Machine Learning Acceleration of the DXA2FEM Pipeline for Femoral Bone Strength Prediction

Joel Ekebro & Morgan Sahlberg

2026

LUND UNIVERSITY

Master's Thesis in Biomedical Engineering

Faculty of Engineering, LTH

Department of Biomedical Engineering

Supervisors:

Lorenzo Grassi & Christian Antfolk

Abstract

Fragility fractures are a major health concern and accurate estimation of bone strength is important for fracture risk assessment. Finite element (FE) models can provide reliable estimates of femoral bone strength, but traditional approaches typically require three-dimensional imaging and computationally intensive model generation. Previous research has enabled reconstruction of subject-specific FE models by only using dual-energy X-ray absorptiometry (DXA) images, but the optimization procedure used to estimate the model parameters can require several hours per patient, which is not feasible in a clinical scenario.

The aim of this thesis was to investigate whether machine learning can be used to accelerate these reconstruction pipelines. Two neural network architectures were developed: a baseline convolutional neural network (CNN) and a transfer learning network based on EfficientNetV2B2. The networks were trained using DXA images from the MrOS Sweden cohort to either predict the statistical shape and appearance model (SSAM) parameters used for reconstruction or predict bone strength directly from the images. Artificially generated digitally reconstructed radiographs were also evaluated as synthetic training data.

The results show that neural networks can predict SSAM parameters and approximate bone strength estimates with a substantially lower computational cost; potentially saving several hours in computation time. The baseline CNN achieved slightly lower prediction errors than the transfer learning model in most metrics, both predicting reconstruction parameters and bone strength. The most accurate bone strength results were obtained when predicting bone strength directly from DXA images. Although the predicted SSAM parameters likely cannot yet replace the optimization procedure used in existing methods, they could be used to initialize the optimization algorithm. Future work will investigate whether such initialization can reduce convergence time and thereby accelerate the reconstruction process, as well as whether direct bone strength predictions are sufficiently accurate for clinical use.

Acknowledgements

We would like to thank our supervisors, Lorenzo Grassi and Christian Antfolk, for their guidance throughout this thesis project.

We are especially grateful to Lorenzo Grassi for his support with the DXA2FEM pipeline and related methodology, and to Christian Antfolk for valuable discussions and insights into machine learning.

Authors' Contributions

This thesis was carried out as an equal collaboration between the coauthors Joel Ekebro and Morgan Sahlberg. During model development, Morgan Sahlberg implemented the baseline network, while Joel Ekebro implemented the transfer learning network. The writing of the report was divided equally between the authors, and all aspects of the work were discussed and developed collaboratively.

Use of Generative AI

Generative AI (ChatGPT) was used during the preparation of this thesis as a writing aid to correct typos, improve grammar, and enhance the flow of the text. AI-generated suggestions were carefully reviewed and edited by the authors to ensure that they accurately reflected the authors' intended meaning. All ideas, analysis, results, and conclusions presented in this thesis are entirely the authors' own.

Contents

1 Introduction and Aim

1.1 Introduction

Fractures related to skeletal fragility represent a major public health concern. Reduced bone strength substantially increases the risk of low impact trauma fractures, which can occur when subjected to minor falls, bumps, or even normal movement. Over a lifetime, approximately one in three women and at least one in six men will experience such a fracture. Fragility fractures place a significant burden on healthcare systems, with the total cost in EU countries estimated to exceed €56 billion annually [1]. Beyond the economic impact, fractures often lead to pain, reduced mobility, loss of independence, and a diminished quality of life. Fractures of the spine and femur are particularly severe, carrying a high risk of long-term disability and mortality. Each year, approximately 250,000 deaths in Europe are attributed directly to these fractures [1]. Because of this it is of value to accurately estimate fracture-risk in order to identify people that have a high risk of fracture and take precautionary measures.

Currently in clinical practice, fracture risk is predicted by combining clinical data with approximate measures of bone strength and risk of falling. In the case of bone strength the standard method used today is an approximation based on the areal bone mineral density (aBMD) calculated from a dual-energy x-ray absorptiometry (DXA). Using aBMD as a surrogate for bone strength is cheap and fast which makes it suitable for clinical practice. However, many fractures occur in individuals who are not classified as high risk according to the standard aBMD assessment [2]. It is therefore of importance to find alternative methods to predict bone strength in order to provide a more accurate fracture risk and prevention strategy [3].

Using finite element models has been shown to be a method that can accurately predict bone strength in the femur and give a better assessment of fracture risk if used instead of aBMD [4]. The issue with this method is that it requires a 3D computed tomography (CT) image of the patient to retrieve a 3D-model of the bone. A CT scan is more expensive and exposes the patient to more radiation than a DXA-scan. A trained engineer is also required to build the FE model which further increases the cost [5].

Methods to obtain 3D models of the femur from a single DXA image have been developed in order to achieve the same accuracy of FE-models but without having to do a CT-scan. One of these methods is DXA2FEM which uses a statistical shape and appearance model (SSAM) that represents the variation of shape and structure of a bone in a population. The reconstruction of the 3D-model is done with an optimization algorithm trying to minimize the difference between the DXA-image and a 2D projection of the 3D reconstruction [6].

Although DXA-based FE approaches have demonstrated improved fracture prediction compared to aBMD, the clinical adoption remains limited due to their computationally intensive pipelines. Model generation, FE simulation, and strength estimation can require several hours per patient, posing a significant barrier to routine clinical application [7].

One possible solution to reduce the reconstruction time is to use convolutional neural networks to replace some of the more computationally heavy parts of the pipeline. Specifically, the optimization of the SSAM parameters, but also directly predicting bone strength, skipping the SSAM and FE modeling completely.

1.2 Aim

The aim of this thesis is to investigate whether machine learning can be used to accelerate the DXA2FEM pipeline while maintaining comparable predictive performance. Specifically, convolutional neural networks are trained to estimate biomechanical properties of the femur directly from DXA images.

The primary objective is to predict the SSAM parameters used in DXA2FEM to reconstruct subject-specific three-dimensional femur models. If successful, these predictions could replace the computationally intensive optimization procedure currently used in DXA2FEM, potentially reducing reconstruction time from several hours to seconds.

A secondary objective is to investigate whether bone strength can be predicted directly from DXA images without reconstructing the intermediate FE model. This approach would bypass both the SSAM reconstruction and the finite element simulations, enabling near-instant estimates of bone strength.

2 Theory

2.1 DXA2FEM

DXA2FEM is a method for estimating subject-specific bone strength from dual-energy X-ray absorptiometry (DXA) images. It has been shown to improve hip fracture prediction compared to standard areal bone mineral density (aBMD) measurements [7]. The method reconstructs a three-dimensional (3D) model of the patient's femur and pelvis using a Statistical Shape and Appearance Model (SSAM) [6]. From the reconstructed geometry, a finite element (FE) mesh is generated and used to simulate bone strength, which serves as a biomechanical predictor of fracture risk [7].

2.1.1 SSAM

SSAM stands for Statistical Shape and Appearance Model and is used to represent anatomical variability in both geometry and density distribution. The model is typically constructed using principal component analysis (PCA) applied to a training set of segmented CT images [6]. PCA provides a compact representation of anatomy by reducing high-dimensional geometry and density information to a small number of statistically independent modes, enabling efficient description, reconstruction, and optimization of realistic anatomies during model fitting [8].

In DXA2FEM, the 3D reconstruction of the femur is described by 17 SSAM parameters, while the pelvis is described by 10 parameters [7]. The model is based on training data from 59 CT scans of femurs and 14 CT scans of pelvises [7].

2.1.2 Optimization

To estimate the subject-specific SSAM parameters from a 2D DXA image, an optimization procedure is performed. Initially, instances of the femoral and pelvic SSAMs are generated and roughly aligned to the DXA image. Digitally reconstructed radiographs (DRRs) are then created by projecting the 3D reconstructions onto the image plane.

The similarity between the DRR and the measured DXA image is quantified using a cost function. This cost function includes the sum of absolute differences between projected and measured areal bone mineral density, constraints on anatomical positioning, and mesh quality measures [6]. A genetic algorithm is used to iteratively update the SSAM parameters, rigid transformations, and scaling in order to minimize the cost function. The optimization continues until convergence.

This optimization process is computationally demanding and may require several hours per patient. For the dataset used in this study the average optimization time is over seven hours.

2.1.3 Bone Strength

Bone strength estimation follows the DXA2FEM framework proposed in Grassi et al. [7]. After reconstruction of the subject-specific 3D proximal femur geometry, a corresponding finite element (FE) model is automatically generated. The reconstructed femur FE mesh consists of linear tetrahedral elements with element-specific volumetric bone mineral density (vBMD) values, which are converted to quadratic elements to improve numerical accuracy [7]. The vBMD values are transformed into elastic moduli using previously validated empirical density–elasticity relationships, and elements at the periosteal surface are assigned a minimum Young's modulus of 5 GPa to compensate for possible reconstruction artifacts [7].

An anatomical reference system is defined by warping a template femur to the reconstructed geometry and automatically identifying anatomical landmarks, enabling consistent application of boundary conditions [7]. Linear elastic, quasi-static simulations are then performed in Abaqus to mimic a sideways fall, the most common mechanism for hip fracture [7]. Ten different loading configurations are considered, spanning 0°–30° in both adduction and internal rotation, including a commonly used reference configuration of 10° adduction and 15° internal rotation [7].

Fracture load is estimated using a principal strain limit criterion, where failure is assumed to occur when predefined tensile or compressive strain thresholds are exceeded [7]. The predicted fracture load is typically normalized by body weight. FE-predicted strength obtained from this 2D-to-3D reconstruction framework has been shown to correlate more strongly with incident hip fractures than areal bone mineral density (aBMD) alone [7].

We emphasize that the FE modeling and strength estimation procedure described above was not implemented in this thesis but is adopted directly from [7].

2.2 Machine Learning

2.2.1 Artificial Neural Networks

Artificial neural networks (ANNs) are parametric models that approximate nonlinear functions by composing multiple layers of linear transformations and nonlinear activation functions. In supervised learning, the network is trained to approximate a mapping

$$f_\theta : \mathbb{R}^n \to \mathbb{R}^m, \tag{2.1}$$

where θ denotes the set of trainable parameters. The presentation in this section follows the general framework described by Goodfellow et al. in *Deep Learning* [9].

Network Structure

A feedforward neural network consists of a sequence of layers that transform an input vector $\mathbf{x}$ into an output prediction $\hat{\mathbf{y}}$. Each layer performs an affine transformation followed by a nonlinear activation function:

$$\mathbf{h}^{(l)} = g^{(l)}\left(\mathbf{W}^{(l)}\mathbf{h}^{(l-1)} + \mathbf{b}^{(l)}\right), \tag{2.2}$$

where $\mathbf{W}^{(l)}$ and $\mathbf{b}^{(l)}$ are the weight matrix and bias vector of layer l, and $g^{(l)}(\cdot)$ denotes the activation function. The input is defined as $\mathbf{h}^{(0)} = \mathbf{x}$.

By stacking multiple layers, the network represents a composition of functions, enabling the modeling of complex nonlinear relationships. The number of layers defines the network depth, while the dimensionality of intermediate representations defines the width. In general, deeper and wider networks have greater representational capacity and can achieve higher accuracy, but they also increase the risk of overfitting if the amount of training data or regularization is insufficient.

Activation Functions

Nonlinear activation functions are necessary to allow the network to represent nonlinear mappings. Without nonlinearities, multiple linear layers would reduce to a single linear transformation.

Common activation functions include the sigmoid function, hyperbolic tangent, and the rectified linear unit (ReLU). In modern deep networks, ReLU is frequently used due to its computational efficiency and stable gradient behavior:

$$\mathrm{ReLU}(z) = \max(0, z) \tag{2.3}$$

Loss Functions

Training is formulated as the minimization of a loss function that quantifies the discrepancy between predictions $\hat{\mathbf{y}}$ and ground truth targets $\mathbf{y}$. Given a dataset of N samples $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$, the network parameters θ are obtained by minimizing the average loss over the training data:

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \ell \left(f_\theta(\mathbf{x}_i), \mathbf{y}_i \right), \tag{2.4}$$

where $\ell(\cdot, \cdot)$ denotes a per-sample loss function. The function $\mathcal{L}(\theta)$ defines the optimization objective and depends on the model parameters through the network output $f_\theta(\mathbf{x}_i)$.

For regression tasks, commonly used loss functions include the mean squared error (MSE),

$$\ell_{\text{MSE}} = \|\hat{\mathbf{y}} - \mathbf{y}\|_2^2, \tag{2.5}$$

the mean absolute error (MAE),

$$\ell_{\text{MAE}} = \|\hat{\mathbf{y}} - \mathbf{y}\|_1, \tag{2.6}$$

and the Huber loss, which combines quadratic and linear behavior to reduce sensitivity to outliers. The choice of loss function influences robustness to noise and the geometry of the optimization problem.

Optimization and Training

The network parameters are optimized using gradient-based methods. Gradients of the loss function with respect to the parameters are computed using backpropagation, which applies the chain rule to efficiently propagate derivatives through the network.

In practice, stochastic gradient descent (SGD) or adaptive variants such as Adam [10] are used. Parameter updates are performed iteratively using mini-batches of data:

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_\theta \mathcal{L}_{\mathcal{B}}(\theta^{(t)}), \tag{2.7}$$

where η is the learning rate and $\mathcal{L}_{\mathcal{B}}$ denotes the loss computed over a mini-batch. The learning rate determines how large the parameter updates are during each iteration of training. A higher learning rate can accelerate training but may lead to unstable optimization or prevent convergence if it is too large.

The mini-batch size determines the number of training samples used to estimate the gradient at each update step. Smaller batch sizes introduce stochasticity in the gradient estimate, which can improve generalization but may lead to noisier optimization. Larger batch sizes provide more stable gradient estimates at the cost of increased memory usage and potentially slower convergence in terms of generalization performance. The batch size is

therefore an important hyperparameter that influences both computational efficiency and training dynamics.

Training proceeds for multiple epochs until convergence or until a predefined stopping criterion is satisfied.

Regularization

Due to their high capacity, neural networks are prone to overfitting, particularly when training data is limited. Regularization techniques are therefore applied to improve generalization.

One common method is L2 regularization (weight decay), which adds a penalty term to the objective function:

$$\mathcal{L}_{\text{reg}}(\theta) = \mathcal{L}(\theta) + \lambda \|\theta\|_2^2, \tag{2.8}$$

where λ controls the regularization strength.

Additional strategies include dropout, early stopping based on validation performance, and data augmentation. These methods reduce overfitting and improve robustness to unseen data.

2.2.2 Convolutional Neural Networks

Convolutional neural networks (CNNs) are the most commonly used deep learning models for image-based learning tasks, because they are designed to extract features from grid-based structures such as images. A CNN typically contains three types of layers, the convolutional layers, pooling layers and fully connected layers. The main operation of the convolutional layers is the discrete convolution, where a set of learnable filters (or kernels) is applied to the image [11]. For a two-dimensional input image I and a filter kernel K, the convolution operation $(*)$ producing a feature map S is given by

$$S(i,j) = K(m,n) * I(m,n) = \sum_m \sum_n K(m,n)\, I(i-m, j-n). \tag{2.9}$$

Each filter learns to respond to specific local patterns in the image, such as edges or texture variations. By stacking multiple convolutional layers, the network can represent increasingly abstract features with larger receptive fields.

A key structural property of convolutional neural networks is weight sharing, where the same set of filter parameters is applied across all spatial locations of the image. This

reduces the number of trainable parameters significantly compared to fully connected architectures and introduces the assumption of translational invariance in images. If a pattern is relevant in one region of the image, it is likely to be relevant elsewhere. This assumption is appropriate for medical images such as DXA scans, where structural features of interest may appear at varying spatial locations [12, 13].

Each convolutional layer typically produces multiple feature maps, corresponding to different learned filters. If a layer has C_{in} input channels and C_{out} filters, then C_{out} feature maps are produced. These feature maps are commonly followed by a nonlinear activation function [11].

Pooling layer

Pooling layers are frequently used between convolutional layers to downsample feature maps. Pooling layers are used to reduce the spatial dimensions of the feature maps, but still keep the most important information. This is done by sliding a filter over the feature map and depending on what type of pooling is used it will summarize it in different ways. The primary types of pooling are max-pooling which reduces each region to its largest value, and average-pooling which takes the average value of the region [14].

The purpose of pooling layers is to reduce the number of feature components to process which makes the model faster and more efficient.

By stacking multiple convolutional and pooling layers, CNNs construct hierarchical feature representations. Early layers typically learn local features such as edges and intensity gradients, while deeper layers capture increasingly abstract structures formed by combinations of local patterns. This means that the size of the receptive field i.e. what parts of the image are used as information for different features, grows with depth, allowing later layers to encode global structural information [14].

Fully connected (dense) layer

Fully connected (FC) layer also known as dense are layers where each of the neurons are connected to each of the neurons in the previous layer as well as to each of the neurons in the next layer. In CNNs FC layers are usually put after the convolutional and pooling layers to convert the feature maps into a final prediction.

2.2.3 Transfer Learning

Transfer learning refers to the practice of initializing a model with parameters learned on a different, often more general, source problem. The formal definition is that, given a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$, knowledge acquired on a source domain $\mathcal{D}_S$

and learning task $\mathcal{T}_S$ is used to improve the predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ [15]. Cases where the target task is different from the source task are referred to as inductive transfer learning [15]. Inductive transfer learning can be used to learn feature representations that are subsequently adapted to the target task.

When the target domain is significantly smaller than the source domain, transfer learning can be a powerful tool to enable the training of large networks without severe overfitting [16]. For image classification and feature extraction, a common source domain is the ImageNet dataset [16, 17]. It consists of over one million labeled images across 1000 object categories [17]. In deep learning for images, it has been observed that the initial layers of convolutional networks learn similar low-level features such as edges and color blobs [16]. This makes the initial layers particularly useful for transfer learning. The middle and final layers are more task-specific and often require fine-tuning, especially when the target task differs from the source task [16]. When transfer learning leads to degraded performance compared to training from scratch, it is referred to as negative transfer [15].

A recent image learning architecture is EfficientNetV2. It is pretrained on ImageNet and achieves shorter training times and improved accuracy compared to earlier convolutional architectures [18]. The model has also demonstrated strong transfer learning performance on multiple downstream datasets, indicating that the learned representations generalize beyond the ImageNet domain [18].

2.2.4 Machine Learning on Medical Imaging

Convolutional neural networks have become the dominant methodology in medical image analysis, demonstrating superior performance compared to traditional approaches based on handcrafted features across tasks including classification, detection, and segmentation in MRI, CT, X-ray, and ultrasound images [19, 20]. A key challenge in this domain is the limited availability of large, annotated datasets due to the cost of expert annotation and data privacy constraints. Transfer learning has therefore emerged as a widely adopted strategy, where models pretrained on large-scale natural image datasets such as ImageNet are adapted to medical imaging tasks [19, 20]. A comprehensive review of 121 studies found that architectures such as ResNet and Inception are most frequently used as backbone models, and that no single architecture consistently outperforms others across all imaging modalities and clinical tasks. Instead, performance depends heavily on dataset size, anatomical region, and number of output classes [20].

Transfer learning with EfficientNet architectures has demonstrated strong performance across a range of medical imaging applications. Marques et al. applied an EfficientNet-B4 model to chest X-ray classification for automated COVID-19 diagnosis, achieving an average accuracy of 99.62% for binary classification, outperforming prior architectures including VGG, ResNet, and MobileNet [21]. Shah et al. similarly demonstrated that a fine-tuned EfficientNet-B0 model for MRI-based brain tumor classification achieved a validation accuracy of 98.87%, outperforming several well-known CNN architectures while

maintaining a smaller model size and fewer parameters [22]. These results illustrate that EfficientNet-based transfer learning can provide accurate and computationally efficient solutions, even in settings with limited data.

Beyond classification, deep learning has been applied to the regression of statistical shape model parameters from medical images; a task closely related to the one addressed in this thesis. Ha et al. proposed a framework for 2D–3D reconstruction of the femur from a single X-ray image using a deep transfer learning network integrated with a statistical shape model [23]. By directly predicting SSM deformation parameters from X-ray images, the method eliminates the need for conventional calibration, 2D–3D registration, and iterative optimization. The approach achieved reconstruction errors of approximately 1.1–1.2 mm RMS point-to-surface distance, demonstrating accuracy comparable to methods relying on multiple images or iterative optimization. Ha et al. also evaluated several pretrained architectures for this task, including Xception, EfficientNet, VGG16, ResNet152V2, NASNetLarge, and InceptionResNet; finding that their performance was broadly comparable, with a slight advantage for Xception [23]. This suggests that multiple modern architectures are suitable for medical image-based parameter regression and that the choice of backbone may be guided by considerations such as training stability and computational efficiency.

Deep learning has also been applied to the reconstruction of 3D volumetric structures from 2D radiographic inputs more broadly. Chen et al. proposed a framework for reconstructing 3D vertebral structures from bi-planar X-ray images, achieving a Dice score of 89.92% and significantly outperforming prior volumetric reconstruction methods [24]. Bottini et al. investigated generating synthetic CT volumes of the spine from biplanar radiographs using both generative adversarial networks and CNN-based implicit neural representations, demonstrating that deep learning can approximate 3D CT reconstruction from limited 2D imaging, with GAN-based approaches currently offering the most perceptually realistic results [25]. These works demonstrate the broader feasibility of the goal pursued in this thesis: using 2D projection images as input to predict or reconstruct clinically useful 3D representations of bone anatomy.

3 Material and Methods

The aim of this thesis is to investigate whether machine learning can be used to accelerate the DXA2FEM pipeline while maintaining comparable predictive performance. To achieve this, two different neural network architectures were developed: a baseline convolutional neural network (CNN) and a transfer learning–based network. The models were trained using femur DXA images from elderly male patients as input.

Two different prediction strategies were evaluated. In the first approach, the networks predicted SSAM parameters, which were subsequently used in the DXA2FEM pipeline to estimate bone strength from the reconstructed FE mesh. In the second approach, the networks predicted bone strength directly from the images.

The use of artificially generated digitally reconstructed radiographs (DRRs) images as synthetic training data was also investigated by generating synthetic images and training the networks on this data.

The results were evaluated at multiple levels in order to compare the performance of the proposed models with each other and with the original DXA2FEM results.

3.1 Material

Two datasets are used in this thesis project: the MrOS and the MrPeak cohorts. These datasets are chosen because they have been used in previous DXA2FEM studies, which means there already exists reconstructions and bone strengths which can be used as output for our models [7].

The primary dataset is the Swedish MrOS cohort, which consists of 3014 men aged 69–80 years [26]. The cohort includes clinical data, physical performance measurements, and DXA images.

Complete data entries, with DXA images and corresponding SSAM reconstruction parameters, only exist for 1583 of the subjects in the cohort. Thus, 1583 subjects are included for machine learning model development for this thesis.

The MrPeak cohort is used as a complementary dataset. It consists of 1052 men aged 18–28 years [27]. As this group includes young individuals, no fall-related fractures are present; however, the dataset provides additional DXA images and corresponding SSAM parameters suitable for evaluation of 3D reconstruction performance. From this cohort, 1097 DXA–SSAM pairs are available. This dataset was only used for validation during

the initial phase of model development and thus no results for MrPeak will be presented.

3.1.1 DXA Images

Figure 3.1 shows three example DXA images before preprocessing. The images represent proximal femur scans acquired in a standardized clinical setting. The femoral neck (Figure 3.2a), greater trochanter (Figure 3.2b), and parts of the femoral shaft (Figure 3.2c) are clearly visible. The cortical bone appears with higher intensity values, whereas the trabecular regions exhibit a more heterogeneous texture.

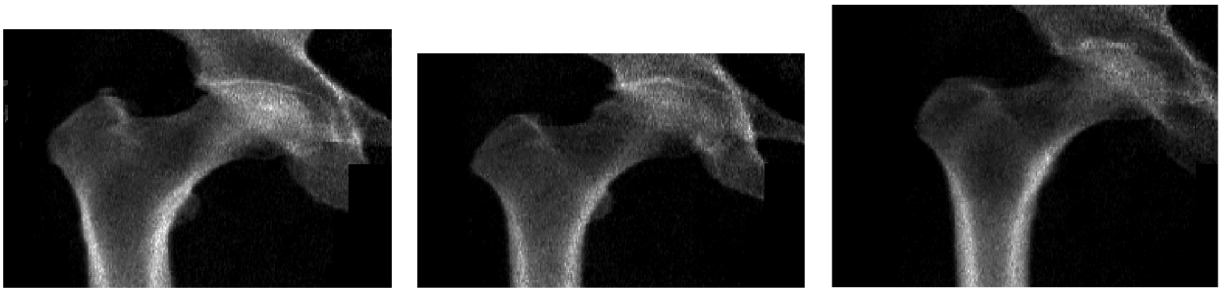


Figure 3.1: Example DXA images from the MrOS training dataset before preprocessing.

(a) Femoral neck. **(b)** Greater trochanter. **(c)** Femoral shaft.

Figure 3.2: Anatomy of the femur.

3.1.2 Reconstructions

The reconstruction files with the SSAM parameters used come from DXA2FEM optimizations. In total 40 parameters are estimated for each subject. The parameters include SSAM parameters for the femur and pelvis as well as scale factors and positions of the bones for aligning the projected DRR to the original images during DXA2FEM optimization, see Table 3.1.

For our dataset parameters 27, 28 and 29 have the value -10 (the bottom end of the allowed range) for all samples. Thus, we treat these as constants for our predictions. These parameters are translations of the pelvis for the 2D projection used for optimization and their final values are not included when calculating the FE mesh for bone strength prediction.

Parameter Group	**#**	**Type**	**Range**
Femur rotations $(\theta_x, \theta_y, \theta_z)$	1–3	Rotation parameters	[-10, 10]
Femur translations (t_x, t_y)	4–5	Translation parameters	[-5, 5]
Femur scale	6	Scale factor	[0.8, 1.2]
Femur SSAM modes	7–23	Shape and appearance coefficients	[-3, 3]
Pelvis rotations $(\theta_x, \theta_y, \theta_z)$	24–26	Rotation parameters	[-10, 10]
Pelvis translations (t_x, t_y, t_z)	27–29	Translation parameters	[-10, 10]
Pelvis scale	30	Scale factor	[0.8, 1.2]
Pelvis SSAM modes	31–40	Shape and appearance coefficients	[-3, 3]

Table 3.1: Summary of parameters contained in the reconstruction vector $\mathbf{y}$.

3.2 Methods

Two main models were developed and evaluated in this study: a baseline convolutional neural network (CNN) model and a transfer learning model. The network architecture for both models can be found in Appendix A. The models were developed independently, and results are presented for the final versions of each approach.

Model development followed an iterative experimental workflow including data preprocessing, architecture design, training, and evaluation. Multiple architectural configurations and training strategies were explored during development.

Hyperparameters were selected through structured experimentation, guided by domain knowledge and standard machine learning practice, with final configurations chosen based on validation performance.

Due to the substantial computational demands of training deep neural networks on large image datasets, exhaustive grid search or automated hyperparameter optimization methods were not feasible within the available resources.

3.2.1 Preprocessing

Before the DXA images can be used as input for the neural networks, they are preprocessed to obtain a consistent input format across datasets.

The DXA images from the MrOS dataset have widths in the range $[250, 285]$ pixels and heights in the range $[119, 202]$ pixels. Since the neural networks require square images of fixed size, an image resolution of 250×250 pixels was defined. All MrOS images with excessive width were cropped on the right-hand side. To obtain the correct height, zero-padding was applied at the bottom of the images.

The MrPeak dataset exhibits larger variation in image dimensions, with widths in the range $[250, 567]$ pixels and heights in the range $[103, 866]$ pixels. A subset of 35 images had a width of 567 pixels and was isotropically rescaled to a width of 250 pixels. The remaining images already had the correct width. Some of the larger images had heights exceeding their width; these images were cropped at the bottom to obtain square dimensions without removing clinically relevant anatomical structures. After rescaling and cropping, all MrPeak images were padded with zeros to reach the final size of 250×250 pixels.

The MrOS dataset has pixel intensities in the range $[0, 9.12]$, while the MrPeak dataset spans $[0, 15.58]$. Visual inspection of the images and the corresponding intensity distributions showed that pixel intensities above 5 are primarily artifacts (bright white shapes in otherwise dark regions of the images). Consequently, all pixel values greater than or equal to 5 were clipped and set to zero.

Since the pretrained network is trained on RGB images, the grayscale DXA images were converted to three-channel images by stacking the grayscale channel three times. The pixel intensities were then linearly rescaled to the range $[0, 255]$ to match the expected input format of the pretrained model.

Finally, the parameter arrays extracted from the reconstruction files were matched with the corresponding DXA images using patient identifiers to ensure correct pairing between input images and target parameters.

Before model development, both datasets were divided into training (70%), validation (10%), and test (20%) sets using a patient-wise split, ensuring that no images from the same patient appeared in more than one subset.

Given the total MrOS cohort of 1583 patients, this resulted in 1108 patients for training, 158 for validation, and 317 for testing. This distribution was selected to provide sufficient data for model optimization while preserving a statistically meaningful and fully independent test cohort for reliable evaluation of generalization performance. While no universally optimal split exists, the chosen ratio represents a commonly adopted and well-balanced compromise in deep learning studies.

3.2.2 Baseline Network

The baseline model consists of a compact convolutional neural network used as a shared feature extractor, followed by multiple regression heads. The convolutional backbone comprises four convolutional blocks, each consisting of a two-dimensional convolution, batch normalization, and a ReLU activation function. All convolutional layers use a kernel size of 7×7 with stride 1. Each block is followed by a max-pooling layer with kernel size 3×3 and stride 2, progressively reducing the spatial resolution while increasing the number of feature channels (32, 64, 128, and 256).

After the final convolutional block, global average pooling is applied, reducing each of the 256 feature maps to a single scalar value. This produces a 256-dimensional feature vector representing the input image.

The resulting feature vector is then processed by a set of independent regression heads. Each head is implemented as a two-layer multilayer perceptron (MLP) consisting of a fully connected layer with 128 hidden units followed by a ReLU activation and a final linear layer producing a single scalar output. One regression head is used per target parameter, resulting in 40 independent prediction branches.

The use of independent regression heads allows each output parameter to learn parameter specific combinations from the shared feature vector. This reduces interference between parameters and enables the model to specialize for each target.

For training all target parameters were standardized with mean and standard deviation for each of the parameters from the training set. Standardization stops the network from prioritizing large-scale parameters as these would otherwise have larger errors and gradients.

Huber-loss was used as the loss function with no custom weights. The Adam optimizer is used. The model is trained with batch size 16 and a base learning rate of $0.05 \cdot \frac{BatchSize}{256} = 0.031$ and scheduling which warms up from 0 to 0.003125 in 5 epochs and then decaying by a factor of 0.5 when validation loss has not decreased in seven epochs. The training stops when the learning rates reaches its minimum which is set to 10^{-5}.

3.2.3 Transfer Network

The transfer learning network is based on the EfficientNet architecture, specifically EfficientNetV2B2 [18]. Several other pretrained architectures were also evaluated, including EfficientNetB2 [28], MobileNetV2 [29], and Xception [30].

Although these architectures differ in their internal design, they are all well-established convolutional neural networks pretrained on large-scale natural image datasets and commonly used for transfer learning. In practical applications, they follow the same overall principle of learning general visual features that can be adapted to domain-specific tasks

such as medical imaging.

Based on both our experimental results and findings reported in previous studies [19, 23], there is no consistent evidence that any of these architectures is inherently superior for medical imaging tasks in general. Performance differences are typically task- and dataset-dependent rather than architecture-dependent. In our experiments, the models achieved comparable results in terms of test loss and mean absolute error (MAE), with EfficientNetV2B2 showing a slight but consistent advantage. It was therefore selected as the primary pretrained backbone for this study.

The full transfer network consists of the pretrained EfficientNetV2B2 network, without the fully connected classification layer, connected to a regression head comprising three dense layers. The output layer contains 40 nodes, one for each SSAM parameter, and uses a linear activation function. To reduce overfitting, input images are augmented during training using small random rotations, translations, and additive noise.

The target parameters were normalized prior to training using a hybrid strategy. The scale factors were standardized using Z-score normalization. All remaining SSAM parameters were normalized to the range [0,1] using min–max scaling. Normalization parameters were computed from the training set and applied consistently to validation and test data.

The loss function used for training is a custom weighted Huber loss. The femoral SSAM parameters have the greatest influence on the reconstructed 3D geometry and were therefore assigned higher weights. The first SSAM parameter corresponds to the first principal component and thus represents the largest mode of variation. Consequently, it was assigned the highest weight (17), with weights decreasing exponentially to 1 for the final femoral SSAM parameter. All remaining parameters were assigned a weight of 1. The Huber loss was chosen due to its robustness to outliers while maintaining sensitivity to small errors.

Training was performed in two phases. In the first phase, all pretrained backbone weights were frozen and only the regression head was trained. In the second phase, the final block of the pretrained network was unfrozen to fine-tune the model. The batch size used for this network is 16. The Adam optimizer was used, with an initial learning rate of $5 \cdot 10^{-4}$ during the first phase and $5 \cdot 10^{-5}$ during fine-tuning. During training, the learning rate was reduced when the validation loss plateaued, and training was terminated early if performance ceased to improve or the learning rate reached a predefined minimum.

3.2.4 Bone Strength Estimation

From the predicted SSAM parameters, subject-specific 3D proximal femur geometries were reconstructed and converted into finite element (FE) meshes using the DXA2FEM pipeline implemented in MATLAB. The resulting meshes consisted of tetrahedral elements with element-specific volumetric bone mineral density (vBMD) values derived from the reconstruction. In accordance with the procedure described in Section 2.1.3, the elements

were converted to quadratic tetrahedra and the vBMD values were transformed into elastic moduli using established density–elasticity relationships. A minimum Young's modulus was enforced at the periosteal surface to reduce the impact of potential reconstruction artifacts. An anatomical reference system was defined automatically to ensure consistent alignment and application of boundary conditions across subjects.

Linear elastic, quasi-static FE simulations were then performed in Abaqus to mimic sideways fall loading conditions. Ten loading configurations were considered, spanning 0°–30° in both adduction and internal rotation, including the commonly adopted 10° adduction and 15° internal rotation configuration. Fracture load was determined using a principal strain limit criterion, where failure was assumed when predefined strain thresholds were exceeded. For each subject, the resulting fracture load from the simulations was used as the predicted bone strength.

Since the DXA2FEM finite element simulations are computationally demanding, an alternative approach was also investigated. Instead of predicting SSAM parameters followed by mesh generation and FE simulation, the neural networks were trained to predict bone strength directly from the DXA images.

To enable direct bone strength prediction, minor architectural modifications were applied to both networks. For the baseline network, the 40 parallel regression heads used for SSAM parameter prediction were replaced with ten regression heads producing ten scalar outputs corresponding to each configuration of the FEM simulation. For the transfer network, the original 40-dimensional output layer was replaced by a fully connected layer connected to ten linear output nodes.

This direct regression approach bypasses SSAM parameter estimation, mesh reconstruction, and FE simulation, resulting in a substantially reduced computational cost.

3.2.5 Evaluation Methods

The performance of the neural networks was evaluated at multiple levels. First, model performance was assessed using standard regression metrics computed on the predicted SSAM parameters. Second, the geometric accuracy of the reconstructed femur was evaluated by comparing the predicted FE mesh to the reference DXA2FEM mesh. Third, the biomechanical validity of the predictions was assessed by comparing predicted bone strength to the corresponding DXA2FEM results.

For all evaluations, the DXA2FEM pipeline was used as the reference method and DXA2FEM results are treated as ground truth.

Loss-Based Evaluation

During model development, training and validation loss were used to compare different network architectures and hyperparameter configurations. Performance was quantified using mean square error (MSE), mean absolute error (MAE), and Huber loss. In addition, prediction errors for individual SSAM parameters were analyzed.

FE Mesh Evaluation

To assess geometric reconstruction accuracy, the predicted SSAM parameters were used to generate subject-specific FE meshes. Since the predicted and reference meshes share identical topology (same number of nodes and elements), direct node-to-node comparison was possible. The mean node-to-node distance and the mean element-wise volumetric bone density (ρ) error were used as quantitative performance metrics.

Bone Strength Evaluation

To evaluate biomechanical consistency, FE simulations identical to those used in the DXA2FEM [7] pipeline (Section 2.1.3) were performed using the predicted meshes. The resulting bone strength values were compared to the corresponding DXA2FEM derived strengths.

3.2.6 Synthetic DRRs

During each iteration of the DXA2FEM optimization procedure, digitally reconstructed radiographs (DRRs) are generated by projecting candidate 3D reconstructions onto a 2D plane. The initial 3D reconstructions are constructed from randomly initialized SSAM parameter configurations.

In principle, this mechanism can be exploited to generate synthetic training data. By sampling anatomically plausible SSAM parameter vectors and projecting the corresponding 3D reconstructions, synthetic DRRs can be created together with their known ground-truth parameters. If the parameters are sampled carefully within realistic bounds, both the reconstructed geometries and the resulting DRRs remain anatomically consistent.

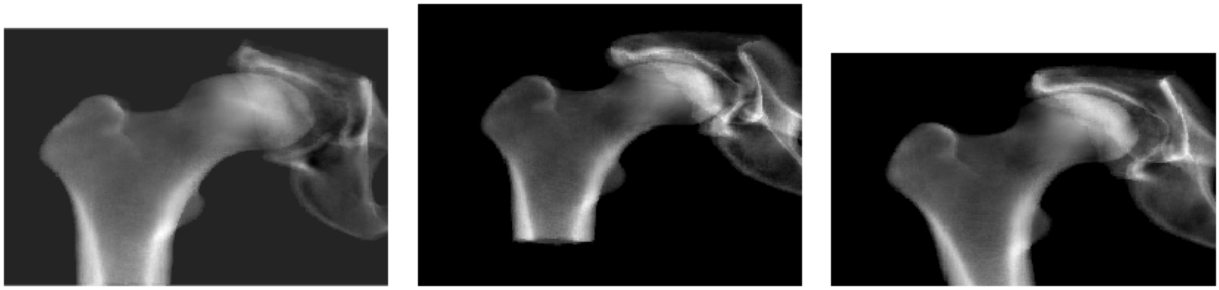


Figure 3.3: Examples of artificially generated DRRs.

To investigate this approach, 10 000 SSAM parameter vectors were generated. Each parameter was sampled independently from a normal distribution estimated from the empirical distribution of the real dataset. For each sampled parameter vector, a corresponding DRR was generated using the same forward projection function employed in the DXA2FEM optimization procedure (Figure 3.3).

Two experimental setups were evaluated. In the first experiment, the synthetic DRRs were split into training, validation, and test sets. The network was trained and evaluated exclusively on synthetic data. Performance was assessed using the same loss functions and 3D reconstruction error metrics as for the real DXA images. In the second experiment, all 10 000 synthetic DRRs were used as training data, while the original MrOS DXA images were used for validation and testing. This setup evaluates whether pretraining on synthetic DRRs improves generalization to real DXA images.

It should be clarified that this was not intended as a new research objective, but rather as an exploratory analysis within the existing framework. The motivation was to assess whether synthetic DRRs could be used directly as training data to mitigate the limited availability of labeled DXA images. The experiments showed that additional adjustments, such as improved realism and domain adaptation, would likely be necessary for effective transfer to real DXA data. Developing and validating such modifications is outside the scope of this thesis. Nevertheless, the approach is of methodological interest, and selected results are therefore presented.

4 Results

4.1 SSAM Parameters

The first stage of the evaluation concerns the prediction of SSAM parameters. During model development, architectures and hyperparameters were selected based on training, validation, and test loss, in addition to regression metrics such as MSE and MAE. Across all reported metrics, the transfer network produced larger errors compared to the baseline network (increase of $15.6\% - 19.9\%$), see Table 4.1.

Analysis of individual SSAM parameters showed that certain parameters were predicted more accurately than others. In particular, the global scale factors and the parameters corresponding to the first femoral principal components exhibited lower prediction error across both models, see Appendix B. Conversely, parameters associated with higher-order modes of variation tended to show larger deviations.

For both networks, the predicted SSAM parameter distributions were generally more concentrated around the population mean compared to the reference values, indicating a tendency toward conservative predictions. No clear signs of overfitting were observed, the loss was similar for the training, validation and test datasets.

Model	**MAE**	**MSE**	**Huber** $\delta = 1$
Baseline Network	0.705	1.778	0.433
Transfer Network	0.823	2.056	0.519

Table 4.1: Comparison of parameter prediction metrics on the test set.

4.2 FE Mesh

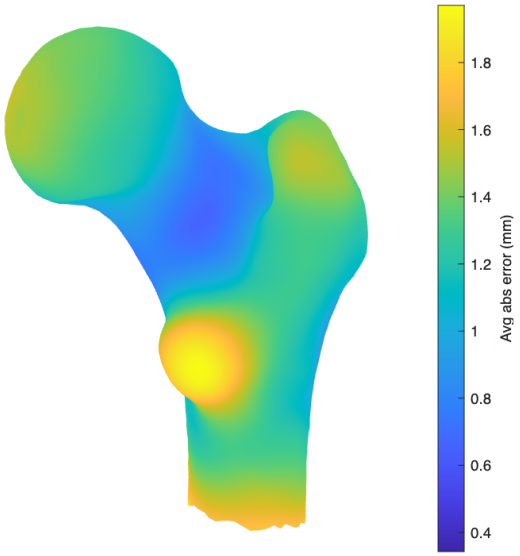
The baseline network achieved lower mean node-to-node distance and lower mean element-wise density error compared to the transfer network. Although the overall performance difference is modest, the transfer model consistently produced slightly larger (+13%) reconstruction errors.

Table 4.2 summarizes the geometric reconstruction performance of the FE meshes generated from the predicted SSAM parameters. The minimum and maximum values correspond to the best and worst-performing test samples, respectively.

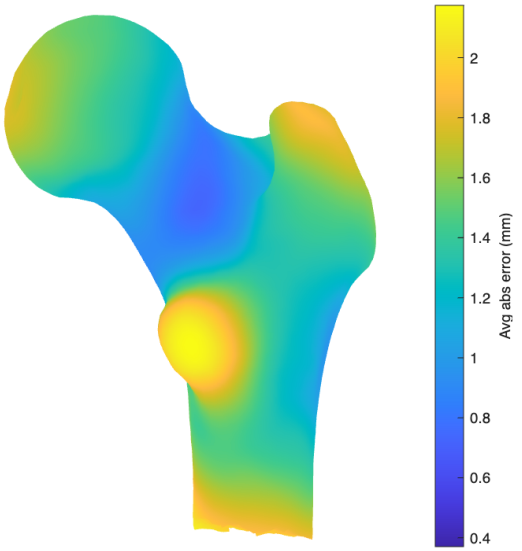
Metric	Baseline Network	Transfer Network
Mean distance [mm]	1.132	1.283
Min distance [mm]	0.355	0.453
Max distance [mm]	3.548	3.399
Mean ρ MAE (elem-wise) [g/cm^3]	0.0438	0.0490
Min ρ MAE [g/cm^3]	0.0142	0.0237
Max ρ MAE [g/cm^3]	0.0937	0.1172

Table 4.2: FE mesh reconstruction performance metrics for the baseline network and transfer network evaluated on the test set.

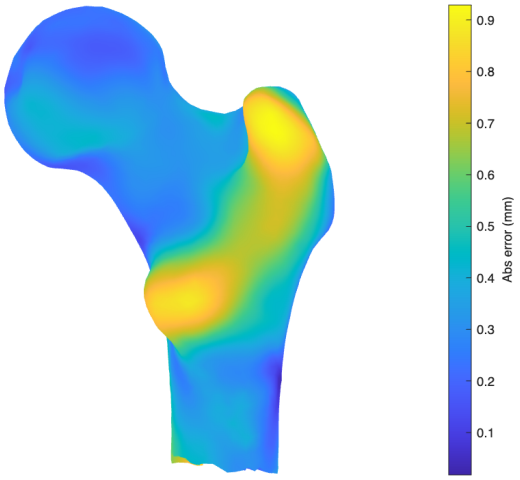
For both models, the reconstructions were most accurate in the femoral neck region, where the lowest mean node-to-node errors were observed. Larger errors were primarily located in the surrounding trochanteric regions. The first column of Figure 4.1 shows the spatial distribution of the mean node-to-node error across all test samples. The second and third columns present examples of the best and worst reconstructions for each model, respectively, based on mean node-to-node error (note the varying scale).



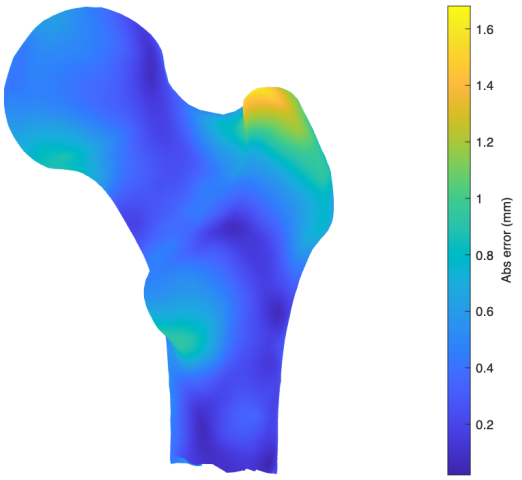
Baseline – Mean



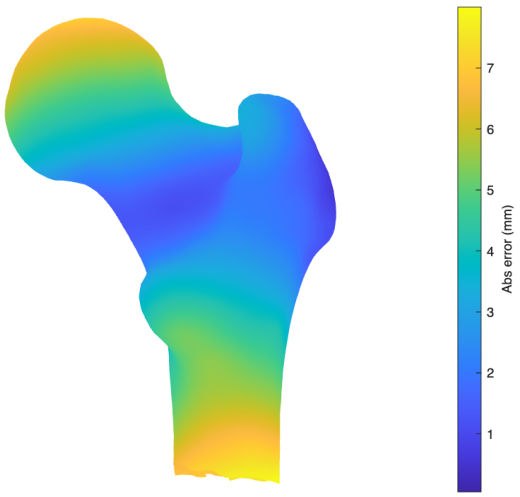
Transfer – Mean



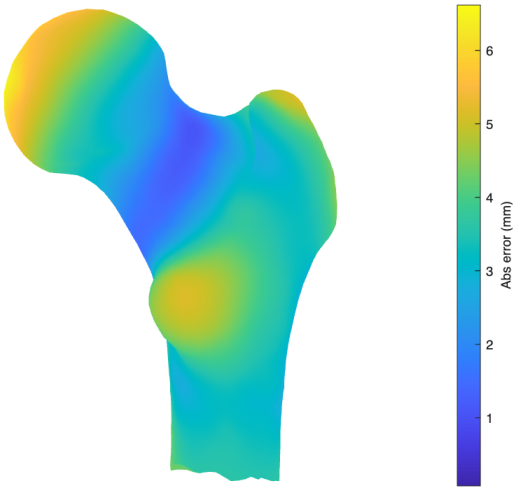
Baseline – Best



Transfer – Best



Baseline – Worst



Transfer – Worst

Figure 4.1: Node-to-node reconstruction error on the FE mesh.

The distribution of the mean node-to-node distance across all test samples for the two networks is shown in Figure 4.2.

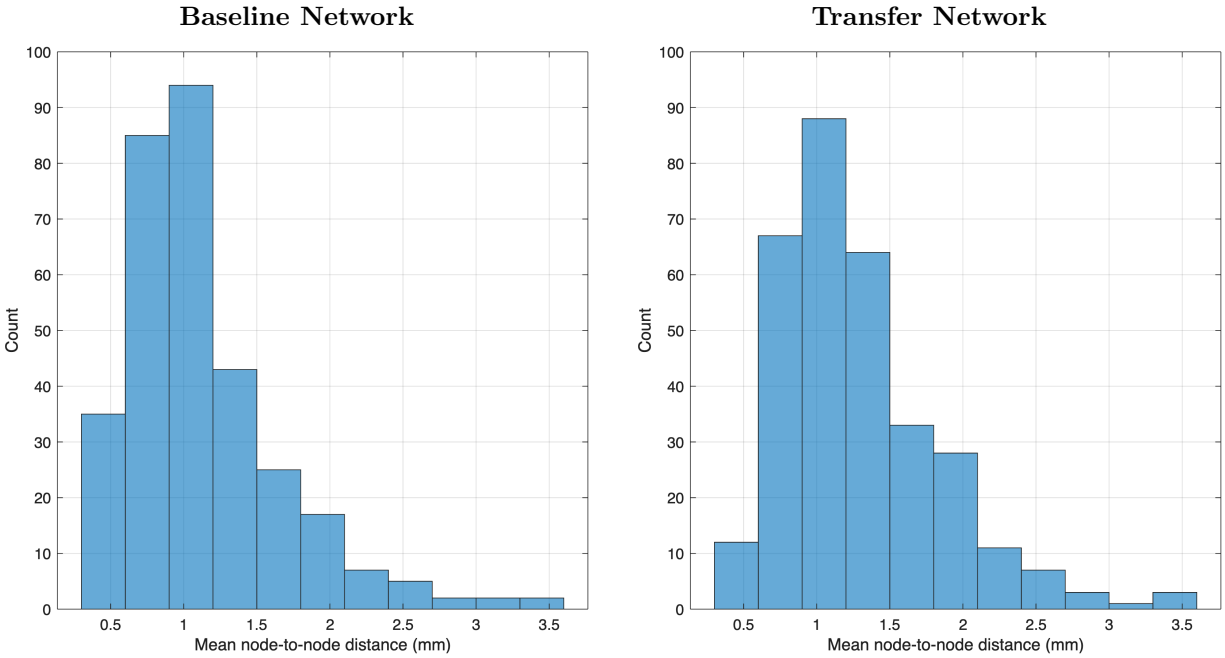


Figure 4.2: Mean node-to-node error of each patient.

4.3 Bone Strength

The bone strength predictions in this section come from FE simulations using the standard loading configuration of 10° adduction and 15° internal rotation, the rest of the configurations are shown in Appendix C. The average error metrics and the coefficient of determination (R^2) for the standard configuration are shown in Table 4.3 and the mean for all the configurations are shown in Table 4.4.

With FE simulations, the two models performed similarly with a mean absolute error of 0.666 kN for the baseline network and 0.664 kN for the transfer network. When directly predicting bone strength, the models achieved a mean absolute error of 0.397 kN for the baseline network and 0.495 kN for the transfer network.

Model	**MAE** [**kN**]	**RMSE** [**kN**]	**MAPE** [%]	R^2
Baseline Network (FEM)	0.666	0.801	13.12	0.725
Transfer Network (FEM)	0.664	0.843	12.55	0.428
Baseline Network (Direct)	0.397	0.503	7.31	0.761
Transfer Network (Direct)	0.495	0.633	9.03	0.731

Table 4.3: Comparison of predicted and reference bone strength for all evaluated models. (10° add., 15° int. rot.)

In terms of average error metrics across all configurations, the baseline model performs better than the transfer model and both models perform better when directly predicting bone strength, see Table 4.4.

Model	**MAE** [**kN**]	**RMSE** [**kN**]	**MAPE** [%]	$\mathbf{R^2}$
Baseline Network (FEM)	0.587	0.712	11.97	0.707
Transfer Network (FEM)	0.605	0.775	12.11	0.420
Baseline Network (Direct)	0.381	0.486	7.53	0.739
Transfer Network (Direct)	0.469	0.603	9.14	0.607

Table 4.4: Comparison of mean prediction performance across all loading configurations for the evaluated models.

4.3.1 FEM Predictions

As can be seen in Table 4.3 the models perform similarly in terms of average error. Figure 4.4 and Figure 4.3 shows trend of predictions versus label as well as distribution of the errors. The baseline network has a R^2 of 0.725 and has a clear bias towards overestimating which can be seen in Figure 4.4 where the majority of prediction errors are positive (see also Appendix C.1). The transfer network has an R^2 of 0.428 without a clear bias. Meaning the baseline network has better correlation than the transfer network but because of the bias they still have approximately the same average errors.

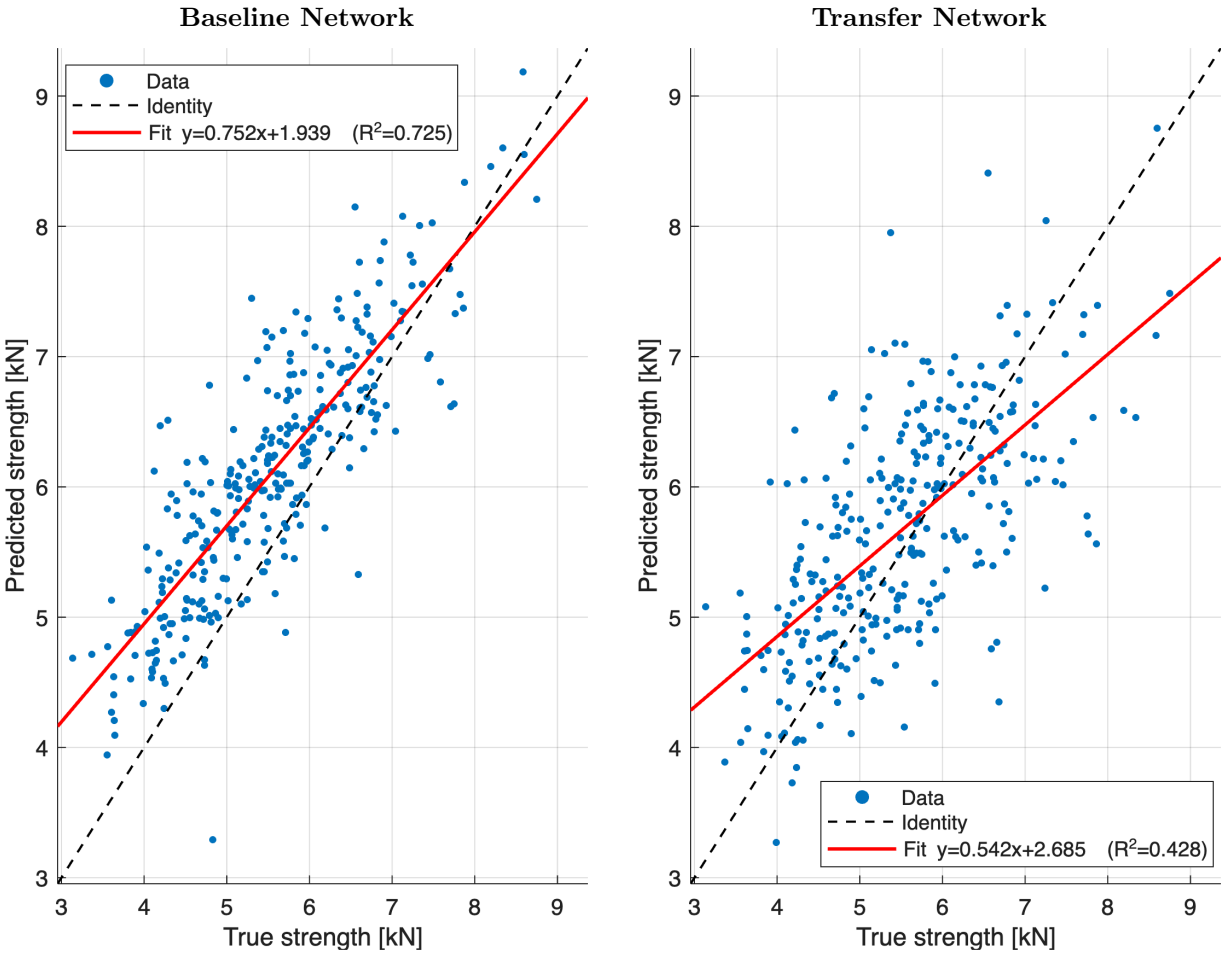


Figure 4.3: Scatter plot of the true and predicted bone strengths using the DXA2FEM pipeline for both models. (10° add., 15° int. rot.)

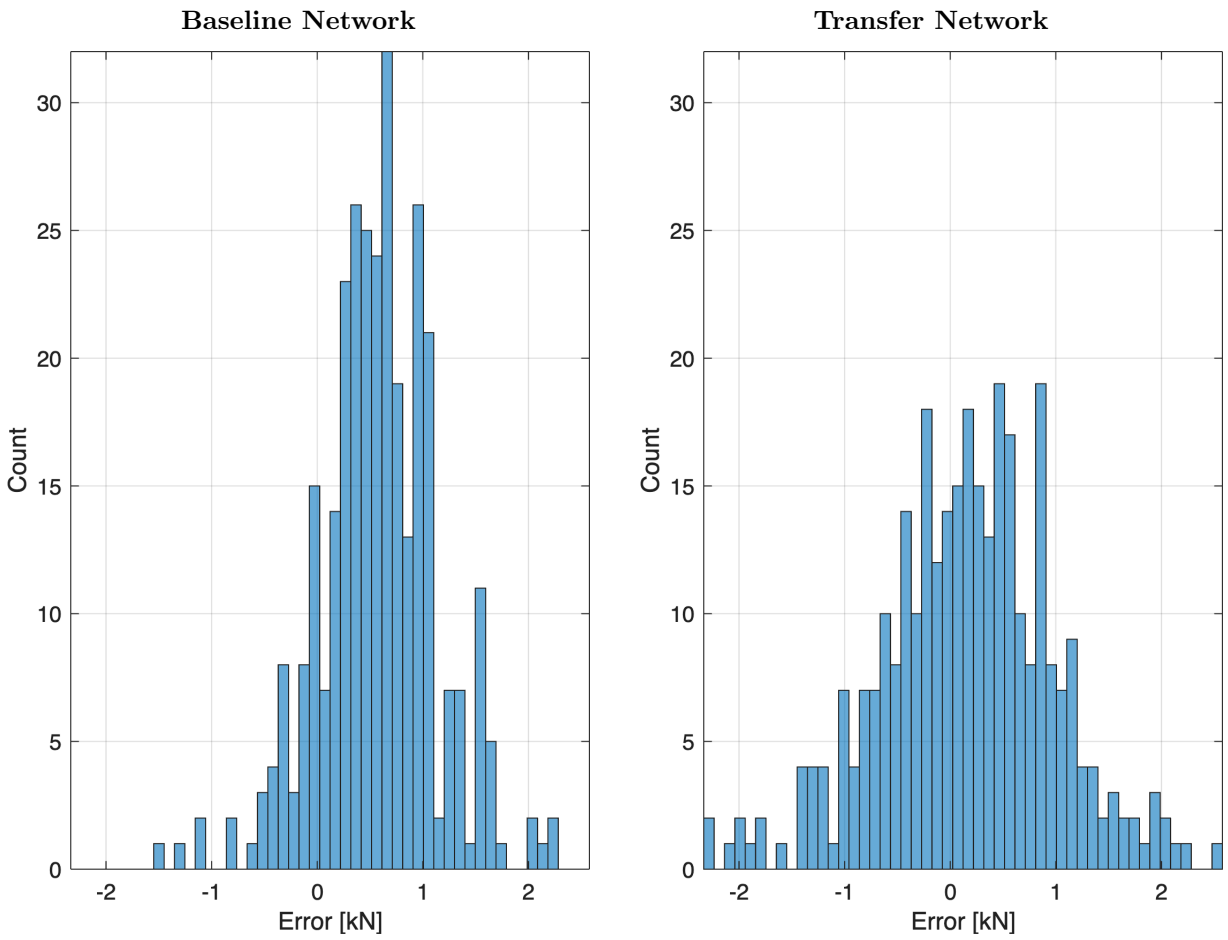


Figure 4.4: Errors of the FEM bone strength predictions. (10° add., 15° int. rot.)

4.3.2 Direct Predictions

Figure 4.5 shows the relationship between predicted bone strength and the DXA2FEM reference values for the baseline and transfer networks. The figure shows the results for the simulations with the initial configuration of the femur of 10° adduction and 15° internal rotation. The results for the other configurations are similar, see Appendix C.2. A linear regression fit is included for each model.

For both networks, predicted strength values follow the overall trend of the reference values without any clear bias, see Appendix C. The regression slope, intercept, and coefficient of determination (R^2) indicate stronger agreement for the baseline network compared to the transfer network.

The distribution of the errors is centered around 0 for both networks, see Figure 4.6.

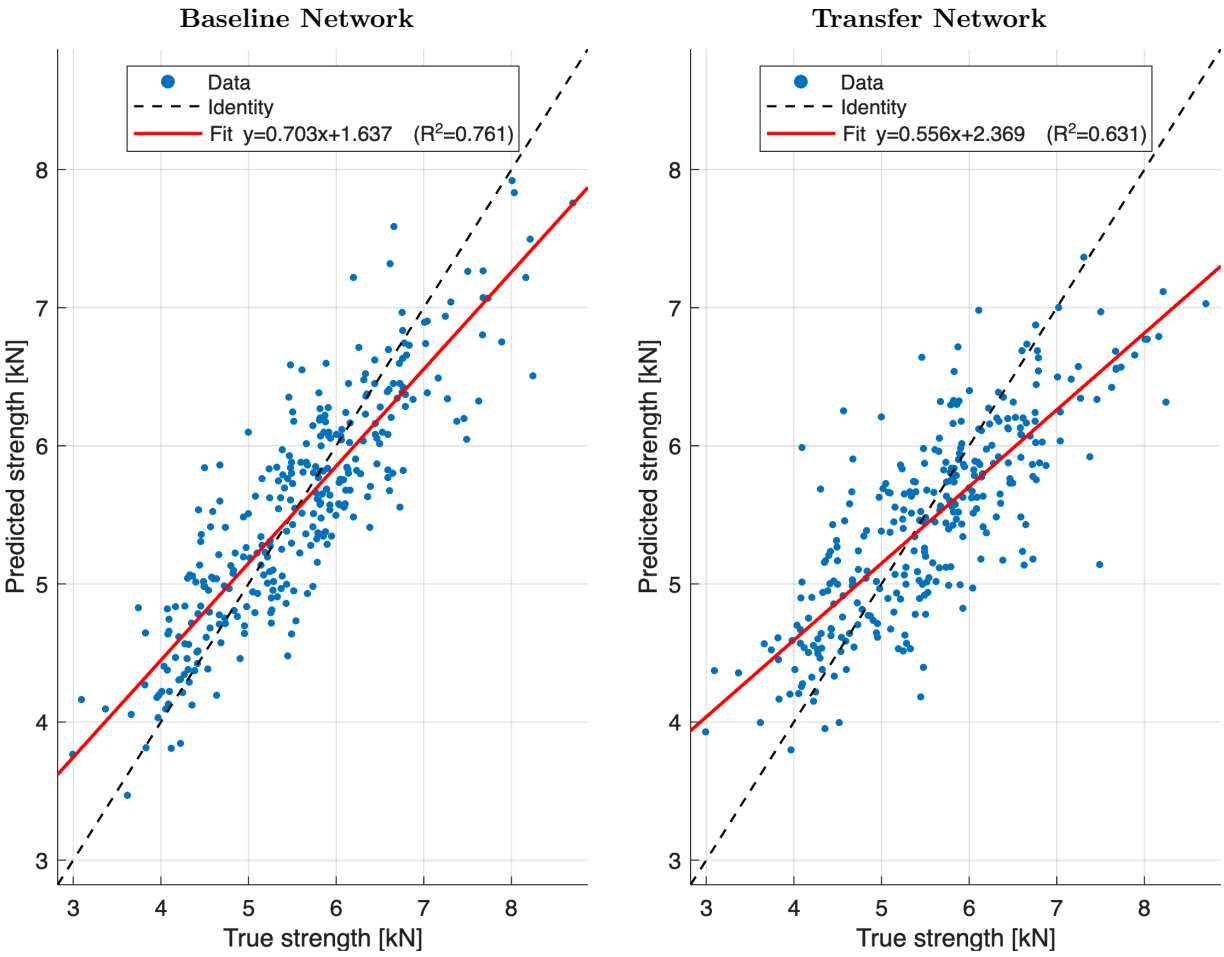


Figure 4.5: Scatter plot of the true and predicted bone strengths, with predictions directly from DXA images. (10° add., 15° int. rot.)

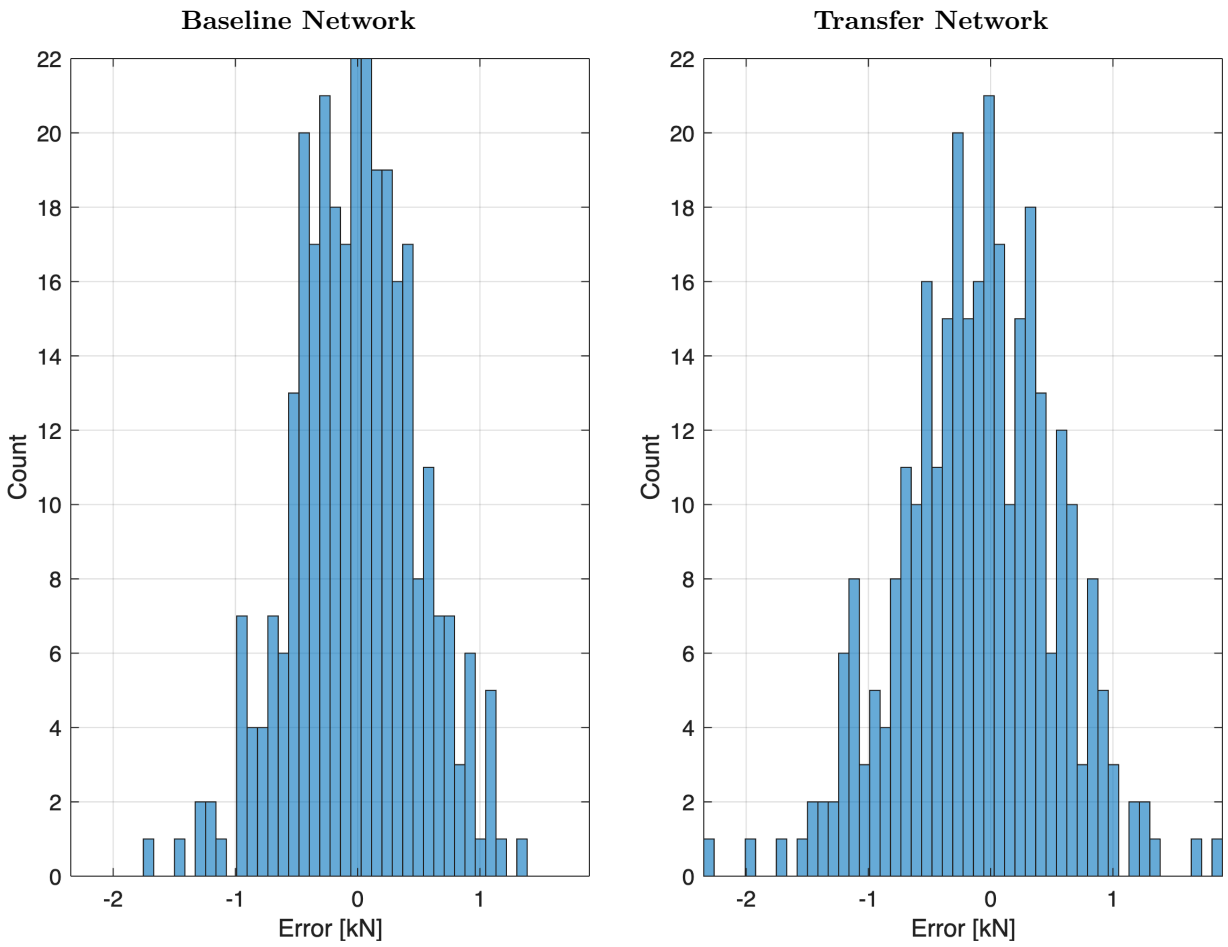


Figure 4.6: Errors of the bone strength predictions directly from the DXA images. (10° add., 15° int. rot.)

4.4 Synthetic DRRs

To evaluate the effect of training on synthetic data, parameter prediction performance was compared across models trained and tested on synthetic DRRs and real DXA images. Table 4.5 summarizes the results, with the corresponding DXA → DXA performance included for reference.

When both training and testing were performed on synthetic DRRs (DRR → DRR), both the baseline and transfer networks achieved substantially lower prediction errors compared to the DXA → DXA setting. The MAE is reduced by 78.9% for the baseline network and 31.7% for the transfer network.

When training was performed on synthetic DRRs and testing on real DXA images (DRR → DXA), prediction errors increased relative to the DXA → DXA configuration. However, the performance degradation was smaller for the transfer network than for the baseline

network. The MAE increased with 43.5% for the baseline network and only 10.7% for the transfer network. In this setting, the transfer model achieved lower prediction error than the baseline model, with a MAE of 0.911 compared to 1.012 for the baseline network.

Model	**MAE**	**MSE**	**Huber** $\delta = 1$
Baseline Network (DRR → DRR)	0.149	0.086	0.036
Transfer Network (DRR → DRR)	0.562	0.902	0.299
Baseline Network (DRR → DXA)	1.012	3.368	0.694
Transfer Network (DRR → DXA)	0.911	2.617	0.600
Baseline Network (DXA → DXA)	0.705	1.778	0.433
Transfer Network (DXA → DXA)	0.823	2.056	0.519

Table 4.5: Comparison of parameter prediction metrics on the test set with models trained on synthetic DRR images.

5 Discussion

5.1 Prediction of SSAM Parameters

The results show that the models can detect features that predict SSAM parameters. This is particularly evident for the first femoral principal components (see Appendix B) which is both expected and promising since these are the parameters that affect the reconstruction the most. The spatial distribution of the reconstruction errors shows that the smallest errors occur in the femoral neck region. This could be due to a lower variation in the shape of this part of the bone within a population. Another explanation is that the target function of the optimization algorithm in DXA2FEM prioritizes this specific region. As a result, features in the femoral neck may have a stronger correlation with the optimized parameter values, leading to improved reconstruction accuracy.

As for the scale of the errors, it is difficult to evaluate how "good" it is. The reconstruction depends on the combination of 40 parameters and subsequently the predicted bone strength depends on the mesh consisting of hundreds of thousands of nodes. This makes the connection between parameter prediction error and the final result difficult to interpret quantitatively. What can be seen is that between the two models, the baseline network is approximately 15% better than the transfer network in terms of parameter prediction errors and about 13% better in the reconstruction both in terms of node-to-node distance and element-wise volumetric bone density error. However, improvements in parameter prediction are not expected to translate into proportional improvements in reconstruction, as some parameters have a greater influence on reconstruction than others. The parameter prediction error is definitely an indication of how good the model is, but the reconstruction metrics are needed to truly evaluate it.

In relation to the work of Ha et al. [23], a direct quantitative comparison of prediction accuracy is not feasible. First, their model regresses only five SSM deformation parameters, whereas the present work predicts 40 SSAM parameters. The dimensionality and complexity of the regression task are therefore substantially different. Second, the reported reconstruction error in Ha et al. is expressed as RMS point-to-surface distance, while this study evaluates reconstruction using node-to-node distance and element-wise volumetric bone density error. These metrics are not directly interchangeable and reflect different aspects of geometric fidelity. Finally, Ha et al. validated their framework on femoral phantoms, whereas the present study is based on clinical patient data, which introduces additional variability due to anatomical diversity, imaging conditions, and optimization noise in the target parameters. Consequently, although both approaches address regression of statistical model parameters from radiographic images, methodological and experimental differences preclude a direct numerical comparison of accuracy.

A fundamental problem for these models is that the target parameters are obtained through an optimization algorithm and are therefore not guaranteed to correspond to the global minima. Because of this, the same or very similar input features may sometimes be associated with different parameter values depending on the outcome of the optimization process. This creates noise in the training labels which will disturb the network's ability to predict accurately. It is difficult to quantify how much of a factor this is, as the reliability of the optimization algorithm in reaching the global minima is unknown.

5.2 Bone Strength Prediction

The FEM-based predictions from the baseline model exhibit strong correlation across all configurations ($R^2 > 0.683$, see Appendix C). However, the mean absolute error remains relatively high compared to the baseline model that directly predicts bone strength (MAE = 0.587 vs. 0.381, see Table 4.4).

The average R^2 values for the two methods are considerably closer (0.707 vs. 0.739), indicating that both models explain a similar proportion of the variance in the data. This suggests that the elevated error in the FEM-based approach is primarily due to systematic bias rather than an inability to capture underlying relationships. Consequently, if the source of this bias can be identified and mitigated, the FEM-based approach has the potential to achieve comparable performance in terms of average error.

Even though a model that directly predicts bone strength would be the most effective in time and complexity, we believe that the most accurate results can be achieved through parameter prediction. It is more probable that the CNN for the direct prediction method learn features that correlate with image-derived bone density measures, such as DXA-based aBMD, rather than capturing the complexity of 3D-reconstruction and FE simulations. The parameter prediction operates on the same input as the optimization algorithm in the DXA2FEM pipeline and should therefore be able to capture the features connected to the parameters much easier.

5.3 Baseline and Transfer Network Performance

The two evaluated networks showed similar performance, although the baseline CNN achieved slightly better results for almost all metrics (see Tables 4.1, 4.2, 4.4). The potential benefits of transfer learning include improved generalization when training data is limited, shorter training times, and a reduced risk of overfitting. In our case, these advantages did not have a substantial impact compared to the baseline CNN.

The DXA images are highly standardized, with the femur located close to the center of the image in almost all cases. This allows a relatively shallow CNN trained from scratch

to learn where relevant features are located without needing strong spatial invariance. Networks pretrained on ImageNet are designed to recognize objects under varying orientations and positions, for example identifying an object even if it is mirrored or appears in a different part of the image. Such general functionality is not necessary for this task. Instead, pretrained feature representations optimized for natural RGB images may be less suited for grayscale radiographic textures.

Shorter training times were also not an important factor in this study. Training on personal computers required approximately one to two hours for both networks, while training on the LUNARC computing cluster took less than fifteen minutes in both cases. The difference in training time therefore had little practical impact.

The reduced risk of overfitting associated with transfer learning was also limited. For both models, the predicted parameters tended to be closer to the population mean than the reference values. These conservative predictions suggest that the main challenge for the networks is extracting sufficient information from the images rather than preventing overfitting.

When training on artificial DRR images and evaluating on DXA images, the transfer network performed better than the baseline network, see Table 4.5. In this case, the improved generalization of the pretrained model becomes beneficial since the test images differ substantially from the training data.

5.4 Dataset Limitations

Although further optimization of both models would likely have been possible, larger performance improvements could probably be achieved by increasing the amount and quality of available training data. A dataset consisting of 1583 images is relatively small for image-based learning tasks. For comparison, large-scale image datasets such as ImageNet are trained on more than one million images. The limited dataset size therefore restricts the ability of the networks to learn more robust feature representations.

Since the models are trained using results generated by the DXA2FEM pipeline, the goal was not to improve the predictive accuracy of DXA2FEM itself. Instead, the objective was to investigate whether similar results could be reproduced using machine learning while significantly reducing the computational cost. If the training data had instead consisted of FE meshes derived from CT scans, it might have been possible to improve the accuracy beyond that of DXA2FEM. However, CT-based reconstructions are considerably more difficult and expensive to obtain, involve higher radiation exposure for patients, and require manual work from a trained engineer to fit the FE mesh to the 3D geometry. In addition, if the number of available samples is limited, the potential benefits of higher-quality CT data may not outweigh the advantages of using the more accessible DXA images.

Another limitation is that only images from male patients were used for training. Although the DXA2FEM pipeline is applicable to both men and women, there is no guarantee that the trained networks would generalize equally well to female patients. Anatomical differences and differences in bone density distributions may require additional training data to achieve comparable performance.

Furthermore, the available datasets primarily consist of specific age groups, which may also limit generalization to broader clinical populations. Additional data covering a wider demographic range would likely improve model robustness.

5.5 Synthetic DRRs

The results from the networks trained on synthetic DRRs show promising performance, although further work is required before the approach can fully replace training on real DXA data.

The DRR $\rightarrow$ DRR results are significantly better than the DXA $\rightarrow$ DXA results for both models, see Table 4.5. The lack of noise, acquisition artifacts, and other imperfections in the synthetic DRRs is likely a major reason for this performance improvement. In addition, the larger training dataset probably contributed to the lower prediction errors (Table 4.5). The very low prediction error achieved by the baseline network suggests that this architecture benefits strongly from increased data availability. This observation is consistent with the expected advantages of transfer learning primarily appearing when training data is limited.

The DRR $\rightarrow$ DXA results are worse than the DXA $\rightarrow$ DXA results, but the performance gap is relatively small for the transfer network ($10.7\% - 27.3\%$ worse compared to $43.5\% - 89.4\%$ for the baseline network, see Table 4.5). This indicates improved robustness to domain shift when using pretrained features. Several factors could likely improve the DRR $\rightarrow$ DXA performance. Independent sampling of SSAM parameters may produce anatomically unrealistic combinations, and sampling that preserves parameter covariance could improve the realism of the generated femur models. In addition, the synthetic DRRs are considerably cleaner than real DXA images, which include scanner noise, soft tissue variation, and acquisition artifacts. Incorporating more realistic noise modeling or image augmentation during DRR generation would likely reduce this domain gap.

Overall, the results indicate that training with synthetic DRRs is a promising approach. This is particularly relevant in medical imaging applications where access to large annotated datasets is often limited.

5.6 Practical Considerations and Future Applications

Before the methods presented in this thesis can be used in a practical setting, improvements are required in several areas. A recent study showed that although DXA2FEM achieves a higher AUC for hip fracture prediction than aBMD (0.74 compared to 0.69), the difference is not statistically significant [31]. Since the mean absolute percentage error (MAPE) of our bone strength predictions ranges from 6.19% to 15.79%, it is unlikely that any of our proposed methods would achieve fracture risk predictions that are more accurate than those obtained using aBMD alone. In such a case, the trade-off of reduced accuracy in exchange for faster computation would not be justified. To confirm this, and find out what accuracy is required, fracture risks would need to be calculated and compared to predictions from other methods and the actual outcomes.

This hypothesis could be further evaluated by calculating the absolute fracture risk (ARF0) [31] using bone strength values predicted by our models. However, this analysis was not performed due to the limited number of hip fracture cases available in our test set.

Another potential application of the predicted SSAM parameters is to use them as initial estimates for the DXA2FEM optimization procedure. Providing improved initial parameters could accelerate the convergence of the optimization algorithm while preserving the accuracy of the final reconstruction. We did not test this because some minor adjustments to the DXA2FEM code would be required.

5.7 Ethics

Applying machine learning to medical prognosis raises several ethical considerations related to data use, model reliability and potential practical applications.

The data used in this study were anonymized prior to access. No personally identifiable information was available to models, and subjects were only represented by an index along with the date when the DXA-scan was taken.

A significant limitation of this study is the composition of the training data, which consists exclusively of male subjects within relatively narrow age ranges. This introduces a risk of model bias, as this may not generalize to female populations or individuals outside the studied age groups. From an ethical perspective, deploying such models without addressing these limitations could lead to unequal performance across patient groups and potentially contribute to disparities in clinical decision-making. Expanding the dataset to include more diverse populations is therefore not only a technical improvement but also an ethical necessity.

When using any predictive model, it is important to consider the accuracy of the predictions before drawing conclusions or applying them in practice. Even when major errors

are rare, it can have significant consequences in a medical context. Incorrect predictions may lead to inappropriate clinical decisions, such as unnecessary treatment or failure to provide needed care. For this reason, clinical decisions with substantial consequences should always rely on the most accurate and validated methods available.

This consideration is particularly important for machine learning models such as those developed in this study. These models are inherently difficult to interpret. As a result, there is an increased reliance on empirical validation rather than understanding how the model came to its prediction. Without sufficient validation, there is a risk that the model may contribute to incorrect conclusions that can negatively affect patient outcomes. If a model such as these are used in practice the results must be so clear that the only reason wrongful treatment is given is if a clinician has misinterpreted the data.

When using any model that predicts an outcome it is important to be aware of the accuracy of the prediction when drawing conclusions. Incorrect predictions even if they are rare can have large consequences if acted upon. For example in a medical setting this could result in giving unnecessary care or even no care at all to a subject. Therefore the care given should always be based on the most accurate method if a decision with large consequences is made. More specifically for machine learning models such as the one developed in this study where it is difficult if not even impossible to derive why it made a certain prediction It is extremely important that the model has went through rigorous testing of its robustness in order to ensure that it doesn't contribute to a conclusion that will effect the patient negatively.

6 Conclusions and Future Work

6.1 Conclusions

This thesis investigated whether machine learning can be used to accelerate the DXA2FEM pipeline while maintaining comparable predictions of bone strength from DXA images.

The results show that neural networks are capable of extracting relevant structural information from DXA images and predicting parameters related to femur geometry and bone strength. Although the predicted SSAM parameters were not sufficiently accurate to fully replace the DXA2FEM optimization procedure, they may already be useful as initial estimates for the optimization algorithm, potentially reducing the overall reconstruction time.

The most accurate bone strength predictions were obtained when predicting bone strength directly from DXA images. This approach bypasses both SSAM reconstruction and finite element simulations, allowing bone strength to be estimated almost instantly. However, the DXA2FEM pipeline remains important for generating reliable reference values used during model training.

Finally, the results indicate that artificially generated DRRs can be used as training data for neural networks. While further work is required to improve the realism of the synthetic images and reduce the domain gap to real DXA images, this approach shows promise as a way to address the limited availability of labeled medical imaging data.

Overall, the results suggest that machine learning has the potential to significantly reduce the computational cost of DXA2FEM while preserving much of its predictive capability.

6.2 Future Work

One natural continuation of this work is to evaluate fracture risk predictions based on the bone strength values predicted by our models. This requires a dataset containing a sufficient number of hip fracture cases. In Grassi et al. [7], the predictive ability of DXA2FEM was evaluated using a test set consisting of 120 hip fracture cases and 240 control cases from the MrOS cohort. A similar evaluation could be performed using bone strength predictions from our models. This could either be done using a 10-fold cross-validation setup or by training the models on the remaining data and computing the absolute fracture risk (ARF0) from the predicted bone strengths.

Another direction for future work is to further explore the use of synthetic DRRs for training the neural networks. One potential improvement is to introduce realistic noise models to the generated DRRs in order to better match the characteristics of real DXA images. Additionally, generating SSAM parameters using covariance-based sampling instead of independent sampling could produce more anatomically realistic reconstructions, which may improve model performance.

The synthetic DRR approach could also be extended by performing FEM simulations on the generated reconstructions in order to obtain corresponding bone strength values. Neural networks could then be trained on artificial DRRs to predict bone strength directly. Although generating such a dataset would require substantial computational resources, it may be worthwhile since the best bone strength prediction performance in this study was obtained when predicting bone strength directly from the images.

Finally, the predicted SSAM parameters from the baseline network could be used to initialize the DXA2FEM genetic algorithm optimization. Instead of starting from randomly sampled parameters, the initial population could be generated around the predicted parameter values. This may reduce the number of optimization iterations required and thereby shorten the overall reconstruction time.

Bibliography

[1] John A. Kanis, Nick Norton, Nicholas C. Harvey et al. "SCOPE 2021: a new scorecard for osteoporosis in Europe". In: *Archives of Osteoporosis* 16.1 (2021), p. 82. DOI: 10.1007/s11657-020-00871-9.

[2] Ethel S. Siris, Ying Chen, Thomas A. Abbott, Elizabeth Barrett-Connor, Paul D. Miller, Lois E. Wehren and Michael L. Berger. "Bone Mineral Density Thresholds for Pharmacological Intervention to Prevent Fractures". In: *Archives of Internal Medicine* 164.10 (2004), pp. 1108–1112. DOI: 10.1001/archinte.164.10.1108.

[3] Katie L. Stone, Dana G. Seeley, Li-Yung Lui, Jane A. Cauley, Kristine Ensrud, Warren S. Browner, Michael C. Nevitt and Steven R. Cummings. "BMD at Multiple Sites and Risk of Fracture of Multiple Types: Long-Term Results From the Study of Osteoporotic Fractures". In: *Journal of Bone and Mineral Research* 18.11 (Nov. 2003), pp. 1947–1954. DOI: 10.1359/jbmr.2003.18.11.1947. URL: https://doi.org/10.1359/jbmr.2003.18.11.1947.

[4] Dianna D. Cody, Gary J. Gross, Fu J. Hou, Horace J. Spencer, Steven A. Goldstein and David P. Fyhrie. "Femoral strength is better predicted by finite element models than QCT and DXA". In: *Journal of Biomechanics* 32.10 (1999), pp. 1013–1020. ISSN: 0021-9290. DOI: 10.1016/S0021-9290(99)00099-8. URL: https://www.sciencedirect.com/science/article/pii/S0021929099000998.

[5] Marco Viceconti, Muhammad Qasim, Prateek Bhattacharya et al. "Are CT-Based Finite Element Model Predictions of Femoral Bone Strengthening Clinically Useful?" In: *Current Osteoporosis Reports* 16.3 (June 2018), pp. 216–223. DOI: 10.1007/s11914-018-0438-8. URL: https://doi.org/10.1007/s11914-018-0438-8.

[6] Sami P. Väänänen, Lorenzo Grassi, Gunnar Flivik, Jukka S. Jurvelin and Hanna Isaksson. "Generation of 3D Shape, Density, Cortical Thickness and Finite Element Mesh of Proximal Femur from a DXA Image". In: *Medical Image Analysis* 24 (2015), pp. 125–134. DOI: 10.1016/j.media.2015.06.002. URL: https://doi.org/10.1016/j.media.2015.06.001.

[7] Lorenzo Grassi, Sami P. Väänänen, Lars Jehpsson, Östen Ljunggren, Björn E. Rosengren, Magnus K. Karlsson and Hanna Isaksson. "3D Finite Element Models Reconstructed from 2D Dual-Energy X-Ray Absorptiometry (DXA) Images Improve Hip Fracture Prediction Compared to Areal BMD in Osteoporotic Fractures in Men (MrOS) Sweden Cohort". In: *Journal of Bone and Mineral Research* 38.9 (2023). Open access under Creative Commons Attribution License, pp. 1258–1267. DOI: 10.1002/jbmr.4878. URL: https://doi.org/10.1002/jbmr.4878.

[8] Lorenzo Grassi, Sami P. Väänänen and Hanna Isaksson. "Statistical Shape and Appearance Models: Development Towards Improved Osteoporosis Care". In: *Current Osteoporosis Reports* 19 (2021). Section Editors: H. Isaksson and S. Boyd, pp. 676–687. DOI: 10.1007/s11914-021-00711-w. URL: https://doi.org/10.1007/s11914-021-00711-w.

[9] Ian Goodfellow, Yoshua Bengio and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[10] Diederik P. Kingma and Jimmy Lei Ba. "Adam: A Method for Stochastic Optimization". In: *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. San Diego, CA, USA, 2015. URL: https://arxiv.org/abs/1412.6980.

[11] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324. DOI: 10.1109/5.726791.

[12] Qingchen Zhang, Laurence T. Yang, Zhikui Chen and Peng Li. "A survey on deep learning for big data". In: *Information Fusion* 42 (2018), pp. 146–157. ISSN: 1566-2535. DOI: 10.1016/j.inffus.2017.10.006.

[13] Yibo Cui, Chi Zhang, Kai Qiao, Linyuan Wang, Bin Yan and Li Tong. "Study on Representation Invariances of CNNs and Human Visual Information Processing Based on Data Augmentation". In: *Brain Sciences* 10.9 (2020), p. 602. DOI: 10.3390/brainsci10090602.

[14] François Chollet and Matthew Watson. *Deep Learning with python*. Manning publications, 2018.

[15] Sinno Jialin Pan and Qiang Yang. "A Survey on Transfer Learning". In: *IEEE Transactions on Knowledge and Data Engineering* 22.10 (2010), pp. 1345–1359. DOI: 10.1109/TKDE.2009.191.

[16] Jason Yosinski, Jeff Clune, Yoshua Bengio and Hod Lipson. "How Transferable Are Features in Deep Neural Networks?" In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 27. 2014.

[17] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision* 115.3 (2015), pp. 211–252. DOI: 10.1007/s11263-015-0816-y.

[18] Mingxing Tan and Quoc V. Le. "EfficientNetV2: Smaller Models and Faster Training". In: *Proceedings of the 38th International Conference on Machine Learning (ICML)*. 2021, pp. 10096–10106.

[19] Ahmad Waleed Salehi, Shakir Khan, Gaurav Gupta, Bayan Ibrahimm Alabduallah, Abrar Almjally, Hadeel Alsolai, Tamanna Siddiqui and Adel Mellit. "A Study of CNN and Transfer Learning in Medical Imaging: Advantages, Challenges, Future Scope". In: *Sustainability* 15.7 (2023), p. 5930. DOI: 10.3390/su15075930. URL: https://doi.org/10.3390/su15075930.

[20] Hee E. Kim, Alejandro Cosa-Linan, Nandhini Santhanam, Mahboubeh Jannesari, Mate E. Maros and Thomas Ganslandt. "Transfer learning for medical image classification: a literature review". In: *BMC Medical Imaging* 22.1 (2022), p. 69. DOI: 10.1186/s12880-022-00793-7. URL: https://doi.org/10.1186/s12880-022-00793-7.

[21] Gonçalo Marques, Deevyankar Agarwal and Isabel de la Torre Díez. "Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network". In: *Applied Soft Computing* 96 (2020), p. 106691. ISSN: 1568-4946. DOI: 10.1016/j.asoc.2020.106691. URL: https://doi.org/10.1016/j.asoc.2020.106691.

[22] Hasnain Ali Shah, Faisal Saeed, Sangseok Yun, Jun-Hyun Park, Anand Paul and Jae-Mo Kang. "A Robust Approach for Brain Tumor Detection in Magnetic Resonance Images Using Finetuned EfficientNet". In: *IEEE Access* 10 (2022), pp. 65426–65435. DOI: 10.1109/ACCESS.2022.3184113. URL: https://doi.org/10.1109/ACCESS.2022.3184113.

[23] Ho-Gun Ha, Jinhan Lee, Gu-Hee Jung, Jaesung Hong and HyunKi Lee. "2D–3D Reconstruction of a Femur by Single X-Ray Image Based on Deep Transfer Learning Network". In: *IRBM* 45 (2024), p. 100822. DOI: 10.1016/j.irbm.2024.100822. URL: https://doi.org/10.1016/j.irbm.2024.100822.

[24] Zheye Chen, Lijun Guo, Rong Zhang, Zhongding Fang, Xiuchao He and Jianhua Wang. "BX2S-Net: Learning to reconstruct 3D spinal structures from bi-planar X-ray images". In: *Computers in Biology and Medicine* 154 (2023), p. 106615. DOI: 10.1016/j.compbiomed.2023.106615. URL: https://doi.org/10.1016/j.compbiomed.2023.106615.

[25] Massimo Bottini, Olivier Zanier, Raffaele Da Mutten, Maria L. Gandia-Gonzalez, Erik Edström, Adrian Elmi-Terander, Luca Regli, Carlo Serra and Victor E. Staartjes. "Generation of synthetic CT–like imaging of the spine from biplanar radiographs: comparison of different deep learning architectures". In: *Neurosurgical Focus* 59.1 (2025), E13. DOI: 10.3171/2025.4.FOCUS25170. URL: https://thejns.org/doi/10.3171/2025.4.FOCUS25170.

[26] Eva L. Ribom, Elin Grundberg, Hans Mallmin, Claes Ohlsson, Mattias Lorenzon, Eric Orwoll, Anna H. Holmberg, Dan Mellström, Östen Ljunggren and Magnus K. Karlsson. "Estimation of physical performance and measurements of habitual physical activity may capture men with high risk to fall—Data from the Mr Os Sweden cohort". In: *Archives of Gerontology and Geriatrics* 49.1 (2009), e72–e76. ISSN: 0167-4943. DOI: https://doi.org/10.1016/j.archger.2008.09.003. URL: https://www.sciencedirect.com/science/article/pii/S0167494308001957.

[27] Emma Lindgren, Björn E. Rosengren and Magnus K. Karlsson. "Does peak bone mass correlate with peak bone strength? Cross-sectional normative dual energy X-ray absorptiometry data in 1052 men aged 18–28 years". In: *BMC Musculoskeletal Disorders* 20.1 (2019), p. 404. DOI: 10.1186/s12891-019-2785-8.

[28] Mingxing Tan and Quoc V. Le. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". In: *Proceedings of the 36th International Conference on Machine Learning (ICML)*. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 6105–6114. URL: https://proceedings.mlr.press/v97/tan19a.html.

[29] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov and Liang-Chieh Chen. "MobileNetV2: Inverted Residuals and Linear Bottlenecks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 4510–4520. DOI: 10.1109/CVPR.2018.00474. URL: https://doi.org/10.1109/CVPR.2018.00474.

[30] François Chollet. "Xception: Deep Learning with Depthwise Separable Convolutions". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1251–1258. DOI: 10.1109/CVPR.2017.195. URL: https://doi.org/10.1109/CVPR.2017.195.

[31] Alessandra Aldieri, Lorenzo Grassi, Pinaki Bhattacharya, Margaret Paggiosi and Richard Eastell. "DXA-derived three-dimensional finite element models of the femur: validation against CT-based models". In: *Bone* 206 (2026), p. 117830. DOI: 10.1016/j.bone.2026.117830.

Appendix A

Network Architectures

Baseline Network

Layer	**Output Shape**	**Notes**
Input Layer	$250 \times 250 \times 1$	Grayscale DXA input
Conv–BN–ReLU	$250 \times 250 \times 32$	7×7 conv, stride 1, padding 3
MaxPool	$124 \times 124 \times 32$	3×3 max pool, stride 2
Conv–BN–ReLU	$124 \times 124 \times 64$	7×7 conv, stride 1, padding 3
MaxPool	$61 \times 61 \times 64$	3×3 max pool, stride 2
Conv–BN–ReLU	$61 \times 61 \times 128$	7×7 conv, stride 1, padding 3
MaxPool	$30 \times 30 \times 128$	3×3 max pool, stride 2
Conv–BN–ReLU	$30 \times 30 \times 256$	7×7 conv, stride 1, padding 3
MaxPool	$14 \times 14 \times 256$	3×3 max pool, stride 2
Global Average Pooling	256	Feature vector
Shared Feature Vector	256	Output of CNN backbone
Regression Heads ($\times 40$)	1×40	MLP: $256 \to 128 \to 1$
Concatenation of Heads	40	Final output vector
Total Parameters	**3,464,425 (all trainable)**	

Table A.1: Architecture of the baseline multi-head CNN used for SSAM parameter regression.

Transfer Network

Layer	**Output Shape**	**Notes**
Input Layer	$250 \times 250 \times 3$	RGB input image
Data Augmentation	$250 \times 250 \times 3$	Rotation, translation, brightness, contrast, Gaussian noise
EfficientNetV2B2 (ImageNet)	$8 \times 8 \times 1408$	Pretrained backbone
Global Average Pooling	1408	Feature vector
Dense (ReLU)	250	Fully connected layer
Batch Normalization	250	Feature normalization
Dropout	250	$p = 0.4$
Dense (ReLU)	128	Fully connected layer
Batch Normalization	128	Feature normalization
Dropout	128	$p = 0.3$
Output (Linear)	40	Final output vector
Total Parameters	**9,160,166 (6,630,054 trainable)**	

Table A.2: Architecture of the transfer learning network based on EfficientNetV2B2.

Appendix B

SSAM Parameter Predictions

Baseline Network

Parameter	Slope	Intercept	R^2
1	0.2215	3.3096	0.1070
2	0.1178	2.4023	0.0499
3	0.1536	-1.3240	0.1494
4	0.5078	1.5293	0.4945
5	0.2111	-2.6834	0.0310
6	0.0193	0.9463	-0.1563
7	0.6733	0.4589	0.4760
8	0.5557	-0.1697	0.5684
9	0.6514	0.4789	0.6843
10	0.2984	0.2411	0.2509
11	0.3654	-0.8064	0.1696
12	0.2742	-0.2715	0.0542
13	0.2883	-0.0774	0.1333
14	0.3457	-0.2178	0.2428
15	0.5113	-0.3595	0.3287
16	0.3655	-0.2929	0.1581
17	0.4814	-0.1457	0.3420
18	0.2789	0.1552	0.2358
19	0.3292	-0.5584	0.1729
20	0.3455	0.4263	0.2776
21	0.3819	0.1789	0.2988
22	0.1936	0.1814	0.1486
23	0.4032	-0.1930	0.2378
24	0.1250	0.0087	0.0412
25	0.1701	0.9374	0.1442
26	0.0906	1.0239	0.0199
30	0.0797	1.0921	-0.0898
31	0.3385	1.6246	0.0908
32	0.3702	-1.3131	0.2531
33	0.0091	2.9331	-0.1225
34	0.0081	2.9398	-0.1526
35	0.1413	-1.5539	0.0228
36	0.2718	0.4455	0.1851
37	0.0033	2.9871	-0.1308
38	0.1927	0.6516	0.0876
39	0.1962	1.4330	0.1296
40	0.0601	2.3511	-0.0945

Table B.1: Linear regression coefficients (prediction vs. ground truth) and R^2 for each SSAM parameter using the baseline network on the test set.

Transfer Network

Parameter	Slope	Intercept	R^2
1	0.0411	3.6021	0.0440
2	0.0205	2.3174	-0.0317
3	0.0101	-1.1383	0.0055
4	0.2689	1.9593	0.2909
5	0.0255	-2.4946	0.0197
6	0.0062	0.9613	-0.0663
7	0.3696	0.2202	0.3658
8	0.2029	-0.1537	0.2268
9	0.4018	0.6227	0.4443
10	0.0339	0.2275	-0.0067
11	0.1363	-0.8242	0.1542
12	0.0639	0.0457	0.0706
13	0.1549	0.2123	0.1394
14	0.0182	-0.0379	0.0094
15	0.0959	-0.3711	0.1238
16	0.0493	-0.2463	0.0164
17	0.0599	-0.6756	0.0765
18	0.0271	0.3569	0.0120
19	0.0781	-0.5379	0.0728
20	0.0870	0.5807	0.0809
21	0.0341	0.4239	0.0411
22	0.0185	0.2716	0.0090
23	0.0311	-0.2651	0.0427
24	0.0164	0.0239	0.0142
25	0.0504	0.9086	0.0450
26	0.0043	-0.2168	-0.0278
30	0.0200	1.1577	-0.0113
31	0.1930	0.9823	0.1201
32	0.0711	-1.8201	0.0945
33	0.0010	2.4182	-0.0418
34	0.0268	2.4831	-0.0582
35	0.0016	-1.8509	-0.0099
36	0.0303	0.2599	0.0276
37	0.0244	2.8447	-0.2532
38	0.0153	0.9160	0.0040
39	0.0326	1.7721	0.0244
40	0.0330	1.9007	0.0170

Table B.2: Linear regression coefficients (predict vs. true) and R^2 for each SSAM parameter using the transfer network on the test set.

Appendix C

Bone Strength Results

C.1 FEM Predictions

Configuration (Adduction, Internal Rotation)	**Bias** [kN]	**MAE** [kN]	**RMSE** [kN]	**MAPE** [%]	**Slope**	**Intercept** [kN]	R^2
Baseline (0°, 0°)	0.976	1.097	1.316	15.79	0.802	2.469	0.714
Transfer (0°, 0°)	0.193	0.999	1.282	13.75	0.552	3.563	0.426
Baseline (0°, 15°)	0.685	0.840	1.013	13.74	0.730	2.491	0.710
Transfer (0°, 15°)	0.178	0.862	1.099	13.45	0.515	3.426	0.412
Baseline (0°, 30°)	0.440	0.566	0.688	11.16	0.714	1.998	0.696
Transfer (0°, 30°)	0.088	0.601	0.759	11.30	0.467	2.990	0.396
Baseline (15°, 0°)	0.630	0.721	0.870	13.98	0.790	1.803	0.735
Transfer (15°, 0°)	0.111	0.699	0.897	12.89	0.561	2.559	0.439
Baseline (30°, 0°)	0.232	0.418	0.513	11.54	0.673	1.510	0.689
Transfer (30°, 0°)	0.022	0.491	0.622	13.02	0.457	2.147	0.426
Baseline (10°, 15°)	0.571	0.666	0.801	13.12	0.752	1.939	0.725
Transfer (10°, 15°)	0.153	0.664	0.843	12.55	0.542	2.685	0.428
Baseline (15°, 15°)	0.529	0.607	0.730	12.89	0.743	1.841	0.716
Transfer (15°, 15°)	0.165	0.591	0.750	12.09	0.552	2.457	0.432
Baseline (15°, 30°)	0.269	0.370	0.458	9.41	0.676	1.630	0.678
Transfer (15°, 30°)	-0.016	0.402	0.509	9.65	0.480	2.165	0.408
Baseline (30°, 15°)	0.171	0.341	0.427	10.14	0.647	1.459	0.683
Transfer (30°, 15°)	0.032	0.412	0.512	11.74	0.518	1.792	0.464
Baseline (30°, 30°)	0.074	0.242	0.307	7.97	0.663	1.160	0.721
Transfer (30°, 30°)	0.034	0.331	0.472	10.63	0.541	1.515	0.370
Baseline (Mean)	**0.458**	**0.587**	**0.712**	**11.97**	**0.719**	**1.830**	**0.707**
Transfer (Mean)	**0.096**	**0.605**	**0.775**	**12.11**	**0.519**	**2.530**	**0.420**

Table C.1: Prediction performance for the ten FE strength loading configurations evaluated on the test set.

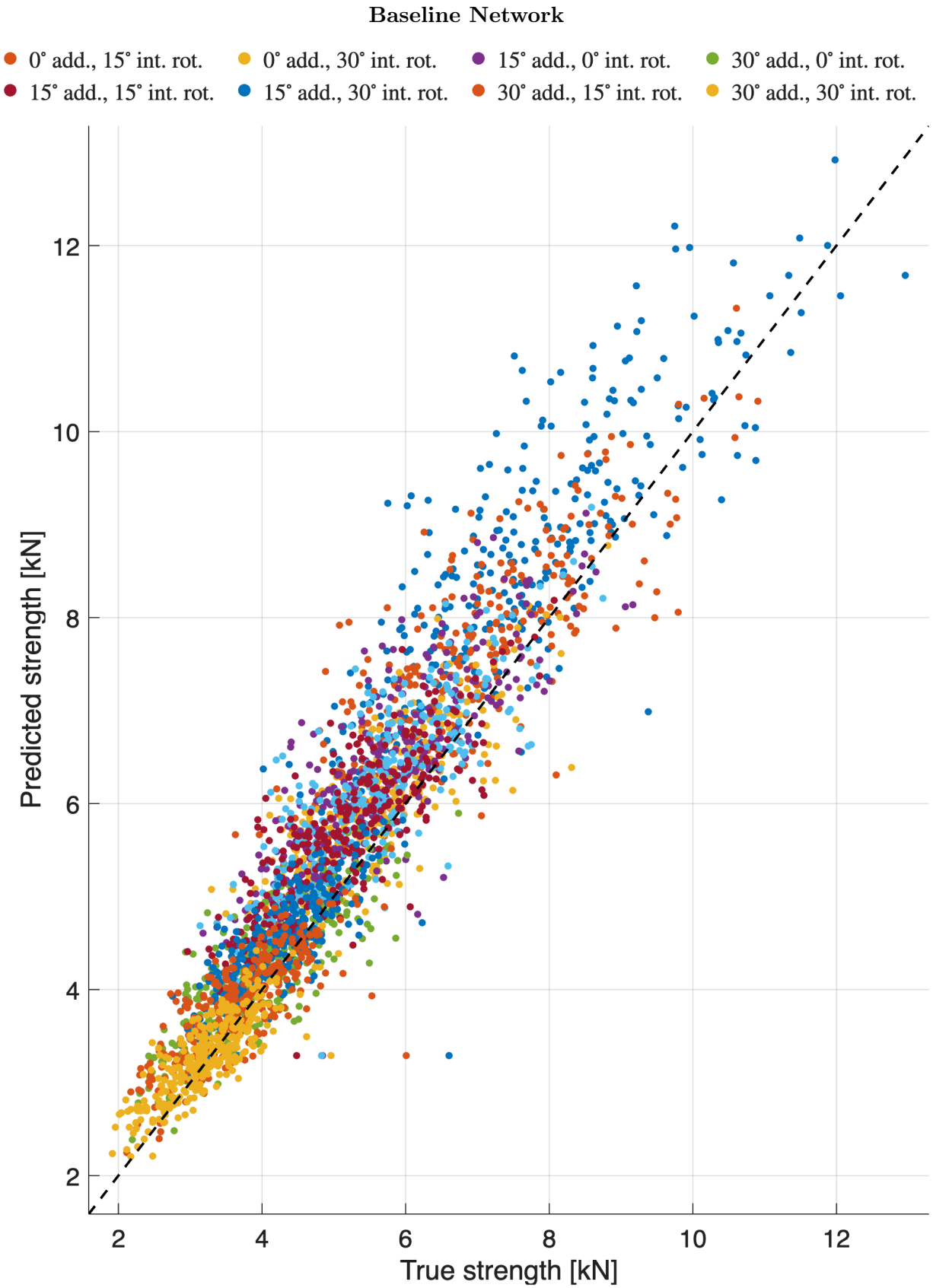


Figure C.1: Baseline errors of the bone strength predictions.

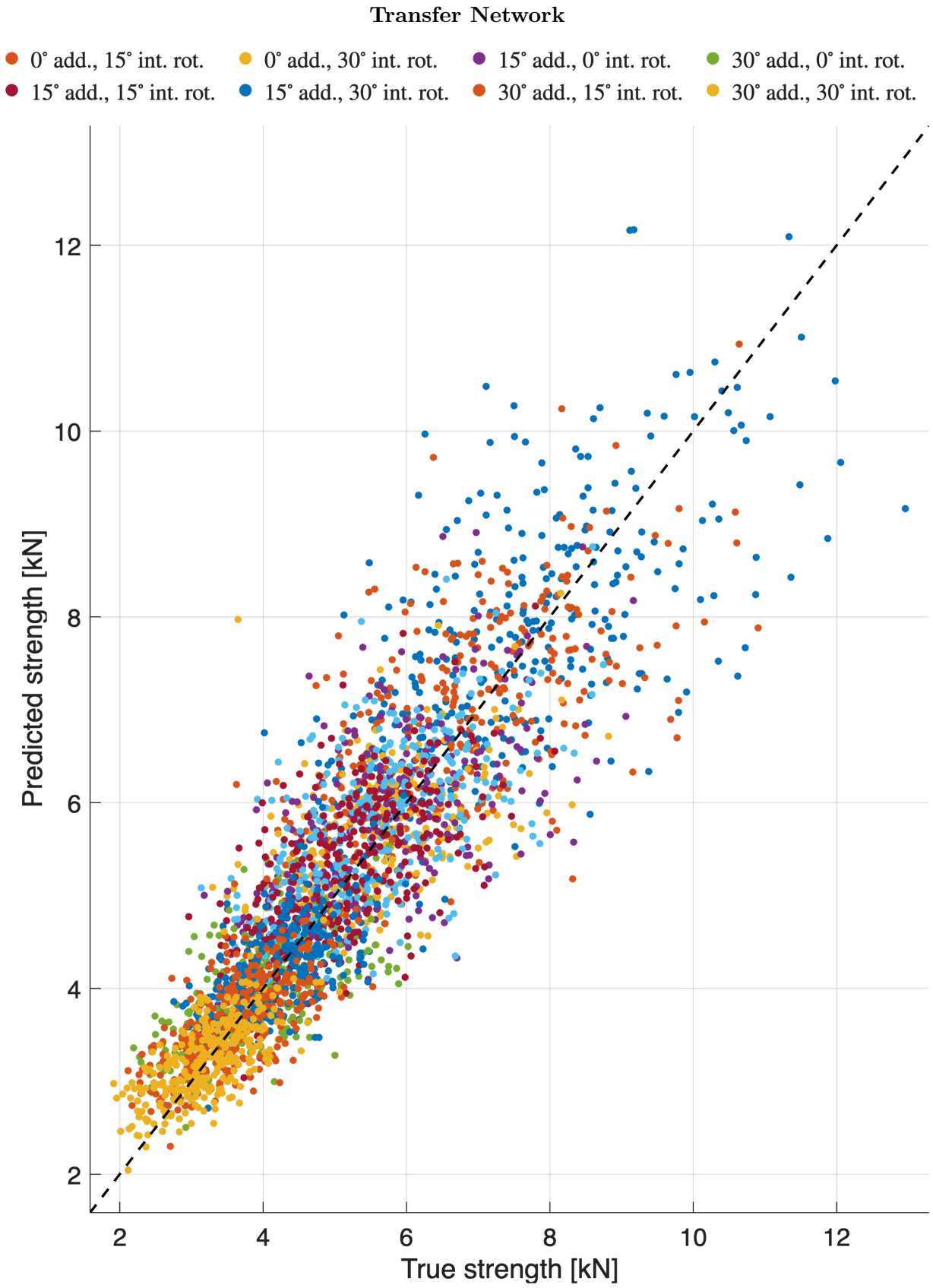


Figure C.2: Transfer errors of the bone strength predictions.

C.2 Direct Predictions

Configuration (Adduction, Internal Rotation)	**Bias [kN]**	**MAE [kN]**	**RMSE [kN]**	**MAPE [%]**	**Slope**	**Intercept [kN]**	R^2
Baseline (0°, 0°)	-0.124	0.640	0.854	8.39	0.597	2.932	0.761
Transfer (0°, 0°)	-0.254	0.812	1.076	10.54	0.467	3.793	0.617
Baseline (0°, 15°)	-0.018	0.525	0.663	8.01	0.662	2.257	0.750
Transfer (0°, 15°)	-0.130	0.646	0.824	9.78	0.524	3.077	0.623
Baseline (0°, 30°)	-0.019	0.387	0.491	7.12	0.753	1.336	0.707
Transfer (0°, 30°)	-0.097	0.453	0.589	8.24	0.596	2.116	0.586
Baseline (15°, 0°)	-0.057	0.422	0.547	7.55	0.650	1.907	0.791
Transfer (15°, 0°)	-0.133	0.552	0.711	9.76	0.509	2.620	0.640
Baseline (30°, 0°)	0.029	0.351	0.437	9.47	0.635	1.450	0.652
Transfer (30°, 0°)	-0.049	0.416	0.515	10.98	0.490	1.937	0.518
Baseline (10°, 15°)	-0.016	0.397	0.503	7.31	0.703	1.637	0.761
Transfer (10°, 15°)	-0.099	0.495	0.633	9.03	0.556	2.369	0.632
Baseline (15°, 15°)	-0.004	0.340	0.432	6.79	0.744	1.310	0.766
Transfer (15°, 15°)	-0.077	0.431	0.549	8.50	0.588	2.036	0.632
Baseline (15°, 30°)	-0.011	0.288	0.361	6.90	0.827	0.716	0.693
Transfer (15°, 30°)	-0.062	0.322	0.418	7.62	0.656	1.386	0.573
Baseline (30°, 15°)	0.026	0.260	0.329	7.54	0.732	1.003	0.733
Transfer (30°, 15°)	-0.030	0.327	0.407	9.31	0.571	1.531	0.593
Baseline (30°, 30°)	-0.007	0.196	0.244	6.19	0.815	0.589	0.777
Transfer (30°, 30°)	-0.029	0.240	0.305	7.63	0.654	1.087	0.651
Baseline (Mean)	**-0.020**	**0.381**	**0.486**	**7.53**	**0.712**	**1.514**	**0.739**
Transfer (Mean)	**-0.096**	**0.469**	**0.603**	**9.14**	**0.561**	**2.195**	**0.607**

Table C.2: Direct-to-strength prediction performance for the ten FE strength loading configurations evaluated on the test set.

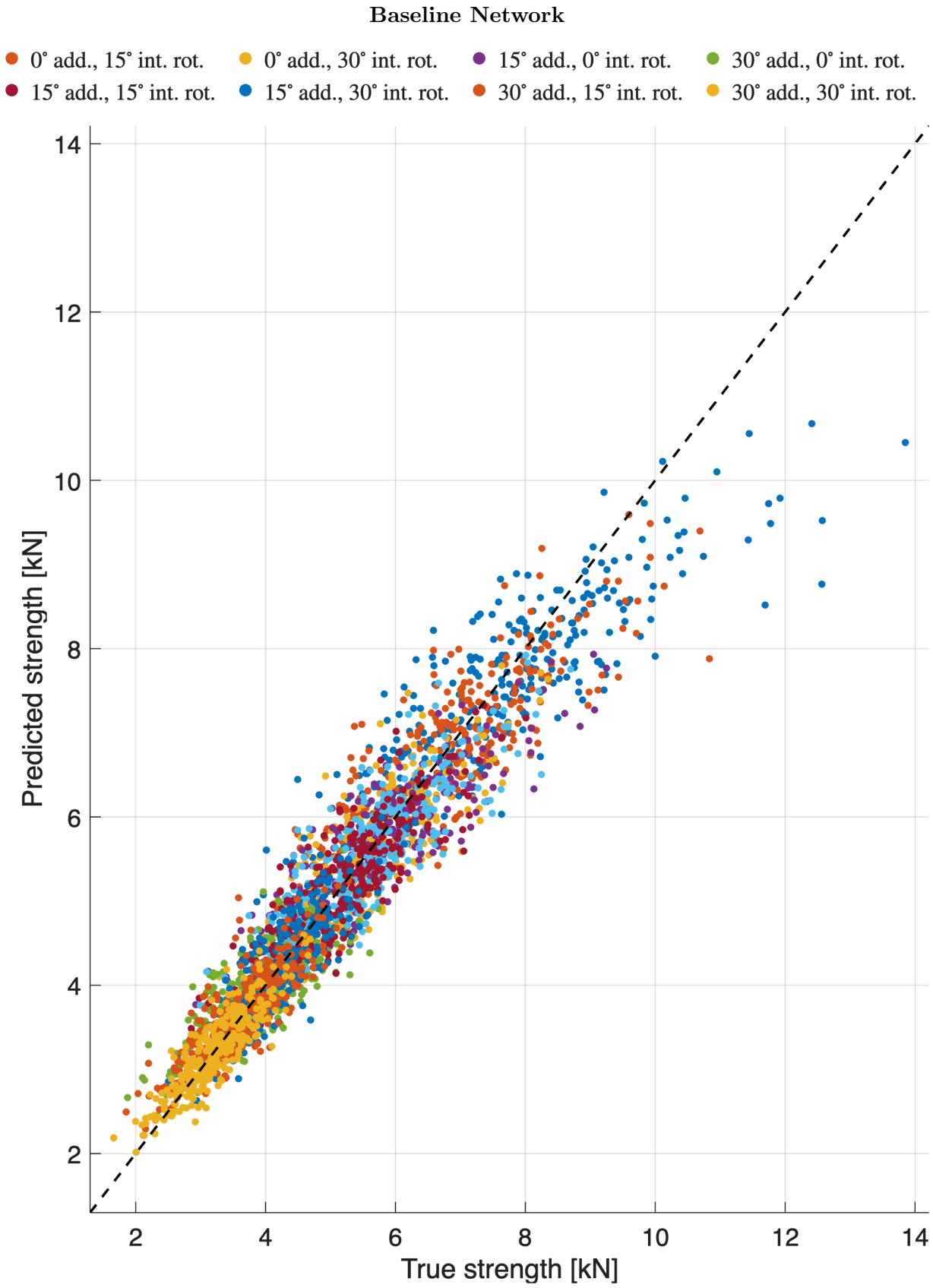


Figure C.3: Baseline errors of the bone strength predictions.

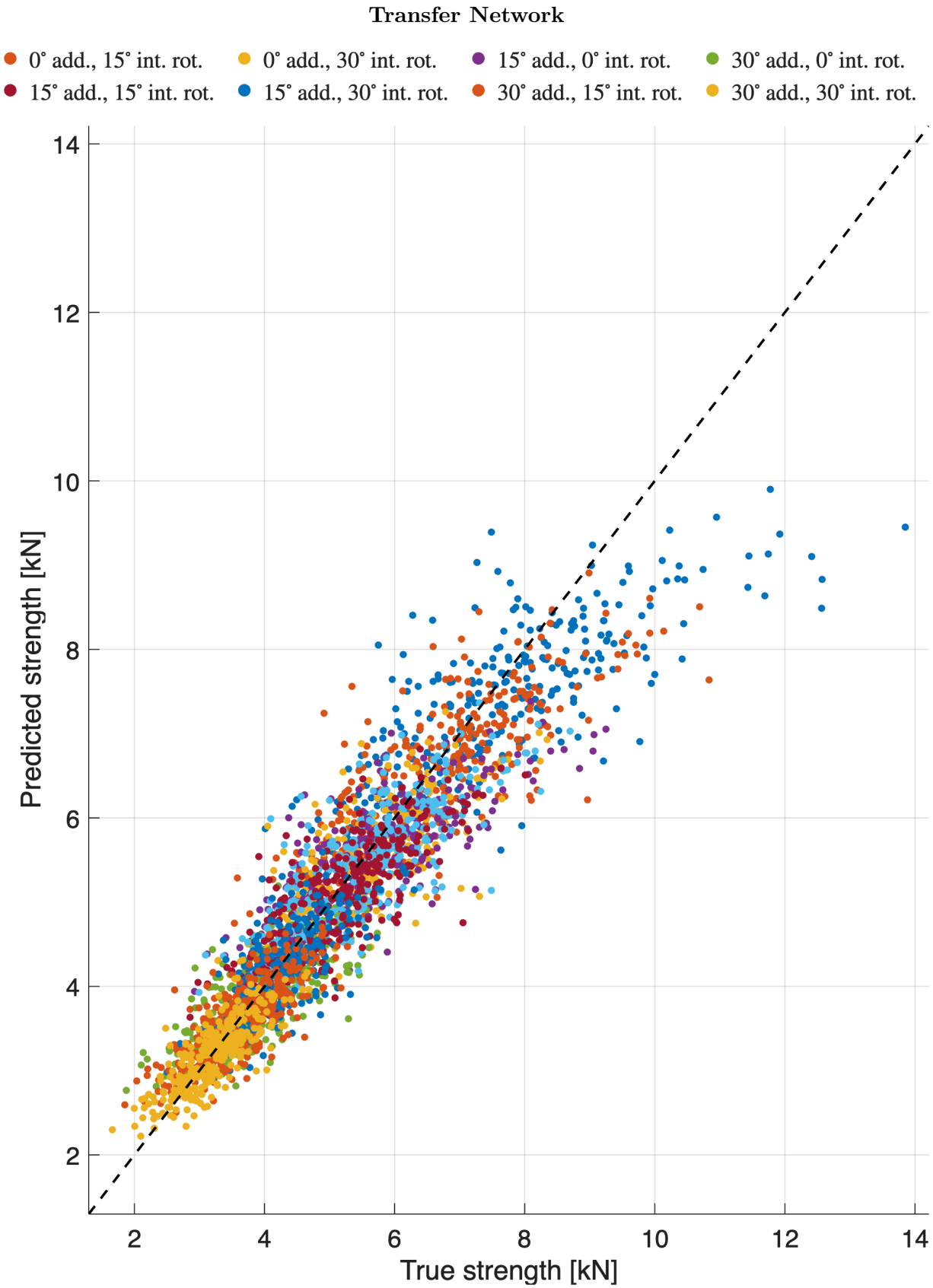


Figure C.4: Transfer errors of the bone strength predictions.