Decoding. *Information* **2024**, *15*, 120. https://doi.org/10.3390/info15020120

Academic Editor: Xiaoshuang Shi

Received: 17 January 2024
Revised: 8 February 2024
Accepted: 9 February 2024
Published: 19 February 2024

1. Introduction

Degenerative pathology of the spine such as osteoporotic fractures of vertebral bodies, spinal canal stenosis, spondylolisthesis, and other degenerative deformations of vertebrae are a common healthcare risk in the elderly population [1]. Early detection of degenerative diseases may enable preventative measures to reduce the risk of chronic severe back pain and disability [2]. In recent years, deep learning has emerged as a powerful tool for many medical applications. Advances in deep learning in computer-aided diagnosis have paved the way for more timely interventions and improved patient outcomes. In this study, we specifically focus on the intersection of deep learning and the diagnosis of vertebral body fractures.

So far, most deep learning-based approaches for classification of vertebral body fractures are trained on intensity images in an end-to-end manner. Even after the VerSe dataset/benchmark for segmentation and localization of vertebrae was introduced in 2019 and 2020 [3–5], most introduced methods only rely on the localization of vertebrae instead of leveraging the available segmentation masks [6–8]. The amount of available ground truth segmentation masks of vertebrae was further increased by the TotalSegmentator dataset [9,10]. To date, only few recently proposed methods [11,12] employ the information provided by spinal segmentation masks in diagnostic downstream tasks.

We identified three relevant challenges that are not sufficiently addressed by existing methods and have so far prevented a wider clinical adoption of vertebral fracture detection.

(a) Image-based classifiers are prone to deterioration under domain shifts; i.e., they are limited in their adaptability for variations of the image intensity, scanner acquisition settings, and other shifts in the data distribution. Furthermore, 3D convolutional neural networks (CNN) trained in a fully-supervised setting tend to overfit and require a larger amount of labeled data of sufficient quality. (b) Surface-based geometric learning models [13,14] have so far been less expressive than 3D CNNs and so far do not achieve the required accuracy on limited labeled data. (c) Shape encoder–decoder architectures [11,12] may help to overcome the label scarcity by shifting a larger proportion of the training to an unsupervised setting for which even healthy spines can be employed, but they may still fail to learn representative latent space representations, due to an over-pronounced weight of the decoder (that is later discarded for the classification task).

We suggest making the following improvements: (a) Instead of training image-based classifiers, we propose to leverage information from a preceding segmentation model and directly operate on shape information of vertebrae (surface contour) for the classification task. This allows trained deep learning models to be independent of shifts in image intensities and moves the demand for labeled data away from the classification task. (b) Leveraging the vast amount of available segmentation masks, unsupervised learning can help geometric learning models overcome problems related to limited labeled data. (c) To ensure a more representative latent space representation, we introduce a novel decoder architecture that ensures that most learned parameters are located in the encoder model and thus relevant for the classification task.

Contributions: (1) We believe that the effectiveness of recently proposed auto-encoder (AE) models is limited due to the sub-optimal design of encoder–decoder components. Therefore, we perform an in-depth analysis of the effectiveness of various AE architectures for the diagnosis of vertebral body fractures. (2) For this purpose, we develop a modular encoder–decoder framework that allows arbitrary combinations of point- and image-based encoder and decoder architectures. (3) To address the problem of over-pronounced weights in the decoder, we designed a novel point-based shape decoder model that employs a differentiable sampling step to bridge the gap between point- and voxel-based solutions. (4) By performing extensive experiments to analyze various combinations of encoder and decoder architectures, we verified that the detection of vertebral fractures, which by definition is a geometric problem, is better solved using shape-based methods compared to image-intensity-based ones. (5) The results of our experiments furthermore demonstrate the particular advantages of employing our novel point-based shape decoder compared to other models.

Outline: In the subsequent Section 2, we provide a comprehensive review of the related literature, emphasizing the existing disparity between image- and point-based analyses. Following this, our comprehensive shape-based classification framework (Section 3.1), incorporating the proposed point-based shape decoder (Section 3.2), is meticulously detailed in methods Section 3. The experiments and results Section 4 elaborates on our setup, encompassing data particulars and implementation details (Section 4.1). Afterwards, the results of our experiments are presented in Sections 4.2 and 4.3, followed by a thorough discussion of the outcomes in Section 5 and a concise summary of our findings in Section 6.

2. Related Works

Radiologists commonly assess osteoporotic fractures using the Genant score [15]. The Genant score is a semiquantitative measurement by which a fracture is defined by the height reduction of the vertebral body (in lateral X-Ray) and categorized into 0—healthy, 1—mildly fractured, 2—moderately fractured, and 3—severely fractured. In 3D computed tomography (CT) scans, this is more complicated as an optimal slice for measuring the height difference needs to be determined.

Recent approaches: When categorizing recently proposed methods for computer-aided detection of vertebral compression fractures, we can observe that most recently proposed methods operate directly on the CT intensity data, e.g., [6,7,16]. The method

proposed by Nicolaes et al. [6] suggests a two-step approach that detects vertebral fractures in CT scans using a voxel-based CNN to perform voxel-level classification on patches containing vertebrae. Yilmaz et al. [7] present a pipeline in which the vertebrae are firstly detected using a hierarchical neural network and in which secondly a CNN is applied on the identified patches that are extracted from the original CT scan. The grading loss as proposed by Husseini et al. [16] addresses the problem of severe data imbalance and minor differences in appearance between healthy and fractured vertebrae in CT scans by exploring a representation learning-based approach that was inspired by Genant grades. These methods do not exploit the shape information provided by the segmentation mask. Consequently, the shape of the vertebrae and the associated fracture status need to be learned implicitly, which leads to a complication of the learning process.

Recent approaches using self-supervised learning: As a remedy, two related previous works aim to explicitly use the shape of vertebrae. Firstly, the Spine-VAE [11] employs the masked image data of vertebrae as input to a conditioned VAE to capture the large variations of healthy and fractured vertebra shapes. The conditioning is performed by concatenating information about the labels of the shapes corresponding to the vertebra (T1-L5 split into five groups) as one-hot-encoded vector. After training the VAE until the reconstruction loss converges, the encoder model is further employed in conjunction with a multilayer perceptron to classify vertebral body fractures. Secondly, the purely geometric approach proposed by Sekuboyina et al. [12] employs a surface point-based probabilistic AE to learn shape representations of vertebrae. The task of detecting vertebral fractures is then treated as an out-of-distribution detection task by computing the reconstruction error based on a model that was only trained on healthy vertebrae subjects. Both methods rely on accurate segmentation masks of vertebrae during test time and do not involve unsupervised pre-training on a larger separate dataset to learn shape features.

Geometric Learning: In many computer vision applications, geometric deep learning methods play a crucial role in extracting meaningful features from 3D data by capturing spatial relationships for shape analysis. Notably, two state-of-the-art techniques in this domain are PointNet [13] and DGCNN (Dynamic Graph Convolutional Neural Network) [17]. PointNet stands out for its ability to process unstructured point clouds directly, showcasing robust performance in tasks like shape recognition and segmentation. DGCNN, on the other hand, utilizes dynamic graph structures to capture both local and global geometric relationships within point cloud data, enhancing the model's capacity to discern intricate patterns. These geometric deep learning approaches have significantly advanced the accuracy and efficiency of medical image analysis, particularly in scenarios where a nuanced understanding of spatial relationships and complex structures within volumetric data is essential. Geometric learning approaches are already being applied in medical imaging tasks, e.g., assessment for spinal curvature in X-ray [18] and detection of pedicles for spinal surgery in CT scans [19]. Furthermore, the required landmarks for the measurement of vertebral height in 2D lie on the surface of the vertebral body in 3D, which is why we believe that geometric learning approaches are beneficial for learning the important shape features from the contour of vertebrae.

3. Methods

To investigate the potential of shape features for detecting vertebral fractures, we explore several encoder–decoder architectures and deploy their latent space vector for classification. In particular, we compare combinations of encoder and decoder trained to reconstruct shape features of vertebrae based on the surface of their respective segmentation mask. The training of our AE architecture comparison pipeline is split into two stages. In the first stage, we train our AE models in an unsupervised manner on a large-scale dataset without classification labels or a specific high occurrence of spine diseases to generate meaningful shape features in the latent space. During the second stage, we employ the generated shape features of the encoder (freezing the encoder layers) and train an MLP for detection of fractures based on these shape features on a smaller labeled dataset.

The problem setup can be defined as follows: Based on multi-label segmentation masks of vertebrae performed on a CT scan, extract patches $\mathcal{S}_{in} \in \mathbb{R}^{D \times H \times W}$ of the segmentation mask contour that is computed using an edge-filter. For geometric learning methods, we extract a point cloud representation $\mathcal{P}_{in} \in \mathbb{R}^{3 \times N}$ of N 3D points by employing a threshold on the grid. Firstly, train an AE model comprising a shape encoder f and a shape decoder g on an unsupervised reconstruction task by minimizing the smooth L1-Loss of the Chamfer distance between the input S_{in} or P_{in} and the prediction S_{out} or P_{out}. Secondly, using the latent space vector $\boldsymbol{z} \in \mathbb{R}^M$ computed by the shape encoder f as input, train an MLP as classifier h to predict the fracture status (healthy or fractured) by minimizing the cross-entropy loss function between the target class c_{target} and the predicted class c_{pred}. As encoders we employ 3 approaches, a simple convolutional encoder f_{conv}, a point encoder f_{point} based on the PointNet architecture [13] and a graph encoder f_{graph} based on the DGCNN [17]. As decoders, we employ 3 approaches, a convolutional decoder g_{conv}, our point decoder g_{point}, and a novel point-based shape decoder $g_{pbShape}$. The resulting framework thus comprises 9 combinations of encoder–decoder architectures. The main idea of the general framework for the employed image- or point-based vertebrae auto-encoder (AE) framework is depicted in Figure 1 Section A. Our newly proposed decoder architecture is visualized in Figure 2 Section B.

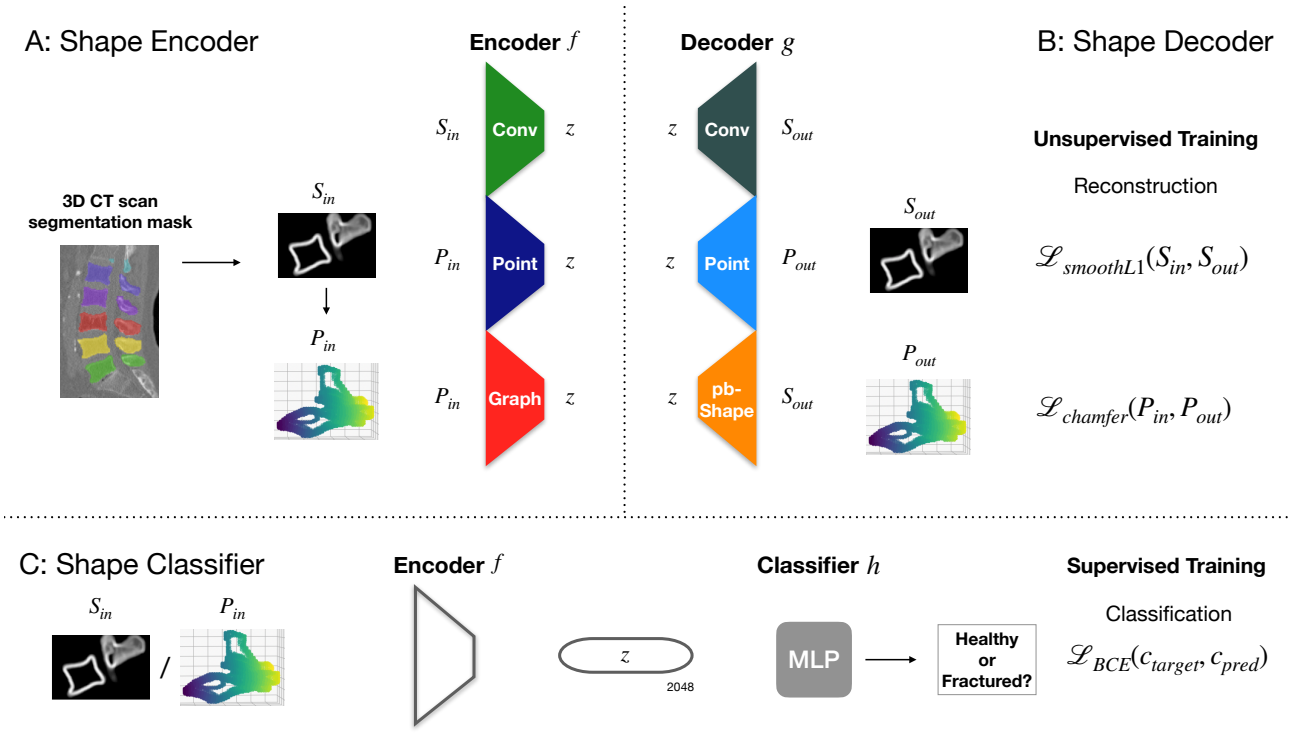

Figure 1. To determine the most suitable architecture for our task, we employ combinations of several encoder–decoder architectures including traditional convolutional methods and geometric methods. The AEs are trained to reconstruct either a point cloud representation or a volumetric surface representation of vertebrae, which are derived from the previously computed segmentation mask. As **Shape Encoder** (**A**), we employ a convolutional method, as well as a point-based and a graph-based method to predict the embedding z. As **Shape Decoder** (**B**), we employ a convolutional method as well as a point-based method and propose a novel point-based shape decoder. The **Shape Classifier** (**C**) is then trained separately on the embedding z for each encoder–decoder combination using the same multilayer perceptron (MLP) model. Note that only the weights of the MLP are trained in a supervised manner, whereas the weights of the encoder are fixed.

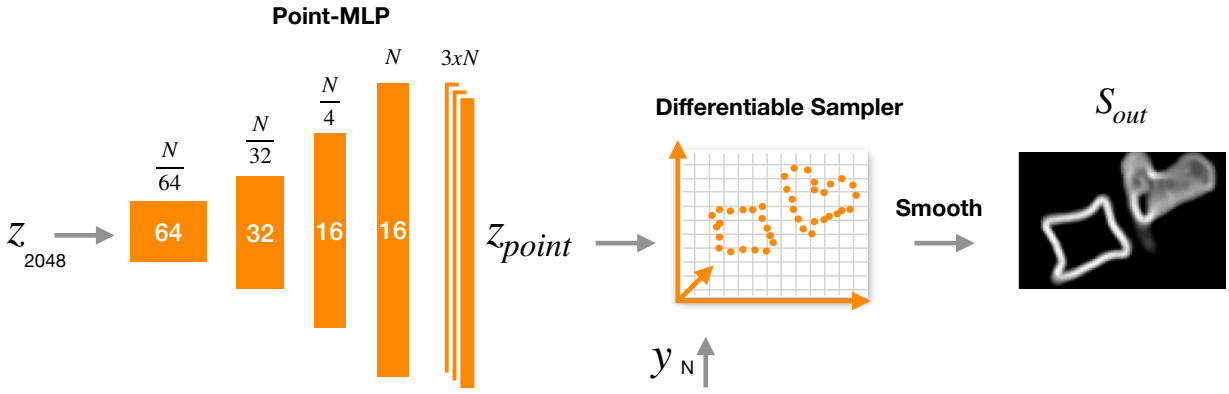

Figure 2. Point-based shape decoder: From the embedding vector z, a point representation of N key points is computed using an MLP. The layers each consist of a 1D convolution with the channel size denoted by white font within the blocks, InstanceNorm and ReLU. The number on top of the blocks denotes the size of the dimensionality of the point cloud. Afterwards, a differentiable sampling operation is applied on the key points to obtain a volumetric representation. This step requires N additional parameters $\boldsymbol{y}$.

3.1. Architectural Building Blocks

■ **Convolutional encoder:** Given a 3D surface patch $\mathcal{S}_{in} \in \mathbb{R}^{D \times H \times W}$ of depth $\boldsymbol{D}$, height $\boldsymbol{H}$, and width $\boldsymbol{W}$, extracted from an automatic multi-label 3D CT segmentation [9,10] that is separated into binary individual vertebrae contours, we aim to encode the vertebral shape into a low-dimensional embedding $\boldsymbol{z} \in \mathbb{R}^M$ with $\boldsymbol{M} = \mathbf{2048}$ using a fully convolutional encoder network $f_{conv}(\mathcal{S}_{in})$ parameterized by trainable weights $\boldsymbol{w}_{f_{conv}}$.

■ **Point encoder:** Based on the 3D surface patches $\mathcal{S}_{in}$, we extract a 3D point cloud $\mathcal{P}_{in} \in \mathbb{R}^{3 \times N}$ where $\boldsymbol{N}$ corresponds to the number of key points in the point cloud that are sampled from voxel representation by applying a threshold on the values in the voxel grid. Using a PointNet [13] model $f_{point}(\mathcal{P}_{in})$ with weights $\boldsymbol{w}_{f_{point}}$, we generate a low-dimensional embedding $\boldsymbol{z} \in \mathbb{R}^M$ with $\boldsymbol{M} = \mathbf{2048}$.

■ **Graph encoder:** As for the point encoder, the graph encoder utilizes an extracted 3D surface point cloud $\mathcal{P}_{in} \in \mathbb{R}^{3 \times N}$ for each individual vertebra. We employ a DGCNN [17] $f_{graph,k}(\mathcal{P}_{in})$ with parameter $\boldsymbol{k}$ for the k-Nearest Neighbor (kNN) graphs and weights $\boldsymbol{w}_{f_{graph}}$ to compute an embedding $\boldsymbol{z} \in \mathbb{R}^M$ with $\boldsymbol{M} = \mathbf{2048}$.

■ **Convolutional decoder:** After generating the embedding $\boldsymbol{z}$, a convolutional decoder $g_{conv}(\boldsymbol{z})$ with weights $\boldsymbol{w}_{g_{conv}}$ is used to map $\boldsymbol{z}$ back into $\mathcal{S}_{\wr\sqcap\sqcup} \in \mathbb{R}^{D \times H \times W}$. During training, $\mathcal{S}_{out}$ is used to minimize a smooth L1-Loss function $\mathcal{L}_{smoothL1}$ to reconstruct $\mathcal{S}_{in}$. The utilized convolutional decoder model is based on PixelShuffle layers.

■ **Point decoder:** The aim of the point decoder $g_{point}(\boldsymbol{z})$ is to map $\boldsymbol{z}$ back to a 3D point cloud representation $\mathcal{P}_{out} \in \mathbb{R}^{3 \times N}$ where $\boldsymbol{N}$ corresponds to the number of points in $\mathcal{P}_{out}$. Similar to the PixelShuffle layers, this is achieved by subsequently transferring from network channels $\boldsymbol{C}$ into the spatial dimension $\boldsymbol{N}$. In the last step, $\boldsymbol{C}$ is set to 3 to predict $\mathcal{P}'$. During training, we minimize a loss function based on the Chamfer distance $\mathcal{L}_{chamfer}$ between $\mathcal{P}_{in}$ and $\mathcal{P}_{out}$

■ **Point-based shape decoder (pbShape decoder):** Our point-based shape decoder $g_{pb-shape}(\boldsymbol{z})$ can be described as a combination of point decoder and convolutional decoder. In a first step, $\boldsymbol{z}$ is mapped to an embedding in shape of key points $\ddagger_{point} \in \mathbb{R}^{3 \times N}$ using a a 3-layer MLP $g_{point-mlp}(\boldsymbol{z})$. In a second step, a differentiable extrapolation $g_{sampler}(\boldsymbol{z})_{point}$ is performed to map back into a volumetric representation $\mathcal{S}_{out}\mathbb{R}^{D \times H \times W}$. This allows us to then minimize a smooth L1-Loss function $\mathcal{L}_{smoothL1}$ to reconstruct $\mathcal{S}_{in}$ from $\mathcal{S}_{out}$. In contrast to the convolutional decoder and the point decoder, our point-based shape decoder relies on a considerably smaller amount of trainable parameters. A more detailed description of our proposed method is provided in the next section.

MLP: After training of the encoder–decoder models was performed in an unsupervised manner on a large dataset, a small MLP $h(\boldsymbol{z})$ is trained on a smaller dataset with fracture labels to predict a binary fracture status from the embedding $\boldsymbol{z}$. For this step, the parameters of the previously trained encoder $\boldsymbol{w}_f$ are fixed, and only the weights of the MLP $\boldsymbol{w}_c$ are optimized using a cross-entropy loss function $\mathcal{L}_{BCE}$.

Data augmentation: To improve the generalizability of our models, we introduce affine augmentation on our input data. For encoder models that take a volumetric patch as input, the augmentation is performed before the forward pass during training on each newly drawn batch. For our geometric models, we also apply the affine augmentation on the volumetric patch after drawing the batch and sample a random set of key points from the augmented volumetric patch using a threshold.

We provide details about the capacity and computational complexity of our AE models in Table 1. Our proposed pbShape decoder requires 600k fewer computations and has only 62k trainable parameters compared to its convolutional counterpart (155k). During the AE-model training, the encoders and decoders are combined, whereas, during training of the MLP classification, only the MLP parameters are adapted. During the test time of the classification, the encoder and MLP parameters are used without further adaptation.

Table 1. Assessment of architectural building blocks employed within our framework including the number of trainable parameters, total size of the model and computational demands in Mult-Add operations.

	#Parameter	Total Size (MB)	Mult-Adds
Shape encoder:			
Conv	6.9M	69.36	5.56G
Point	2.8M	34.1	1.39G
Graph	4.5M	678.92	7.96G
Shape decoder:			
Conv	155k	24.88	1.37G
Point	62k	0.59	0.9M
Pb- Shape	62k	0.66	0.68M
Shape Classifier:			
MLP	541k	2.18	0.54M

3.2. Point-Based Shape Decoder

Normally, the decoder $g(\boldsymbol{z})$ would follow an inverse architecture comprising transposed 3D convolutions or up-sampling layers and introduce a similar number of additional trainable weights $\boldsymbol{w}_g$ as the encoder $\boldsymbol{w}_f$. This makes the latent space less constrained as information about the shape can also be "encoded" in decoder layers. Consequently, we propose a completely new strategy: As opposed to conventional AEs, we map the latent space to represent geometric 3D point coordinates using a small MLP $g_{point-mlp}(\boldsymbol{z})$ to compute a point embedding $\ddagger_{point} \in \mathbb{R}^{3\times N}$, which is then used as input to a differentiable sampler $g_{sampler}(\boldsymbol{z}_{point})$to reconstruct the original input $\mathcal{S}_{in}$. The trainable weights in w_g are limited to $w_{g_{point-mlp}}$ as $w_{g_{sampler}} \rightarrow \varnothing$ (parameters y are fixed after training).

The structure of our proposed decoder is displayed in Figure 2.

We define $g_{sampler}$ as a differentiable geometric operation that extrapolates a 3D volume from a small number keypoint coordinate and value pairs. Recall that spatial transformer networks [20] perform differentiable interpolation of regular 2D/3D feature maps (or images) given a grid of coordinates that is predicted by a localizer network to obtain sampled values. Each resampled value on an off-grid location depends on 4 (2D) or 8 (3D) neighboring integer locations that are used within bi-/trilinear interpolation. Here, following [21] we employ the reverse operation in that spatial coordinates $\boldsymbol{x} \in \mathbb{R}^3$ and

values $\boldsymbol{y} \in \mathbb{R}$ are used as input to extrapolate a 3D grid using their respective trilinear extrapolation coefficients.

The sampling operation $g_{sampler}$ is implemented as a reverse grid sampling operation in PyTorch, and we compute the derivative for both sampling values $\boldsymbol{y}$ and coordinates $\ddagger_{point}$ to make this step differentiable. To mitigate noise and ensure a smooth output, we apply a cubic spline kernel to spatially filter the extrapolation results. Note that the values $\boldsymbol{y}$ attached to each key point are treated as additional trainable parameters, which are shared across all training samples and can be learned and restricted within a range of 0 to 1 with a sigmoid. This enables our method to "paint" a detailed surface, as a zero value placed close to one effectively erases or overwrites parts of a thicker line.

Furthermore, we hypothesize that only $\boldsymbol{N}$ representative (key point) coordinates $\boldsymbol{x}$ are required to reconstruct a detailed vertebral surface and that those key points can be efficiently predicted from the input using f and $g_{point-mlp}$. Hence, our latent space $\boldsymbol{z}_{point}$ now represents a geometric entity that reflects a compact representation of the vertebral shape. This is useful for understanding the impact of osteoporotic fractures on vertebral geometry and can hence lead to improved classification accuracy. By keeping the number of trainable parameters $w_{g_{point-mlp}}$ small, we ensure that these advantageous properties of $\boldsymbol{z}_{point}$ are mirrored in $\boldsymbol{z}$.

4. Experiments and Results

To explore the efficacy of utilizing unsupervised pre-training of vertebral shapes for detecting vertebral body fractures, we conducted two experiments. In the initial experiment, we assessed and compared the classification outcomes of various AE architectures and end-to-end trained models. We also scrutinized the adaptability of segmentation masks generated through deep learning for subsequent diagnostic tasks. The second experiment aimed to determine the robustness of the AE models by executing a data-hold-out experiment on the labeled dataset, specifically during the supervised training phase of the MLP. Subsequently, we outline the prerequisites for our experiments and delve into the implementation details of both the unsupervised pre-training stage for our AE models and the supervised training phase for the MLP. Additionally, we provide implementation details concerning the models subjected to end-to-end training.

4.1. Datasets and Implementation Details

To conduct our experiments as outlined in Figure 1, we utilize the following two public datasets:

TotalSegmentator dataset [9,10]: The dataset contains 1204 CT scans/patients of different parts of the body and corresponding segmentation masks of 104 volumes of interest including 24 vertebrae (C1-L5). From this dataset, we extracted patches of nearly 13k vertebrae surface masks, which are employed for unsupervised training of our AEs. Using our best classifier model, we estimate that 2.9% are fractured.

VerSe19 dataset [3–5]: The dataset contains 160 patients split into 80/40/40 for training, validation, and test set and originally serves as a benchmark for the vertebrae segmentation and localization task. Since fracture labels are also provided for the dataset, we use them to train and test our vertebral fracture classification. The amounts of extracted vertebrae are 770/385/380, respectively. We have also determined that when excluding mild fractures (grade 1) [15], approximately 10% of vertebrae of each subset are fractured.

Unsupervised Pre-Training: The training of our AE models is performed using the same training regime and hyper-parameters for all encoder–decoder combinations, to ensure comparability. Each model is trained for 15,000 iterations using a batch size of 64. The models are optimized by deploying the Adam optimizer with an initial learning rate of 0.005 and CosineAnnealing scheduler with two warm restarts. During training, we augment our data by applying affine transformations. For geometric approaches, we sampled the number of points to 3840, as we have found that the shape of vertebrae depicted in our given patch size can be accurately represented with this amount of key

points. For the DGCNN (graph-encoder) model, we have fixed the hyper-parameter for graph neighbors to $k = 20$.

Supervised Training: For our MLP classifiers, the training is conducted for 6000 iterations using a batch size of 16. We again use the Adam optimizer with an initial learning rate of 0.0001 and CosineAnnealing learning rate scheduling with warm restarts. To enhance the diversity of our input data to the classification model, we augment the input patch to the encoder using affine transformations and we also keep the parameters of the encoder fixed. For geometric encoder models, we utilize the augmented input to construct a randomly sampled point cloud of the vertebral surface. During test time, this point cloud is sub-sampled by employing farthest point sampling to determine 3840 surface points instead of drawing random points from an intensity threshold. While we use the ground truth (GT) segmentation masks for training our classifier models, we employ automatically generated segmentation masks using the TotalSegmentator model [9,10] at test time.

End-to-End Training: In addition to our AE architecture comparison, we train our encoder models from scratch, namely the convolutional encoder, the PointNet, and the DGCNN, in combination with the MLP classifier in an end-to-end fashion on the Verse19 dataset. Furthermore, the convolutional encoder is trained on image patches containing the intensities values as information as well as on surface patches.

4.2. Automated Fracture Detection Pipeline

We evaluate the accuracy of our shape-based vertebral fracture detection through a binary classification that distinguishes between healthy and fractured vertebrae. Since mild fractures (Genant grade 1) [15] of the vertebral bodies are oftentimes indistinct and thus harder to annotate consistently (higher inter-rater variability), vertebrae with label fracture grade 1 are left out for this analysis.

In Table 2, we show the median area-under-curve (AUC) of fracture detection results for 10 seeds of training the MLP for various encoder–decoder architectures and additional end-to-end trained models on the test set. We report the results both for using ground truth segmentation masks for preprocessing and for automatically generated segmentation masks using the TotalSegmentator [9,10].

The results illustrate the benefits of pre-training on shape reconstruction for the detection of vertebral fractures. Furthermore, the results convey that explicitly training on the shape of vertebrae improves classification performance compared to training on image intensities directly. The highest AUC score in the setup utilizing GT segmentation masks to generate the model input is achieved with the point encoder and pbShape decoder architecture, resulting in an AUC of 0.937. When employing TotalSegmentator (TS) segmentation masks, the convolutional encoder in conjunction with the pbShape decoder attains an AUC of 0.914. Notice that the pbShape decoder consistently achieves the highest AUC for each encoder model, highlighting the benefits of the robustness of our proposed decoder.

When examining the outcomes between AE models and end-to-end trained models, the AE models consistently produce more accurate and robust classification results, especially employing the same convolutional encoder architecture, which outperforms its image-based end-to-end trained counterpart by more than 12.8% in AUC. Even when using segmentation masks from TotalSegmentator, there is a 9.0% improvement.

When considering realistic segmentation errors by adapting TS-segmentation labels, especially in models trained with graph-based encoding, a more severe drop in classification performance can be observed in contrast to other encoder models.

To investigate the performance of our models in greater detail, we present the receiver-operator curves (ROC curves) in Figure 3. The depicted results are computed using TS-generated segmentation masks for preprocessing. From left to right the ROC curves are provided for convolutional-encoder, point-encoder and graph-encoder models.

Table 2. Vertebral body fracture detection results based on auto-encoder and end-to-end models. The median and quartiles of the AUC on the VerSe19 test set over 10 seeds evaluated on ground-truth (GT) segmentation mask patches and automatically generated segmentation masks using the TotalSegmentator (TS) are reported. The highest median AUC is highlighted in bold.

Model	GT-Masks AUC Median (0.25, 0.75)	TS-Masks AUC Median (0.25, 0.75)
End-to-end		
Conv-encoder (img)	0.756 (0.746, 0.769)	-
Conv-encoder (surf)	0.883 (0.861, 0.896)	0.847 (0.840, 0.860)
Point-encoder	0.842 (0.821, 0.861)	0.803 (0.778, 0.822)
Graph-encoder	0.616 (0.613, 0.617)	0.628 (0.621, 0.635)
Conv-encoder		
Conv-decoder	0.890 (0.883, 0.892)	0.814 (0.809, 0.814)
Point-decoder	0.704 (0.689, 0.716)	0.712 (0.697, 0.716)
pbShape decoder	0.924 (0.919, 0.926)	**0.914** (0.909, 0.917)
Point-encoder		
Conv-decoder	0.930 (0.027, 0.031)	0.882 (0.879, 0.886)
Point-decoder	0.911 (0.910, 0.913)	0.859 (0.857, 0.862)
pbShape decoder	**0.937** (0.935, 9.943)	0.902 (0.898, 0.904)
Graph-encoder		
Conv-decoder	0.871 (0.868, 0.874)	0.825 (0.823, 0.828)
Point-decoder	0.843 (0.838, 0.844)	0.761 (0.759, 0.764)
pbShape decoder	0.920 (0.918, 0.921)	0.856 (0.852, 0.865)

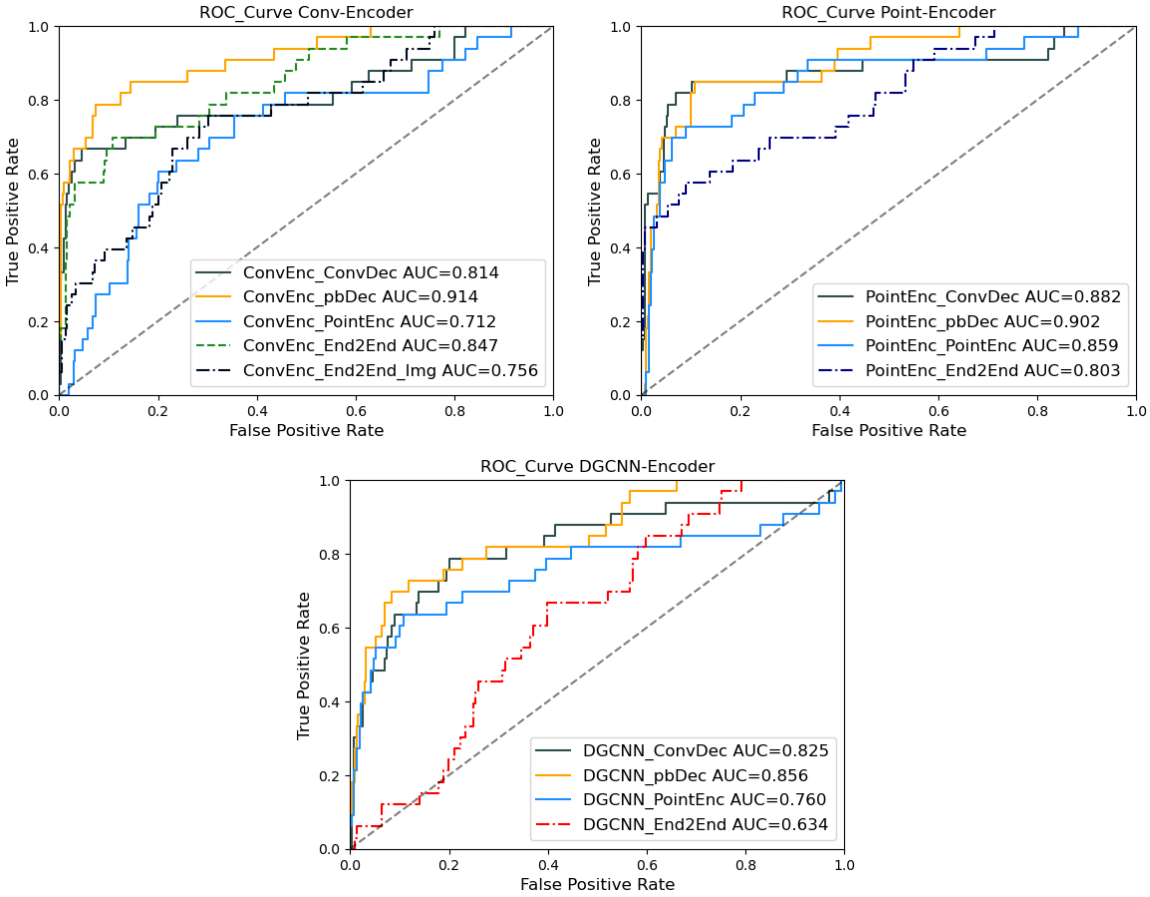

Figure 3. ROC curve and corresponding AUC for encoder–decoder combinations of the median AUC of 10 seeds. The encoders are grouped by color and line style, whereas the decoders are grouped by color and marker. The corresponding area under curve (AUC) is listed inside the legend.

When interpreting the plot for the convolutional encoder, identifying the threshold with the highest possible sensitivity and specificity would be in favor of the ConvEnc-pbDec model, which also reaches the highest overall AUC in this setup (AUC = 0.914).

For ROC curves obtained for both point-encoder and graph-encoder models, we observe that by choosing an optimal threshold either the convolutional decoder model or the pbShape decoder produces the most accurate classification outcome.

4.3. Data-Hold-Out Experiment

In our data-hold-out experiment, we explore the required amount of supervised training data for achieving robust classification outcomes. AUC values, calculated across 10 random seeds for each data split (2%, 5%, 10%, 50%, 75%, and 100%), are presented as both scatter plots and box plots over the 10 seeds (see Figure 4). The experiment demonstrates that the robustness is increased by adding more training data. Notably, employing just 25% of training data already yields as accurate and robust classification results as models trained on the complete training dataset. The increased number of outliers in experiments involving training data below 25% indicates that specific data splits offer more advantageous conditions for the classification task, while others present less favorable circumstances.

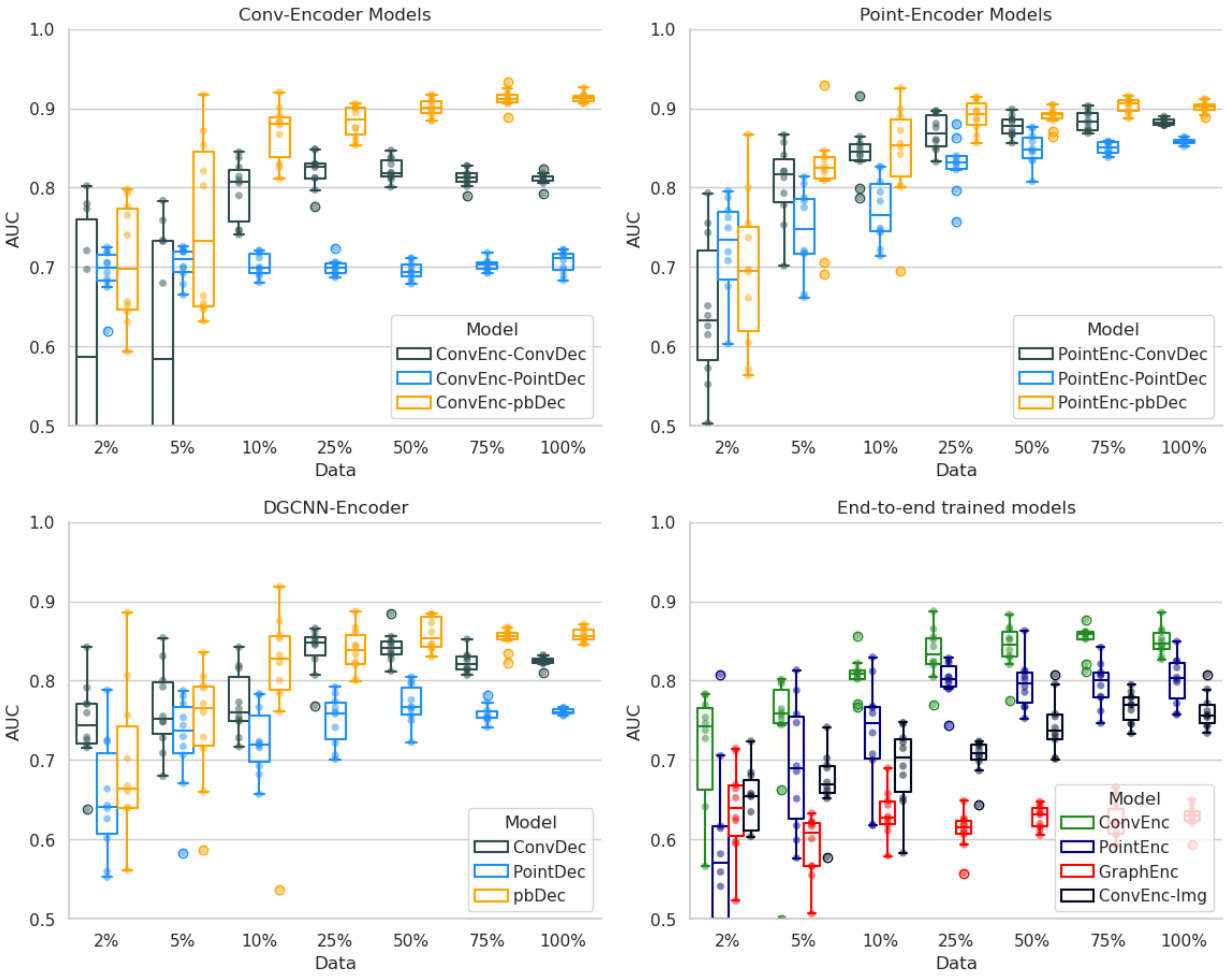

Figure 4. Results of our data-hold-out experiment as boxplots and scatterplots of the AUC obtained for 10 random seeds each. The plots are separated by the employed encoder architecture, and provide the classification results obtained with the respective decoder. **Top-Left:** Convolutional encoder **Top-Right:** Point-encoder models, **Bottom-Left:** Graph-encoder models **Bottom-Right:** End-to-End trained models including the traditional CNN trained on image intensities instead and trained on vertebra surface (denoted as Conv-encoder img and surf).

5. Discussion

In this work, we examined the benefits of directly utilizing shape information in the diagnosis of vertebral body fractures. We introduced a shape encoding approach that takes a volumetric surface representation of a vertebra as input, encoding it into a latent vector, which is decoded by transforming the latent vector into point representations (using an MLP) and then applied a novel differentiable point sampling method: the point-based shape decoder (pbShape decoder).

Within our vertebrae AE framework, we analyze the performance of our proposed decoder and other AE building blocks that are pre-trained on a large-scale dataset to reconstruct the shape of an input vertebra surface patch and additionally employ end-to-end trained models for comparison. Whereas the vast majority of weights in our AE models (the whole encoder part) are trained in an unsupervised fashion, only a light-weight MLP classifier using the compact shape representation from the latent space requires training with fracture labels. The findings of our end-to-end trained models indicate that, with supervised training on surface data, good classification results (AUC above 0.9) can already be obtained using a 3D CNN model, and our results demonstrate that using automatically generated segmentation data yields preferable results in comparison to training on image-intensity data directly. This shows the advantage of explicitly employing shape information from large-scale multi-label segmentation models over implicitly learning shape information in intensity-based CNNs. Furthermore, employing surface information that was previously already learned, for instance by the nnUNet in the TotalSegmentator model [9,10], enhances the robustness of the classification task against domain shifts. The results confirm that the quality and robustness of the generated TotalSegmentator [9,10] masks are adequate for this task. Furthermore, our findings highlight that the choice of architecture influences the model's robustness in the face of inaccuracies in segmentation masks. In a clinical setting, obtaining ground truth segmentation masks can be challenging. The fact that our model achieves an AUC of 0.914 holds significant potential for automated diagnosis of osteoporotic fractures in the spine with limited availability of labeled data.

Our findings further suggest that, when trained in an end-to-end fashion, geometric learning approaches such as PointNet [13] and DGCNN [17] do not perform as well as 3D convolutional approaches. However, when integrated as an encoder into an AE architecture and pre-trained on a large-scale dataset, we demonstrate their ability to achieve more accurate results. Moreover, our findings indicate that models employing a PointNet as an encoder produce accurate and robust results with only a small deviation between the different decoder models (AUC pbShape: 0.902 conv: −2.0% and point: −4.3%) with fewer trainable parameters being incorporated. This underscores the effectiveness and efficacy of the PointNet architecture. Contrarily, the graph-based encoder appears to under-perform on this specific task. This could be attributed to the nature of the task of detecting vertebral fractures, which involves a more global geometric relationship based on height differences, rendering local graph structures less relevant. Consequently, capturing finer geometric features may not contribute significantly to this task but might be of interest when detecting other degenerative deformations of vertebrae.

By performing a data-hold-out experiment, we highlight the effectiveness and robustness of our models and find that an AUC above 0.9 can be obtained even when training on 5% of data (51 healthy and 5 fractured vertebrae), but robustness increases when training on a larger dataset. The results of this experiment furthermore show the importance and impact of data distributions and that the prevalence of vertebral fractures for this task increases the difficulty to compare methods between multiple datasets. We thus consider that our point-encoder, point-decoder model us similar to the approach proposed as pAE by Sekuboyina et al. [12], who also utilize a point cloud generated from ground truth segmentation masks as input into their point-based AE architecture. However, they treat the detection of vertebral body fractures as an anomaly detection task by training the reconstruction of the point cloud on healthy samples and only achieving an AUC of 0.759 with the best model on their dataset using accurate segmentation masks. In comparison, our purely point-based AE

model reaches an AUC of 0.895 (0.911 on ground truth segmentation masks), setting a new state of the art.

6. Conclusions

In summary, our study explores AE architectures for effective shape encoding of vertebrae to diagnose osteoporotic fractures of vertebrae. We introduce a novel approach, utilizing a differentiable geometrical decoder for unsupervised pre-training on the TotalSegmentator dataset. By combining convolutional or point-based encoders with our proposed shape decoder, we achieve a meaningful and robust shape representation in the latent space, facilitating the detection of osteoporotic fractures. Our approach demonstrates robustness through comparisons with ground truth segmentation masks, showcasing commendable classification results even when applying automatically generated segmentation masks. Moreover, this approach can be extended to other shape-related tasks, e.g., diagnosing degenerative deformations, spinal canal stenosis, or lymph nodes.

Author Contributions: Conceptualization, H.H., A.B. and M.P.H.; data curation, H.H. and M.P.H.; formal analysis, H.H., A.B. and M.P.H.; funding acquisition, M.P.H.; investigation, H.H., A.B. and M.P.H.; methodology, H.H., A.B. and M.P.H.; project administration, M.P.H.; resources, H.H., A.B. and M.P.H.; software, H.H., A.B. and M.P.H.; supervision, M.P.H.; validation, H.H., A.B. and M.P.H.; visualization, H.H.; writing—original draft, H.H.; writing—review & editing, H.H., A.B. and M.P.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the German Federal Ministry of Education and Research grant number 01EC1908D.

Informed Consent Statement: Not applicable.

Data Availability Statement: The source code is made available on GitHub: https://github.com/multimodallearning/shape_matters, accessed on 14 February 2024. The VerSe Dataset is publicly available at: https://osf.io/923ap/, accessed on 14 February 2024, Ethics approval: TUM Proposal 27/19 S-SR, Licence: CC BY-SA 4.0. The TotalSegmentator Dataset is publicly available at: https://zenodo.org/records/6802614, accessed on 14 February 2024, Ethics approval: Ethics Committee Northwest and Central Switzerland (EKNZ BASEC Req-2022-00495), Licence: CC BY 4.0.

Conflicts of Interest: Author Alexander Bigalke was employed by the company Dräger, Drägerwerk AG & Co. KGaA. The remaining authors declare no 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.

References

1. Ballane, G.; Cauley, J.; Luckey, M.; El-Hajj Fuleihan, G. Worldwide prevalence and incidence of osteoporotic vertebral fractures. *Osteoporos. Int.* **2017**, *28*, 1531–1542. [CrossRef] [PubMed]
2. Papaioannou, A.; Watts, N.B.; Kendler, D.L.; Yuen, C.K.; Adachi, J.D.; Ferko, N. Diagnosis and management of vertebral fractures in elderly adults. *Am. J. Med.* **2002**, *113*, 220–228. [CrossRef] [PubMed]
3. Liebl, H.; Schinz, D.; Sekuboyina, A.; Malagutti, L.; Löffler, M.T.; Bayat, A.; El Husseini, M.; Tetteh, G.; Grau, K.; Niederreiter, E.; et al. A computed tomography vertebral segmentation dataset with anatomical variations and multi-vendor scanner data. *Sci. Data* **2021**, *8*, 284. [CrossRef] [PubMed]
4. Löffler, M.T.; Sekuboyina, A.; Jacob, A.; Grau, A.L.; Scharr, A.; El Husseini, M.; Kallweit, M.; Zimmer, C.; Baum, T.; Kirschke, J.S. A vertebral segmentation dataset with fracture grading. *Radiol. Artif. Intell.* **2020**, *2*, e190138. [CrossRef] [PubMed]
5. Sekuboyina, A.; Husseini, M.E.; Bayat, A.; Löffler, M.; Liebl, H.; Li, H.; Tetteh, G.; Kukačka, J.; Payer, C.; Štern, D.; et al. VerSe: A vertebrae labelling and segmentation benchmark for multi-detector CT images. *Med. Image Anal.* **2021**, *73*, 102166. [CrossRef] [PubMed]
6. Nicolaes, J.; Raeymaeckers, S.; Robben, D.; Wilms, G.; Vandermeulen, D.; Libanati, C.; Debois, M. Detection of vertebral fractures in CT using 3D convolutional neural networks. In Proceedings of the Computational Methods and Clinical Applications for Spine Imaging: 6th International Workshop and Challenge, CSI 2019, Shenzhen, China, 17 October 2019; Springer: Cham, Switzerland , 2020; pp. 3–14.
7. Yilmaz, E.B.; Buerger, C.; Fricke, T.; Sagar, M.M.R.; Peña, J.; Lorenz, C.; Glüer, C.C.; Meyer, C. Automated deep learning-based detection of osteoporotic fractures in CT images. In Proceedings of the Machine Learning in Medical Imaging: 12th International

Workshop, MLMI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, 27 September 2021; Springer: Cham, Switzerland, 2021; pp. 376–385.
8. Zakharov, A.; Pisov, M.; Bukharaev, A.; Petraikin, A.; Morozov, S.; Gombolevskiy, V.; Belyaev, M. Interpretable vertebral fracture quantification via anchor-free landmarks localization. *Med. Image Anal.* **2023**, *83*, 102646. [CrossRef] [PubMed]
9. Wasserthal, J.; Breit, H.C.; Meyer, M.T.; Pradella, M.; Hinck, D.; Sauter, A.W.; Heye, T.; Boll, D.T.; Cyriac, J.; Yang, S.; et al. TotalSegmentator: Robust Segmentation of 104 Anatomic Structures in CT Images. *Radiol. Artif. Intell.* **2023**, *5*, e230024. [CrossRef] [PubMed]
10. Isensee, F.; Jaeger, P.F.; Kohl, S.A.; Petersen, J.; Maier-Hein, K.H. nnU-Net: A self-configuring method for deep learning-based biomedical image segmentation. *Nat. Methods* **2021**, *18*, 203–211. [CrossRef]
11. Husseini, M.; Sekuboyina, A.; Bayat, A.; Menze, B.H.; Loeffler, M.; Kirschke, J.S. Conditioned variational auto-encoder for detecting osteoporotic vertebral fractures. In Proceedings of the Computational Methods and Clinical Applications for Spine Imaging: 6th International Workshop and Challenge, CSI 2019, Shenzhen, China, 17 October 2019; Springer: Cham, Switzerland, 2020; pp. 29–38.
12. Sekuboyina, A.; Rempfler, M.; Valentinitsch, A.; Loeffler, M.; Kirschke, J.S.; Menze, B.H. Probabilistic point cloud reconstructions for vertebral shape analysis. In Proceedings of the Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, 13–17 October 2019; Proceedings, Part VI; Springer: Cham, Switzerland, 2019; pp. 375–383.
13. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
14. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph. (Tog)* **2019**, *38*, 1–12. [CrossRef]
15. Genant, H.K.; Wu, C.Y.; Van Kuijk, C.; Nevitt, M.C. Vertebral fracture assessment using a semiquantitative technique. *J. Bone Miner. Res.* **1993**, *8*, 1137–1148. [CrossRef] [PubMed]
16. Husseini, M.; Sekuboyina, A.; Loeffler, M.; Navarro, F.; Menze, B.H.; Kirschke, J.S. Grading loss: A fracture grade-based metric loss for vertebral fracture detection. In Proceedings of the Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, 4–8 October 2020; Proceedings, Part VI; Springer: Cham, Switzerland, 2020; pp. 733–742.
17. Zhang, M.; Cui, Z.; Neumann, M.; Chen, Y. An end-to-end deep learning architecture for graph classification. *Proc. AAAI Conf. Artif. Intell.* **2018**, *32*, 4438–4445. [CrossRef]
18. Huo, L.; Cai, B.; Liang, P.; Sun, Z.; Xiong, C.; Niu, C.; Song, B.; Cheng, E. Joint spinal centerline extraction and curvature estimation with row-wise classification and curve graph network. In Proceedings of the Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, 27 September–1 October 2021; Proceedings, Part V; Springer: Cham, Switzerland, 2021; pp. 377–386.
19. Bürgin, V.; Prevost, R.; Stollenga, M.F. Robust vertebra identification using simultaneous node and edge predicting Graph Neural Networks. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention: 26th International Conference, Vancouver, BC, Canada, 8–12 October 2023; Proceedings, Part IX; Springer: Cham, Switzerland, 2023; pp. 483–493.
20. Jaderberg, M.; Simonyan, K.; Zisserman, A.; Kavukcuoglu, K. Spatial transformer networks. *Adv. Neural Inf. Process. Syst.* **2015**. Available online: https://proceedings.neurips.cc/paper_files/paper/2015/hash/33ceb07bf4eeb3da587e268d663aba1a-Abstract.html (accessed on 14 February 2024).
21. Heinrich, M.P.; Bigalke, A.; Großbröhmer, C.; Hansen, L. Chasing Clouds: Differentiable Volumetric Rasterisation of Point Clouds as a Highly Efficient and Accurate Loss for Large-Scale Deformable 3D Registration. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 8026–8036.