Documentation of TIDAQ: Tissue Identification and Quantification (v. 1.4)

September 23, 2019
Sokratis Makrogiannis, Ph.D. (smakrogianniss@desu.edu)
Division of Physics, Engineering, Mathematics, and Computer Science
Delaware State University

1 Contents

2 Introduction 2
3 Installation 2
  3.1 Windows 2
  3.2 Linux 2
4 Application Description 5
  4.1 Main window 5
    4.1.1 Operation Panel 5
    4.1.2 Parameter Panel 5
    4.1.3 Subject List Panel 6
    4.1.4 Results Panel 6
5 Software Model 8
6 Algorithm Description 9
    6.1.1 4% tibia: 9
    6.1.2 38% tibia: 10
    6.1.3 66% tibia: 11
    6.1.4 Middle thigh 12
    6.1.5 Computed Features for each Tissue Type 13
7 Appendix 14

7.1.1 Parameter file 14
7.1.2 Identified Tissue Types 15
7.1.3 Structural and Material Measures 15
7.1.4 Acronyms 16
8 References 17

2 Introduction

TIDAQ (Tissue Identification and Quantification) –formerly known as TiBesT- is a software tool that analyzes pQCT (peripheral Quantitative Computed Tomography) [1],[2] images of the tibia, and CT images of the middle thigh. The main processes are delineation of hard and soft tissue types at selected cross-sectional sites of the lower leg and computation of material and structural properties of each tissue type. The goal is to use this software for analysis of large scale datasets in clinical studies.

This guide provides information about installation and use of the software. Then it briefly describes the underlying software model, the image processing/analysis algorithms and finally discusses more specialized topics in the appendix.

3 Installation

3.1 Windows

1. Install Microsoft Visual C++ 2008 SP1 Redistributable Package (x86), if not already installed (English version in http://www.microsoft.com/download/en/details.aspx?id=5582)
2. Download the appropriate mipav version according to your MS Windows operating system from http://mipav.cit.nih.gov/clickwrap.php
3. Run mipav, then select Plugins->Install Plugin.
4. Navigate to the directory where “PlugInTIDAQ.jar” is located, select the file and install it.
5. Make sure that under the Plugins directory you see a submenu DSU-NIA->TIDAQ.
6. Close and restart mipav, then run TIDAQ under plugins menu.

3.2 Linux

1. Download the appropriate mipav version according to your Linux distribution from http://mipav.cit.nih.gov/clickwrap.php
2. Run mipav, then select Plugins->Install Plugin.
3. Navigate to the directory where “PlugInTIDAQ.jar” is located, select the file and install it.
4. Make sure that under the Plugins directory you see a submenu DSU-NIA->TIDAQ.
5. Create a directory named “results” under the mipav root directory.
6. Close and restart mipav, then run TIDAQ under plugins menu.

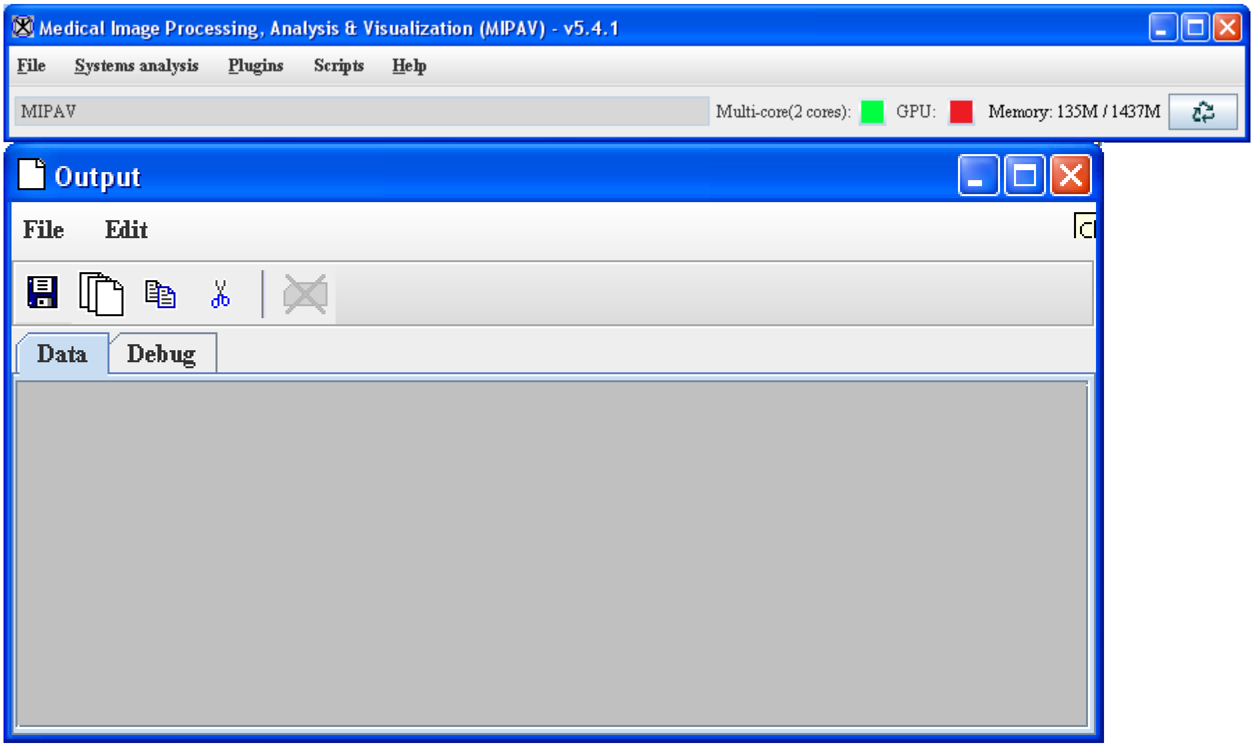
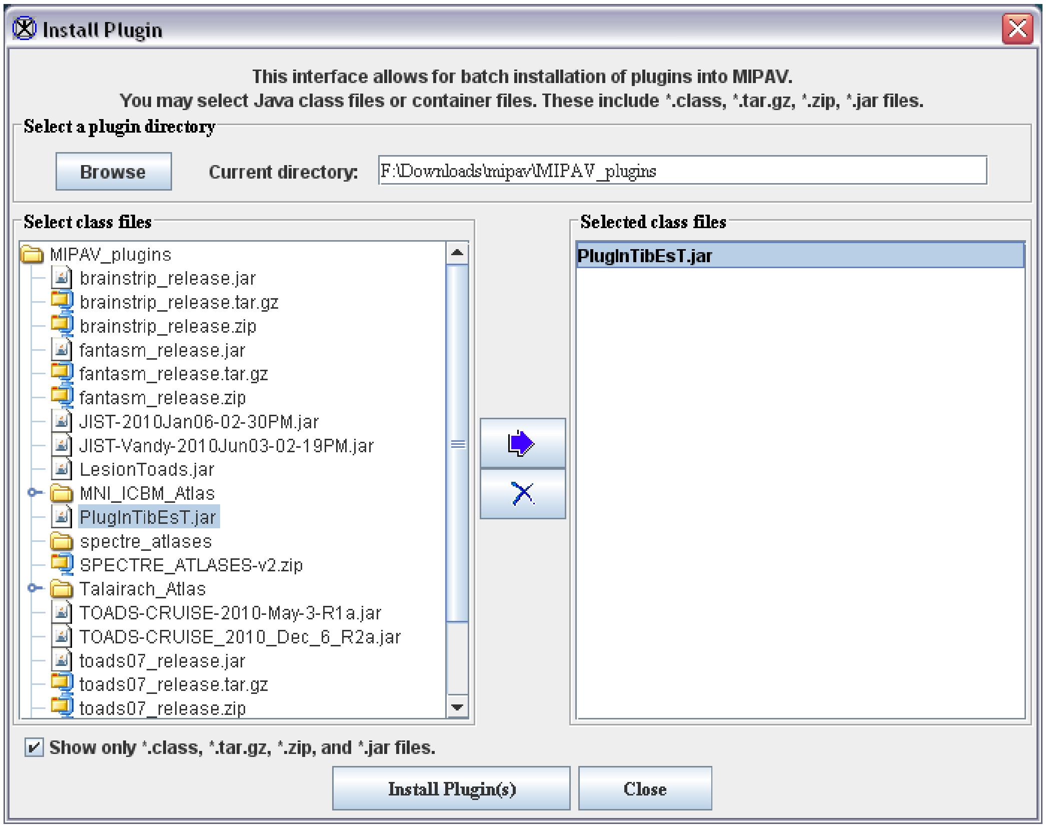


Figure 1 MIPAV screenshot.



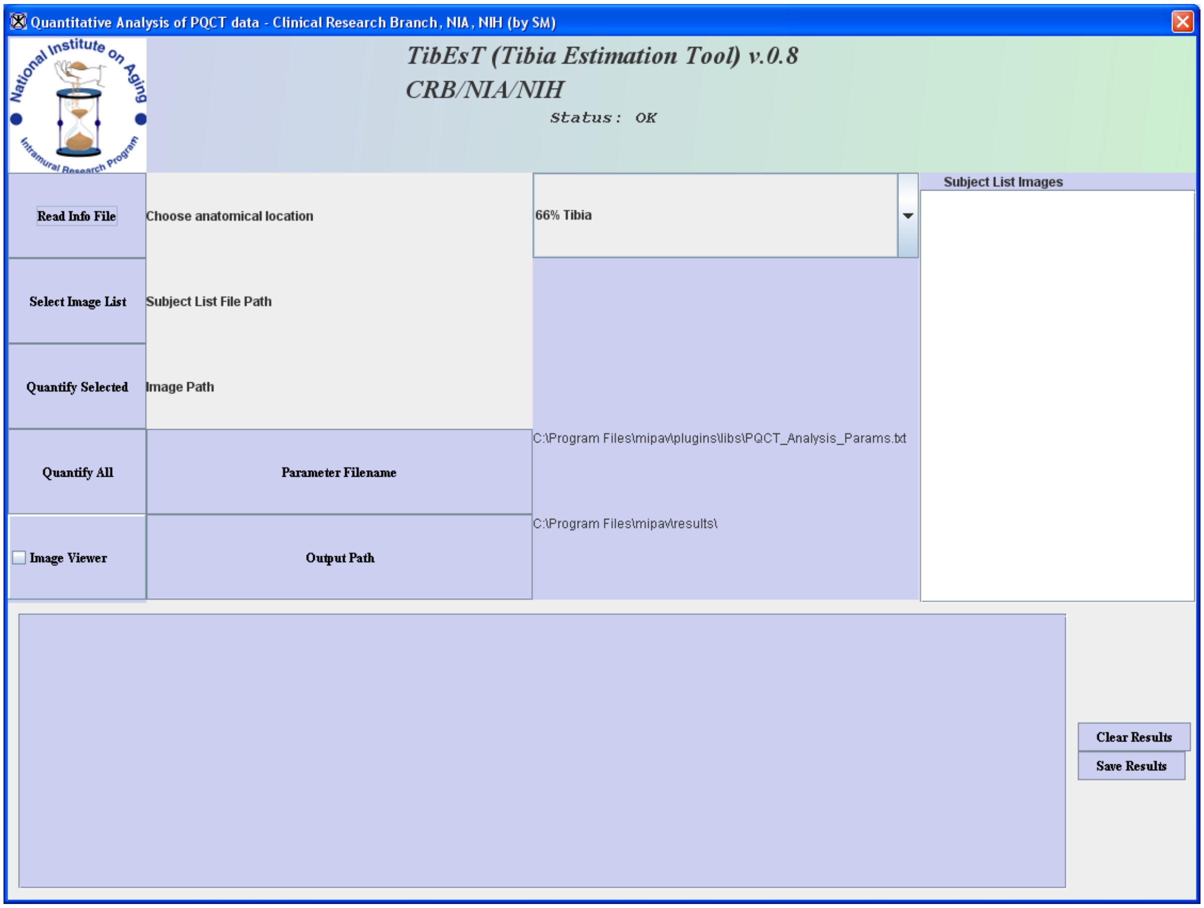
This interface allows for batch installation of plugins into MIPAV.
You may select Java class files or container files. These include *.class, *.tar.gz, *.zip, *.jar files.

Current directory: F:\Downloads\mipav\MIPAV_plugins

Selected class files: PlugInTibEsT.jar

☑ Show only *.class, *.tar.gz, *.zip, and *.jar files.

Figure 2 Plugin Installation.



TibEsT (Tibia Estimation Tool) v.0.8
CRB/NIA/NIH
Status: OK

Choose anatomical location: 66% Tibia

Subject List File Path

Image Path

Parameter Filename: C:\Program Files\mipav\plugins\libs\PQCT_Analysis_Params.txt

Output Path: C:\Program Files\mipav\results\

Figure 3 TIDAQ screenshot.

4 Application Description

This application can analyze pQCT images of the lower leg acquired at 4%, 38% and 66% tibia sites. In short, the 4% and 38% involve hard tissue quantification, while analysis at 66% tibia aims for soft tissue quantification.

The user is able to execute the workflows at the above 3 sites through a GUI frontend without having to know many details about the algorithmic aspect. The first step is to define an image list through a text info file, or by directly selecting a list of images. Then the user selects the appropriate tibia site and can execute the analysis of one, or many images. The quantification results are reported in a text area and the user can export them in a delimited-format text file and then import them into a spreadsheet. There is an option to view segmentation results to examine the accuracy of segmentation. We detail the application layout and main functions in the next sections.

4.1 Main window

Once the user starts the plugin, the main window will appear as displayed in Figure 3. The plugin frame is divided into the Operation Panel (left side), Parameter Panel (center), Subject List Panel (right side) and Quantification Panel (bottom).

4.1.1 Operation Panel

Read Info File: Opens a dialog box for reading a text file with the root directory followed by the list of raw CT images to be analyzed.

Select Image List: User navigates to the data directory and directly selects the raw CT images to be analyzed. No text info file is needed here.

Quantify Selected: First select an image from the subject list on the right, and then push this button to run the analysis on the selected image.

Quantify All: First select an image from the subject list on the right, and then push this button to run the analysis on all the members of the list from the selected image downwards.

Image Viewer: Push this button to toggle the image viewer state. When the state is ON, the original and segmented images will be displayed in separate windows. Important note: images need to be analyzed first in order to be viewed.

4.1.2 Parameter Panel

Anatomical Location: User must set the tibia level to be analyzed so that the program runs the correct analysis algorithm. Options are 4%, 38% and 66% tibia. The last option named "ANONYMIZATION", removes the identification information from the original input images and saves them in the native scanner format.

Subject List File Path: Displays the path to text file with the data root and the image list after completing "Read Info File" operations.

Image Path: Displays the path to group of images that were selected for analysis after completing the “Select Image List”, or “Read Info File” operations.

Parameter Filename: The parameter filename used in analysis. The user can change the filename, or edit the selected filename to change the default settings (this needs to be done with caution).

Output Path: The destination directory to which all segmentation and quantification files are copied. The user can select another directory if needed.

4.1.3 Subject List Panel

This panel displays the list of images that have been selected for segmentation and quantification using the “Select Image List”, or “Read Info File” operations.

4.1.4 Results Panel

Quantification Pane: This text area displays information about retrieved from the CT image header along with measurements of structural and material properties produced by the analysis. This information is also stored in text and image files under the output path described above.

Clear Results: Clears the quantification pane content. This is usually selected to clear results from old analyses, before running a new one.

Save Results: Saves the quantification pane content in a text file selected by the user. When an analysis session is completed, the user can save the results in this text format and then import them into a spreadsheet program as data of delimited length.

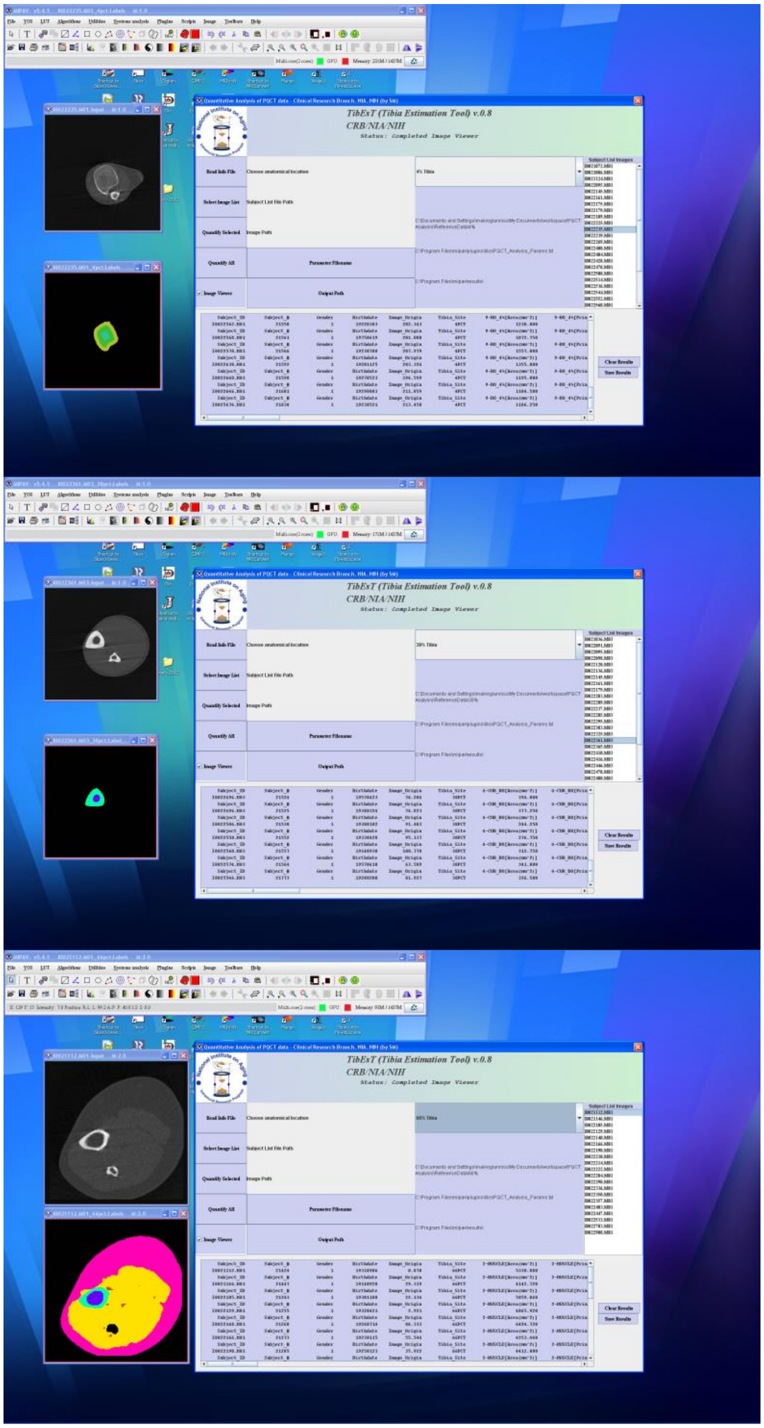


Figure 4 Screenshot of analysis of a list of subjects at 4%,38% and 66% tibia.

5 Software Model

Specifications

The system should:

1. produce automated and reproducible measurements from input data
2. be executed in batch mode to reduce user's operations
3. meet accuracy requirements set by clinical experts
4. include a user-friendly interface, image viewer and quantification review and text output capabilities
5. run on multiple platforms and be easy to deploy.

Algorithm development

1. Divide the problem into parts and solve the quantification problem at 4%, 14% and 66% of tibia height.
2. Rapid prototyping (matlab,etc) and implementation (C++).

Software engineering

1. Design of execution model.
2. Development of an application with GUI elements.
3. Execution of algorithms from the application.

Implementation

1. The algorithm module is a shared dynamic library that is loaded during runtime. Image analysis and feature computation algorithms were implemented in C++ using the ITK software library [3].
2. The C++ Runtime Application can be run for test purposes and in batch mode.
3. The Java Runtime Application loads and executes algorithms library using Java’s [4] Native Interface (JNI) [5].
4. The Java GUI app loads algorithms as a dynamic library and uses MIPAV's API [6] to run as a plugin.

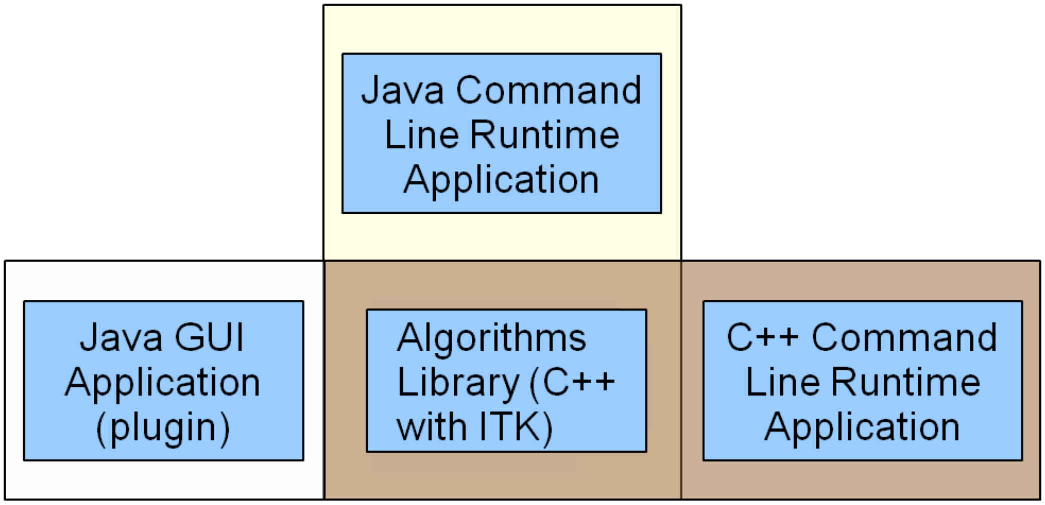

Java Command Line Runtime Application
Java GUI Application (plugin)
Algorithms Library (C++ with ITK)
C++ Command Line Runtime Application

Figure 5 TIDAQ main software modules.

6 Algorithm Description

This module introduces and applies image processing, analysis and computer vision-based techniques [7] for segmentation of soft and hard tissues.

6.1.1 4% tibia:

1. Tissue clustering using K-Means (5 classes) [8].
2. Form whole leg region by subtracting the air from all identified tissue groups.
3. Identify bone regions using prior HU and size knowledge.
4. Compute centroid of bone regions.
5. Apply fast marching [9] followed by geodesic active contour segmentation [10] to identify tibia.
6. Compute Distance Transform on tibia region followed by distance values ranking to define 50% and 10% of tibia.
7. Label whole leg, whole tibia, 50% tibia and 10% tibia regions.
8. Compute material and structural features over labeled regions.

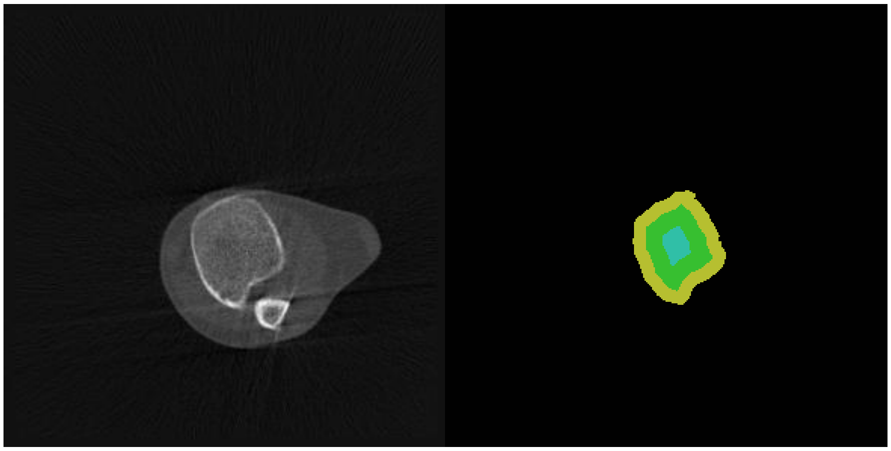


Figure 6 Segmentation at 4% tibia: whole area, 50% area and 10% area.

6.1.2 38% tibia:

1. Tissue segmentation using central clustering into 5 classes {AIR, FAT, MUSCLE, BONE INTERIOR, CORTICAL BONE}.
2. Form whole leg region by subtracting the air from all identified tissue groups.
3. Identify bone regions and divide them into cortical bone and bone interior.
4. Label whole leg, and cortical and interior tibia regions.
5. Compute material and structural features over labeled regions.

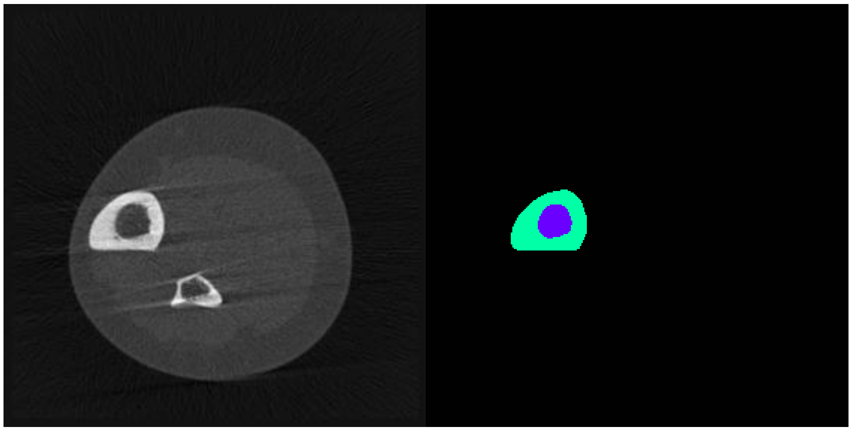


Figure 7 Segmentation at 38% tibia: cortical bone and bone marrow.

6.1.3 66% tibia:

1. Tissue segmentation using central clustering into 5 classes {AIR, FAT, MUSCLE, BONE INTERIOR, CORTICAL BONE}.

2. Form whole leg region by subtracting the air from all identified tissue groups.

3. Separate inter-muscular from subcutaneous fat by

    a. connected component analysis (default-recommended), or

    b. Geodesic Active Contour segmentation (experimental).

4. Identify tibia, fibula and divide into cortical bone and bone interior.

5. Label whole leg, muscle, fat, cortical tibia and tibia interior.

6. Compute material and structural features over selected regions.

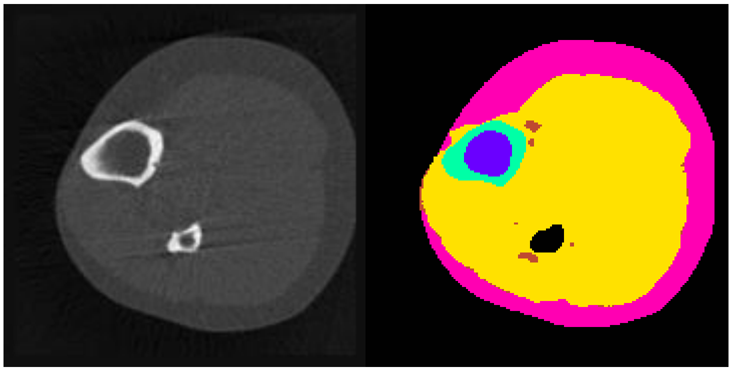


Figure 8 Segmentation at 66% tibia: subcutaneous fat, inter-muscular fat, and muscle.

6.1.4 Middle thigh

1. Automatically remove patient table and select left leg.

2. Tissue segmentation using central clustering into 5 classes {AIR, FAT, MUSCLE, CORTICAL BONE_LOW, CORTICAL_BONE_HIGH}.

3. Separate inter-muscular from subcutaneous fat by

    a. connected component analysis (default-recommended), or

    b. Geodesic Active Contour segmentation (experimental).

4. Identify interior bone component.

5. Re-classify inter-muscular fat and muscle pixels using k-means clustering into 2 classes.

6. Morphological dilation of subcutaneous fat region to address partial voluming effect.

7. Compute material and structural features over selected regions.

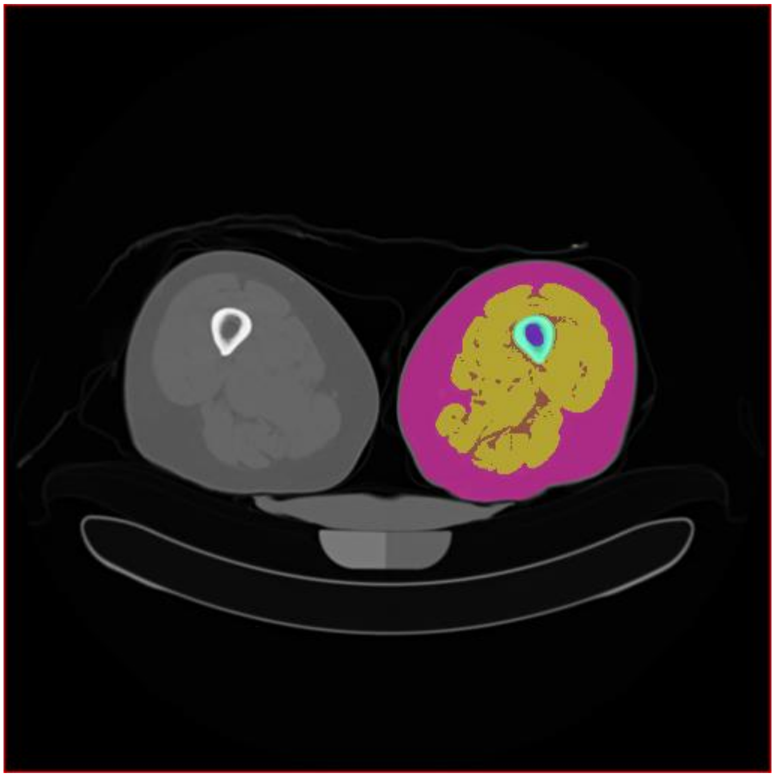


Figure 9 Segmentation of middle thigh: subcutaneous fat, inter-muscular fat, muscle, cortical femur, femur interior.

6.1.5 Computed Features for each Tissue Type

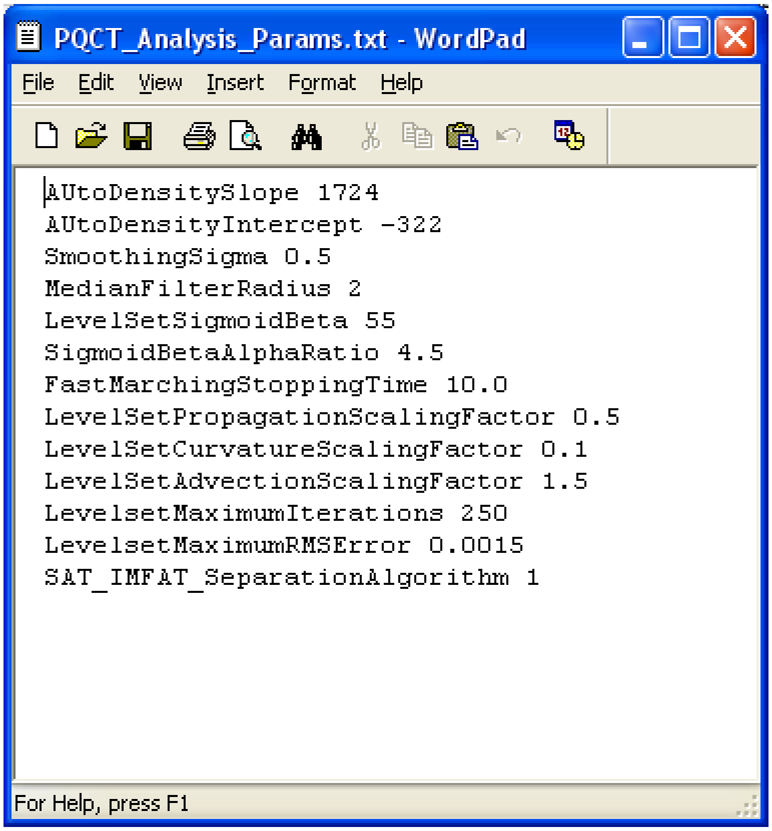
Material: average density, standard deviation of density.

Structural: area, principal moments in 2D, equivalent radius.

7 Appendix

7.1.1 Parameter file

AUtoDensitySlope: Slope parameter in conversion from attenuation units to density values.
AUtoDensityIntercept: Intercept parameter in conversion from attenuation units to density values.
SmoothingSigma: Standard deviation of Gaussian blurring filter before computation of gradient in level sets.
MedianFilterRadius: Radius of median denoising filter.
LevelSetSigmoidBeta: Alpha parameter of Sigmoid function used to compute speed image in GAC segmentation.
SigmoidBetaAlphaRatio: Beta/Alpha parameter ratio of Sigmoid function used to compute speed image in GAC segmentation.
FastMarchingStoppingTime: Stopping time in fast marching contour initialization used in 4% tibia segmentation,
LevelSetPropagationScalingFactor: Propagation scaling factor in GAC segmentation.
LevelSetCurvatureScalingFactor: Curvature scaling factor in GAC segmentation.
LevelSetAdvectionScalingFactor: Advection scaling factor in GAC segmentation.
LevelsetMaximumIterations: Maximum iterations in GAC segmentation.
LevelsetMaximumRMSError: Maximum rms error in level set propagation used as a convergence criterion in GAC segmentation.
SAT_IMFAT_SeparationAlgorithm: 1: connected component analysis, 2: GAC


PQCT_Analysis_Params.txt - WordPad
File Edit View Insert Format Help
For Help, press F1

```
AUtoDensitySlope 1724
AUtoDensityIntercept -322
SmoothingSigma 0.5
MedianFilterRadius 2
LevelSetSigmoidBeta 55
SigmoidBetaAlphaRatio 4.5
FastMarchingStoppingTime 10.0
LevelSetPropagationScalingFactor 0.5
LevelSetCurvatureScalingFactor 0.1
LevelSetAdvectionScalingFactor 1.5
LevelsetMaximumIterations 250
LevelsetMaximumRMSError 0.0015
SAT_IMFAT_SeparationAlgorithm 1
```

Figure 10 The default parameter file as opened by WordPad. If values are edited (with caution) the TIDAQ will read the modified values every time that a new analysis is executed.

7.1.2 Identified Tissue Types

"AIR":	Air
"FAT":	Fat
"MUSCLE":	Muscle
"COR_BO":	Cortical Tibia Compartment
"BO_INT":	Bone Interior
"SUB_FA":	Subcutaneous Fat
"IM_FA":	Inter-muscular Fat
"BO_4%":	4% Tibia-Whole Region
"BO_4%50%":	4% Tibia-50% region around centroid
"BO_4%10%":	4% Tibia-10% region around centroid
"TOT_AR":	Total Cross-sectional Leg Area

7.1.3 Structural and Material Measures

“Area(mm^2)”:	Area in squared millimeters
“Princ.Mom.1”:	1st principal moments component
“Princ.Mom.2”:	2nd principal moments component
“Eq.Radius”:	Radius of equal area circle
“Den.M.”:	Density Mean
“Den.SD.”:	Density Standard Deviation

7.1.4 Acronyms

pQCT: peripheral Quantitative Computed Tomography

GAC: Geodesic Active Contours

ITK: Insight toolkit.

8 References

[1] Guglielmi, G.; Schneider, P.; Lang, T. F.; Giannatempo, G. M.; Cammisa, M. & Genant, H. K. (1997), 'Quantitative computed tomography at the axial and peripheral skeleton.', *Eur Radiol* **7 Suppl 2**, S32--S42.

[2] Rüegsegger, P.; Elsasser, U.; Anliker, M.; Gnehm, H.; Kind, H. & Prader, A. (1976), 'Quantification of bone mineralization using computed tomography.', *Radiology* **121**(1), 93--97.

[3] http://www.itk.org/

[4] http://www.java.com/en/

[5] http://java.sun.com/docs/books/jni/

[6] http://mipav.cit.nih.gov/

[7] Shapiro, L. G.; Stockman, G. C.; Shapiro, L. G. & Stockman, G. (2001), *Computer Vision*, Prentice Hall.

[8] Jain, A. K.; Murty, M. N. & Flynn, P. J. (1999), 'Data clustering: a review', *ACM Comput. Surv.* **31**(3), 264--323.

[9] Sethian, J. A. (1996), 'A fast marching level set method for monotonically advancing fronts', *Proceedings of the National Academy of Sciences*, 1591--1595.

[10] Caselles, V.; Kimmel, R. & Sapiro, G. (1997), 'Geodesic Active Contours', *International Journal of Computer Vision* **22**(1), 61--79.