ORIGINAL ARTICLE

SSDL—an automated semi-supervised deep learning approach for patient-specific 3D reconstruction of proximal femur from QCT images

Jamalia Sultana[1] · Mahmuda Naznin[1] · Tanvir R. Faisal[2]

Abstract

Deep Learning (DL) techniques have recently been used in medical image segmentation and the reconstruction of 3D anatomies of a human body. In this work, we propose a semi-supervised DL (SSDL) approach utilizing a CNN-based 3D U-Net model for femur segmentation from sparsely annotated quantitative computed tomography (QCT) slices. Specifically, QCT slices at the proximal end of the femur forming ball and socket joint with acetabulum were annotated for precise segmentation, where a segmenting binary mask was generated using a 3D U-Net model to segment the femur accurately. A total of 5474 QCT slices were considered for training among which 2316 slices were annotated. 3D femurs were further reconstructed from segmented slices employing polynomial spline interpolation. Both qualitative and quantitative performance of segmentation and 3D reconstruction were satisfactory with more than 90% accuracy achieved for all of the standard performance metrics considered. The spatial overlap index and reproducibility validation metric for segmentation—Dice Similarity Coefficient was 91.8% for unseen patients and 99.2% for validated patients. An average relative error of 12.02% and 10.75% for volume and surface area, respectively, were computed for 3D reconstructed femurs. The proposed approach demonstrates its effectiveness in accurately segmenting and reconstructing 3D femur from QCT slices.

Keywords Segmentation · Semi-supervised deep learning (SSDL) · 3D U-Net · 3D femur reconstruction

1 Introduction

Three-dimensional (3D) reconstructed bone models are widely used in orthopedics for visually inspecting bone morphology and joint structure and assessing pathological conditions. In particular, to prevent hip fracture of elderly people, a 3D reconstructed femur image can play a crucial role in orthopedic evaluation consisting of mechanical, morphological, and densitometric properties for conducting an early and non-invasive assessment of osteoporotic fracture risk [1, 2]. Imaging modalities such as computed tomography (CT) and magnetic resonance imaging (MRI) can provide more detailed spatial information on the target anatomical structure [3, 4], which are used in mechanistic computational models to functionally understand, examine, and predict the performance of individual structure. In particular, quantitative computed tomography (QCT) has been used to predict the fracture risk [5] in proximal femur considering 3D structural and geometric parameters with bone mineral density (BMD) distributions in finite element analysis (FEA)-based computational modeling [6–13]. Traditionally, the 3D reconstruction of femur requires expensive and advanced biomedical image analysis software, extensive well-trained user intervention, and advanced engineering and image processing knowledge that restrict the clinical use of mechanistic models as a non-invasive assessment tool till today. Therefore, the primary objective of this study is to develop a fully automated data-driven modeling framework for obtaining 3D femurs to facilitate preclinical and clinical diagnosis and assessment of hip fracture.

The major challenge of 3D reconstruction of the femur lies within the segmentation of the proximal femur. Koh et al. [14] proposed a 3D template-based reconstruction for

✉ Tanvir R. Faisal
tanvir.faisal@louisiana.edu

[1] Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh

[2] Department of Mechanical Engineering, University of Louisiana at Lafayette, Lafayette, LA 70503, USA

predicting patient-specific 3D femur geometry using X-ray and sparse CT images requiring manual segmentation as well as higher reconstruction time. Concurrent segmentation processes mainly involve statistical and semi-automated techniques such as primitive shape recognition methods [15], atlas-based segmentation [16–18], graph-cut approaches [19, 20], active model [21, 22], and statistical shape models [19], which usually require time consuming user monitoring and significant manual intervention. Commonly used techniques often face challenges that affect their accuracy and usability. Conventional 3D reconstruction requires manual segmentation, which is a time-consuming labor-intensive process [23]. Semi-automated methods such as atlas-based segmentation and region growing techniques provide partial automation [24] but with significant user intervention. Some methods are also sensitive to variations in imaging conditions, such as changes in contrast, resolution, or noise [25], which impact the accuracy and robustness of segmentation and reconstruction algorithms. The introduction of artifacts in volumetric reconstruction due to interpolation techniques [26] as well as the errors and inaccuracies in surface meshes from volumetric data [27] may lead to a loss of detail and accuracy in the constructed 3D model. The intricate connectivity of the femur with surrounding anatomical structures poses challenges in accurate segmentation [28] as well.

A model-based iterative reconstruction algorithm was proposed to work with QCT images to perform automatic segmentation of tissues to estimate material properties [29]. In addition, a gradient-based method [30] was also used for pelvic bone segmentation. However, these algorithms are very sensitive to region growing procedures and thresholding to get a well segmented bone. Kalshetti et al. [31] proposed an interactive medical image segmentation framework combining mathematical morphology and Grab-Cut algorithm [32]. The performance and accuracy of this model highly depend on the distinguishing boundary of the region of interest and its contrast with the background. To automate the segmentation in the 3D construction framework, an iterative algorithm but with high computational cost was used [33]. Krčah et al. [34] utilized statistical shape and intensity modeling that requires heavy post-processing including morphological erosion for satisfactory results. To eliminate the bottleneck, data-driven approach such as deep learning (DL)-based algorithm can be incorporated in segmentation framework [35–37] to isolate proximal femur from other anatomical features. Hence, the objective of this work is to develop an automated segmentation framework and the 3D reconstruction of the proximal femur with minimal expert interventions, mitigating the challenges associated with other techniques. However, the acquisition of high-quality, well-annotated diverse, and comprehensive dataset for 3D femur reconstruction using DL is also a major challenge [38, 39].

Deep learning is a subset of machine learning (ML) that focuses on training artificial neural networks with multiple layers, known as deep neural networks, to automatically learn and extract features from data. In the context of medical image segmentation, DL techniques involve training deep neural networks to identify and delineate specific structures or regions of interest within medical images, such as identifying and segmenting tumors in radiological scans [40]. Convolutional Neural Network (CNN)-based 3D U-Net architecture has been recently developed for pelvic bone segmentation from dual energy CT (DECT) images [41]. U-Net architecture, one of the popular DL models, has been widely adopted by the medical imaging community and performs with excellence in the field of biomedical image segmentation. Although U-Net [42], CNN-based model, and data augmentation techniques [43–45] have shown promising performance in automated bone segmentation from QCT images, the performances of these frameworks are limited, because segmenting a femur surrounded by pelvic bone with similar bone density is difficult to comprehend without manual intervention. A performance enhancement was observed in femur segmentation utilizing edge detection, deep CNN, and Conditional Generative Adversarial Network [46–48]. However, new or unseen patient-specific performance of these frameworks was not evaluated outside of the study cohorts, and thereby limiting their performance measure.

The overarching objective of our research is to leverage the 3D reconstructed femur as a pivotal instrument for analyzing its mechanical, morphological, and densitometric properties. This comprehensive analysis serves as a foundation for the early and non-invasive evaluation of fracture risk, with a particular focus on mitigating hip fractures among the elderly population. Thus, the primary aim of this study is to develop a data-driven automated framework to segment as well as reconstruct 3D femurs from QCT images without or with minimum expert interventions and human-on-the-loop corrections. In this proposed framework, our objectives were to (1) develop an automated end-to-end semi-supervised deep learning (SSDL)-based framework combining both unlabeled and labeled QCT slices for training, (2) preserve Digital Imaging and Communications in Medicine (DICOM) metadata of unseen patients, and (3) retain the spatial data with properties to reconstruct 3D proximal femur from QCT slices. The proposed SSDL network employs an end-to-end CNN-based 3D U-Net model to segment and reconstruct the 3D femur in a fully automated fashion.

2 Data acquisition and preparation

The proposed CNN-based 3D U-Net model was developed utilizing the QCT dataset obtained in DICOM format. The semi-supervised learning framework was developed combining both unlabeled (3158) and labeled (2316) QCT

slices for training. A schematic diagram of the overall workflow is shown in Fig. 1.

2.1 Data acquisition

The QCT image dataset of patients in DICOM format used in this study was previously obtained removing all personal information from the Great-West Life PET/CT center in Winnipeg, Canada, following the institutional guidelines. The DICOM format combines image data with metadata that typically describes a patient, imaging procedure, and spatial referencing information, which are essential to reconstruct as well as to conduct clinical and morphological studies of the femur [49]. The QCT images were obtained using a SIEMENS S5VB40B CT scanner (Siemens Medical Solution, Malvern, USA) with acquisition and reconstruction parameters of 120 kVp and 244 mAs, respectively [8, 50]. To ensure an accurate estimation of BMD and correct scanner drift, a calcium hydroxyapatite calibration phantom (Mindways Inc., Austin, TX, USA) was mounted during imaging.

In this preliminary study, a group of 18 anonymous adults, consisting of ten male and eight female were selected. The average age of male and female patients was 66.0 ± 8.01 and 67.85 ± 5.27 years, respectively. The average weight of the male and female patients was 82.19 ± 7.83 and 69.7 ± 7.52 kg, respectively. The dataset comprised a total of 6789 (13,578 QCT slices after splitting into left and right femurs) high-resolution QCT slices in DICOM format. Each QCT image contains an in-plane resolution of 512×512 pixel array with approximately $1 \times 1 \times 1 \text{ mm}^3$ of voxel size.

2.2 Data annotation

QCT slices closure to the femoral head were primarily annotated using MIALab—a free medical image analysis toolkit (Linköping University, Sweden) [51]. In hip joint near the proximal end of the femur, the femur head and acetabulum form a ball and socket joint, having nearly identical Hounsfield units (HU) that measure bone density [52].

Consequently, differentiating the femur from the pelvic bone based on bone density alone is challenging. Therefore, we opted for a minimum number of slices in the proximity of the hip joint for labeling. To create the labeled dataset for training, we annotated between 88 and 178 (a total of 2316) QCT slices of the patient cohorts in the training dataset to identify the femur profile at the proximity of femoral head and acetabulum. The variation of slice numbers was attributed to the differences in femur length primarily varying with patients' height. The annotation was performed in a semi-automated process, where a femur boundary in a slice was first identified along the femur profile via built-in interactive region selection option. However, the annotated slices were examined carefully in consultation with an experienced orthopedic surgeon for accuracy and corrected manually as necessary. This semi-automated annotation process ensured that the resulting annotated dataset was of high quality and suitable to use in training a DL-based model for femur bone segmentation. Using the labeled data, binary masks were obtained to represent the area of the image labeled as true (i.e., contained femur bone) or false (i.e., did not contain femur bone) (Fig. 2).

2.3 DICOM data pre-processing

The DICOM dataset was pre-processed in several steps (Fig. 3) to facilitate data annotation and for developing training dataset. First, the DICOM images were split into half to separate left and right femurs. After splitting, each QCT image contains an image matrix of 512×256 pixels with approximately $1 \times 1 \times 1 \text{ mm}^3$ of voxel size. The number of images was doubled after the splitting, and a total number of 13,578 QCT images were considered for the proposed framework. Second, the training data was batched according to their instance numbers in the QCT slices so that the correct DICOM images were trained against their annotated labels. After that, the image and text metadata were separated, because since only image data was required to train the adopted CNN-based U-Net. However, the metadata is
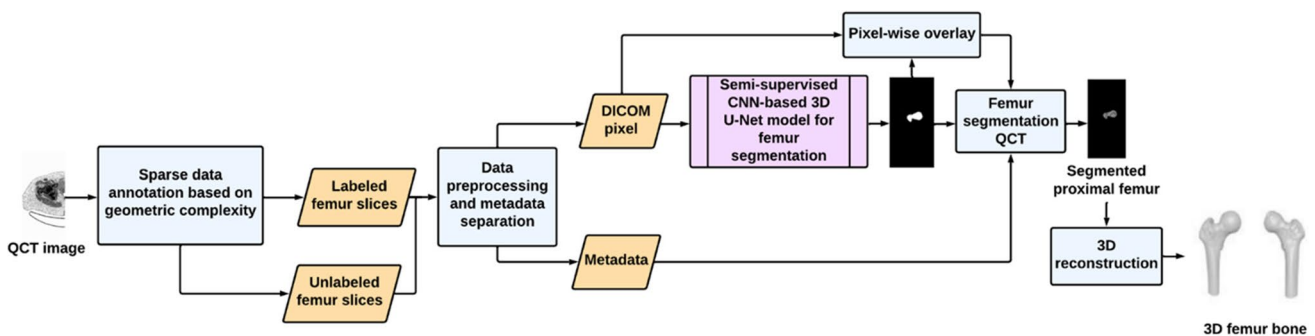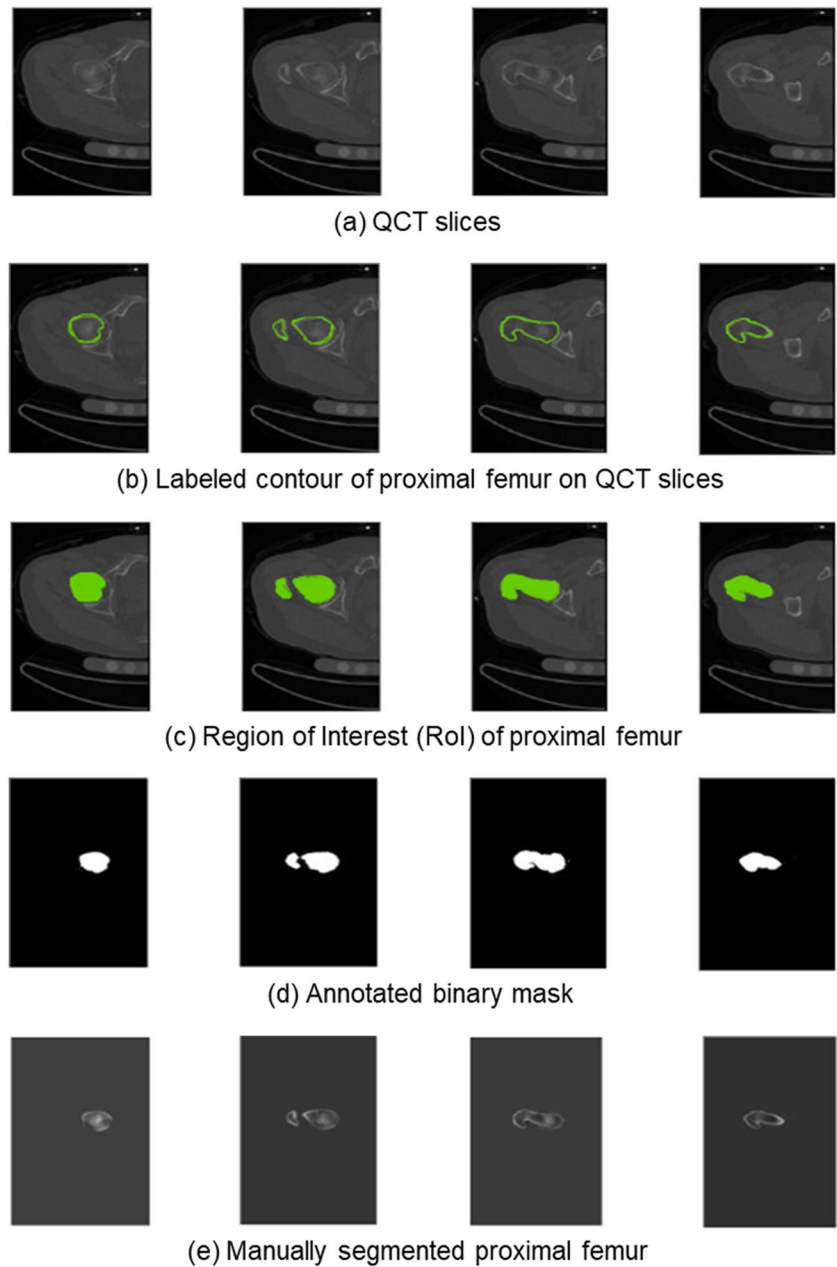

Fig. 1 An overview of the workflow for segmentation and 3D reconstruction of a femur from QCT image dataset

Fig. 2 Example of annotation process for segmenting femur. **a** QCT slices belonging to different locations, **b** labeled contour of the femur, **c** region of interest corresponds to the contour profile, **d** binary mask, and **e** segmented femur

(a) QCT slices

(b) Labeled contour of proximal femur on QCT slices

(c) Region of Interest (RoI) of proximal femur

(d) Annotated binary mask

(e) Manually segmented proximal femur

required for the 3D reconstruction of the proximal femur. While separating the image and textual metadata, the instance numbers were correctly embedded from the QCT slices to each DICOM image. It is to be noted that instance number was used and extreme care was taken to avoid any mismatch of metadata and image during re-encoding the metadata with segmented images. Next, the pixel value of DICOM images was converted into integer16-based pixel array. After that, the pixel values were normalized to train the U-Net model [40]. Finally, DICOM pixel array was stored and converted into DL readable image channels of $512 \times 256 \times 1$.

3 Proposed methodology of segmentation

3.1 CNN-based 3D U-Net model

To segment intricate structures like the proximal femur, the U-Net architecture emerges as highly suitable [33–35] since its characteristic "U" shape facilitates the capture of nuanced details, while preserving contextual information—a pivotal factor for precise segmentation [42]. The CNN-based U-Net is purposefully crafted for semantic segmentation—a quality particularly beneficial
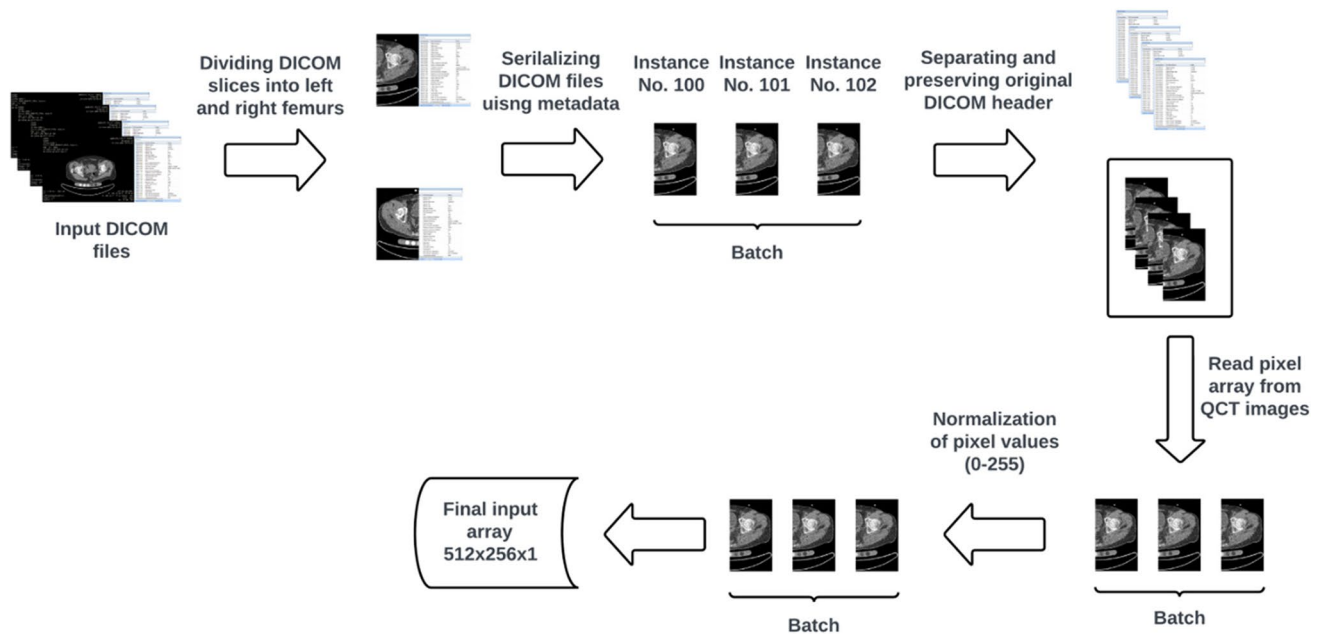
Fig. 3 Workflow of data pre-processing steps to separate image and text metadata from the input DICOM dataset

for the isolation of the femur from neighboring tissues [42]. Its proficiency lies in the classification of each pixel within an image, attributing it to the pertinent class, be it femur, muscles, fat, or other bones. The U-Net model was also adopted in this study for its demonstrated robust performance even with relatively limited training dataset [35–37].

The proposed CNN-based 3D U-Net architecture (Fig. 4) encodes the latent and high-level features of the input pixel array of raw QCT images and decodes the features to generate the final segmentation results in a binary pixel array of true and false. The U-Net architecture developed herein consists of six deep learning components and two paths—contracting (*downsampling*) and expanding (*upsampling*) paths. The contracting path is the encoder and captures context using stacked *convolutional* units and *max pooling* layers. In contrast, the expanding path is the decoder and allows for precise localization using upsampling. Figure 4 shows the CNN-based architecture of our 3D U-Net SSDL model with pre-processed QCT images as the input and the generated binary mask as the output.

3.1.1 Downsampling

In this 3D U-Net model, max pooling layers along with 2D convolutional layers, kernel initializer "he normal" [53], and rectified linear activation unit (ReLU) [40] were used for downsampling the dimensionality of feature maps. Convolutional layers containing several filters were applied to the feature input.

Increasing speed by rectangular kernel Since the orientation of the DICOM slices was known and the feature map of segmentation was taller than its width, a kernel of 6×3 matrix size in convolution layers was utilized to restrict fast learning. This rectangular kernel matrix maps each input point using two times more along vertical than horizontal neighbors. Given the similar bone density distribution in pelvis, our 3D U-Net model utilized the rectangular kernel to identify discriminative features in the acetabular regions compared to the shaft region of the femur. Specifically, the model devotes twice as much attention to the acetabular regions. Moreover, ReLU activation sets all the negative numbers in the matrix to zero, keeping the non-negative input numbers unchanged. Since all pixel values associated with the femur bone are greater than zero, the ReLU activation function is suitable for femur segmentation.

Managing the memory constraints To address memory constraints, we adopted a patch-based model, where each volume patch was $512 \times 256 \times 1$ voxels with an overlap of $256 \times 128 \times 64$ voxels. This patch size captures the entire femoral head, which is the most critical section of the proximal femur, and downsamples after each use of max pooling (here, only the integer values of the most prominent features of the previous feature map were left for patch dimensions). To address image shrinkage and the reduction of pixel size at the image edges during convolution, an extra border was added to the image by padding it with zeroes and an odd filter dimension. This additional border around the image ensures and maintains the same size of the input and output
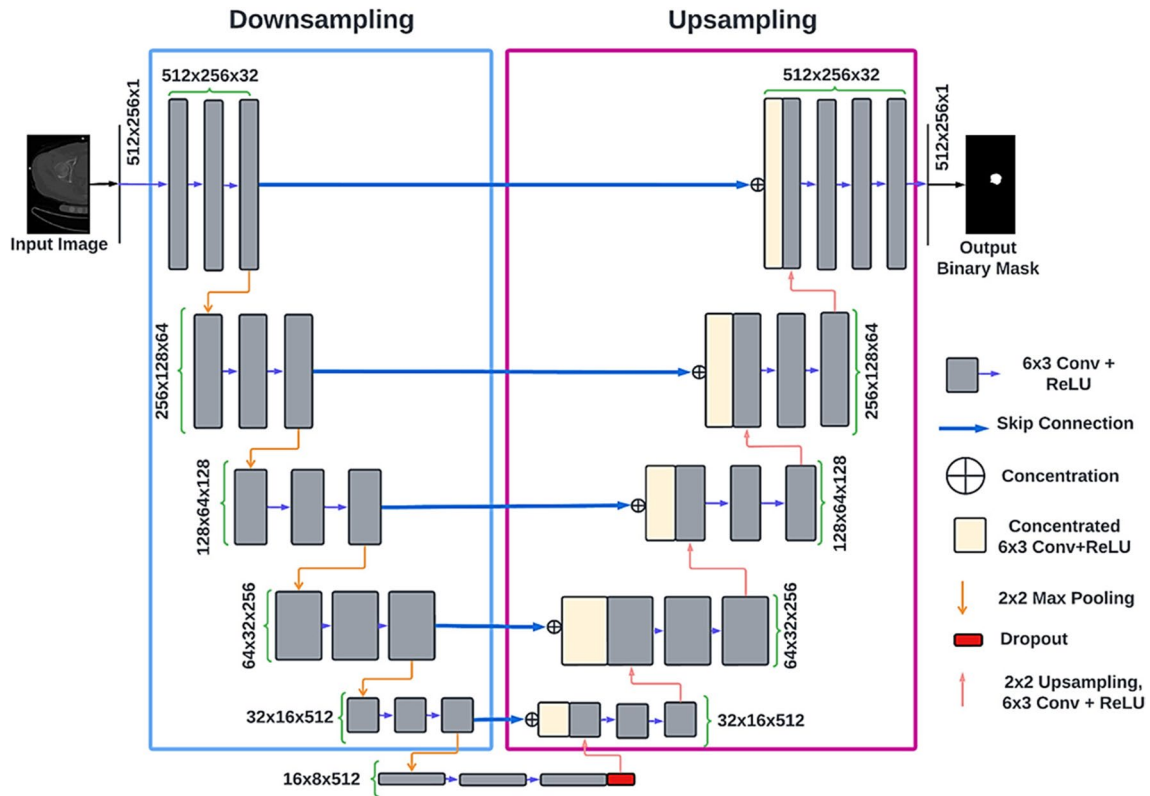
Fig. 4 CNN architecture of end-to-end 3D U-Net model. The network consists of a downsampling path and an upsampling path

images. During training, volume patches were randomly selected from the image volume and subsequently fed into the neural network with a batch size of 10. As the image patch is downsampled in the contracting path, the number of channels increases, allowing for the extraction of more complex features. In addition, we added a dropout layer before transferring the feature maps to the upsampling path to prevent the model from overfitting [54]. Skip connections were also used to pass features from the contracting path to the expanding path to recover the lost spatial information during downsampling.

3.1.2 Upsampling

Upsampling path in our 3D U-Net architecture decodes downsampled pixel array. We utilized the 2D upsampling layers along with the convolutional layers to create the decoding path of this end-to-end U-Net model. At each step in the decoder, a skip connection allows for the concatenation of the output of the transposed convolutional layer with the corresponding feature map from the encoder. Skip connections tackle vanishing gradient problems by using uninterrupted gradient flow from the first convolutional layer to the last convolutional layer [55]. They pass features from the encoder path to the decoder path. Hence, skipping

connections can recover any lost spatial information during downsampling and stabilize gradient updates in deep architectures. Concatenative skip connections ensure feature risibility of the same dimensionality from the earlier layers. The convolutional layer is subsequently able to produce a more precise output based on this information.

Finding binary mask A U-Net output layer made of convolutional layer with $1 \times 1 \times 1$ kernel and *Sigmoid* activation function [56] was created. The Sigmoid function can map predicted values to probabilities between 0 and 1. This layer then transfers the feature maps into a probability image with the resolution of 512 to generate the binary output mask. The output of each voxel is the probability that the voxel in the pixel array contains the proximal femur. Finally, the probability value of each voxel was rounded to the nearest integer value of 1 and 0, where 1 means true, indicating femur presence, and 0 means background (no femur). Thus, we converted the output probability map into a binary segmentation mask of true and false.

3.2 Performance metrics and model training

To evaluate the accuracy and robustness of our 3D U-Net segmentation model, we considered several performance

metrics, including *Mean Intersection over Union* (mean IoU) [57], *Dice Similarity Coefficient* (DSC) [58], *average precision* [59], *sensitivity* [60], and *specificity* [60]. These metrics were also used for measuring the matching between the automated segmented femur and the manually annotated ground truth of the femur. To ensure a more rigorous evaluation, we incorporated a cross-validation process as well.

The performance indicators were mathematically evaluated on two classes (femur and background) based upon *true positives* (TP), *true negatives* (TN), *false positives* (FP), and *false negatives* (FN). In this problem domain, TP measures the proposed model's ability to detect the pixel positions on the femur foreground, TN denotes our model's ability to detect the pixel positions on background, FP is the model's ability to identify pixels on femur where the background is situated, and FN represents our model's ability of detecting background where the femur pixels are situated. The performance indicators are described as follows.

The *mean IoU* evaluates the accuracy of the predicted image mask by calculating the intersection between the predicted and actual images [57]. This metric assesses the effectiveness of segmentation or accuracy of detecting objects in images in segmentation models.

$$mean\ IoU = \sum_{c \in classess} \frac{TP}{TP + FP + FN} \tag{1}$$

DSC has been widely used in various bone segmentation applications [35, 37, 61]. The DSC (Eq. 2) measures the similarity between the predicted segmentation mask and ground truth mask by considering both TP and TN rates [58]. The numerator in Eq. 2 represents the number of pixels that are correctly labeled as the object of interest in both the predicted and ground truth masks, while the denominator represents the total number of pixels labeled as the object of interest in either of the masks. The DSC varies between 0 and 1, with a higher value indicating a better segmentation performance.

$$DSC = \frac{2 \times TP}{(TP + FP)(TP + FN)} \tag{2}$$

Average precision (Eq. 3) is used to evaluate the efficiency of ML models in identifying positive samples [59]. *Sensitivity* (Eq. 4) evaluates the model's ability to correctly identify positive instances of the pixel and position of the femur in the QCT image. *Specificity* (Eq. 5) metric measures the model's ability to correctly identify negative instances. In this study, specificity and sensitivity metrics were computed based on pixel. This process involves a binary mask, generated by the U-Net model to perform a pixel-wise overlay on the input QCT slices to extract the segmented femur. The subsequent computation of specificity and sensitivity relies on pixel-level comparisons within

this framework. A higher sensitivity value demonstrates an increased segmentation ability of our adopted framework, whereas a high specificity score indicates the model's ability to detect negative instances more accurately, thereby making it an essential metric in evaluating the overall performance of ML models. Specificity loss functions also quantify the model performance at each iteration of each epoch to adjust the parameters of the gradient descent algorithm during backpropagation.

$$Average\ Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{4}$$

$$Specificity = \frac{TN}{TN + FP} \tag{5}$$

The validation set allowed us to gain insight into the performance of the model on unseen data so that hyperparameters could be adjusted accordingly. For 18 patients in our dataset, we used a *leave one (patient) out cross-validation* (LOOCV) [62] approach, with eight patients for training and the remaining ten patients for testing. Among the testing patients, a total of 4434 QCT slices of four patients were validated to the U-Net segmentation model training through validation, whereas the other six were new independent set of unseen patients with 6898 QCT slices. *Adam optimizer* [63] and *binary cross-entropy* loss function were used for model compilation. To ensure optimal performance, with the patience of ten iterations and an improvement threshold of 0.0001, early stopping criteria were employed based on the maximum validation of DSC. Additionally, a model checkpoint was set based on the maximum validation of DSC. To further enhance the learning process, the learning rate was reduced for the model based on validation loss, with a lower bound set at 0.00001, and the learning rate was decreased by a factor of 0.2.

The *binary cross-entropy* loss function (Eq. 6) was used to optimize the neural network model by calculating the differences between the predicted and true probability distributions.

$$H_P(q) = -\frac{1}{N} \sum_N \left(y_i \cdot \log \left(p(y_i) \right) + \left(1 - y_i \right) \cdot \log \left(1 - p(y_i) \right) \right) \tag{6}$$

where y_i and $\log(p(y_i))$ represent the true and predicted labels, respectively.

3.3 Femur segmentation and metadata inclusion

After obtaining the DICOM files of the segmented proximal femur, we isolated the femur by performing a pixel-wise multiplication (Eq. 7) using binary segmentation masks

generated from our 3D U-Net model. To segment the femur in DICOM images, we retain the pixel value in the DICOM images if the corresponding value in the binary mask array is 1. Conversely, if the value in the binary mask array is 0, we discard the corresponding area of the DICOM image.

$$f(x) = \begin{cases} D(x), if\ D(x) * M(True). \\ 0, D(x) * M(False). \end{cases} \qquad (7)$$

where $D(x)$ represents the pixels of DICOM image array, $M(True)$ and $M(False)$ represent the values 1 and 0, respectively, in the generated binary mask array.

After performing pixel-wise multiplication, the resulting array turned into 16-bit integer format. We also updated the metadata values for the newly generated segmented femur QCT files, ensuring that the metadata accurately reflects the segmentation process. This step helps maintain the integrity of the metadata associated with the segmented femur images. Additionally, the largest image pixel and smallest image pixel values were also updated for the newly generated DICOM images. Finally, we added preserved metadata from the original DICOM files to the newly generated femur bone.

For the 3D reconstruction of the femur (Section 4), we also performed post-processing cleanup on the DICOM slices. We removed any undesired pixels (noise) outside the range of cancellous and trabecular bones by replacing the out of bound pixel values with the HU value of air. In this denoising step, morphology masking based on HU variation was used to discard the outer variances from the slices. This eventually ensures the 3D reconstruction of the femur bone only.

4 Reconstruction of 3D femur

In the last step of this workflow, 3D femur was reconstructed from the segmented CT slices preserved in DICOM format. The femur reconstruction process was built upon voxel sizes and a number of metadata information primarily including intercept slope, rescale slopes, image position of the patient, location of the femur in the slice, and DICOM slice thickness. Furthermore, to calculate the intercept and slope, we stacked the pixel arrays of the scans using the instance numbers and converted the pixel values to HU values and calculated the intercept and slope of the segmented femur, considering the HU values of air as the reference.

4.1 Stacking of the slices

The slices were resampled according to the spacing obtained from QCT imaging. The newly segmented DICOM slices were stacked on top of each other based on the new spacing values. During the 3D femur reconstruction from 2D images,

cubic spline interpolation [64] was used to estimate the voxel values in between the actual image slices. To obtain the cubic spline interpolation function $f(x, y, z)$, a set of voxel data $v_i(x_i, y_i z_i)$ from the segmented QCT slices were considered as follows:

$$f(x, y, z) = \sum_{i=0}^{n} N_i(x, y, z) \cdot v_i \qquad (8)$$

where n is the number of input data points, and $N_i(x, y, z)$ is the cubic B-spline polynomial function, which is a continuous function having continuous first and second-order derivatives. The cubic spline interpolation method was used to estimate the voxel values at any point in between the actual image slices, allowing for smoother 3D reconstruction of the image data. The cubic spline interpolation method provides a smooth and continuous interpolation of data, which is important for an accurate 3D reconstruction of a femur from the segmented DICOM slices.

4.2 Volumetric reconstruction

For volumetric reconstruction, we utilized Python "Vedo" libraries [65], which is primarily designed for 3D visualization and image processing built on top of VTK (Visualization Toolkit) library functions [66]. In particular, the volume class from the Vedo package was used for the 3D reconstruction from 2D images. The 3D volume regeneration process involves stacking of the 2D segmented DICOM images along the z-axis based on the origin, resampled voxel spacings, and original dimensions. We also utilized static cast functions to convert the void pointer to correct the scalar type, if there is any. This static-cast function converts the pixel array to float64 to estimate the voxel values in between the actual image slices. The cubic spline interpolation method was used to estimate the voxel values at any point in between the actual image slices, allowing for a smoother 3D reconstruction.

4.3 Surface mesh generation

To generate the 3D surface mesh of the femur, the *iso-surface* function of the Vedo libraries was employed. The generated 3D isosurface represents the boundary of the material of interest in the volume. To create a 3D mesh that can be rendered and manipulated, we triangulated the isosurface by subdividing the surface area into triangles connecting the vertices of the surface. This isosurface uses Delaunay triangulation. In generating surface mesh, *cell size limit* was used as the characteristic size, where we specified a threshold value to control the maximum and minimum cell size of the elements in the generated mesh. This process returns an isosurface

representation as a *vedo.Mesh* having triangles connected the vertices of the surface. In this method, the function takes one argument "value," which can be a single value or a list of values and is used to compute the isosurface. Here, we considered a threshold value of 20, which was obtained after grid searching the threshold values across the isosurface method. Furthermore, we conducted up to 200 iterations to obtain smooth the surface mesh. Finally, the VTKPolyData, a 3D data structure, was used in this process, and the VTK marching cubes algorithm [67] was employed to convert the resulting VTKPolyData into a NumPy array of float64 to generate STL format—a suitable 3D mesh format.

4.4 Quality enhancement via noise removal

The last step of the 3D reconstruction process focuses on the quality improvement of the generated volume by eliminating unwanted voxels, if there is any. We utilized the "Islands Removal" algorithm [68] to eliminate any detached islands of (unwanted) voxels around the 3D femur bone. This algorithm was applied along with a threshold value that determined the size of the isolated "islands" of voxels, keeping the largest connected island of voxels by the filter. To obtain an optimal 3D femur surface while maintaining the femur geometry, manually optimized threshold values, ranging from 0.75 to 45.25 along the three axes, were used to eliminate isolated noise above the surface for each patient. The quality of the 3D surface was further enhanced by applying a median *smoothening* function to close any gaps by taking the median voxel values from the neighboring voxels. This process also helped eliminate additional noise in the 3D femur surface. Finally, we converted our reconstructed 3D femur in STL format to allow for further visualization and inspection.

4.5 Cross-validation

Cross-validation is a vital step of any ML/DL experiment, and an appropriate data split can significantly impact the model's performance. In this study, we employed LOOCV method. This approach involves dividing the dataset into multiple subsets, with each subset consisting of one sample as the testing set and the rest as the training set. During the experiments, an eight-fold cross-validation was used, where the data was divided into eight subsets of patients, and each subset served as a testing set once for one patient, while the remaining seven subsets served as the training set. By leveraging the LOOCV technique, we ensure that our model is trained and evaluated on a diverse set of data, and the results are robust to variations in the data distribution.
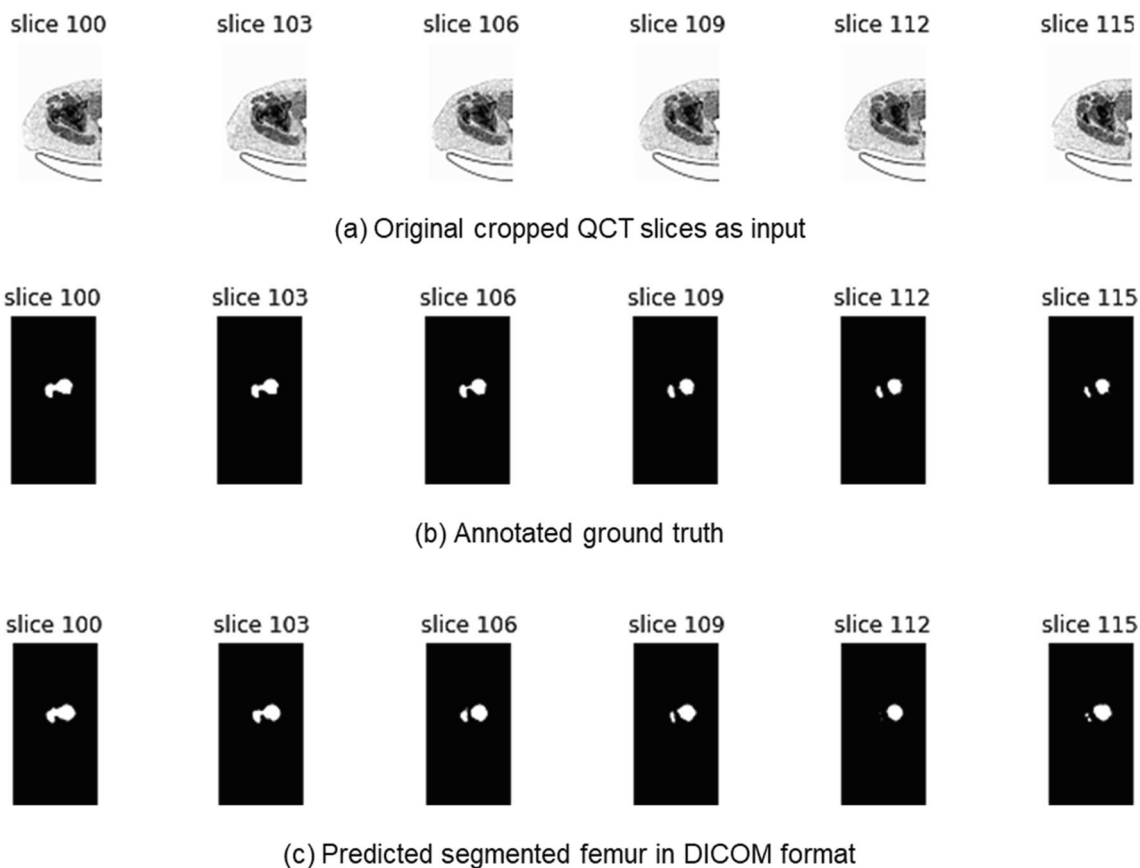
4.6 Testbed description

The testing environment for our experiments involved a personal laptop computer with an Intel Core i7-7500U CPU running at 2.90 GHz, 4 MB cache, and 8 GB of RAM. In addition, we utilized an NVIDIA GeForce GTX 940 MX GPU with 8094 MB of memory and a refresh rate of 60 Hz. We chose to use the Python3 programming language [69] due to its ease of use and availability of relevant libraries. For the implementation of our model, several Python libraries were incorporated in this work. We used TensorFlow [70], an open-source software library for accessible and programmable DL functions for segmentation, dataflow and differentiable tasks, PyDicom [71], which is a pure Python package for working with DICOM files, and SciPy [72] library for scientific and technical computing. We also used Skimage, which is a collection of algorithms for image processing and computer vision as well as the libraries associated with Google Colab, the cloud-based platform for data analysis and machine learning, and Visual Studio Code—widely used for Python development.

5 Results and discussion

The goal of this work was to develop an automated end-to-end SSDL model for the patient-specific 3D reconstruction of proximal femur. We have evaluated the performance of our model with a testing dataset 11,332 QCT slices of 20 femurs belonging to ten patients. Four among these ten patients with 4434 slices were known to the 3D U-Net segmentation model through validation process (validated dataset), and the other six patients with 6898 were completely new, unseen to our segmentation model, and independent of training and validation process. The outcomes of this work are two folds—2D segmented QCT images of femur and 3D reconstructed femur. Hence, the efficacy of our model was evaluated for both outcomes separately. The performance of both the outcomes was evaluated qualitatively and quantitatively for validated as well as for unseen patients to assess the robustness of our proposed SSDL model.

5.1 Segmented femur

The visual comparison depicts that the contours of the segmented femur at different locations via our proposed 3D U-Net method match very well with the ground truth binary mask (Fig. 5). This qualitative comparison suggests that our model for segmentation is accurate in isolating the regions of interest in the QCT slices. The SSDL model is able to capture the complex geometrical structure at the trochanteric regions and preserve the structural integrity of

slice 100 slice 103 slice 106 slice 109 slice 112 slice 115

(a) Original cropped QCT slices as input

slice 100 slice 103 slice 106 slice 109 slice 112 slice 115

(b) Annotated ground truth

slice 100 slice 103 slice 106 slice 109 slice 112 slice 115

(c) Predicted segmented femur in DICOM format

Fig. 5 Segmentation of femur on different QCT images in DICOM format using 3D U-Net segmentation model. **a** Input QCT slices, **b** annotated ground truth, and **c** predicted segmented femur

the femur's internal anatomy such as the hollow region of the femoral shaft is turned into solid with gradually changing bone density at the proximal end comprising trochanter and femoral head. Furthermore, the framework is capable of accurately predicting the voxel values later used for 3D femur reconstruction.

The quantitative comparison between the area of the segmented femurs obtained from our 3D U-Net model and the ground truth (Fig. 6) shows evidently that the proposed framework is able to segment femurs with comparatively high accuracy. The segmentation algorithm can accurately identify the boundaries of the femurs in the images. The maximum difference between the area values obtained from automated segmentation and manual segmentation (ground truth) is small, ranging from approximately 0.430 to 2.574 cm^2. This suggests that the segmentation algorithm can identify the femur area accurately and reliably.

Table 1 shows the quantitative segmentation performance of our proposed 3D U-Net model for both validated and unseen patients' data. All the performance indicators exhibited accuracy over 90% even with a limited dataset, indicating that the proposed model is capable of segmenting femurs

from DICOM images of unseen patients with comparable accuracy. The difference in performance indicators between the validated and unseen patients is ~ 7%, which may have originated from the variation of bone density distributions typically affected by osteoporosis, bone necrosis, etc. However, we do not have prior knowledge of any disease for the patients in the considered dataset. The 3D U-Net model was hyper tuned using validation data. The mean IoU and specificity show a good agreement in performance for both validated and unseen patients. Hence, we argue that with high specificity, our model can detect the pixels for non-femur values with high correctness even though a slight variation is observed for DSC, average precision, and sensitivity. In addition, the performance metrics in Table 1 also suggests that our automated model is generic such that it can perform any femur segmentation, and this model can be applied to any patient with geo-diversity like patients with different races, ethnicity, and age groups, etc.

The violin plot in Fig. 7 shows the distributions of the performance indicators (DSC, mean IoU, average precision, sensitivity, and specificity) as a continuous approximation of the probability density function (PDF), computed using
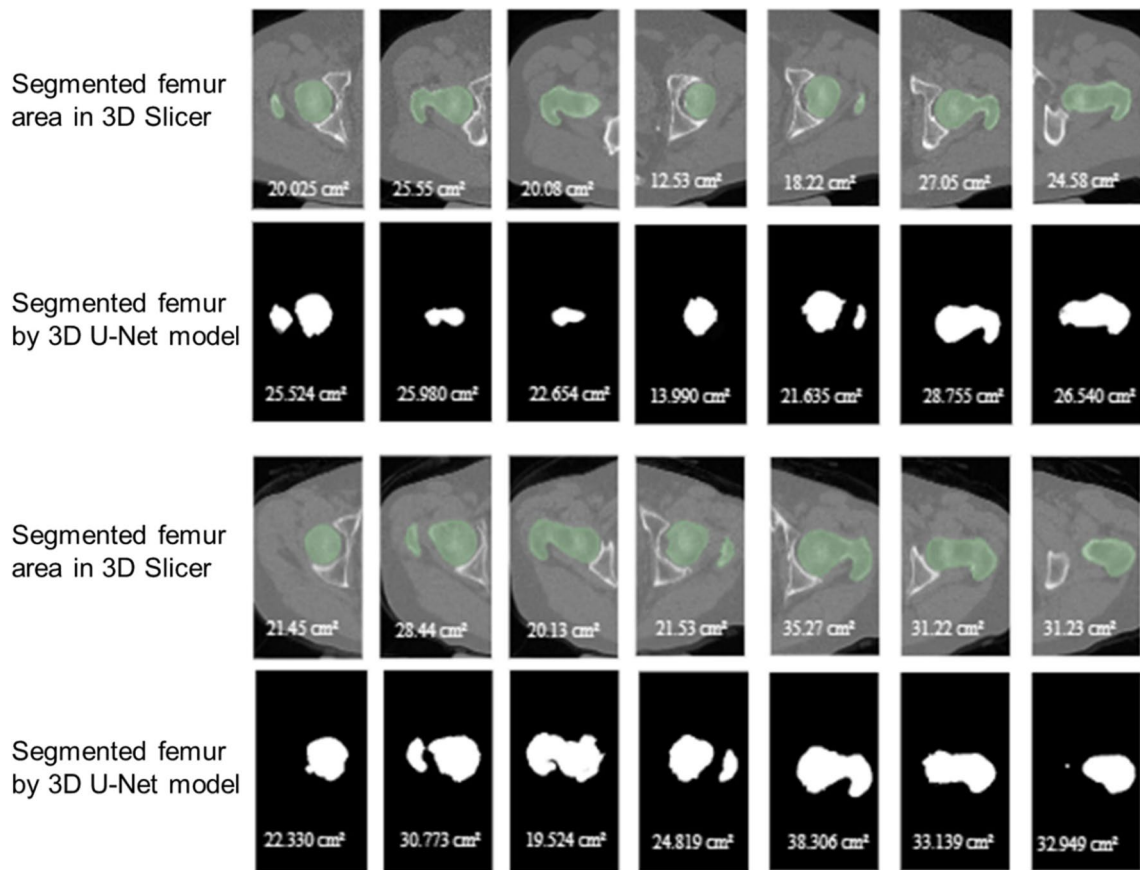
Fig. 6 Comparison of femur area between (manually/semi-automatically) selected femur contour (green shaded area) on original DICOM image using 3D Slicer and fully automated segmented femur in DICOM format using proposed 3D U-Net segmentation model. The QCT slices in the proximity of the hip joint are shown here as examples considering the most critical location in terms of segmentation

Table 1 Quantitative performance measure of femur segmentation via our 3D U-Net model. Patients' status validated, patients' data was visible to SSDL model training through validation process. Patients' status unseen, new patients' data that were not introduced in any DL process (only testing)

Patient's status	DSC	Mean IoU	Average precision	Sensitivity	Specificity
Validated	0.992 ± 0.008	0.995 ± 0.005	0.996 ± 0.003	0.995 ± 0.004	0.998 ± 0.002
Unseen	0.918 ± 0.025	0.983 ± 0.016	0.923 ± 0.024	0.927 ± 0.02	0.999 ± 0.002

kernel density estimation (KDE). The wider regions of the density plots indicate the performance indicator values occur more frequently, and the narrower sections represent a lower probability, indicating performance indicator values that occur less frequently. The violin plot uses KDE to compute an empirical distribution of training data, and therefore, it better reveals the information contained in the training and more convincingly suggests multimodality, if there is any. It is apparent that the distributions of all performance metrics are normal, indicating mean and median are same.

The SSDL model was trained with a total of 5474 QCT slices among which approximately 42% were annotated

slices and demonstrated its ability to generate a mask image from the CT scans input to isolate the femur from the other hip bones. Furthermore, by creating different mask images to segment the femur from the CT scan images, we can conserve the image data without any modification. To the best of our knowledge, the automated construction of femur using a DL framework is yet to be achieved successfully. It must be mentioned that it is considerably difficult and laborious task to annotate each DICOM file for supervised learning. Thus, we decided to use a semi-supervised DL model trained with a combination of annotated and raw DICOM slices. Most
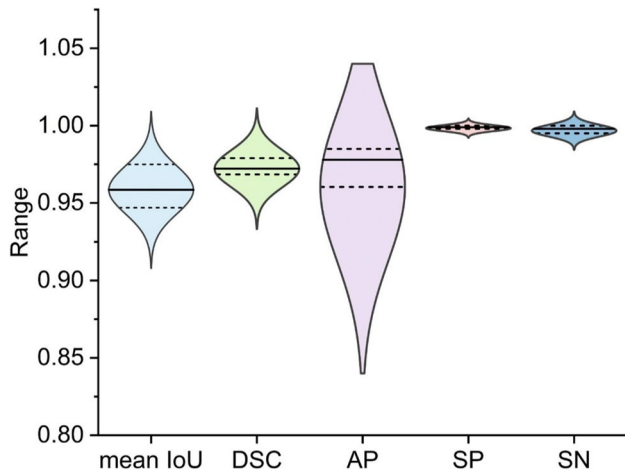
Fig. 7 Violin plots showing the distributions of performance indicator values for the segmentation results of femurs in testing dataset using 3D U-Net model. The solid line represents median; dotted lines are for 1st (bottom) and 3rd (top) quartiles. Here, AP. average precision; SP, specificity; SN, sensitivity

importantly, our segmentation framework works without any image augmentation, preserving all the metadata associated with original DICOM images. Table 2 shows a comparative analysis of the performance metrics of SSDL method with other existing works found in literature that align with our study. The results are promising and demonstrate its potential for accuracy improvement by increasing the training data. Accuracy can be further enhanced by resampling segmentation, cropping, padding, smoothing, etc., if necessary [47, 61].

5.2 3D reconstructed femur

With this adopted SSDL model, 3D proximal femurs can be reconstructed without manual intervention. 3D reconstructed femurs were analyzed both qualitatively and quantitatively.

5.2.1 Qualitative analysis

Figure 8 shows a 3D reconstructed femur on different orientations and planes. As observed, our model successfully segmented all the unwanted components of the QCT images, including muscle and fat, especially bone marrow, which is reflected as the hollow shaft section as shown in Fig. 8c (bottom view). To the best of our knowledge, no prior works considering different ML/DL models were able to generate a hollow shaft [14, 73] that gradually vanishes toward the proximal end and turns into solid mimicking trabecular bone at the proximal end. The bone models were also compared qualitatively with manually reconstructed femur using the widely used and publicly available software, 3D Slicer (https://www.slicer.org/). However, we observe a difference in shaft wall thickness between the automated and manually reconstructed femurs, and the natural variations in shaft wall thickness have also been reflected on the quantitative analyses.

The comparative analyses (Figs. 9 and 10) have been done based on the two separated datasets—validated and unseen patients' dataset as mentioned earlier. Figure 9 shows the anterior view of the 3D reconstructed left and right femurs of a validated patient (Fig. 9a) and an unseen patient
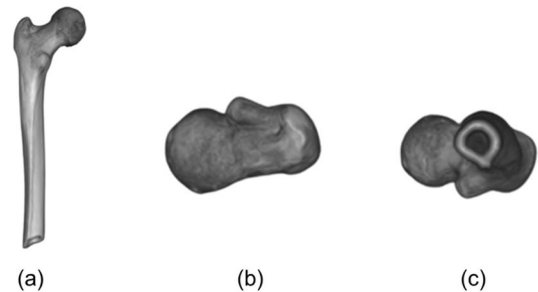
Fig. 8 An example of a 3D reconstructed proximal femur with our proposed SSDL model. **a** Posterior view on coronal plane, **b** top view, and **c** bottom view on transverse plane

Table 2 Comparative study of the performance metrics of SSDL method with other existing works found in literature. (Here, DSC, Dice Similarity Coefficient; ACC, accuracy; SN, sensitivity; SP, specificity)

Articles	Total patients	DSC	ACC	SN	SP
Deng et al. (endosteal surface segmentation) [47]	100 + augmentation	0.96		0.98	0.92
Chang et al. [73]	60 + augmentation	0.92			
Krcah et al. [34]	197		0.81		
Chen at el. [46]	300 + augmentation	0.96			
Carballido-Gamio et al. [16]	88 + augmentation	0.976			
Younes et al. [15] (SSM)	18 + augmentation	0.89			
Xia et al. [20]	52 + augmentation	0.81			
Jeuthe et al. [74] (semi-supervised)	30 + augmentation	0.914			
SSDL (validated) (our method)	18	0.992		0.995	0.998
SSDL (unseen) (our method)	18	0.918		0.927	0.999

(Fig. 9b). Figure 10 shows a greater detail by representing both anterior and posterior views of femurs of two unseen patients. A smoother surface on the anterior side is apparent than the posterior side of the femur, which may be impacted by the geometrically complex posterior anatomy.

5.2.2 Quantitative analysis

The 3D reconstructed femurs by the CNN-based 3D U-Net model have been quantitatively compared with manually reconstructed 3D femurs using 3D Slicer based on surface area and volume (Fig. 11). The differences in surface area and volume between the automated and manually generated 3D proximal femurs have been found as $10.749 \pm 7.979\,cm^2$ and $12.018 \pm 6.520\,cm^3$, respectively. We argue that the quantitative variations are fair enough because a relatively small dataset was considered for training and testing. However, the deviations might be originated from *smoothing* operation, which was rigorously used in manual intervention as well as due to the hollow section and its transition from the femoral shaft to the greater trochanteric region. Figure 12 shows the violin plots of the distributions of differences of volume (Fig. 12a) and surface area (Fig. 12b) between the femurs generated by SSDL and manually. It is evident that both plots exhibit normal distribution and a wider distribution near the median with nearly equal mean and median values, indicating a very good matching.

To further quantify the performance of our framework, we assessed several metrics, including *root mean square error* (RMSE), *mean average error* (MAE), *relative error* (RE), R^2, and *correlation coefficient* values for both the surface area and volume of the 3D reconstructed femurs (Table 3). The small values of RMSE, MAE, and RE indicate that


Fig. 9 Examples of 3D femur reconstruction from segmented DICOM files via 3D U-Net model. Anterior views of right and left femurs of a validated patient (**a**) and an unseen patient (**b**)
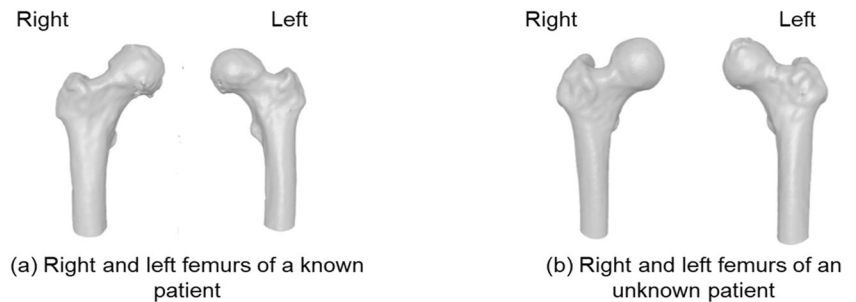

Fig. 10 Examples of 3D reconstructed femurs (right only) of unseen patients (1 and 2) with posterior and anterior views
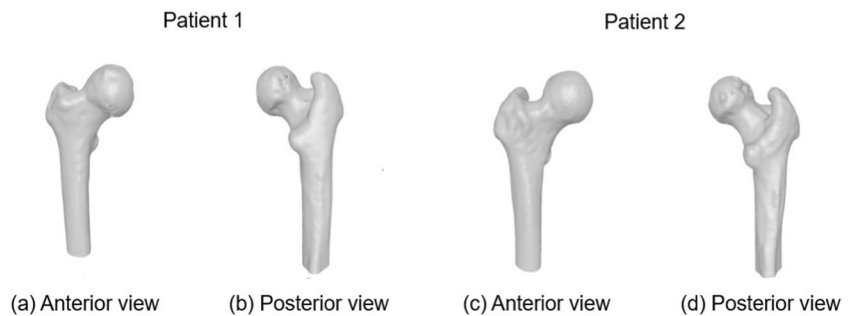
Fig. 11 Comparison of surface area (**a**) and volume (**b**) between 3D reconstructed femurs via SSDL framework and manually reconstructed femur via 3D Slicer (ground truth). Bar plots within solid boundary represent the validated patients, and the bar plots within dashed boundary represent the unseen patients
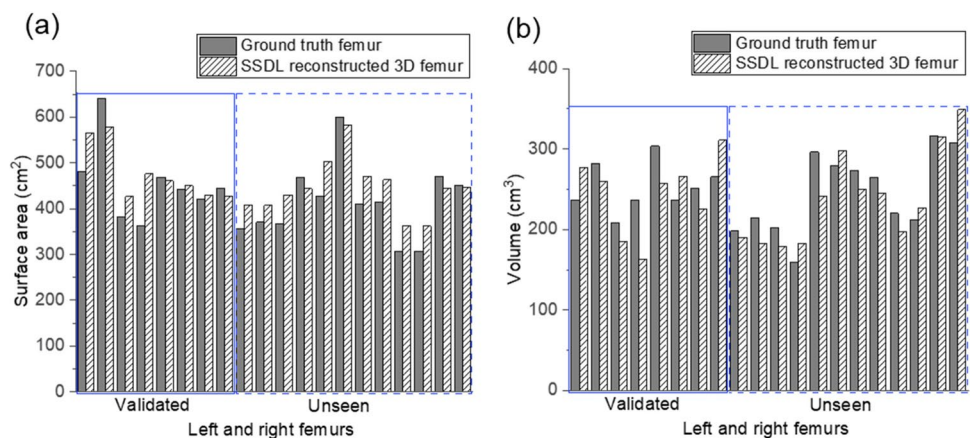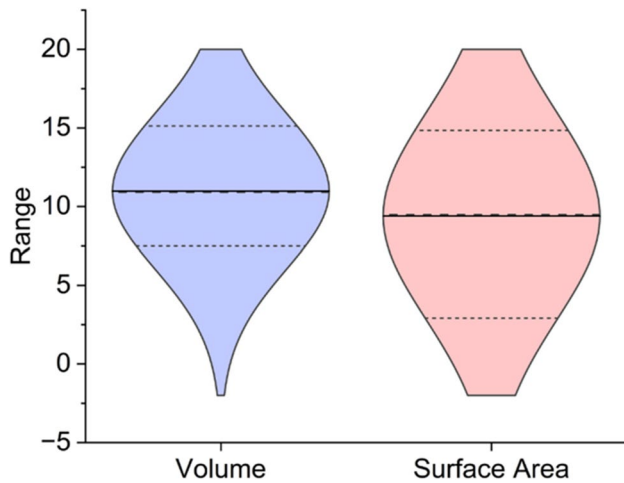
Fig. 12 Violin plots showing the distributions of the differences in the volume of 3D reconstructed femurs and the surface area of segmented femur obtained via SSDL approach and manually (ground truth)

Table 3 Performance measure of 3D reconstructed femur by SSDL approach in comparison with manually reconstructed femur. Here, RMSE, root mean square error; MAE, mean average error; RE, relative error

	RMSE	MAE	RE (%)	R^2	Correlation coefficient
Volume (cm^3)	33.820	29.556	6.610	0.338	0.771
Surface area (cm^2)	51.839	43.077	12.082	0.592	0.846

our framework exhibits low error and better accuracy in generating 3D femurs. R^2 values are used to assess how well our framework fits the purpose of generating 3D femurs. Our 3D femur reconstruction framework shows low to moderate evaluation of variants with values close to 0.34 (for volume) and 0.6 (for surface area). However, with strong correlation coefficient values of 0.77 (for volume) and 0.85 (for surface area), our 3D femur reconstruction framework shows relatively small errors and a strong positive relationship between the evaluated variables, indicating that it performs well in the reconstruction of 3D femurs from our segmented DICOM files.

Further, the proposed framework requires significantly less time, approximately 85 ± 10 s to reconstruct the 3D bone once the QCT image dataset is loaded. On the other hand, the same process takes between 30 and 45 min with a conventional image processing software, 3D Slicer, by a well-trained undergraduate research assistant who has been working for nearly 3 years [6]. Therefore, our model presents a more efficient and time-saving alternative to the existing conventional methods.

5.3 Limitations and challenges

Although this work shows promising performance in developing an automated DL-based framework, the proposed SSDL approach has few limitations. First, an accurate annotation of QCT image dataset for training is a prerequisite for better performance. The annotation process for labeling bone profile on each QCT slice is time-consuming, expensive, and requires some level of expertise. Since the annotations were manually conducted, we occasionally encountered mismatch of the femur boundary due to conservative annotation as well as difficulty in identifying the bone profiles, requiring redo of the annotation process. Although extreme care has been taken, a small percentage of annotation errors may be anticipated in the training dataset, and this error might impact the performance of our framework. We have noticed some negligible errors near the femoral head than the other parts of the femur. Due to conservative annotation, the model may predict femur head less clearly. Thus, a cascading error in the 3D reconstruction can be possible. We have further noticed that our framework in some cases shows suboptimal 3D femur reconstruction in proximal areas such as Fig. 9a. We believe that this limitation will be resolved with more accurately annotated femur slices for the training dataset. Second, finding a patient-specific optimum segmentation threshold is difficult, and bone properties clustering does not yield good results [74]. These thresholding values also contribute to producing isolated islands during the 3D model generation. We need to conduct further analysis with more patients to observe the patterns before applying optimal threshold values. Our primary goal is to develop an automated segmentation framework tailored for the proximal femur with specific relevance to musculoskeletal applications. Our dataset used in this model lacks images of fractured femurs, limiting our assessment of the framework's performance across varying fracture severities. While it may excel in hairline fractures as the overall bone structure is minimally affected, its efficacy remains uncertain for severe fractures with substantial disruption of bone continuity due to insufficient data. Thus, datasets with more patients and different races, ethnicity, medical conditions such as with and without osteoporosis, as well as fractured bone should be included in both training and testing to increase its fidelity to work with a wide variety of patients.

6 Conclusion

In conclusion, we have presented an efficient, faster automated semi-supervised deep leaning-based model for femur segmentation and 3D femur reconstruction. To minimize the cost and time required for data

pre-processing, the proposed approach aimed at keeping the annotation effort to a minimum without compromising the annotation accuracy, which is a critical step for a supervised or semi-supervised framework. Only the QCT slices at the proximal end where multiple bones such as femur, acetabulum, and pelvis are in proximity were annotated. The ability to preserve DICOM metadata interwoven with the segmented QCT images was one of the key aspects of this framework. By selecting a minimal number of slices to annotate and generating binary masks from these annotations, the proposed approach was able to leverage the power of deep learning while minimizing the effort required for data pre-processing. Overall, this approach offers a promising solution for medical image analysis via reconstructing 3D model, which is a crucial step in the development of accurate and effective automated analysis tools for clinical practice.

Declarations

Competing interests The authors declare no competing interests.

References

1. Sheehan SE, Shyu JY, Weaver MJ, Sodickson AD, Khurana B (2015) Proximal femoral fractures: what the orthopedic surgeon wants to know. Radiographics 35(5):1563–1584

2. Siebenlist S, Torsiglieri T, Kraus T, Burghardt R, Stöckle U, Lucke M (2010) Comminuted fractures of the proximal ulna—preliminary results with an anatomically preshaped locking compression plate (LCP) system. Injury 41(12):1306–1311

3. Grignon B, Oldrini G, Walter F (2016) Teaching medical anatomy: what is the role of imaging today? Surg Radiol Anat 38:253–260

4. Kasban H, El-Bendary M, Salama D (2015) A comparative study of medical imaging techniques. Int J Information Sci Intell Syst 4(2):37–58

5. Ohnaru K, Sone T, Tanaka K, Akagi K, Ju Y-I, Choi H-J et al (2013) Hip structural analysis: a comparison of DXA with CT in postmenopausal Japanese women. Springerplus 2(1):1–8

6. Anez-Bustillos L, Derikx LC, Verdonschot N, Calderon N, Zurakowski D, Snyder BD et al (2014) Finite element analysis and CT-based structural rigidity analysis to assess failure load in bones with simulated lytic defects. Bone 58:160–167

7. Benca E, Synek A, Amini M, Kainberger F, Hirtler L, Windhager R et al (2019) QCT-based finite element prediction of pathologic fractures in proximal femora with metastatic lesions. Sci Rep 9(1):10305

8. Faisal TR, Luo Y (2017) Study of the variations of fall induced hip fracture risk between right and left femurs using CT-based FEA. Biomed Eng Online 16:1–17

9. Lee Y, Ogihara N, Lee T (2019) Assessment of finite element models for prediction of osteoporotic fracture. J Mech Behav Biomed Mater 97:312–320

10. Mirzaei M, Keshavarzian M, Naeini V (2014) Analysis of strength and failure pattern of human proximal femur using quantitative computed tomography (QCT)-based finite element method. Bone 64:108–114

11. Travascio F, Buller LT, Milne E, Latta L (2021) Mechanical performance and implications on bone healing of different screw configurations for plate fixation of diaphyseal tibia fractures: a computational study. Eur J Orthop Surg Traumatol 31:121–130

12. Solitro GF, Welborn MC, Mehta AI, Amirouche F (2024) How to optimize pedicle screw parameters for the thoracic spine? A biomechanical and finite element method study. Global. Spine J 14(1):187–194

13. Solitro GF, Mainnemare F, Amirouche F, Mehta A (2019) A novel technique with reduced computed tomography exposure to predict vertebral compression fracture: a finite element study based on rat vertebrae. Med Biol Eng Comput 57:795–805

14. Koh K, Kim YH, Kim K, Park WM (2011) Reconstruction of patient-specific femurs using X-ray and sparse CT images. Comput Biol Med 41(7):421–426

15. Ben Younes L, Nakajima Y, Saito T (2014) Fully automatic segmentation of the femur from 3D-CT images using primitive shape recognition and statistical shape models. Int J Comput Assist Radiol Surg 9:189–196

16. Carballido-Gamio J, Bonaretti S, Saeed I, Harnish R, Recker R, Burghardt AJ et al (2015) Automatic multi-parametric quantification of the proximal femur with quantitative computed tomography. Quant Imaging Med Surg 5(4):552

17. Chu C, Bai J, Wu X, Zheng G (2015) MASCG: Multi-atlas segmentation constrained graph method for accurate segmentation of hip CT images. Med Image Anal 26(1):173–184

18. Xia Y, Fripp J, Chandra SS, Schwarz R, Engstrom C, Crozier S (2013) Automated bone segmentation from large field of view 3D MR images of the hip joint. Phys Med Biol 58(20):7375

19. Chandra SS, Xia Y, Engstrom C, Crozier S, Schwarz R, Fripp J (2014) Focused shape models for hip joint segmentation in 3D magnetic resonance images. Med Image Anal 18(3):567–578

20. Xia Y, Chandra SS, Engstrom C, Strudwick MW, Crozier S, Fripp J (2014) Automatic hip cartilage segmentation from 3D MR images using arc-weighted graph searching. Phys Med Biol 59(23):7245

21. Gilles B, Magnenat-Thalmann N (2010) Musculoskeletal MRI segmentation using multi-resolution simplex meshes with medial representations. Med Image Anal 14(3):291–302

22. Korfiatis VC, Tassani S, Matsopoulos GK (2017) An independent active contours segmentation framework for bone micro-CT images. Comput Biol Med 87:358–370

23. Despotović I, Goossens B, Philips W (2015) MRI segmentation of the human brain: challenges, methods, and applications. Comput Math Methods Med 2015

24. Ahmed SM, Mstafa RJ (2022) A comprehensive survey on bone segmentation techniques in knee osteoarthritis research: from conventional methods to deep learning. Diagnostics. 12(3):611

25. Besler BA, Michalski AS, Kuczynski MT, Abid A, Forkert ND, Boyd SK (2021) Bone and joint enhancement filtering: application to proximal femur segmentation from uncalibrated computed tomography datasets. Med Image Anal 67:101887

26. Santarelli C, Argenti F, Uccheddu F, Alparone L, Carfagni M (2020) Volumetric interpolation of tomographic sequences for accurate 3D reconstruction of anatomical parts. Comput Methods Programs Biomed 194:105525

27. Grassi L, Hraiech N, Schileo E, Ansaloni M, Rochette M, Viceconti M (2011) Evaluation of the generality and accuracy of a new mesh morphing procedure for the human femur. Med Eng Phys 33(1):112–120

28. Schmid J, Kim J, Magnenat-Thalmann N (2011) Robust statistical shape models for MRI bone segmentation in presence of small field of view. Med Image Anal 15(1):155–168

29. Kardell M, Magnusson M, Sandborg M, Alm Carlsson G, Jeuthe J, Malusek A (2016) Automatic segmentation of pelvis for brachytherapy of prostate. Radiat Prot Dosimetry 169(1-4):398–404

30. Vasilache S, Najarian K (2008) Automated bone segmentation from pelvic CT images. In: 2008 IEEE International Conference on Bioinformatics and Biomeidcine Workshops. IEEE, pp 41–47

31. Kalshetti P, Bundele M, Rahangdale P, Jangra D, Chattopadhyay C, Harit G et al (2017) An interactive medical image segmentation framework using iterative refinement. Comput Biol Med 83:22–33

32. Stawiaski J, Decenciere E, Bidault F (2008) Interactive liver tumor segmentation using graph-cuts and watershed. In: 11th international conference on medical image computing and computer assisted intervention (MICCAI 2008)

33. Malusek A, Magnusson M, Sandborg M, Alm CG (2017) A model-based iterative reconstruction algorithm DIRA using patient-specific tissue classification via DECT for improved quantitative CT in dose planning. Med Phys 44(6):2345–2357

34. Krčah M, Székely G, Blanc R (2011) Fully automatic and fast segmentation of the femur bone from 3D-CT images with no shape prior. In: 2011 IEEE international symposium on biomedical imaging: from nano to macro. IEEE, pp 2087–2090

35. Lopez-Jimenez F, Attia Z, Arruda-Olson AM, Carter R, Chareonthaitawee P, Jouni H et al (2020) Artificial intelligence in cardiology: present and future. Mayo Clinic Proc 95(5):1015–1039

36. Peña-Solórzano CA, Albrecht DW, Bassed R, Gillam J, Harris P, Dimmock M (2020) Semi-supervised labelling of the femur in a whole-body post-mortem CT database using deep learning. Comput Biol Med 122:103797

37. Rokaya D, Kongkiatkamon S, Heboyan A, Dam VV, Amornvit P, Khurshid Z et al (2022) 3d-printed biomaterials in biomedical application. In: Functional Biomaterials: Drug Delivery and Biomedical Applications. Springer, pp 319–339

38. Luca AR, Ursuleanu TF, Gheorghe L, Grigorovici R, Iancu S, Hlusneac M et al (2022) Impact of quality, type and volume of data used by deep learning models in the analysis of medical images. Inform Med Unlocked 29:100911

39. Yu AC, Mohajer B, Eng J (2022) External validation of deep learning algorithms for radiologic diagnosis: a systematic review. Radiology. Artificial Intelligence 4(3):e210064

40. Ronneberger O, Fischer P, Brox T (2015, Proceedings, Part III 18) U-net: convolutional networks for biomedical image segmentation. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015. In: 18th International Conference. Springer, Munich, Germany, pp 234–241

41. Sánchez JCG, Magnusson M, Sandborg M, Tedgren ÅC, Malusek A (2020) Segmentation of bones in medical dual-energy computed tomography volumes using the 3D U-Net. Phys Med 69:241–247

42. Klein A, Warszawski J, Hillengaß J, Maier-Hein KH (2019) Automatic bone segmentation in whole-body CT images. Int J Comput Assist Radiol Surg 14:21–29

43. Noguchi S, Nishio M, Yakami M, Nakagomi K, Togashi K (2020) Bone segmentation on whole-body CT using convolutional neural network with novel data augmentation techniques. Comput Biol Med 121:103767

44. Pham T-T, Le M-B, Le LH, Andersen J, Lou E (2021) Assessment of hip displacement in children with cerebral palsy using machine learning approach. Med Biol Eng Comput 59(9):1877–1887

45. Wani IM, Arora S (2020) Computer-aided diagnosis systems for osteoporosis detection: a comprehensive survey. Med Biol Eng Comput 58:1873–1917

46. Chen F, Liu J, Zhao Z, Zhu M, Liao H (2017) Three-dimensional feature-enhanced network for automatic femur segmentation. IEEE J Biomed Health Inform 23(1):243–252

47. Deng Y, Wang L, Zhao C, Tang S, Cheng X, Deng H-W et al (2022) A deep learning-based approach to automatic proximal femur segmentation in quantitative CT images. Med Biol Eng Comput 60(5):1417–1429

48. Zhu L, Han J, Guo R, Wu D, Wei Q, Chai W et al (2020) An automatic classification of the early osteonecrosis of femoral head with deep learning. Current medical imaging 16(10):1323–1331

49. Awal R, Ben Hmida J, Luo Y, Faisal T (2022) Study of the significance of parameters and their interaction on assessing femoral fracture risk by quantitative statistical analysis. Med Biol Eng Comput 60(3):843–854. https://doi.org/10.1007/s11517-022-02516-0

50. Faisal TR, Luo Y (2016) Study of stress variations in single-stance and sideways fall using image-based finite element analysis. Biomed Mater Eng 27(1):1–14

51. Wang C, Dahlström N, Fransson S-G, Lundström C (2015) Real-time interactive 3D tumor segmentation using a fast level-set algorithm. J Med Imaging Health Inform 5(8):1998–2002

52. Khan S, Warkhedkar R, Shyam A (2014) Analysis of Hounsfield unit of human bones for strength evaluation. Procedia Mater Sci 6:512–519

53. He K, Zhang X, Ren S, Sun J (2015) Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision, pp 1026–1034

54. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15(1):1929–1958

55. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778

56. Dubey SR, Singh SK, Chaudhuri BB (2022) Activation functions in deep learning: a comprehensive survey and benchmark. Neurocomputing 503:92–108

57. Rezatofighi H, Tsoi N, Gwak J, Sadeghian A, Reid I, Savarese S (2019) Generalized intersection over union: a metric and a loss for bounding box regression. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 658–666

58. Sudre CH, Li W, Vercauteren T, Ourselin S, Jorge CM (2017) Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017. Springer, Québec City, QC, Canada Proceedings 3, pp 240–248

59. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. Int J Comput Vis 88:303–338

60. Ting KM (2010) Sensitivity and specificity. In: Sammut C, Webb GI (eds) Encyclopedia of Machine Learning. Springer US, Boston, MA, pp 901–902

61. Bjornsson PA, Helgason B, Palsson H, Sigurdsson S, Gudnason V, Ellingsen LM (2021) Automated femur segmentation from computed tomography images using a deep neural network. In: Medical Imaging 2021: Biomedical Applications in Molecular, Structural, and Functional Imaging. SPIE, pp 324–330

62. Sammut C, Webb GI (2010) Leave-one-out cross-validation. In: Encyclopedia of machine learning, pp 600–601

63. Kingma DP, Ba J. Adam (2014) A method for stochastic optimization. arXiv preprint arXiv:14126980

64. Wolberg G, Alfy I (2002) An energy-minimization framework for monotonic cubic spline interpolation. J Comput Appl Math 143(2):145–188

65. Musy M, Jacquenot G, Dalmasso G, de Bruin R, Pollack A, Claudi F et al (2021) Vedo: a python module for scientific analysis and visualization of 3D objects and point clouds. Zenodo

66. Chaudhary A, Jhaveri SJ, Sanchez A, Avila LS, Martin KM, Vacanti A et al (2019) Cross-platform ubiquitous volume rendering using programmable shaders in VTK for scientific and medical visualization. IEEE Comput Graph Appl 39(1):26–43

67. Lorensen WE, Cline HE (1987) Marching cubes: a high resolution 3D surface construction algorithm. ACM Siggraph Computer Graphics 21(4):163–169

68. Tan Y, Hua J, Dong M (2007) 3D reconstruction from 2D images with hierarchical continuous simplices. Vis Comput 23(9):905–914. https://doi.org/10.1007/s00371-007-0157-0

69. Guido VR, Drake F Jr (2009) Python 3 reference manual. CreateSpace, Scotts Valley

70. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C et al (2016) Tensorflow: large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:160304467

71. Mason D (2011) SU-E-T-33: pydicom: an open source DICOM library. Med Phys 38(6 Part 10):3493

72. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D et al (2020) SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat Methods 17(3):261–272

73. Zhao C, Keyak JH, Tang J, Kaneko TS, Khosla S, Amin S et al (2020) A deep learning-based method for automatic segmentation of proximal femur from quantitative computed tomography images. arXiv preprint arXiv:200605513

74. Jeuthe J, Sánchez JCG, Magnusson M, Sandborg M, Tedgren ÅC, Malusek A (2021) Semi-automated 3D segmentation of pelvic region bones in CT volumes for the annotation of machine learning datasets. Radiat Prot Dosimetry 195(3-4):172–176

Jamalia Sultana received B.Sc. Engineering degree in Computer Science and Engineering (CSE) from the Bangladesh University of Engineering and Technology (BUET) and completed M.Sc. Engineering degree also from the same department in 2023 with a perfect CGPA. She started her PhD in CS in SUNY at Stonybrook, NY, USA from Fall 2023. She published international journal and conference papers from her B.Sc. and MSc. thesis including IEEE COMPSAC, IEEE ICMLA, IEEE Access. She was the first Graduate Fellow of the Department of CSE, BUET. Her research interests include deep learning, medical image processing, mathematical models, and data security.

Dr. Mahmuda Naznin received her B.Sc. and M.Sc. degrees in Computer Science and Engineering from Bangladesh University of Engineering and Technology (BUET), and M.S. and Ph.D. degrees in CS from North Dakota State University (NDSU), USA. She is working as a professor and currently as the Head of the Department of CSE, BUET. She has published more than 50 research articles in peer reviewed journals and in proceedings of international conferences by IEEE, ACM, Springer. She is involved in different professional activities and serves as a reviewer for internationally reputed journals. She has served as TPC co-chair, TPC member, and panelist in different international conferences and workshops.

Dr. Tanvir R. Faisal is an Assistant Professor at the University of Louisiana at Lafayette and the Principal Investigator of the Musculoskeletal Mechanics & Multiscale Materials (4M) Laboratory. His research focuses on computational and experimental biomechanics, especially focusing on orthopedics. His research goal is to understand the root cause of degenerative joint diseases such as osteoporosis and osteoarthritis. His research focus is to develop a data-driven computational framework to develop predictive model for early detection of degenerative diseases.