August 2022

Python-Based Analysis to Segment Bone and Soft Tissue in a Healing Callus

Roberto Lopez
University of Wisconsin-Milwaukee

PYTHON-BASED ANALYSIS TO SEGMENT BONE AND SOFT TISSUE IN A HEALING CALLUS

by

Roberto Lopez

A Thesis Submitted in

Partial Fulfillment of the

Requirements for the Degree of

Master of Science

in Engineering

at

The University of Wisconsin – Milwaukee

August 2022

ABSTRACT

PYTHON-BASED ANALYSIS TO SEGMENT BONE AND SOFT TISSUE IN A HEALING CALLUS

by

Roberto Lopez

The University of Wisconsin-Milwaukee, 2022
Under the Supervision of Professor Priyatha Premnath

The main objective of this study was to produce a Python script that would help facilitate the segmentation process of both bone and soft tissue. The proposed script, in tangent with ImageJ and Mimics, was successful in producing viable results when the bone and soft tissue sample was placed near hydroxyapatite (HA) phantoms during the image acquisition process. It was important to acquire both the HA phantoms and the sample within the same image sequence as the script functioned by analyzing the statistical distribution of the different HA regions to locate the most ideal thresholding ranges to determine the bone mineral density (BMD) percent composition. When the sample and HA phantoms were in the same set of images, they were both subject to the same type of noise and attenuation, thus allowing for better results to be produced. The script was successful in processing input images and was able to calculate the overall volume and surface area of both the bone and soft tissue, as well as determining the overall bone mineral density of bone. It was also attempted to process bone and soft tissue samples separate of the HA phantoms, but the results were inconclusive.

To all the people who never stopped believing in me,

Thanks for being patient with me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

HA	Hydroxyapatite
BMD	Bone Mineral Density
CT	Computed Tomography
MRI	Magnetic Resonance Imaging
AI	Artificial Intelligence
IACUC	Institutional Animal Care and Use Committee

ACKNOWLEDGEMENTS

I'd like to thank my advisor, Dr. Premnath, for without her I would have never learned about my fascination with coding. She was the one who encouraged me to learn Python, as well as sign up for those graduate level computer science classes even though I had only ever taken the intro 101 MATLAB class beforehand. Thanks to you, I found an activity to do that is both challenging and rewarding.

I'd like to thank Dr. Mahsa Dabagh, the UWM professor who allowed me to use her Mimics software license to process the soft tissue masks. Without this essential software, it would have been more difficult to complete my thesis the way I wanted to.

I'd like to thank PhD student Rahul Khandge Ratz, who helped and taught me how to operate the MicroCT machine. Thanks for your willingness to help whenever you had the time.

I'd like to thank Nou Thao and Humza Siddiqui, who are both UWM undergrad students who worked in Dr. Premnath's lab and were responsible for the mice sample preparation.

A big thanks to my own father, Roberto Lopez Sr., who not only helped me continue on even when I hit massive emotional walls; but also helped in the design and construction of a apparatus that could be used to capture scanned images that contained both the HA phantom and the sample. Although the phantom used in this thesis was not the correct type, with his help I was able to develop a method that could be used in the future with an appropriate phantom.

A big thanks to my girlfriend, Elizabeth Komp, who constantly supported me throughout my entire master's degree and helped with the finishing touches on my paper.

FRONTISPIECE

1. Introduction

1.1 Overview of Bone Properties

When dealing with bones, nobody likes a broken one. However, as this is the real world, there is roughly a 45% chance of a long bone fracture occurring during one's life [1, 2]. While the healing mechanisms of a fracture are understood well enough, there is still a chance that a fracture will never properly heal. When a fracture fails to regenerate it is called a nonunion fracture and in 2016, a national team of orthopedic surgeons investigated the health records of 90.1 million patients with the goal to better describe the epidemiology of fracture nonunions in adults [3].

The researchers determined that nonunion fractures occur at an overall average of 4.93% with significant variation from bone to bone. The bones that were more likely to heal were in the arms, while the bones that had a higher chance of a nonunion fracture occurring were in the legs. With the tibia having a 13.95% chance of a nonunion fracture occurring, trying to gather more information regarding the entire healing process of this bone could provide greater insight into how to better treat these types of injuries [3].

Bone tissue is made of a two-phase composite material, which is mostly collagen and mineral [4]. Collagen makes up around 90% of the organic content of bone, intertwining to form flexible and slightly elastic fibers in the bone. The collagen of mature bones is mixed with a dense inorganic filling component called hydroxyapatite that allows for the hardening of bone. Crystals from the mineral form a calcium phosphate, that impregnates the collagen matrix within bones [5].

Along with being on the higher end of the spectrum for the possibility of nonunion fractures to occur, tibial fractures are also the third most fractured bone [6]. They are in general commonly described by their location (i.e., proximal, middle, distal) and by the fracture configuration (i.e., transvers, oblique, spiral, comminuted, or segmental) [7]. They occur when an external force transfers energy into the bone that is above the maximum that can be absorbed and properly distributed, the bone will break [4].

The amount of force that enters the bone is not the only component that will determine if a bone breaks. The density of trabecular structures also plays an important role in determining the quality of the bone, being directly related to bone strength and loading capabilities. It is measured as bone mineral density (BMD) and is the index most frequently used to quantify the strength of bone, while being defined as bone mass per unit area (g/cm^2) or volume (g/cm^3) [8]. That means if a region of the bone has low BMD, it is already at a greater likelihood of a fracturing occurring at a lower limit of excess external force entering the system.

1.2 Healing process of Bone Fractures

A fracture is defined as a breach in the structural continuity of the bone cortex, also resulting in a degree of injury to nearby soft tissue [9]. Hematopoietic and immunological cells within the bone marrow work in concert with vascular and skeletal cell precursors that are regenerated from the surrounding tissues to aid in the repair of fractures [10]. The healing process for bone fractures occurs throughout three progressive stages: 1.) the inflammatory phase, 2.) the repairing phase, and 3.) the bone remodeling phase [11].

Every healing process begins with the formation of a hematoma during the inflammatory phase, with this initial phase being one of the more significant determinants of the healing outcome [12, 13]. Following the formation of a hematoma, a specific cytokine pattern is initiated that is crucial in guiding and coordinating the cell recruitment, as well as the cellular activity of those cells [14]. This cytokine pattern stimulates critical cellular biology at the site and cells (i.e., macrophages, lymphocytes, etc.) called upon act together removing injured, necrotic tissue while also secreting vascular endothelial growth factors and other cytokines to encourage healing [9].

The second stage, the repairing phase, can be broken up into two different sections: 2.a) the fibrocartilaginous callus formation, and 2.b) the bony callus formation [9]. During the formation of the fibrocartilaginous callus, fibroblasts and osteoblasts enter the area to begin the reformation of bone. The fibroblasts are there to produce collagen fibers that attach the broken bone ends, while the osteoblasts are tasked with the formation of spongy, woven bone [15]. After around 11 days since the formation of the fibrocartilaginous callus, this cartilaginous callus begins undergoing endochondral ossification and the bone healing process moves to the formation of the bony callus [16]. This results in the cartilaginous callus starting to be resorbed and begins to calcify, all while the continuation of woven bone being laid down. This woven bone allows for the newly developed blood vessels to proliferate, promoting the further migration of mesenchymal stem cells [17]. This second stage ends with the formation of a hard, calcified callus of an immature bone.

The third and final stage, the bone remodeling phase, continues with the migration of osteoblasts and osteoclasts which allows the hard callus to undergo coupled remodeling [9]. This remodeling is essentially a balancing act between the resorption by osteoclasts and the formation of new bone by osteoblasts. The process of bone remodeling lasts many months and results in the center of the callus eventually being replaced with compact bone, while the callus edges get replaced with lamellar bone [16].

1.3 Role of Soft Tissue during the Repair of a Bone Fracture

When a fracture occurs, the energy release from a bone breaking gets transferred to the muscles, tendons, vascular structures, nerves and other soft tissue that happen to be in close proximity to the injury site [18]. Immediately after the fracture occurs, the recruitment, multiplication and accumulation of stem cells begins during the development of a soft callus [19]. When the fracture healing reaches the hard callus phase, the tissue becomes vascularized due to the calcification of cartilage and ossification of newly formed bone [20].

Both intramembranous and endochondral ossification contribute to the healing of fractures [19]. Stem cells go through chondrogenesis during endochondral ossification and create an intermediary cartilage structure. The cartilage callus' chondrocytes develop into mature, calcified cartilage that serves as a model for bone growth and promotes angiogenesis. Osteoblasts differentiate from stem cell precursors directly during intramembranous ossification, and bone forms predominantly without the aid of cartilage as a template. It has also recently been demonstrated that chondrocytes can transdifferentiate into osteoblasts directly. The pathway by which this occurs is still undetermined [21].

Recent research has shown that muscle-derived stem cells can develop into cartilage and bone, while also actively aiding in the healing of bone fractures [18]. For fractures that are connected to periosteal injuries, stem cells derived from muscle specifically play a vital role [22].

Even though evidence suggests that stem cells originating from all sources near the injury site contribute during the repair process, one of the primary sources of stem cells does appear to come from the periosteal tissue [19]. For these types of injuries, research have been able to demonstrate through cell lineage tracing that the precursor cell pool that occurs before the start of the fracture healing process is initially accumulated by periosteum-derived stem cells [23].

1.4 Impact of Image Segmentation during Analysis of Bone Fractures

Bone fractures, as well as any other injuries or diseases pertaining to bone, require the use of medical image capturing devices. Computed tomography (CT), or MicroCT, is commonly used to determine values such as bone volume fraction and volumetric BMD, which have been found to have a high correlation with bone loading ability and strength [8, 24]. Evaluations to determine if a certain region of the bone has either high or low BMD requires the use of a bone-equivalent material, such as potassium phosphate (K_2HPO_4) solutions or calcium hydroxyapatite (CaHA) mixtures, for calibration during the imaging acquisition process [25]. These bone-equivalent material contains different concentrations of mineral densities that are used as a reference point in conjunction with the CT imaging. By knowing the percent weights of the different concentrations within the solution phantoms, the equivalent volumetric BMDs can be determined by using image segmentation [26, 27].

Currently, images produced from CT scans must be processed with either manual or semi-automatic segmentation to create a bone surface data set where the important information can be analyzed [28]. Accurate automatic segmentation is difficult since bone surfaces have inconsistent properties (i.e., shape, density, structure, etc.), ambiguous boundaries, and are often accompanied by artifacts and noise requiring more parameter changes depending on the structure [29, 30].

Being able to accurately differentiate between regions with low bone mineral density and high bone mineral density has become essential for determining the overall strength and health of a bone [31]. The main objective of image segmentation is to transform an input image into informative data, thus allowing for the successful analysis of that image. There are a variety of segmentation techniques, with most falling under the classification of the following: 1.) contour-based techniques, 2.) region-based techniques, 3.) feature thresholding, and 4.) clustering [32].

1.5 Current Challenges in Bone and Soft Tissue Image Segmentation

This thesis set out to create an algorithm that could function as a bridge that connected the individually processed image segmentation of bone and individually processed image segmentation of soft tissue. As most other studies simply focus on either the image segmentation of bone or soft tissue while ignoring the other component, the main objective of this thesis was to produce an algorithm capable of segmenting both the bone and the soft tissue [8, 26, 27, 28, 32, 33, 34, 35]. By reading through literature, it became apparent that one of the leading issues of producing viable image segmentation results that included both soft tissue and bone was that it struggled with pure automatic segmentation algorithms.

This becomes even more of an issue when attempting to segment soft tissue along with bone as one of the major disadvantages of CT imaging is that there is inferior soft tissue contrast when compared to the results produced from a magnetic resonance imaging (MRI) machine [33]. That is also why most image segmentation studies that focus solely on soft tissue deal with images produced from an MRI machine as the contrast is better and allows for better automatic segmentation [34]. As segmentation is the process of dividing an input image into regions with similar properties (i.e., gray level, color, brightness, etc.), analyzing and partitioning the different regions accurately is much more complicated when two or more of the regions of interest have similar characteristics. Thus, to accurately segment a sample so that both bone and soft tissue are clearly defined requires some combination of manual and automatic segmentation.

Manual segmentation can be avoided with the help of artificial intelligence (AI) based segmentation by using machine learning [33, 36]. The way this would function is that a large enough dataset would be generated that consisted of many similar image files. This dataset would provide the basis to train the AI based segmentation algorithm on, where it would be capable of processing all the available data presented. Once the analyzing step had concluded, it would have built a pathway where an input image, like the images found on the initial dataset, would be processed and the output produced would be the fully segmented image. This type of segmentation, while being one of the most advanced segmentation techniques capable of producing some of the best results, was not applicable in this study due to time constraints and unavailability of a larger data set.

1.6 Proposed Design

Modern image segmentation methods mostly focus solely on analyzing bone in the callus, while neglecting the soft tissue area surrounding the injury. Most US images are prone to speckle noise, unusual artifacts and pixel intensity value inconsistencies that come from the specific imaging machine’s settings [28]. The unusual artifacts that tend to appear in these images seems to have pixel values that can be very similar in value to that of the pixel values of soft tissue. That means that most automatic image segmentation processes, will struggle without some manual image processing.

In this study, the focus was on not only the image segmentation of bone tissue, as it has been discussed already in great depth by numerous other authors, but to also perform image segmentation on the soft tissue surrounding a bone fracture and include it in the final output. As the biomechanics behind the healing process of fractures has been studied and analyzed to great depth, the focus was on developing a pipeline that would be able to take the input image data generated by a MicroCT scanner and run it through various software and Python scripts. The algorithm would be capable of determining not only the total surface area of bone, total volume of bone, portions of the bone that has low BMD/ medium BMD/ high BMD, as well as being able to calculate the total surface area of soft tissue and the total volume of the soft tissue present. The image segmentation algorithm utilizes region-based and feature thresholding techniques, in combination with edge detection, pixel intensity detection, and a variety of other image processing algorithms. The intention of this study was to shine a light on an area that might not have so much current focus on it, as well as demonstrating a method that would allow for the image segmentation of soft tissue that could be further expanded upon in the future.

1.7 Research Aims

Objectives

The primary objective was to develop an algorithm that could calculate the total volume and surface area of both the bone and soft tissue regions, as well as determine the BMD distribution of bone. This was achieved through the following specific aims:

Specific Aim 1: Develop algorithm to segment and analyze bone and soft tissue

Specific Aim 2: Develop and validate algorithm to determine BMD of bone

Contributions

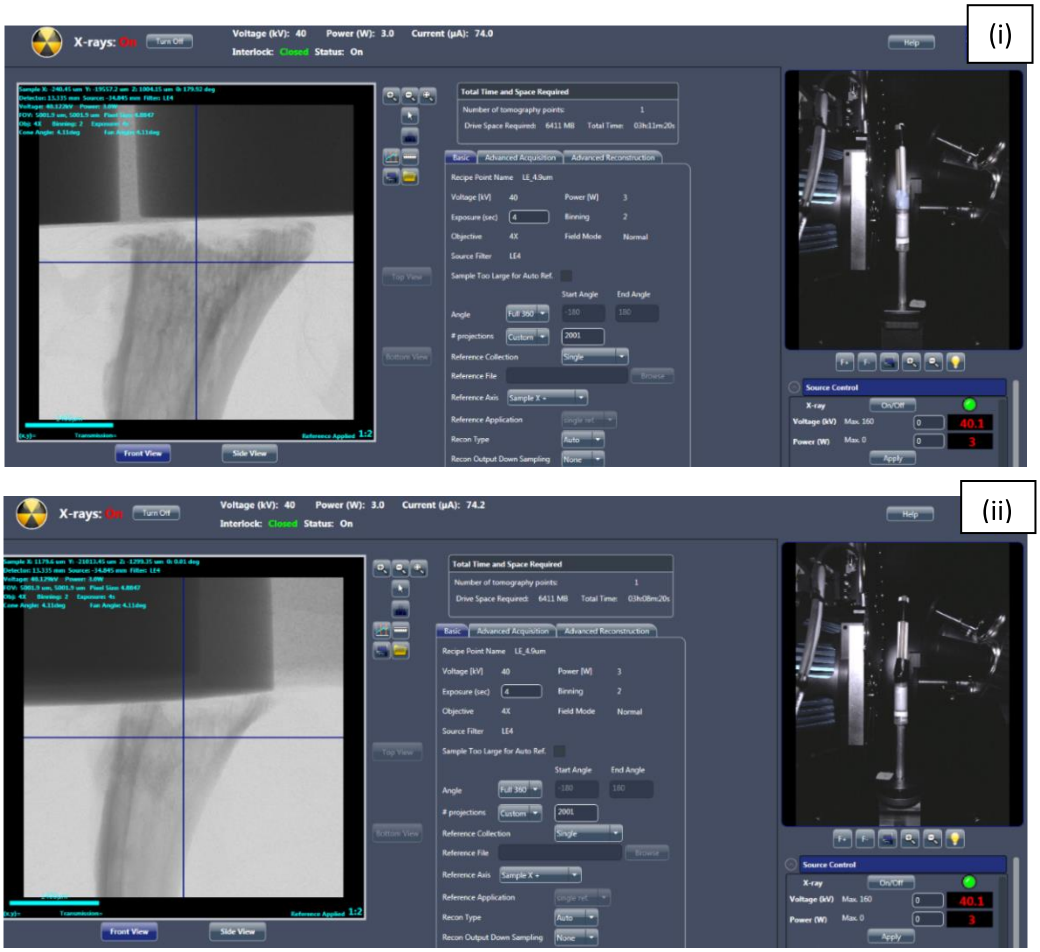
The work done in this study will cause further advancements in science and engineering, as well as having a broader impact in the healthcare field. By developing a method to accurately segment bone and soft tissue samples, future novel healing strategies might be possible to develop with the information provided in this thesis showcase a viable image segmentation strategy.

2. Materials and Methods

2.1 Materials

2.1.1 Input images

Two multi-image TIF files that contained visible bone and soft tissue were chosen to be processed. Both image files were acquired through a MicroCT machine, and the acquisition parameters can be seen in figure 1.



(i)

X-rays: On Turn Off
Voltage (kV): 40 Power (W): 3.0 Current (µA): 74.0
Interlock: Closed Status: On

Sample X: -240.45 um Y: -19557.2 um Z: 1004.15 um θ: 179.92 deg
Detector: 13.335 mm Source: -34.845 mm Filter: LE4
Voltage: 40.122kV Power: 3.0W
FOV: 5001.9 um, 5001.9 um Pixel Size 4.8847
Obj: 4X Binning: 2 Exposure: 4s
Cone Angle: 4.11deg Fan Angle: 4.11deg

Total Time and Space Required
Number of tomography points: 1
Drive Space Required: 6411 MB Total Time: 03h11m20s

Basic | Advanced Acquisition | Advanced Reconstruction

Recipe Point Name: LE_4.9um
Voltage [kV]: 40 Power [W]: 3
Exposure (sec): 4 Binning: 2
Objective: 4X Field Mode: Normal
Source Filter: LE4
Sample Too Large for Auto Ref. □
Angle: Full 360 Start Angle: -180 End Angle: 180
# projections: Custom 2001
Reference Collection: Single
Reference File:
Reference Axis: Sample X +
Reference Application: single ref.
Recon Type: Auto
Recon Output Down Sampling: None

Source Control
X-ray On/Off
Voltage (kV) Max. 160 0 40.1
Power (W) Max. 0 0 3

(ii)

X-rays: On Turn Off
Voltage (kV): 40 Power (W): 3.0 Current (µA): 74.2
Interlock: Closed Status: On

Sample X: 1179.6 um Y: -21013.45 um Z: -1299.35 um θ: 0.01 deg
Detector: 13.335 mm Source: -34.845 mm Filter: LE4
Voltage: 40.129kV Power: 3.0W
FOV: 5001.9 um, 5001.9 um Pixel Size 4.8847
Obj: 4X Binning: 2 Exposure: 4s
Cone Angle: 4.11deg Fan Angle: 4.11deg

Total Time and Space Required
Number of tomography points: 1
Drive Space Required: 6411 MB Total Time: 03h08m20s

Basic | Advanced Acquisition | Advanced Reconstruction

Recipe Point Name: LE_4.9um
Voltage [kV]: 40 Power [W]: 3
Exposure (sec): 4 Binning: 2
Objective: 4X Field Mode: Normal
Source Filter: LE4
Sample Too Large for Auto Ref. □
Angle: Full 360 Start Angle: -180 End Angle: 180
# projections: Custom 2001
Reference Collection: Single
Reference File:
Reference Axis: Sample X +
Reference Application: single ref.
Recon Type: Auto
Recon Output Down Sampling: None

Source Control
X-ray On/Off
Voltage (kV) Max. 160 0 40.1
Power (W) Max. 0 0 3

Figure 1: MicroCT bone and soft tissue sample parameters; image_1 (i) and image_2 (ii)

The first file, also referred to as image_1, had a total of 1002 images and the second file, also referred to as image_2, had a total of 1008 images. As the bone and soft tissue samples were scanned along with the HA phantoms being captured in the same TIF file, this ended up resulting in roughly 2/3 of all thousand image slices containing mostly the x-ray scans of the bone and soft tissue sample. The remaining 1/3 image slices from the original image stack contained the HA phantoms that would be used to determine BMD.

For the image slices within image_1, each image slice was composed of a 1000x1024 pixel array and for the image slices within image_2, each image slice was composed of a 984x1024 pixel array. Both image files were initially converted into grayscale 8-bit files to speed up the processing speed of the Python script developed would take too long to interpret and analyzed the input files. As 8-bit files, the pixel values within the images were scaled down where the fully black pixels found in an image slices would be interpreted as having a value of zero, while fully white pixels would be interpreted as having a value of 255. The remaining various different levels of gray pixels would each have their own unique value somewhere within the range of 1 – 254, where the lower value pixels are darker and the higher valued pixels would be brighter.

2.1.2 Hydroxyapatite Phantom

Initially, it was decided that the original image files provided would be used to develop the Python script. Once the script had been completed, then new bone and soft tissue samples would be analyzed along with a new phantom that contained three distinct HA levels. However upon initial examination of the HA phantom provided, it became apparent that it was different than the one used to develop the algorithm. As can be seen by comparing figure 2 (iii) and figure 2 (iv), it was clear that the three different HA regions that would be used to determine BMD from the HA phantom provided were much farther apart than the HA phantoms used to develop the image segmentation algorithm. The location of the cylindrical HA-containing rods of the new phantom was positioned in such a manner that by acquiring an image with 4x magnification, there was no manner to acquire all three regions in the same set of image slices. As the recipe provided that was used in the acquisition of the initial image files from which the image segmentation algorithm was developed from used 4x magnification, it was important to keep the image acquisition parameters as similar as possible.

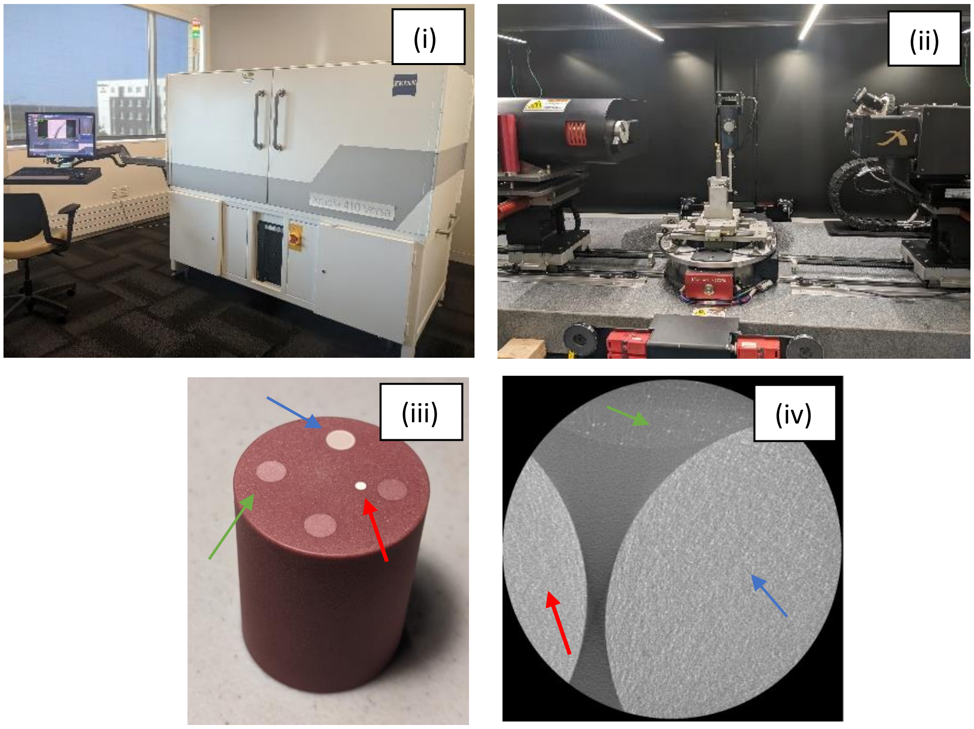


Figure 2: UWM MicroCT machine outside (i), inside (ii); HA phantoms provided (iii), HA phantoms found on the provided image files using 4x magnification (iv)

However, not only was there an issue with the location of the cylindrical HA-containing rods on the new HA phantom, but there was also the issue that there was no apparatus as can be seen on the right side of figure 1 that could be used to hold the HA phantom, as well as the bone and soft tissue sample in such a manner that would allow for the image acquisition to capture both the phantom and the sample. Attempts were made to recreate the HA phantom plus bone and soft tissue sample holder in such a way that would still allow for the best scanned images to be produced and figure 3 demonstrates the final holder developed.

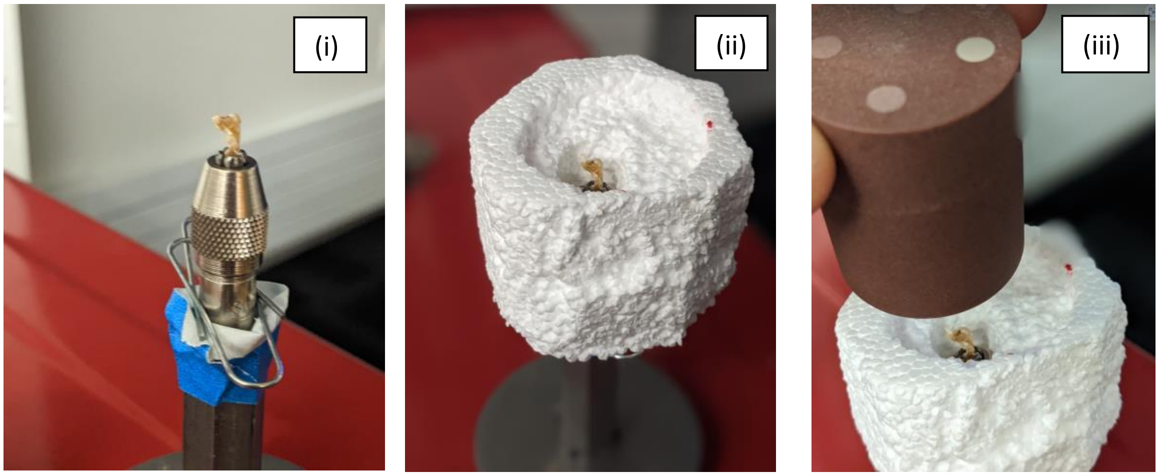


Figure 3: Only bone and soft tissue sample (i), Styrofoam was placed on a paperclip (ii), HA phantom was placed on Styrofoam to be directly above sample (iii)

While the HA phantom plus sample holder was able to produce results that were much better than scans produced of only the HA phantom or the sample, the different HA regions were still simply too far apart, and three different scans had to be taken with different HA regions being rotated above the bone and soft tissue sample. The image files produced in this manner still had discrepancies between the overall brightness and as such, were not included.

Before realizing the importance of producing scans that would also include the HA phantom along with the bone and soft tissue sample in the same set of image slices, a couple scans were produced simply containing the bone and soft tissue sample. While it was eventually determined that the only way to properly test if the algorithm functioned as intended on a new set of scanned image slices where the three different HA regions were all within the same image file, some of the first scanned images only contained the bone and soft tissue sample. These scanned images were captured using the same MicroCT recipe used when producing the image files from which the image segmentation algorithm was developed. From the individually scanned bone and soft tissue sample, the data provided by them was used in determining the pixel-to-length conversion rate.

The HA phantoms were provided by Pure Imaging Phantoms, a 4 centimeter (cm) long cylindrical phantom with a radius of 1.5 cm was used during the imaging process. It had the capacity to contain up to five different densities of CaHA densities within cylindrical inserts [37]. For this thesis, 3 of the cylindrical tubes were filled with the density values of 50 mgHA/cm^3, 200 mgHA/cm^3, and 1200 mgHA/cm^3. Unfortunately, no viable images were able to be produced of the HA phantoms.

2.1.3 Mouse Samples

C57Bl/6 mice purchased from Charles River were used in this study. All procedures were in accordance with the International Animal Care and Use Committee (IACUC) guidelines. Mice were first administered with buprenorphine. They were then placed under isoflurane and a three-point bending injury or burr-hole injury was performed. Mice were checked for fracture and placed back in their cages. They were allowed to ambulate freely. In the case of three-point bending injury, they were administered one more dose of buprenorphine 8 hours post injury and 18 hours post injury. For burr-holes, mice were sacrificed at 7 days post injury. For three-point bending, mice were sacrificed two weeks post injury and the tibiae were dissected. The tibiae were fixed in NBF and then immersed in 75% ethanol. Although the images produced from three-point bending injury mouse samples were excluded, they did provide the valuable pixel-to-length conversion that was used in the last stages of the code.

2.2 Software

2.2.1 Mimics

Mimics was designed to import 2D medical image data and segment the anatomy to generate accurate 3D models. While most of the segmentation for the bone was done via a combination of different Python scripts and with ImageJ, the soft tissue segmentation required a part of it to be done with the assistance of Mimics. Since the manner in which soft tissue regions end up getting processed results in the pixel value of the soft tissue being similar with the pixel values of the background and/or noise, it can cause auto-segmentation methods to produce non-accurate 3D models of the segmented soft tissue. As seen in figure 4, some of the random noise in the background does appear to be similar in pixel value to that of the soft tissue region.

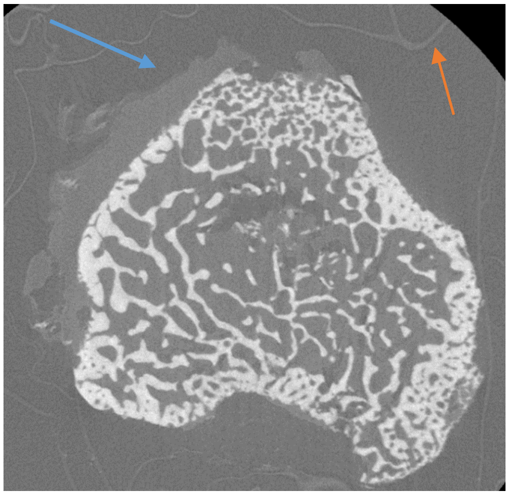


Figure 4: Sample input file, the blue arrow shows the soft tissue region and the orange arrow shows random noise

The Mimics software allows for the combination of auto segmentation and manual segmentation. First, the software allows the user to define a threshold to separate the soft tissue, along with some artifacts/noise unfortunately. Then, the software allows for the manual segmentation of multiple slices at a time. This combination of auto segmentation and manual segmentation allows for creating new input data where the soft tissue has been segmented as accurately as possible. As seen in figure 5, the automatic segmentation was able to remove a good amount of the background noise, however manual segmentation was still required as a lot of pixels that were not actually soft tissue got included during the automatic segmentation.

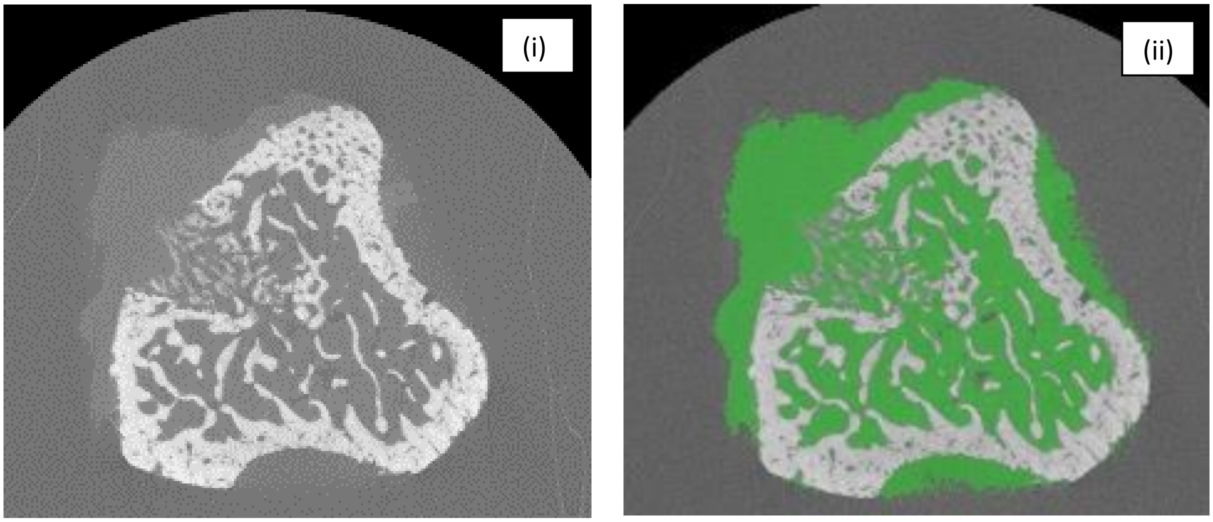


Figure 5: Sample input image (i) and the soft tissue outcome post automatic segmentation (ii)

2.2.2 ImageJ

ImageJ allows users to do basic image processing and manipulation. Drawbacks of ImageJ are that it lacks in selecting specific objects within the image due to the researcher having to manually choose polygon, oval, rectangular or freehand sections [32]. This in turn can affect the accuracy of object selection, thus causing the segmentation process to not be as effective when only using ImageJ.

While ImageJ was not used so much for segmentation, it still proved useful for a variety of its morphological operators and noise removal. The morphological operators that were called upon were *threshold*, *close*, *median*, *dilate*, *erode*, and *fill holes* [38]. *Threshold* was used to convert the input grayscale image into a binary image by setting a lower and an upper limit to remove any pixels not within that specified ranged. *Median* was used to reduce noise in the image by replacing each pixel with the median of the neighboring pixel values, defined by a pre-specified radius. *Erode* removes pixels from the edge of objects in binary images, while *dilate* adds pixels to the edges of objects. *Close* performs a *dilation* operation, followed by an *erosion* operation. This smooths the image object, as well as removing any isolated pixels.

The noise removal functions that were applied and used through this software where *despeckle* and *remove outliers* [38]. *Despeckle* calls upon a median filter to pass over the entire image and replaces each pixel with the median value in its 3x3 neighborhood. This type of filter was good at removing some of the salt and pepper noise. *Remove outliers* replaced a pixel by the median of the pixels in the surrounding if it deviates from the median by more than a certain value. It required an input radius (r) that was used to determine the area used to determine the median and was used to help remove any pixels that the *despeckle* function failed to get.

2.2.3 Python

Version 3.7.6 of Python through the Spyder IDE and Anaconda Navigator was used as it is a language that is widely used to access raw data produced by X-ray detectors [39, 40]. It was used to calculate the surface area and volume of both the bone and soft tissue samples analyzed, the statistical analysis of the different HA regions within the phantoms, and to use the values determined from the statistical analysis to segment the varying BMD regions of the bone sample accordingly.

Calculating the total volume of both the bone and the soft tissue was done in the same manner. The main function that was used during this step was the *measure.regionprops* function from the scikit-image ecosystem within Python [41]. The input image files were batch processed using the *glob* module and the data was stored into a *pandas* data frame [42, 43]. Regionprops functions as a form of Boolean algebra, which simply works by setting X to be an arbitrary non-empty set and let $\mathcal{P}$(X) (the power set of X) be the class of all subsets of X [44]. The way of introducing a Boolean structure into $\mathcal{P}$(X) is as follows, where the distinguished elements are defined by:

$$0 = \textit{false value} \qquad\qquad 1 = \textit{true value,}$$

And, if P and Q are subsets of X, then,

$$P + Q = (P \cap Q') \cup (P' \cap Q) \qquad \textit{and} \qquad PQ = P \cap Q$$

Where the ⋃, ⋂, and ʻ imply union, intersection, and complement. The true values were defined as pixel having a value of 255 and the false values were any of the remaining pixels that did not have a value equal to 255. This allowed the *measure.regionprops* function to count all the true values and return the number of pixels that were either soft tissue or bone, respectively. This was done for every individual image slice, with the total area (i.e., number of true values) being stored into a data frame and once the area had been calculated for each individual image, the values were summed up to determine the total sum of the surface areas of either the bone or soft tissue regions. The total sum of surface areas was then later used to determine the total volume of either the bone or soft tissue regions using the following equation:

$$Volume\ (\mu m^3) = \sum Surface\ Area\ (pixel^2) * \frac{Area\ (\mu m^2)}{pixel^2} * [\#\ image\ slices * pixel\ length\ (\mu m)]$$

Calculating the total surface area for either the bone or soft tissue was done in a similar fashion, but before running the image slices through the *measure.regionprops* function the input images were first processed through a bilateral filter. Through the means of a nonlinear combination of nearby image pixel values, bilateral filtering smooths an image while also preserving its edges [45]. This filter is simply a weighted average of the nearby local samples, where the weights are determined by the radiometric and temporal distances between the central pixel and the surrounding pixels [46, 47].

After being processed through the bilateral filter, the processed image files were run through a canny edge detection algorithm using OpenCV. This algorithm operates in the following order:

1.) smooth the image using a Gaussian filter,

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{\frac{-x^2+y^2}{2\sigma^2}}$$

2.) compute the gradient magnitude and orientation utilizing finite-difference approximations for the partial derivatives,

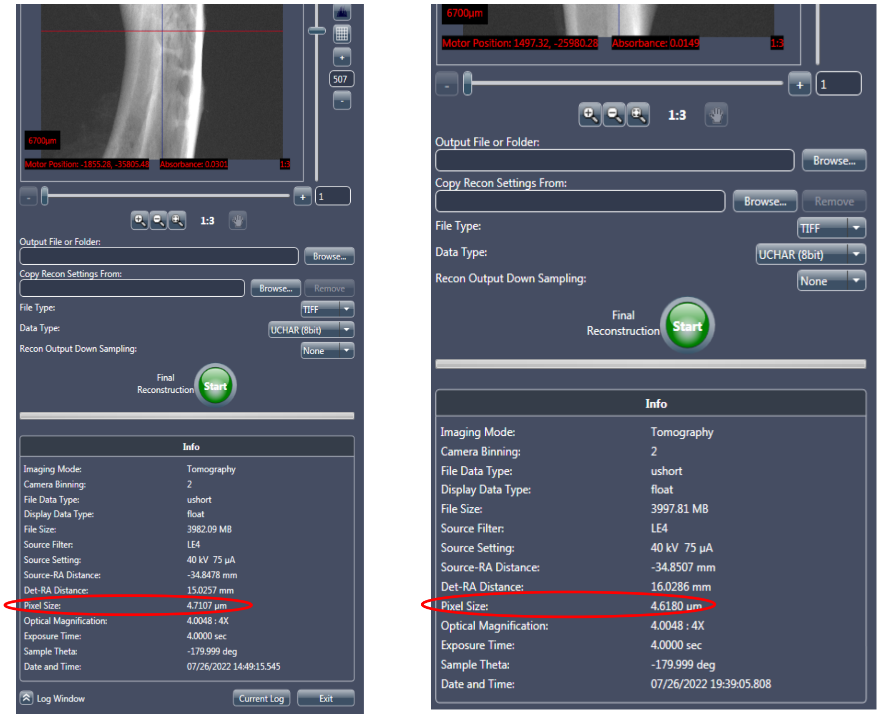
$$G = \sqrt{G_x^2 + G_y^2} \qquad \theta = \arctan\left(\frac{G_y}{G_x}\right)$$

3.) apply nonmaximal suppression to the gradient magnitude, plus use the double thresholding algorithm to detect and link edges [48, 49]. The pixel is designated as an edge pixel if the amplitude of the pixel position exceeds the high threshold. The pixel is excluded if the position's amplitude is below the high threshold. If the pixel has an amplitude between the two thresholds, the pixel is saved only when directly next to a pixel with a value higher than the high threshold.

Once the canny edge detection algorithm has generated new image slices where only the perimeter pixels of either the bone or soft tissue are left with a value of 255, they were processed through the *measure.regionprops* to determine the number of pixels that had a true value. After the number of pixels were counted that would be defined as the perimeter of the individual images, all the values were stored in a data frame before being added up to determine the total sum of the perimeter for either the bone or soft tissue regions. The total sum of perimeter was then later used to determine the total surface area of either the bone or soft tissue regions using the following equation:

$$Surface\ Area\ (\mu m^2) = \sum Perimeter\ (pixel) * \frac{length\ (\mu m)}{pixel} * [\#\ image\ slices * pixel\ length\ (\mu m)]$$

The pixel-to-length conversion was provided by the scanning output produced from two good scans acquired using the same recipe that generated the image files from which the algorithm was developed. The two good scans only contained the bone and soft tissue sample, thus were eventually excluded as processing new scans through the algorithm required a different HA phantom. As the original files did not contain the pixel to length conversion rate, the values determined from the good scans were used instead as the same recipe was used (fig. 6).



Output File or Folder:

Copy Recon Settings From:

File Type: TIFF

Data Type: UCHAR (8bit)

Recon Output Down Sampling: None

Info

Imaging Mode: Tomography
Camera Binning: 2
File Data Type: ushort
Display Data Type: float
File Size: 3982.09 MB
Source Filter: LE4
Source Setting: 40 kV 75 µA
Source-RA Distance: -34.8478 mm
Det-RA Distance: 15.0257 mm
Pixel Size: 4.7107 µm
Optical Magnification: 4.0048 : 4X
Exposure Time: 4.0000 sec
Sample Theta: -179.999 deg
Date and Time: 07/26/2022 14:49:15.545

Output File or Folder:

Copy Recon Settings From:

File Type: TIFF

Data Type: UCHAR (8bit)

Recon Output Down Sampling: None

Info

Imaging Mode: Tomography
Camera Binning: 2
File Data Type: ushort
Display Data Type: float
File Size: 3997.81 MB
Source Filter: LE4
Source Setting: 40 kV 75 µA
Source-RA Distance: -34.8507 mm
Det-RA Distance: 16.0286 mm
Pixel Size: 4.6180 µm
Optical Magnification: 4.0048 : 4X
Exposure Time: 4.0000 sec
Sample Theta: -179.999 deg
Date and Time: 07/26/2022 19:39:05.808

Figure 6: Pixel-to-length sample conversion rate

Following the conversion from pixel^3 and pixel^2 into cubic micronmeters and micronmeters squared for the volume and surface area, the units were converted into cubic centimeter and centimeter squared. This was done to match the values provided from the manufacture for the HA phantom and the conversion rate is as follows:

$$1\ \mu m^3 = 1.0 * 10^{-12}\ cm^3 \qquad 1\mu m^2 = 1.0 *\ 10^{-8}\ cm^2$$

The statistical analysis of the phantom was done by analyzing the distribution of the pixel intensity of the three BMD regions. From this analysis, the thresholding limits for the three unique regions were determined and used to segment the bone accordingly. The percent BMD of the overall bone was calculated by dividing the amount per region by the total bone volume determined previously and is as follows:

$$\%\ BMD = \frac{volume\ of\ one\ BMD\ region}{total\ volume\ of\ bone}$$

2.3 Block Diagram

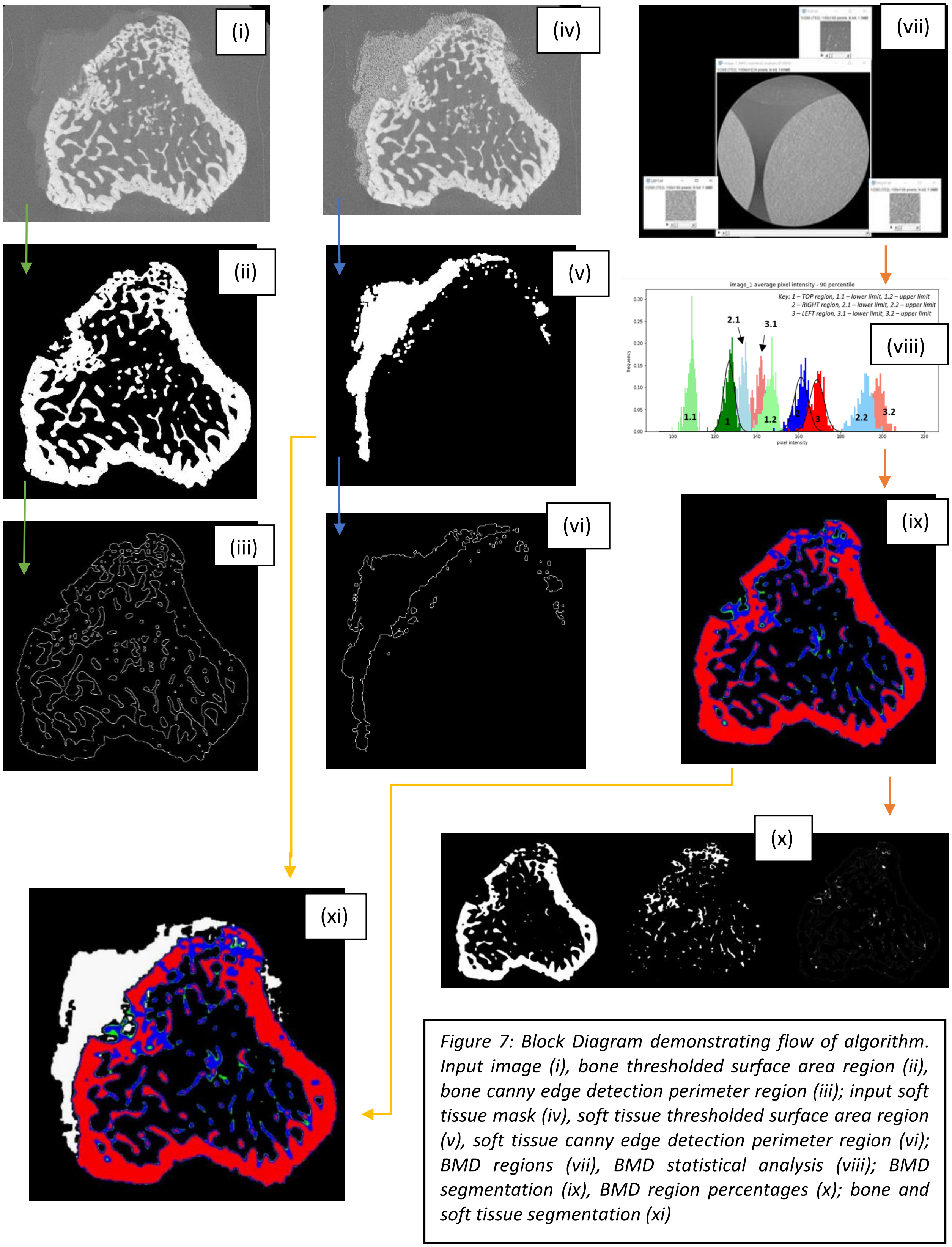

(i)
(iv)
(vii)
(ii)
(v)
image_1 average pixel intensity - 90 percentile
Key: 1 – TOP region, 1.1 – lower limit, 1.2 – upper limit
2 – RIGHT region, 2.1 – lower limit, 2.2 – upper limit
3 – LEFT region, 3.1 – lower limit, 3.2 – upper limit
2.1
3.1
1.1
1
1.2
2
3
2.2
3.2
frequency
0.00
0.05
0.10
0.15
0.20
0.25
0.30
100
120
140
160
180
200
220
pixel intensity
(viii)
(iii)
(vi)
(ix)
(x)
(xi)

Figure 7: Block Diagram demonstrating flow of algorithm. Input image (i), bone thresholded surface area region (ii), bone canny edge detection perimeter region (iii); input soft tissue mask (iv), soft tissue thresholded surface area region (v), soft tissue canny edge detection perimeter region (vi); BMD regions (vii), BMD statistical analysis (viii); BMD segmentation (ix), BMD region percentages (x); bone and soft tissue segmentation (xi)

3 Results

3.1 Range for Image Segmentation

When image segmentation is used to determine the bone characteristics, such as volume or BMD, selecting a certain range of image slices to segment does not necessarily impact the accuracy of the results. However, when analyzing the soft tissue segments still attached to the bone sample, the pixel intensity of this material does have a similarity in pixel value to that of the background, noise and/or attenuation pixels that also appear within the image. Thus a specific range of image slices must be specified to get the most accurate results.

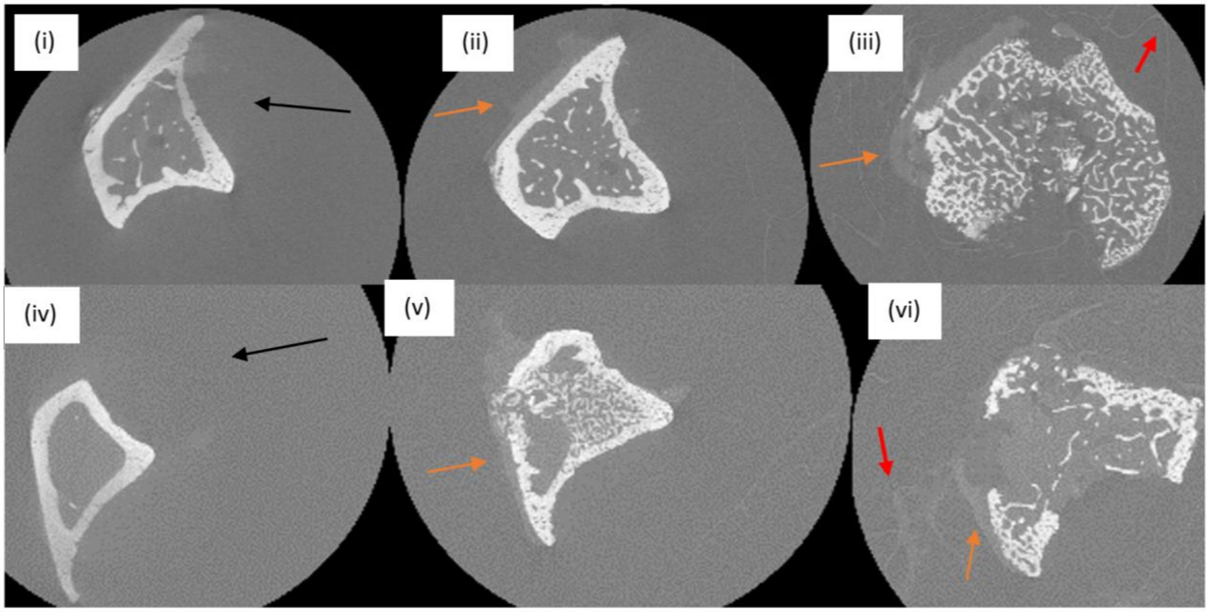


Figure 8: image_1 – image slice number 0000 (i), 0187 (ii), 0550 (iii)
image_2 – image slice number 0000 (iv), 0361 (v), 0568 (vi)

The first 50 image slices from both files were not selected to run through the Python script, as the overall image was darker (fig. 6: i, iv). When the intensity of pixels is used to determine what a certain region is for multiple image slices, having the overall brightness be consistent results in producing better output image files.

The files, image_1 and image_2, were acquired with a HA phantom placed next to it during the image acquisition process. The advantage to acquiring an image in this manner is that all the image slices that are produced have an overall similar texture. The disadvantage of this technique is that the images can produce excess noise during the transition phase between the sample and the HA phantoms. As shown in figure 6 (ii, iii, v, vi), the orange arrows point to regions where soft tissue is still attached to the bone. The red arrows point to the noise that starts to appear in the images during that transition. The final selection for the input image slices of both files was 0050 – 00499, for a total of 450 image slices.

3.2 Calculating Volume and Surface Area

The core concept behind how this part of the script functions is that by utilizing the *skimage.measure* library on Python, the *regionprops* function can be used to process a binary image (i.e., black pixels having a value of 0 and white pixels having a value of 255) to determine the total number of white pixels found on that image. By being able to take the summation of either all the surface areas or all the perimeter values from the different image slices, the total volume or surface area of an object can be determined if one knows the pixel-to-length conversion.

3.2.1 Soft tissue

The process of thresholding the input image slices to fit the requirements so that the soft tissue regions could be processed and analyzed required multiple steps. Both files were initially processed through the Mimics software, generating a mask that highlighted the soft tissue regions. Setting the threshold limits to so a small, specific range produced the best results. Once the mask was generated on Mimics, the file was exported and processed through ImageJ where an appropriate threshold was found where the mask could be properly thresholded. Following that there was a serious of image manipulation steps that occurred to better calculate the total volume and total surface area. The sequence of events is described in table 1.

	Mimics	**ImageJ**
image_1	*Threshold: lower = 130 and upper = 140.* *Removed extra noise manually.*	*Threshold: L=183, U=184 → remove outliers (r=1)* *. → close → median (r=2) → dilate → fill holes*
image_2	*Threshold: lower = 139 and upper =148.* *Removed extra noise manually.*	*Threshold: L=183, U=184 → remove outliers (r=2)* *. → close → median (r=2) → dilate → fill holes*

Table 1: Soft tissue image processing steps

Once the segmentation masks for the soft tissue had been generated, the binary files were put through a Python script that utilized the *regionprops* function. This was how the sum of all of the image slice's surface area was calculated, which was later used to determine the total volume of the soft tissue.

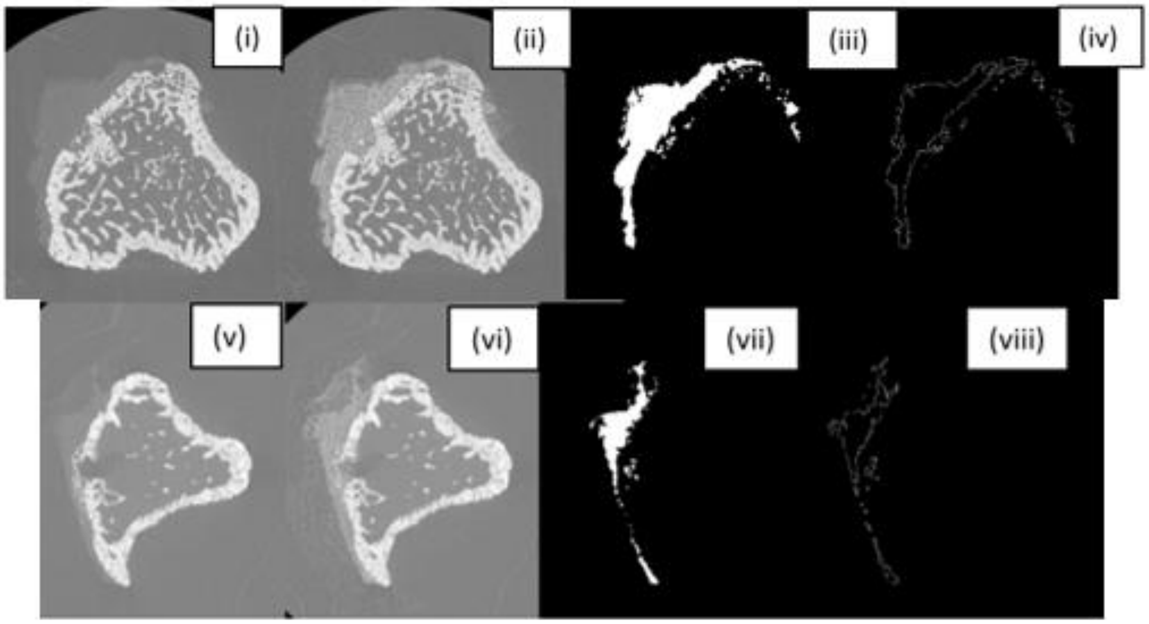


Figure 9: image_1 input (i), soft tissue mask (ii), threshold (iii), canny edge detection (iv) image_2 input (v), soft tissue mask (vi), threshold (vii), canny edge detection (viii)

To determine the total surface area of the soft tissue, the binary images were first put through a canny edge detection utilizing cv2 library. Once the only pixels remaining were the edge ones of the soft tissue, the *regionprops* function was able to determine the perimeter of each individual image slice. Then all the individual image slice's perimeter data was added up to calculate the total sum of the perimeter, which was then used to calculate the total surface area of the soft tissue in pixels. Figure 9 illustrates the outputs from the different steps of this process and table 2 shows the results.

	Σ Surface Area (*pixel*^{2})	**Σ Perimeter (*pixel*)**
image_1	*11,398,194.00*	*3,823,203.00*
image_2	*5,565,704.00*	*1,849,178.00*

Table 2: Calculated sum of soft tissue surface area and sum of perimeter data

3.2.2 Bone

Determining the total volume and the total surface area of the bone was done in a similar fashion as to the way soft tissue was calculated, just without the use of the Mimics software. As the pixel values of bone are much different than all the other pixels in the image, the thresholding was possible to accomplish effectively on ImageJ. Once thresholded, the image slices were processed through noise removal functions to remove any outliers. The process can be found in table 3.

	Threshold Range	**Noise Removal**
image_1	*155 - 255*	*despeckle → remove outliers (r=2)*
image_2	*160 - 255*	*despeckle → remove outliers (r=2)*

Table 3: Bone threshold range and noise removal process

Once the binary image slices had been developed, they were processed on Python to determine total bone volume. The edge detection script was used again to determine the perimeter values of each image slice before being used to determine the total surface area of the bone samples. Table 4 illustrates the calculated sum of surface area and sum of perimeter results, while figure 10 shows the output images during the various steps.

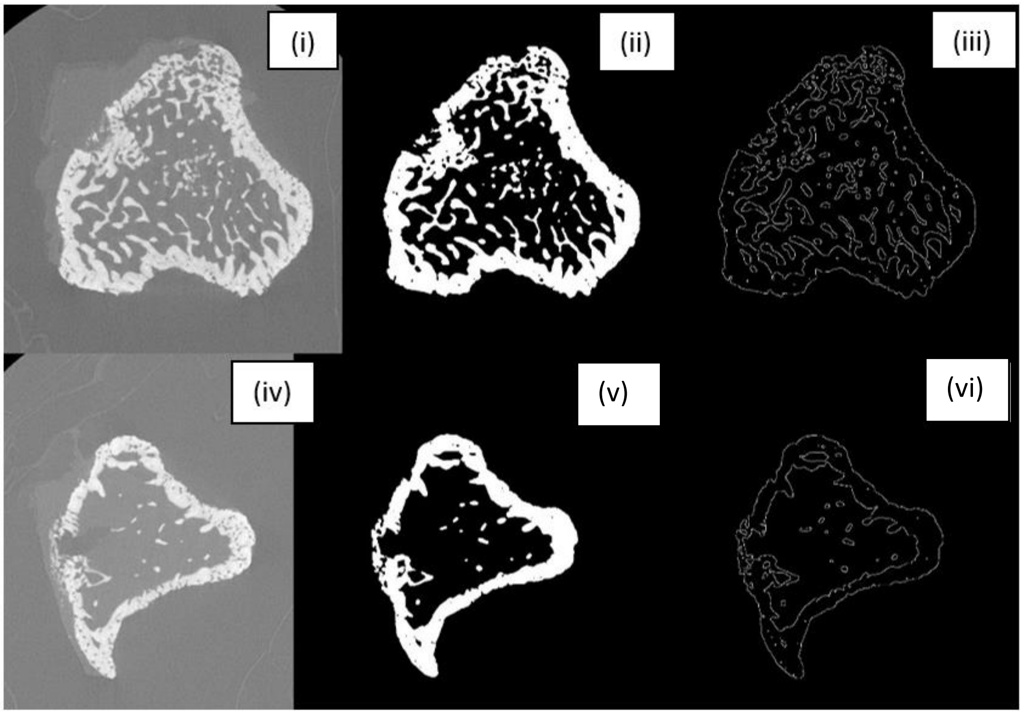


Figure 10: image_1 input (i), threshold (ii), canny edge detection (iii) image_2 input (iv), threshold (v), canny edge detection (vi)

	Σ Surface Area (*pixel*^{2})	**Σ Perimeter (*pixel*)**
image_1	*35,911,474.00*	*4,919,876.00*
image_2	*18,350,921.00*	*2,183,359.00*

Table 4: Calculated sum of bone surface area and perimeter data

3.3 Calculating Bone Mineral Density

The process of determining the overall percent BMD composition of the bone samples was done by analyzing the pixel intensity and distribution of the HA phantoms located towards the end of the image stacks. As this step required the input image slices to be as clear and free of random noise, a selection of 200 images sequential images containing the three regions of interest was selected. From that subset of images, three 100x100 pixel blocks were cropped from the three different regions that contained the different levels of HA (fig. 11). This provided a total of two million different sample pixels to perform a statistical analysis and in doing so, would produce a viable thresholding limits, shown as follows:

$$(200\ input\ images\ x\ [100x100\ pixels]) = 2{,}000{,}000\ pixels$$

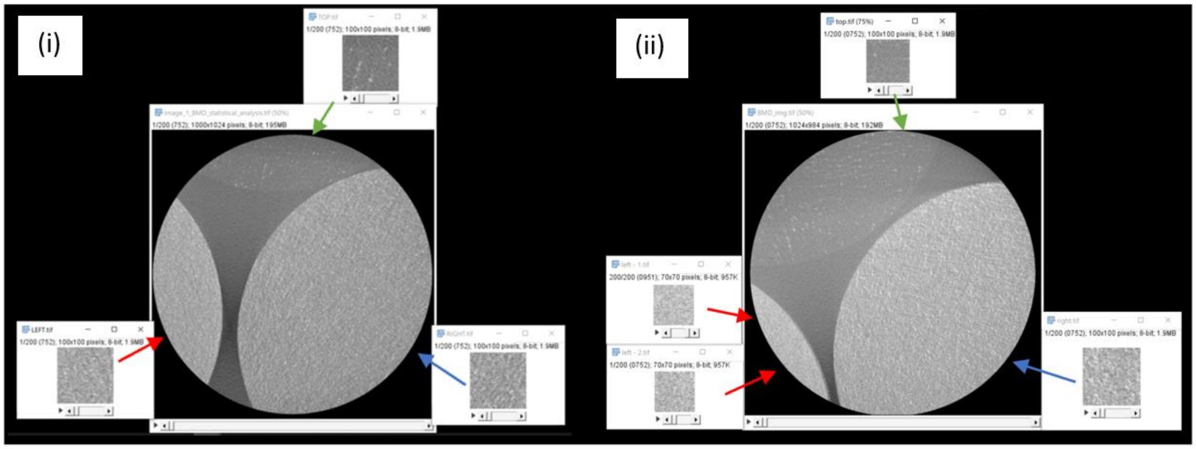

(i)
TOP.tif
1/200 (752); 100x100 pixels; 8-bit; 1.9MB
image_1_BMD_statistical_analysis.tif (50%)
1/200 (752); 1000x1024 pixels; 8-bit; 195MB
LEFT.tif
1/200 (752); 100x100 pixels; 8-bit; 1.9MB
RIGHT.tif
1/200 (752); 100x100 pixels; 8-bit; 1.9MB
(ii)
top.tif (75%)
1/200 (0752); 100x100 pixels; 8-bit; 1.9MB
BMD_img.tif (50%)
1/200 (0752); 1024x984 pixels; 8-bit; 192MB
left - 1.tif
200/200 (0951); 70x70 pixels; 8-bit; 957K
left - 2.tif
1/200 (0752); 70x70 pixels; 8-bit; 957K
right.tif
1/200 (0752); 100x100 pixels; 8-bit; 1.9MB

Figure 11: image_1 (i) and image_2 (ii) different HA regions, where the green arrow is for the low BMD region, the blue arrow is for the medium BMD region and the red arrow is for the high BMD region

For image_2, partitioning a 100x100 pixel region from the bottom left side was impossible due to the initial image file that was captured resulted in that area being extremely narrow. Thus, two 70x70 pixel regions were taken from the that specific region to keep the number of pixels being analyzed from the different regions the same. As the script processed 200 HA input image regions, two 70x70 pixel regions could generate enough sample pixels to analyze that would be in similar in size to the 100x100 pixel blocks also being analyzed, shown as follows:

$$(200\ input\ images\ x\ [70x70\ pixels]) + (200\ input\ images\ x\ [70x70\ pixels]) = 1{,}960{,}000\ pixels$$

The three different image stacks regions were processed with a Python script that could determine the average pixel value, the average lower quartile and the average upper quartile (50- and 90- percentile range), as well as the correlating standard deviation for each individual BMD regions (table 5).

			50 Percentile		**90 Percentile**		
		Average	**Lower (L = 25)**	**Upper (U = 75)**	**Lower (L = 5)**	**Upper (U = 95)**	**Standard deviation**
image_1	TOP	126.73	118.82	134.26	108.29	146.07	11.80
	RIGHT	161.03	148.68	172.85	132.81	190.94	18.01
	LEFT	168.67	156.97	180.12	141.32	196.77	17.20
image_2	TOP	142.32	134.80	149.49	124.92	160.74	11.21
	RIGHT	176.97	164.60	188.75	149.05	206.93	18.05
	LEFT	188.54	177.87	199.15	163.44	213.93	15.77

Table 5: Statistical analysis of the three different BMD regions

The results from the statistical analysis were graphed and a gaussian curve was applied to the average pixel values to better determine the appropriate thresholding limits for the different BMD regions (fig. 12).

After the statistical analysis was completed, the original input images were processed with Python to determine the percent amount from each of the three regions. Taking into consideration the initial thresholding values used to calculate the total volume of bone and the values produced through the statistical analysis of the three different BMD regions, the final limits for thresholding were decided upon (table 6). The discussion section will elaborate on how the final limits were decided upon.

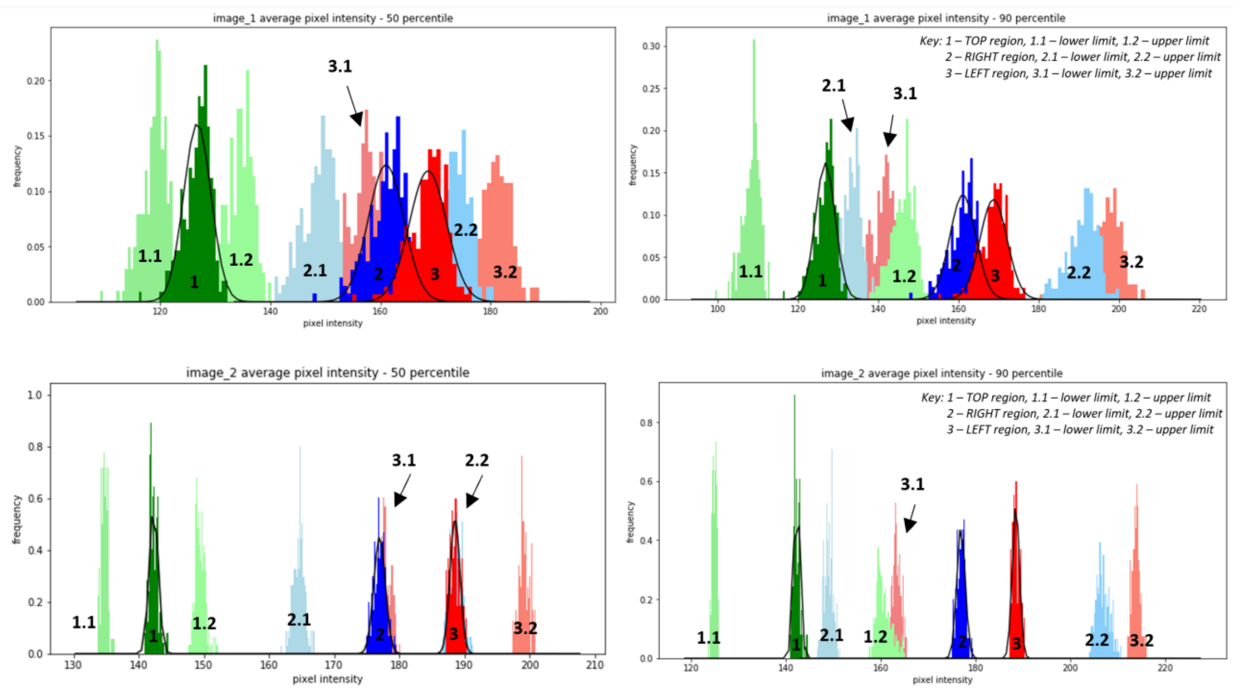


Figure 12: The 50 and 90 percentile distributions of the average pixel intensity per image slice

	Thresholding Limits for the Different HA Regions
image_1	*Nonbone < 155 < TOP < 160 < RIGHT < 190 < LEFT*
Image_2	*Nonbone < 160 < TOP < 165 < RIGHT < 190 < LEFT*

Table 6: BMD segmentation thresholding limits

Once the thresholding limits had been determined, the original input files were processed through Python. The script first assigned each of the pixels from the respective regions a certain color value (green for the low BMD region, blue for the medium BMD region, and red for the high BMD region) as shown in figure 13. All the newly created RBG image slices were then split by their respective colors using the cv2.split function. Once all the appropriate pixels from the three different regions had been separated into their own image, the function regionprops was applied to each image slice and the different regions had all the pixels counted to calculate their respective surface area. After every image slice had been processed, the volume was determined through the summation of all the surface area values (table 7).

		Σ Surface Area (pixel^2)	**Percent of Bone**	**Percent Error**
image_1	TOP	*1,766,996.00*	*4.92%*	
	RIGHT	*10,996,927.00*	*30.62%*	
	LEFT	*26,189,378.00*	*72.93%*	
	Total Sum	*38,955,301.00*	*108.47%*	*8.47%*
image_2	TOP	*617,214.00*	*3.36%*	
	RIGHT	*6,980,672.00*	*38.03%*	
	LEFT	*11,040,397.00*	*60.16%*	
	Total Sum	*18,638,283.00*	*101.57%*	*1.57%*

Table 7: Bone mineral density segmentation results

The percent error was calculated by taking the sum of the three different regions volumes and comparing it with the value calculated for total bone volume initially as follows:

$$\%\ error = \frac{Total\ Sum\ (pixels^3) - Bone\ Volume\ (pixels^3)}{Bone\ Volume\ (pixels^3)} * 100$$

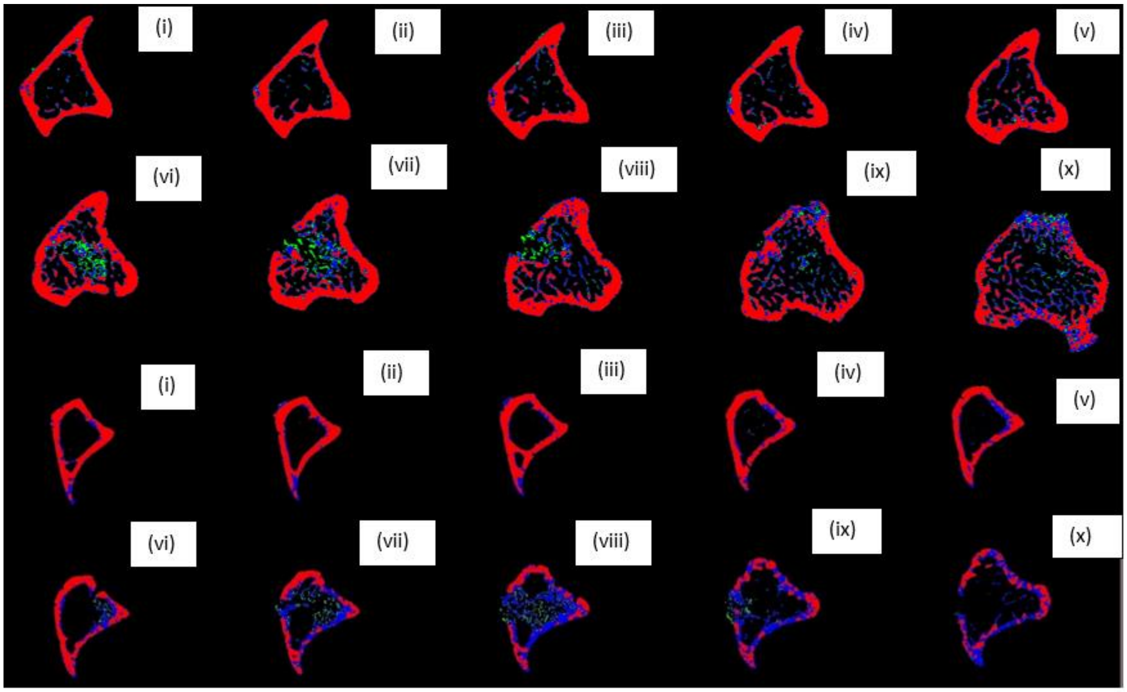


Figure 13: Bone mineral density segmentation output for image_1 (top) and image_2 (bottom), respectively:
Image slice - 50 (i), 99 (ii), 149 (iii), 199 (iv), 249 (v), 299 (vi), 349 (vii), 399 (viii), 449 (ix), 499 (x)

The last part from the analysis section was to convert the values from pixel form into unit length form. Table 8 shows the results from this section.

	Total Soft Tissue Volume	Total Soft Tissue Surface Area	Total Bone Volume	Total Bone Surface Area	Low BMD (green) Volume	Medium BMD (blue) Volume	High BMD (red) Volume
image_1	0.520 cm^3	373.42 cm^2	1.638 cm^3	480.53 cm^2	0.081 cm^3	0.502 cm^3	1.194 cm^3
image_2	0.254 cm^3	180.62 cm^2	0.837 cm^3	213.25 cm^2	0.028 cm^3	0.318 cm^3	0.504 cm^3

Table 8: Total volume and total surface area calculations

3.4 Bone and Soft Tissue Segmentation

The final step took the individually segmented soft tissue segments and combined them with the segmented BMD images to produce the final output result (fig. 14 and fig. 15).

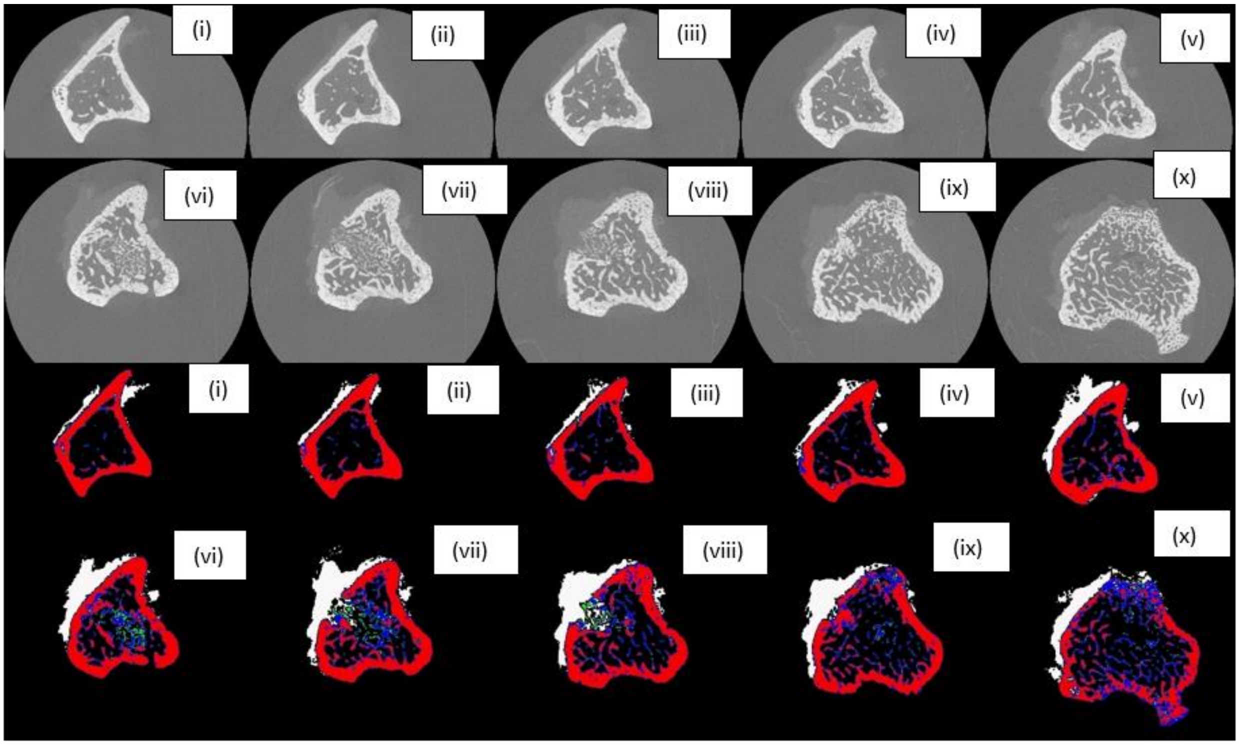


Figure 14: Bone and soft tissue segmentation for image_1, input (top) and output (bottom): Image slices 50 (i), 99 (ii), 149 (iii), 199 (iv), 249 (v), 299 (vi), 349 (vii), 399 (viii), 449 (ix), 499 (x)

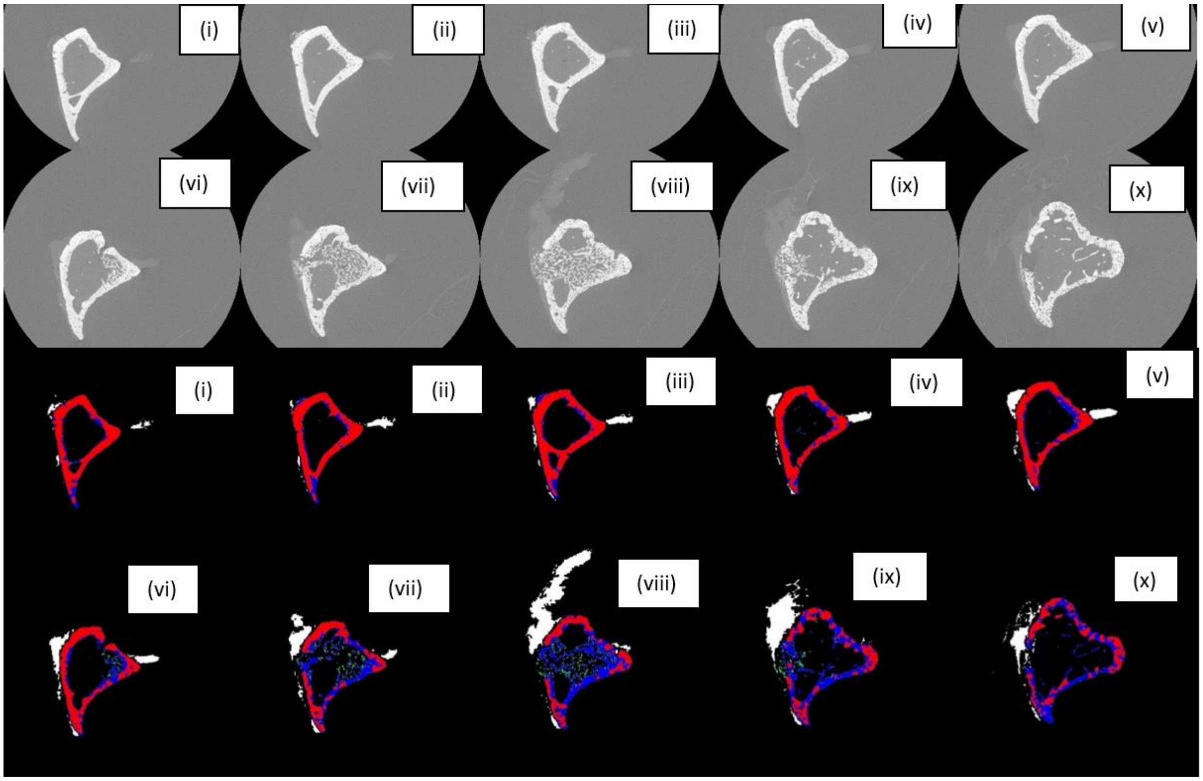


Figure 15: Bone and soft tissue segmentation for image_2, input (top) and output (bottom): Image slices 50 (i), 99 (ii), 149 (iii), 199 (iv), 249 (v), 299 (vi), 349 (vii), 399 (viii), 449 (ix), 499 (x)

4 Discussion

The primary objective of this thesis was to develop an algorithm that would be able to segment both bone and soft tissue sections, while being able to differentiate between three unique BMD regions. The algorithm was successfully developed and could complete the primary objective. However, it was developed in a manner that it would only function correctly and accurately when the input image file included the three unique HA regions along with the bone and soft tissue sample. It was still capable of segmenting the total volume and surface area of both the bone and soft tissue but would produce inaccurate threshold measurements for defining the different BMD regions. This was due to the slight attenuation and brightness difference that occurred when attempting to image the BMD regions through three individual scans.

4.1 Image Acquisition

The preliminary scans that only captured the bone and soft tissue samples unfortunately could not be processed. Figure 16 (i) shows the type of image used to develop the algorithm, while Figure 16 (ii, iii, iv) show example images of some of the initial scans that were produced.

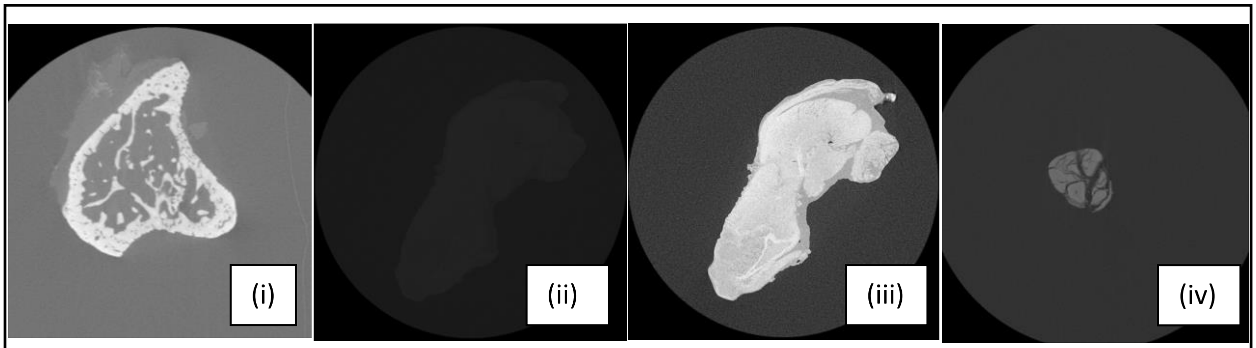


Figure 16: image used to develop algorithm (i), images acquired using 0.4x magnification (ii, iii, iv)

The initial bone and soft tissue samples had excess soft tissue still attached to the bone. While one of the primary objectives was to segment soft tissue along with bone, these samples had excess of muscle tissue as well. When using the 4x magnification it proved near impossible to know where exactly the X-ray source was aiming at. Figure 16 (iii) and Figure 16 (iv) both show a 0.4x magnification, however Figure 16 (iv) was processed after some of the excess soft tissue was removed.

The preliminary attempts at trying to produce separate, viable scans of the three different HA regions on the phantom also ran into some complications. Figure 17 (i) shows the way the HA regions were attempted to be acquired. Figures 17 (ii, iii, and iv) show the supposed three different HA regions; where the green arrow is pointing to what should be the low HA region, the blue arrow is pointing to what should be the medium HA region, and the red arrow is pointing to the high HA region.

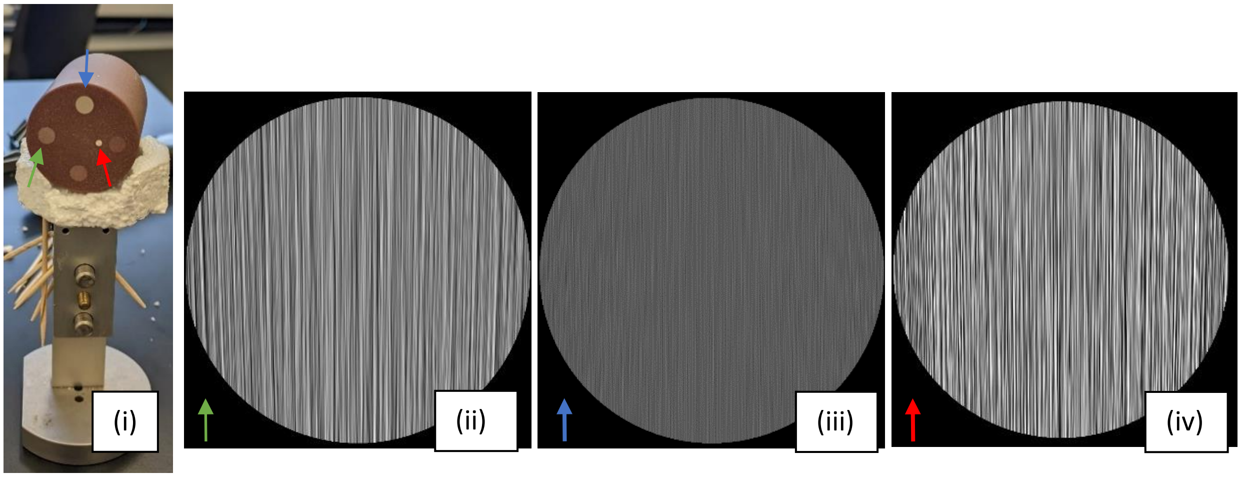


Figure 17: initial attempt to process the three different HA regions (i), supposed low HA region (ii), supposed medium HA region (iii), and supposed high HA region (iv)

The idea behind this attempt was to keep the phantom horizontal and then turn off the rotation mechanism of the MicroCT machine. This, in theory, would capture the three unique regions which would then be processed through the statistical analysis Python script to determine the appropriate thresholding limits for the BMD. However, it is observed that Figure 17 (iii) is darker overall than that of Figures 17 (ii and iv). This region was supposed to have mid-range pixel intensity distribution but ended up having the lowest thresholding limits. When Figures 17 (ii and iv) were analyzed, the pixel distribution produced no viable data and this is mostly likely a result of the streaks of white pixels and streaks of black pixels. Figure 18 shows the type of image that was used to develop the algorithm and just the stark contrasts between the quality of the images produced versus provided.

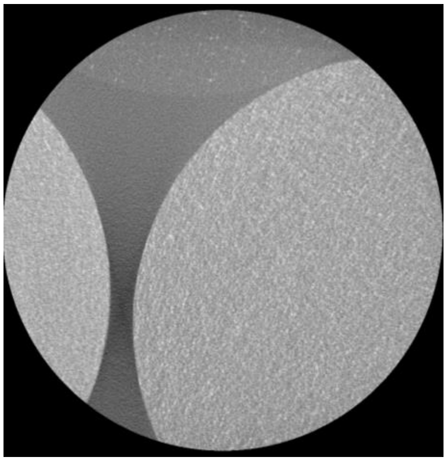


Figure 18: Sample image used to develop BMD segmentation part of algorithm

The secondary attempts at producing viable scanned images of the individual bone and soft tissue improved after excess soft tissue was removed. Figures 19 (i and ii) show that once the bone and soft tissue sample had some of the extra soft tissue removed, the MicroCT recipe provided initially could produce good results. These results were initially analyzed to determine the total volume and surface area of both the bone and soft tissue regions. However, it was apparent that the algorithm would only properly function and produce accurate results if the input image file also contained the three different phantom regions along with the sample.

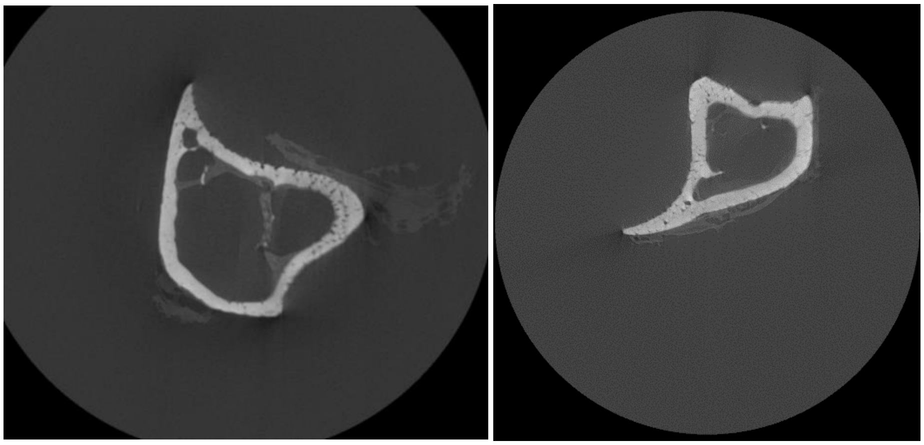


Figure 19: Good scan outputs of only the bone and soft tissue sample

The secondary attempts to produce viable scan results of the HA phantom did attempt to include the HA regions along with the bone and soft tissue sample in output image file. Figure 20 (i) shows the type of holder that was used during the image acquisition of the images used during the development of the algorithm. Figure 20 (ii) shows how the bone and soft tissue sample were placed perpendicular to the HA phantom. In the first trial, the sample was placed horizontal to the phantom as shown in Figure 20 (iii). Not only did placing the sample in this orientation fail to produce viable scans of the bone and soft tissue regions, but ultimately it also still failed at even targeting the right HA region. The output produced is shown in Figure 20 (iv).

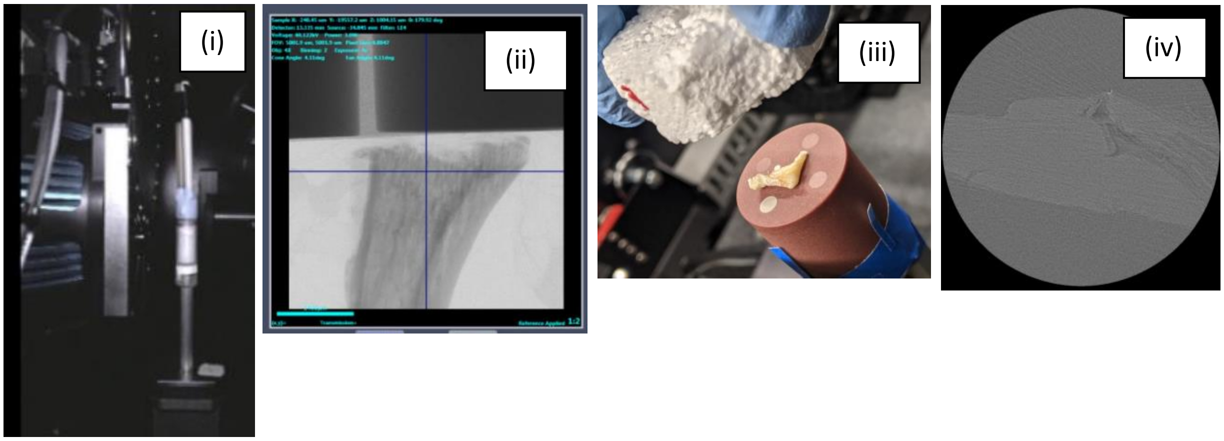


Figure 20: Holder used in the image acquisition of the initial images used to develop the algorithm (i), the orientation of the bone and soft tissue sample in regard to the HA phantom (ii), initial attempt to recreate holder (iii), and output of initial attempts to recreate holder (iv)

The tertiary attempt at producing viable results attempted to recreate the phantom plus sample holder that could produce results like the ones used to construct the algorithm as exact as possible. It was determined that it was essential for the sample to be perpendicular to the phantom. The only way to achieve this was to place the sample in the metal holder so that it would be straight up and then setting the phantom right above it as seen in Figure 20 (ii).

Eventually a holder as shown in Figures 21 (i, ii, and iii) was developed that could produce the best results given that the three different HA regions were far apart. Figure 21 (i) shows the bone and soft tissue sample placed in the metal holder so that it was vertical. It also shows a small paper clip being supported by some tape and this was placed here so that a piece of Styrofoam could be placed on top of it. As Styrofoam does not appear in CT images, this was a viable material that could be used to hold the phantom directly above the bone sample. Figure 21 (ii) shows the Styrofoam placed above the paperclip and Figure 21 (iii) shows the phantom placed above the sample.

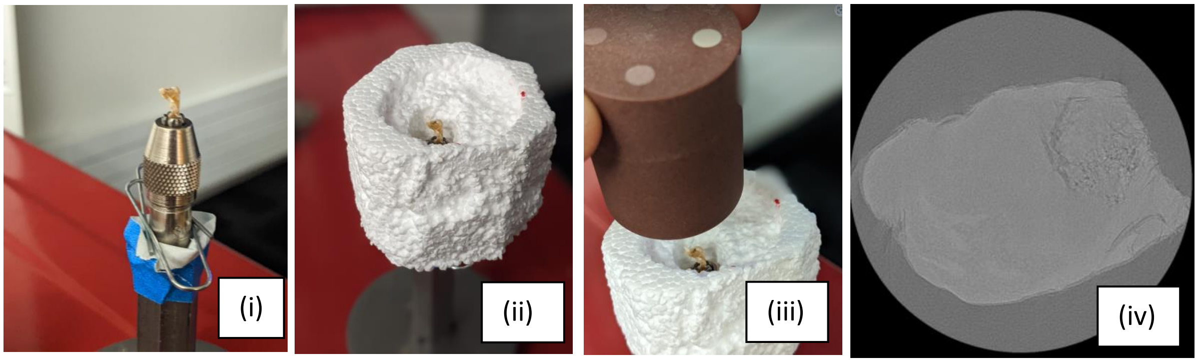


Figure 21: Bone and soft tissue sample in metal holder with paper clip near base (i), Styrofoam holder that would sit on the paper clip and be capable of holding the HA phantom directly above the sample (ii and iii); initial output (iv)

Figure 21 (iv) shows the initial output produced from this scan. While it was overall too gray to be processed, the results did demonstrate that with more precision and further manipulation of the MicroCT parameters, viable results could be obtained. Several attempts were made using different MicroCT parameters. The final results are shown in Figure 22.

While the images do clearly show the different HA regions as well as the bone and soft tissue sample, Figure 22 (iii) does appear to just be an overall darker image. As this image segmentation algorithm functions by analyzing the pixel intensities from the different HA regions on the phantom to determine the appropriate thresholding levels, these type of brightness discrepancies between the three images would have likely resulted in inaccurate results and thus were also excluded from the study.

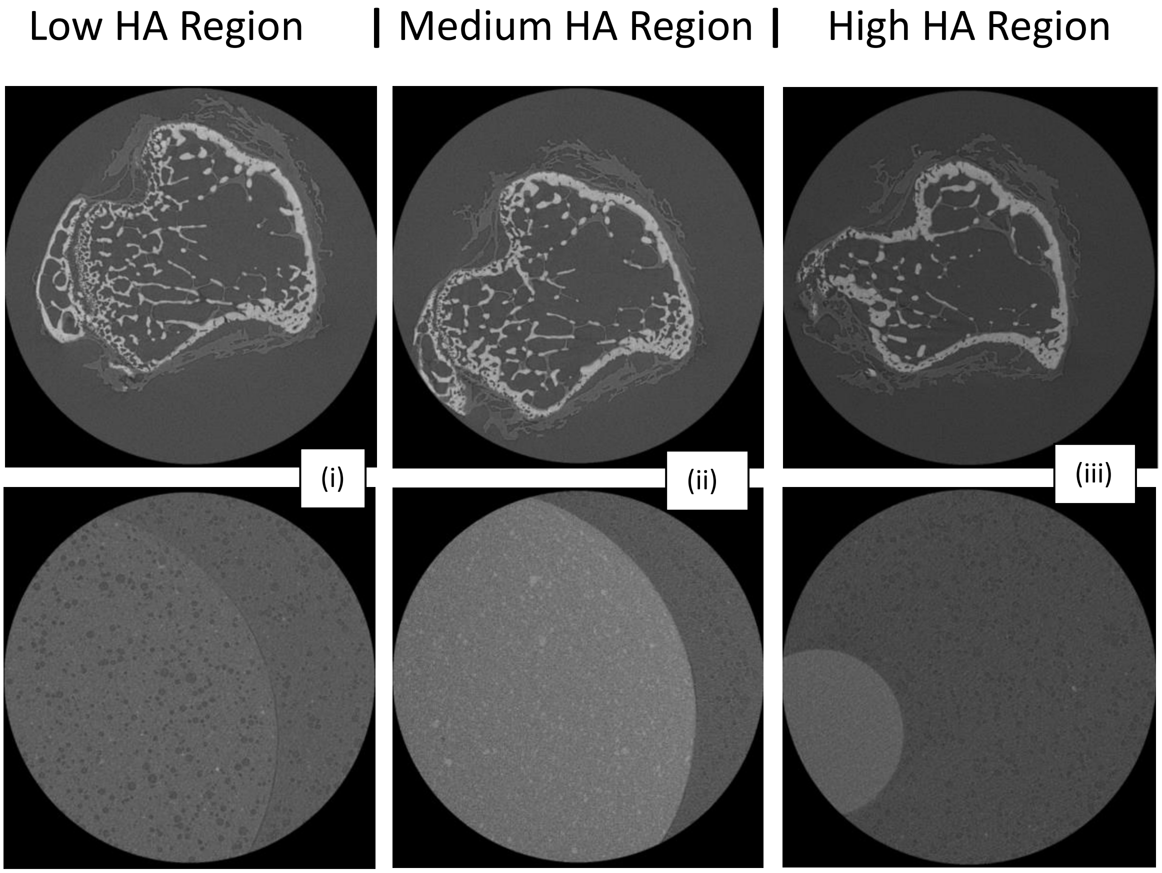


Figure 22: Individually scanned HA regions, along with the bone and soft tissue sample; low HA region and sample (i), medium HA region and sample (ii), high HA region and sample (iii)

4.2 BMD Threshold Limits

The final limits for the BMD segmentation were decided upon by taking into consideration the initial thresholding limits used to eventually determine the total bone volume and the values produced from the statistical analysis from the three unique BMD regions. The bone section of image_1 was set to only include pixel values above 155 and for image_2 it was set for pixel values above 160. These were the values that captured the most amount of bone pixels from both input images, while also not thresholding as much random background noise.

Referencing figure 12, the distribution for the pixel values from the region with low BMD (green parts) was shown to be mostly concentrated below the pre-defined threshold limit used to determine the total volume of bone. When the thresholding limits for the low BMD region (green) were set to parameters like the one found in the 50-percentile range, the image would threshold too many additional pixels that were clearly not bone as seen on Figure 23. Thus, it was determined that the lowest thresholding value for the BMD segmentation had to be the value used to first determine the bone total volume.

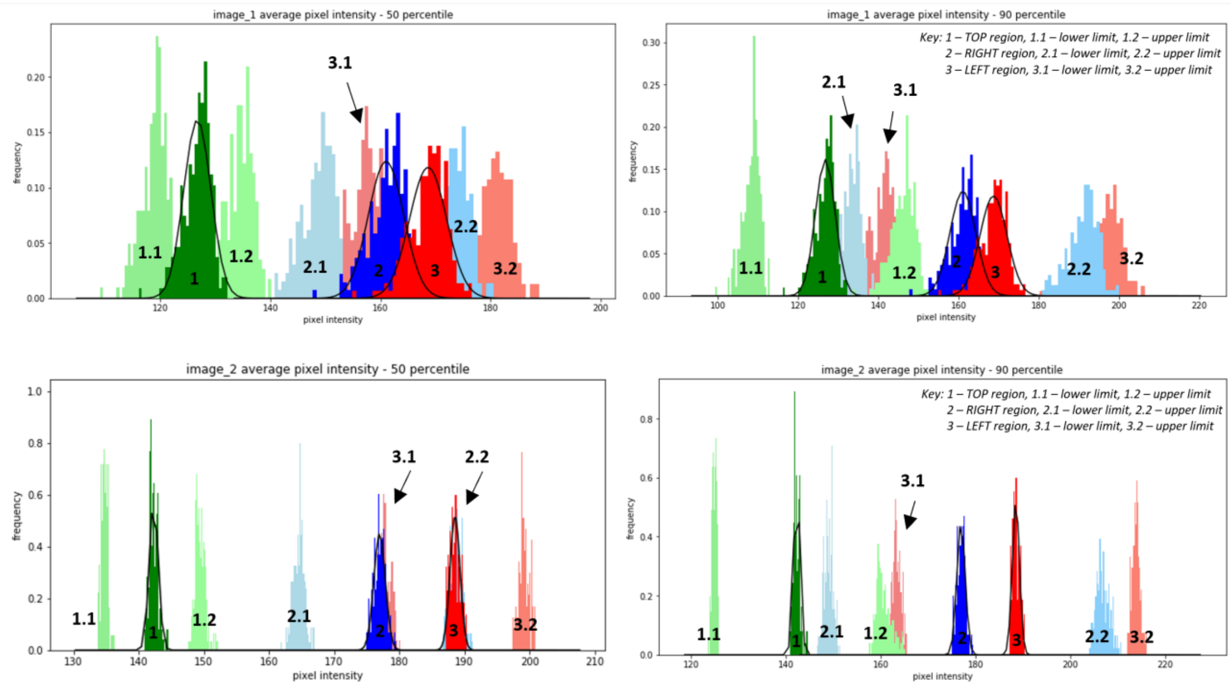


Figure 12: The 50 and 90 percentile distributions of the average pixel intensity per image slice

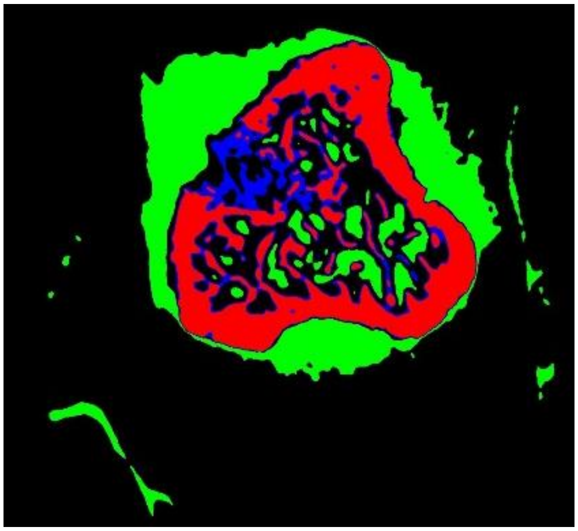


Figure 23: Failed attempt at initially defining BMD segmentation

The output files generated from the code were capable of being uploaded into Mimics to create a 3D model showcasing the different regions as seen on Figure 24. Viewing the output in such a manner allowed for the different segmented regions to be properly visualized.

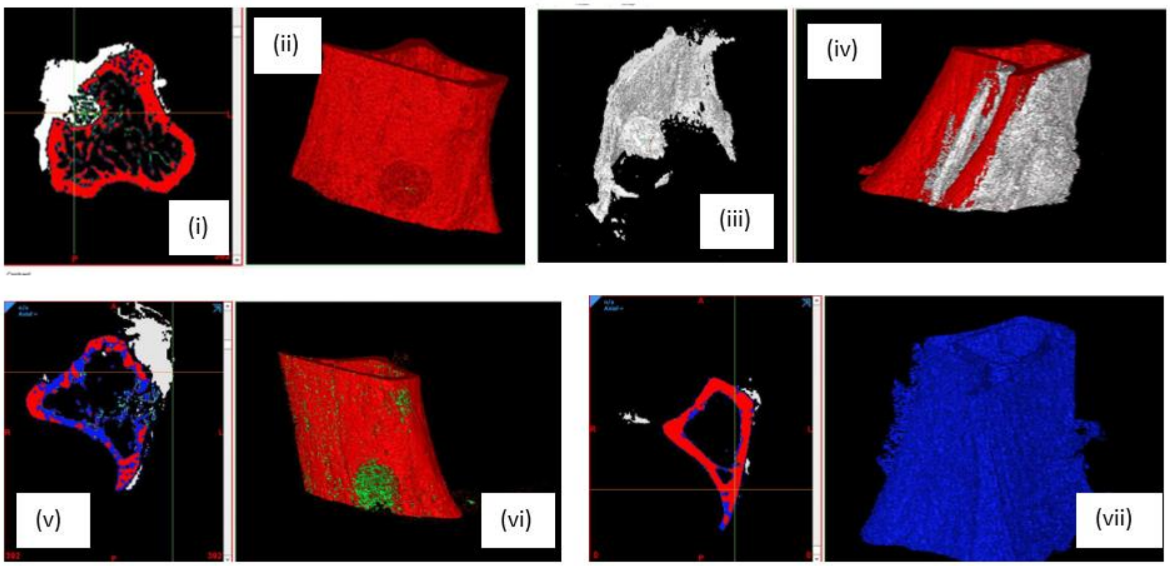


Figure 24: image_1 (i), high BMD region (ii), soft tissue region (iii), high BMD and soft tissue region (iv) image_2 (v), low and high BMD region (vi), medium BMD region (vii)

5 Conclusion

The code developed for the main objective of this thesis functions well and delivers the results expected from producing a Python Script that would help facilitate the image segmentation process of both bone and soft tissue; however, the code was constructed to work under very specific parameters to function as originally intended. Efforts are underway to recreate similar image acquisition parameters.

Although the code was unable to process the new scans completely, it did demonstrate its efficiency in successfully segmenting and analyzing samples where bone and phantom were imaged together. The final percent error produced from the two images fully processed shows that there was not a significant difference between the original number of pixels used to determine the total volume of bone and the total sum of the number of pixels from the three unique BMD regions.

This thesis set out to provide a manner to acquire more information about the soft tissue in relation to a bone fracture and it was successful in doing that. Future work is needed to develop a code that can use the information from multiple different image sources and still be capable of producing viable results. More work could be done in creating a method to better differentiate between the medium BMD region and the High BMD region, as there was overlap observed when analyzing the statistical distribution of pixel intensities from the two regions.

6 Future Work

As the thesis progressed and more knowledge pertaining to coding was acquired, it became apparent that most of the image morphological changes that were applied through ImageJ could have also been included in the final Python script. Further work in developing the Python script to include the image morphological manipulation techniques from ImageJ could potentially produce better results. This is because ImageJ applies a standard 3x3 kernel to the entire image and it's much harder to alter the perimeters on ImageJ. On Python, the user has a lot more flexibility and can better define the size of kernel to apply, as well as being able to manipulate the other parameters to their liking.

Besides improving the Python script, a new HA phantom could be used where the three different HA-containing cylinders are closer together. This, along with a better developed phantom plus sample holder could be produced to allow it to be as close as possible to the bone and soft tissue sample.

Lastly, future work could be done in the development of an image segmentation process that also utilizes machine learning. As only two image files containing the bone, the soft tissue and the three unique HA regions were provided, developing an image segmentation process that could use machine learning was difficult due to the lack of datapoints that could be used to train a model. If the code were to be improved so that the ImageJ parts could be done effectively on Python and if provided with a more files that contained both the sample and the three different HA regions in the same image stack, it should be possible to train a model using machine learning that would be able to more accurately and much more quickly process an input image to produce the results one would expect.

7 References

[1] Currey, J. D. (2009). How well are bones designed to resist fracture? *Journal of Bone and Mineral Research, 18*(4), 591-598. https://doi.org/10.1359/jbmr.2003.18.4.591

[2] Lovejoy, C. O. & Heiple, K. G. (1981). The analysis of fractures in skeletal populations with an example from the Libben site, Ottowa County, Ohio. *American Journal of Physical Anthropology, 55*(4), 529-542. https://doi.org10.1002/ajpa.1330550414

[3] Louisiana State University Health Sciences Center. (2016, September 7). Why bones don't heal: Researchers identify risk factors for nonunion of fractures. *ScienceDaily*. www.sciencedaily.com/releases/2016/09/160907135137.htm

[4] Turner, C. H. (2006). Bone strength: Current concepts. *Annals of the New York Academy of Sciences, 1068*, 429-446. Doi: 10.1196/annals.1346.039

[5] White, T. D., & Folkens, P. A. (2005). *The Human Bone Manual.* Elsevier.

[6] Seaman, J. A. & Simpson, A. M. (2004). Tibial fractures. *Clinical Techniques in Small Animal Practice, 19*(3), 151-167. Doi: 10.1053/j.ctsap.2004.09.007

[7] Gordon, J. E., & O'Donnell, J. C. (2012). Tibia fractures: What should be fixed? *Journal of Pediatric Orthopaedics, 32,* S52-S61. Doi: 10.1097/BPO.0b013e318254c7be

[8] Lin, H. H., Peng, S. L., Wu, J., Shih, T. Y., Chuang, K. S., & Shih, C. T. (2017). A novel two-compartment model for calculating bone volume fractions and bone mineral densities from computed tomography images. *IEEE Transactions on Medical Imaging, 36*(5), 1094-1105. Doi: 10.1109/TMI.2016.2646698

[9] Sheen, J. R. & Garla, V. V. (2022). *Fracture healing overview.* StatPearls Publishing.

[10] Dimitriou, R., Tsiridis, E. & Giannoudis, P. V. (2005). Current concepts of molecular aspects of bone healing. *International Journal of the Care of the Injured, 36*, 1392-1404. Doi: 10.1016/j.injury.2005.07.019

[11] Travia, S. (2021). *Fabrication of double-walled polymeric microspheres for improved bone healing and regeneration.* [Unpublished master's thesis]. University of Wisconsin-Milwaukee.

[12] Kolar, P., Schmidt-Bleek, K., Schell, H., Gaber, T., Toben, D., Schmidmaier, G., Perka, C., Buttgereit, F. & Duda, G. (2010). The early fracture hematoma and its potential role in fracture healing. *Tissue Engineering Part B: Reviews, 16*(4), 427-434.Doi:10.1089/ten.teb.2009.06

[13] Schell, H., Duda, G. N., Peters, A., Tsitsilonis, S., Johnson, K. A. & Schmidt-Bleek, K. (2017). The haematoma and its role in bone healing. *Journal of Experimental Orthopaedics, 4*, 1-11. Doi: 10.1186/s40634-017-0079-3

[14] Hankenson, K. D., Gagne, K., & Shaughnessy, M. (2015). Extracellular signaling molecules to promote fracture healing and bone regeneration. *Advanced Drug Delivery Reviews, 94,* 3-12. Doi: 10.1016/j.addr.2015.09.008

[15] Bone Remodeling and Repair. (2022). In: LibreTexts.

[16] Ghiasi, M.S., Chen, J., Vaziri, A., Rodriguez, E. K., & Nazarian, A. (2017). Bone fracture healing in mechanobiological modeling: A review of principles and methods. *Bone Reports, 6*, 87-100. Doi: 10.1016/j.bonr.2017.03.002

[17] Marsell, R. & Einhorn, T. A. (2011). The biology of fracture healing. *Injury, 42*(6), 551-555. Doi: 10.1016/j.injury.2011.03.031

[18] Shah, K., Majeed, Z., Jonason, J. & O'Keefe, R. (2013). The role of muscle in bone repair: the cells, signals, and tissue responses to injury. *Current Osteoporosis Reports, 11*(2), 130-135. Doi: 10.1007/s11914-013-0146-3

[19] Colnot, C., Zhang, X., & Knothe Tate, M. L. (2012). Current insights on the regenerative potential of the periosteum: molecular, cellular & endogenous engineering approaches. *Journal of Orthopaedic Research, 30*(12), 1869-1878. Doi: 10.1002/jor.22181

[20] Zuscik, M. J., Hilton, M. J., Zhang, X., Chen, D., & O'Keefe, R. J. (2008). Regulation of chondrogenesis and chondrocyte differentiation by stress. *Journal of Clinical Investigation, 118*(2), 429-438. Doi: 10.1172/JCI134174

[21] Marcucio, R., Hu, D., Yang, F., Bahney, C. and Miclau, T. (2016), Transdifferentiation of Chondrocytes to Osteoblasts during Bone Fracture Healing. *The FASEB Journal*, 30: 223.1-223.1. https://doi.org/10.1096/fasebj.30.1_supplement.223.1

[22] Zhang, X., Xie, C., Lin, A. S., Ito, H., Awad, H., Lieberman, J. R., Rubery, P. T., Schwarz, E. M., O'Keefe, R. J., & Guldberg, R. E. (2005). Periosteal progenitor cell fate in segmental cortical bone graft transplantations: implications for functional tissue engineering. *Journal of Bone and Mineral Research, 20*(12), 2124-2137. Doi: 10.1359/JBMR.050806

[23] Colnot, C. (2009). Skeletal cell fate decisions within periosteum and bone marrow during bone regeneration. *Journal of Bone and Mineral Research, 24*(2), 274-282. Doi: 10.1359/jbmr.081003

[24] Nazarian, A., von Stechow, D., Zurakowski, D., Muller, R. & Snyder, B. D. (2008). Bone volume fraction explains the variation in strength and stiffness of cancellous bone affected by metastatic cancer and osteoporosis. *Calcified Tissue International, 83,* 368-379. Doi: 10.1007/s00223-008-9174-x

[25] Faulkner, K. G. & McClung, M. R. (1995). Quality control of DXA instruments in multicenter trials. *Osteoporosis International, 5,* 218-227. Doi: 10.1007/bf01774010

[26] Adams, J. E. (2009). Quantitative computed tomography. *European Journal of Radiology, 71,* 415-424. Doi: 10.1016/j.ejrad.2009.04.074

[27] Karantanas, A. H., Kalef-Ezra, J. A., & Glaros, D. C. (1991). Quantitative computed tomography for bone mineral measurement: technical aspects, dosimetry, normal data and clinical applications. *The British Journal of Radiology, 64,* 298-304. Doi: 10.1259/0007-1285-64-760-298

[28] Šćepanović, D., Kirshtein, J., Jain, A. K., & Taylor, R. H. (2005). Fast algorithm for probabilistic bone edge detection (FAPBED). *Proceedings of the SPIE, 5747,* 1753-1765. Doi: 10.1117/12.596950

[29] Jain, A. K., & Taylor, R. H. (2004). Understanding bone responses in B-mode ultrasound images and automatic bone surface extraction using a Bayesian probabilistic framework. *Proceedings of the SPIE, 5373,* 131-142. Doi: 10.1117/12.535984

[30] Kowal, J., Amstutz, C., Ioppolo, J., Styner, M. & Nolte, L. (2002). Fast automatic bone contour extraction in ultrasound images. *IEEE Transactions on Medial Imaging,* 1-20.

[31] NIH Consensus Development Panel on Osteoporosis Prevention, Diagnosis, and Therapy (2001). Osteoporosis prevention, diagnosis, and therapy. *Journal of the American Medical Association, 285*(6), 785-795. Doi: 10.1001/jama.285.6.785

[32] Sahadun, N. A., Ali, N. A., Haron, H. & Kadir, M. R. (2017). A simple object detection of trabecular bone morphology in 2D image at bovine sample preliminary segmentation. *2017 6th ICT International Student Project Conference (ICT-ISPC),* 1-5. Doi: 10.1109/ICT-ISPC.2017.8075333

[33] Sharma, N., & Aggarwal, L. M. (2010). Automated medical image segmentation techniques. *Journal of medical physics*, *35*(1), 3–14. https://doi.org/10.4103/0971-6203.58777

[34] Mikulka J., Gesheidtova, E., & Bartusek, K. (2012). Soft-tissues Image Processing: Comparison of Traditional Segmentation Methods with 2D active Contour Methods. *Measurement Science Review*, 12, 153-161. Doi: 10.2478/v10048-012-0023-8

[35] Carter, T. J., Sermesant, M., Cash, D. M., Barratt, D. C., Tanner, C., & Hawkes, D. J. (2005). Application of soft tissue modelling to image-guided surgery. *Medical engineering & physics*, *27*(10), 893–909. https://doi.org/10.1016/j.medengphy.2005.10.005

[36] Carter, T. J., Sermesant, M., Cash, D. M., Barratt, D. C., Tanner, C., & Hawkes, D. J. (2005). Application of soft tissue modelling to image-guided surgery. *Medical Engineering & Physics*, *27*(10), 893–909. https://doi.org/10.1016/j.medengphy.2005.10.005

[37] Pure Imaging Phantoms. (2022). Micro CT Phantom Kit. https://www.pureimagingphantoms.com/product/micro-ct/

[38] Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Cardona, A. (2012). Fiji: an open-source platform for biological-image analysis. *Nature Methods*, *9*(7), 676–682. doi:10.1038/nmeth.2019

[39] Gouillart, E., Nunez-Iglesias, J., & van der Walt, S. (2016). Analyzing microtomography data with Python and the scikit-image library. *Advanced Structural and Chemical Imaging, 2*(18), 1-11. Doi: 10.1186/s40679-016-0031-0

[40] Knudsen, E. B., Sørensen, H. O., Wright, J. P., Goret, G., & Kieffer, J. (2013). FabIO: easy access to two-dimensional x-ray detector images in Python. *Journal of Applied Crystallography, 46,* 537-539. Doi: 10.1107/s0021889813000150

[41] Scikit-image (2012). regionprops.py [Source code]. https://github.com/scikit-image/scikit-image/blob/main/skimage/measure/_regionprops.py#L1025-L1323.

[42] Pandas development team (2008). Pandas [Source code]. https://github.com/pandas-dev/pandas.

[43] Python (1991). glob.py [Source code]. https://github.com/python/cpython/blob/3.10/Lib/glob.py.

[44] Halmos, P. R. (1974). Lectures on Boolean algebras. *Undergraduate Texts in Mathematics*. Doi: 10.1007/978-1-4612-9855-7

[45] Tomasi, C., & Manduchi, R. (1998). Bilateral filtering for gray and color images. *Sixth International Conference on Computer Vision,* 839-846. Doi: 10.1109/iccv.1998.710815

[46] Elad, M. (2002). On the origin of the bilateral filter and ways to improve it. *IEEE Transactions on Image Processing*, *11*(10), 1141-1151. Doi: 10.1109/TIP.2002.801126

[47] Singh, H. (2019). Advanced image processing using OpenCV. *Practical Machine Learning and Image Processing,* 63-88. Doi: 10.1007/978-1-4842-4149-3_4

[48] Shrivakshan, G.T. & Chandrasekar, Chandramouli. (2012). A Comparison of various Edge Detection Techniques used in Image Processing. *International Journal of Computer Science Issues*. 9. 269-276.

[49] Xu, Z., Baojie, X., & Guoxin, W. (2017). Canny edge detection based on open CV. *13th IEEE International Conference on Electronic Measurement & Instruments*, 53-56. Doi: 10.1109/ICEMI.2017.8265710