Michalski, Andrew
March 25, 2020

Document Overview

This document is meant to provide an overview of using the OGO Python scripts for performing opportunistic CT screening. The opportunistic screening pipeline includes image segmentation, internal density calibration, bone mineral density (BMD) analysis, finite element (FE) model generation, FE model post-processing, and results tabulation. Each of these steps will be discussed below and how to implement the analyses. This bone analysis pipeline is set up to analyze abdominal CT scans, which includes the right proximal femur, left proximal femur, right pelvis, left pelvis, sacrum, L5 vertebra, L4 vertebra, L3 vertebra, L2 vertebra, and L1 vertebra. All CT scan images can be performed using DICOM or NIFTI file formats. For purposes here, we use the NIFTI file format, as it is a single image file, rather than a directory of multiple files. All example analyses use images from the OGO_Example_Data directory included with the analysis scripts.

Included in the OGO Python scripts are some additional scripts for performing QCT phantom-based density calibration, where different scripts are for different calibration phantoms. These details are at the end of this document.

Relevant File Naming in Example Images

OGO_Abdomen_Example.nii – This is an example abdominal CT scan, converted to NIFTI format from a DICOM image sequence.

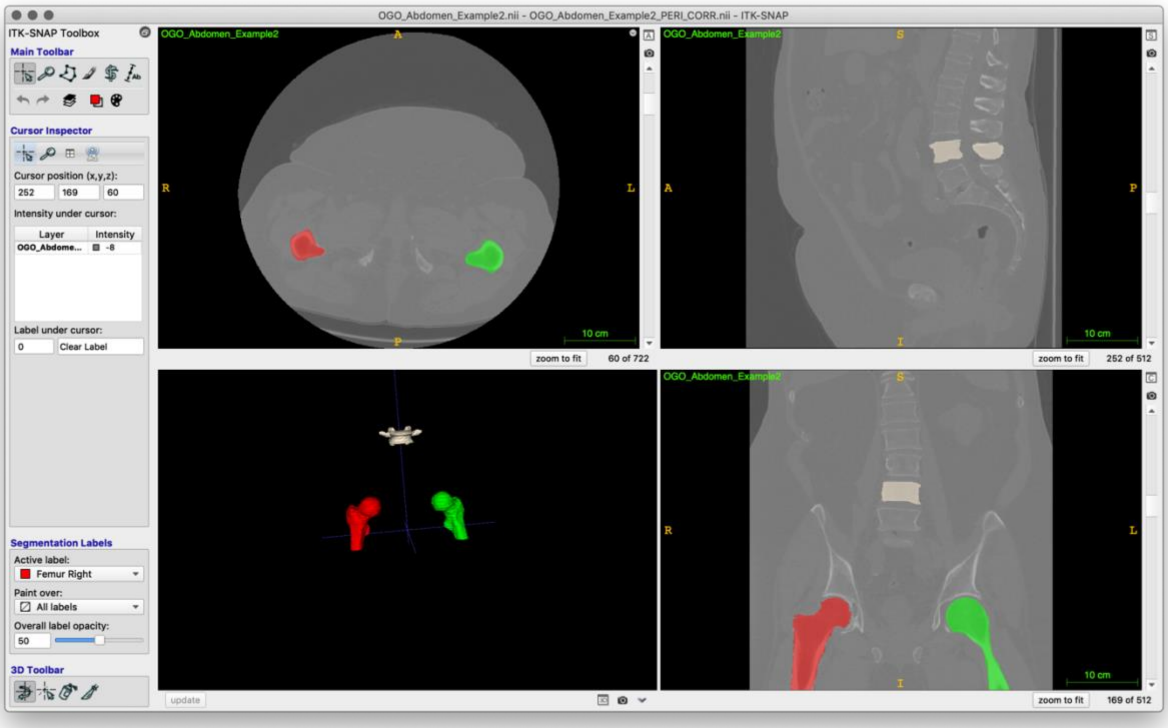
OGO_Abdomen_Example_PERI_CORR.nii – This is the segmentation MASK image, where each different label is a specific bone and a value of 0 is the background.

OGO_Abdomen_Example_IC_MASK.nii – This is the internal density calibration MASK image, where each different label is a specific region-of-interest for the calibration and a value of 0 is the background.

Step 1: Image segmentation to identify bone of interest

Image segmentation can be performed using a number of different tools. For purposes here, we use manual image segmentation in ITK-SNAP. The main necessary component is to stay consistent with the proper bone labels (see included labels text file: 'Bone_ITKSNAP_Labels.txt'). These bone labels are used for all analyses when applying the MASK image.

Label Value	**Description**
0	Clear Label
1	Femur Right
2	Femur Left
3	Pelvis Right
4	Pelvis Left
5	Sacrum
6	L5
7	L4
8	L3
9	L2
10	L1
11	Background
255	Label 255

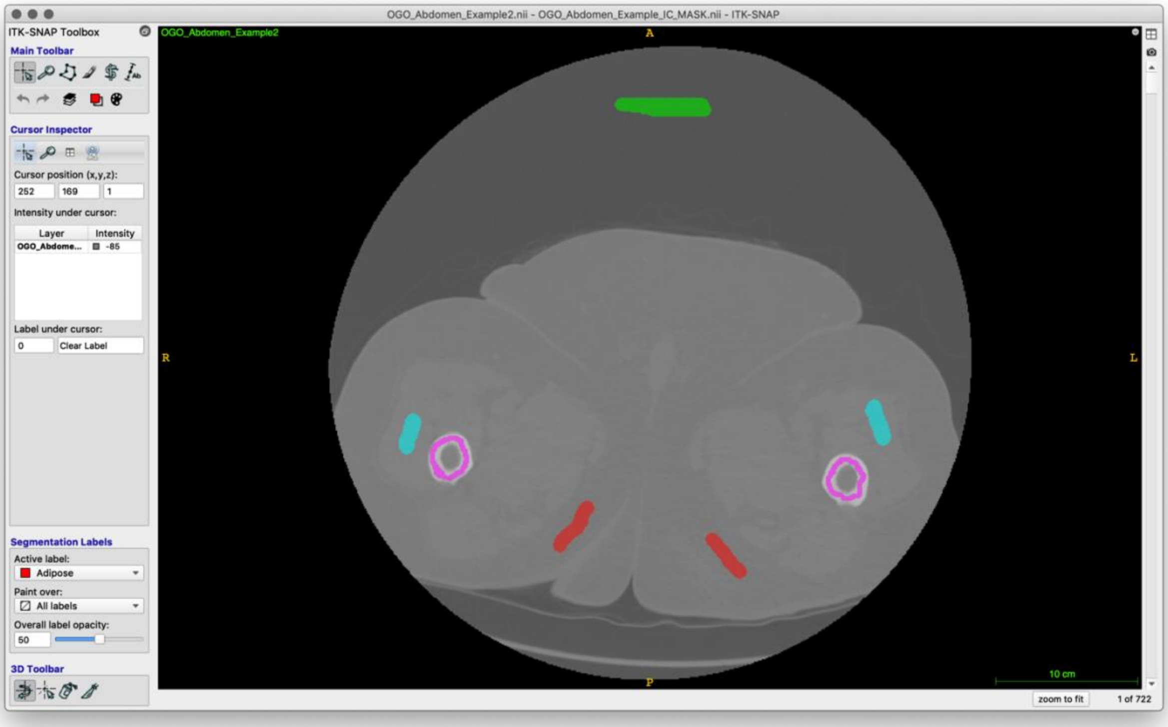


Above is an example screenshot of ITK-SNAP in which the right proximal femur, left proximal femur, and L4 vertebra are segmented (using the example datasets). The output of the image segmentation is provided in NIFTI file format. Output: *OGO_Abdomen_Example_PERI_CORR.nii*

Step 2: Internal density calibration

The internal density calibration step uses the OGO script *ogo_internal_calibration.py*. This script reads in the base image and the internal density calibration MASK image. The internal density calibration MASK image (*OGO_Abdomen_Example_IC_MASK.nii*) is currently generated in ITK-SNAP by manual segmentation, using the provided label files (*Internal-Calibration_ITKSNAP_Labels.txt*).

Label Value	Description
0	Clear label
1	Adipose
2	Air
3	Blood
4	Cortical Bone
5	Skeletal Muscle



Above is an example screenshot of the region definitions using ITK-SNAP. From ITK-SNAP, save the calibration mask as **_IC_MASK.nii* (i.e., *OGO_Abdomen_Example_IC_MASK.nii*).

The internal calibration script is run from a terminal. Activate your analysis environment and change to the OGO script directory. To run the internal calibration, use the command prompt as follows:

$ python ogo_internal_calibration PATH/TO/OGO_Example_Data/OGO_Abdomen_Example.nii PATH/TO/OGO_Example_Data/OGO_Abdomen_Example_IC_MASK.nii

OUTPUTS:

1. **_IntCalibParameters.txt* – Text file containing all of the internal density calibration parameters, as seen below.

OGO_Abdomen_Example_IntCalibParameters.txt

```
ID      OGO_Abdomen_Example
Output File     OGO_Abdomen_Example_IC_K2HPO4.nii
Python Script   ogo_internal_calibration.py
Version 1.0
Date Created    2020-03-25
Image Directory /Users/andrew.michalski/Desktop/Py3/OGO_Example_Data
Image   OGO_Abdomen_Example.nii
Mask Directory  /Users/andrew.michalski/Desktop/Py3/OGO_Example_Data
Mask    OGO_Abdomen_Example_IC_MASK.nii
+++++   +++++
Effective Energy [keV]  96.0
Max R^2 0.9983404998928974
HU-u/p Slope    1.5474534088249448e-05
HU-u/p Y-Intercept      0.1718058762154702 8
HU-Material Density Slope       0.0008884964716273846
HU-Material Density Y-Intercept 0.9655496613171292
Adipose u/p     0.17104
Air u/p 0.15652
Blood u/p       0.17214000000000002
Cortical Bone u/p       0.19297999999999998
Skeletal Muscle u/p     0.1719
K2HPO4 u/p      0.20792
CHA u/p 0.2133
Triglyceride u/p        0.17042000000000002
Water u/p       0.17329999999999998
```

2.**_IC_K2HPO4.nii* – K_2HPO_4 bone equivalent density calibrated image in NIFTI format. This file is used for BMD analysis and FE model generation.

Step 3: Integral BMD analysis for a specific bone

All integral BMD analyses are performed using the density calibrated image (*OGO_Abdomen_Example_IC_K2HPO4.nii)* and the bone mask image (*OGO_Abdomen_Example_PERI_CORR.nii)*. These analyses are performed using the *ogo_bmd_analysis.py* script in the command prompt. Example command lines are:

1. Integral BMD analysis of right femur:

```
$ python ogo_bmd_analysis.py
/Path/to/OGO_Example_Data/OGO_Abdomen_Example_IC_K2HPO4.nii
/Path/To/OGO_Example_Data/OGO_Abdomen_Example_PERI_CORR.nii --mask_threshold 1
```

2. Integral BMD analysis of left femur:

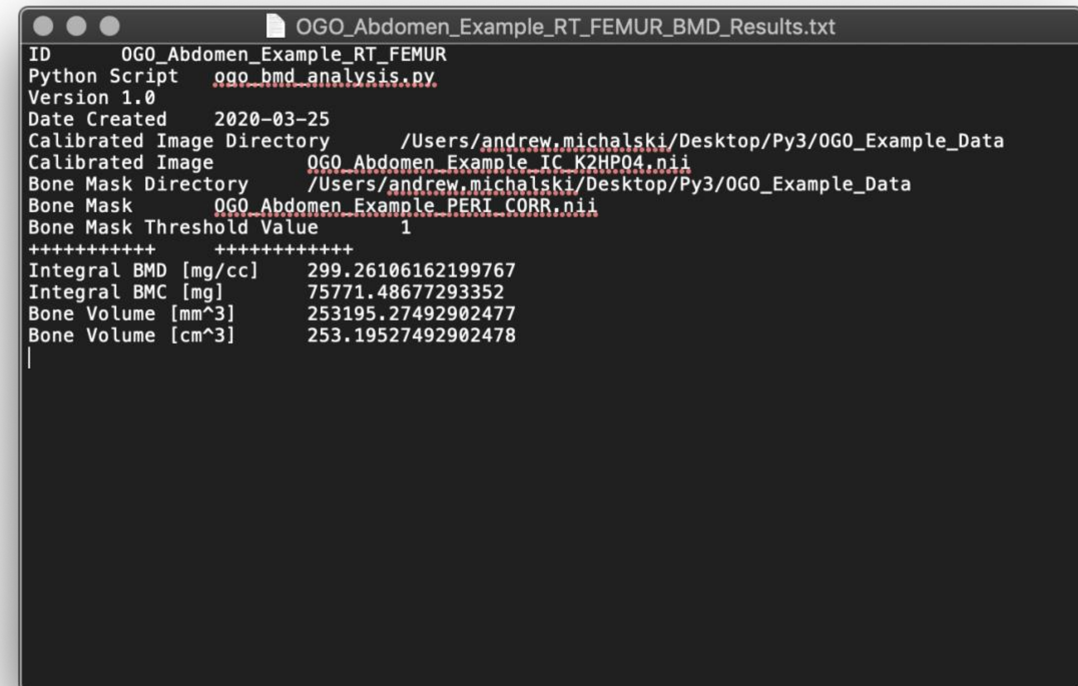
```
$ python ogo_bmd_analysis.py
/Path/to/OGO_Example_Data/OGO_Abdomen_Example_IC_K2HPO4.nii
/Path/To/OGO_Example_Data/OGO_Abdomen_Example_PERI_CORR.nii --mask_threshold 2
```

3. Integral BMD analysis of L4 vertebra:

$ python ogo_bmd_analysis.py
/Path/to/OGO_Example_Data/OGO_Abdomen_Example_IC_K2HPO4.nii
/Path/To/OGO_Example_Data/OGO_Abdomen_Example_PERI_CORR.nii --mask_threshold 7

OUTPUT:

1. *_RT_FEMUR_BMD_RESULTS.txt; *_LT_FEMUR_BMD_RESULTS.txt; OR *_L4_BMD_RESULTS.txt – Text file containing the integral BMD analysis for the specific bone, as shown below.



OGO_Abdomen_Example_RT_FEMUR_BMD_Results.txt

```
ID	OGO_Abdomen_Example_RT_FEMUR
Python Script	ogo_bmd_analysis.py
Version 1.0
Date Created	2020-03-25
Calibrated Image Directory	/Users/andrew.michalski/Desktop/Py3/OGO_Example_Data
Calibrated Image	OGO_Abdomen_Example_IC_K2HPO4.nii
Bone Mask Directory	/Users/andrew.michalski/Desktop/Py3/OGO_Example_Data
Bone Mask	OGO_Abdomen_Example_PERI_CORR.nii
Bone Mask Threshold Value	1
+++++++++++	++++++++++++
Integral BMD [mg/cc]	299.26106162199767
Integral BMC [mg]	75771.48677293352
Bone Volume [mm^3]	253195.27492902477
Bone Volume [cm^3]	253.19527492902478
```

Step 4: FE model generation

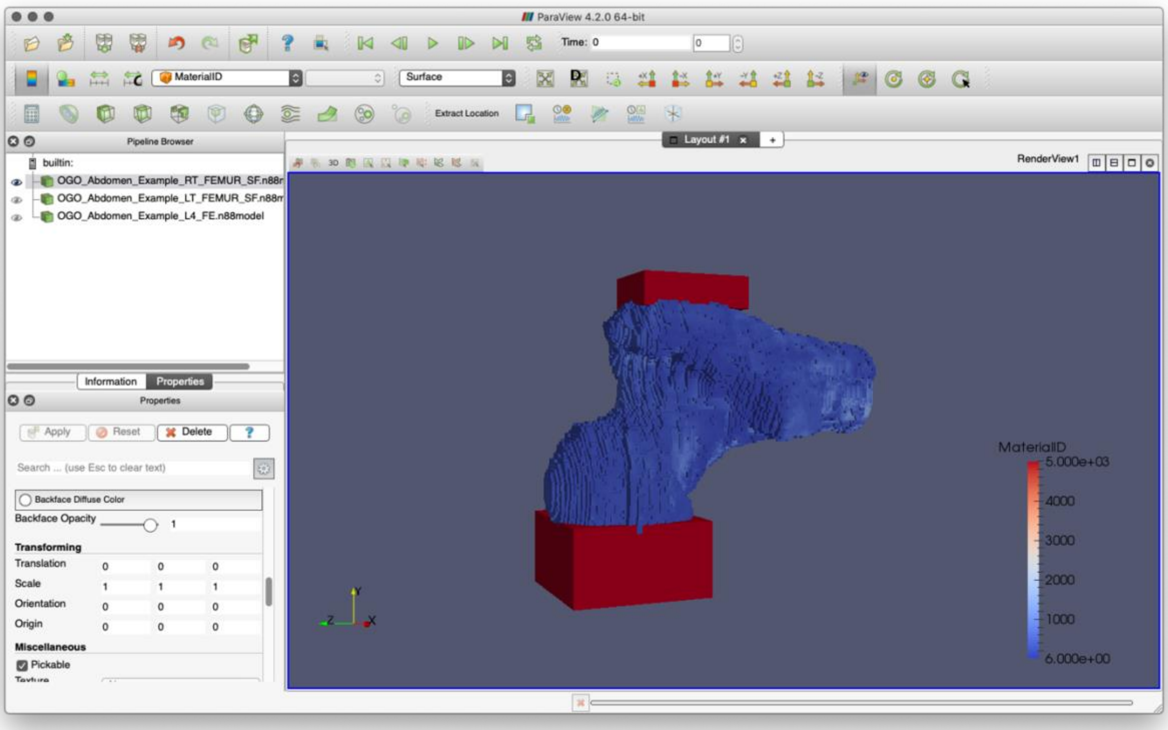
The provided OGO scripts will generate an FE model in either a sideways fall on the hip (for proximal femurs) or a vertebral body compression (for lumbar vertebra). The scripts expect a K_2HPO_4 density image, so if using a HA density image some conversion equations will require adjustment to properly generate the FE model. For the sideways fall models, PMMA caps are added to the femoral head and greater trochanter. For the vertebral compression models, PMMA caps are added to the inferior and superior vertebral body faces.

Sideways fall models are generated using the *ogo_sideways_fall_fe.py* script, and vertebral compression models are generated using the *ogo_l4_compression_fe.py* script. There are multiple optional arguments for each model as well. Example command lines are:

1. Right proximal femur sideways fall model:

$ python ogo_sideways_fall_fe.py /Path/To/OGO_Example_Data/OGO_Abdomen_Example_IC_K2HPO4.nii /Path/To/OGO_Example_Data/OGO_Abdomen_Example_PERI_CORR.nii --mask_threshold 1 --femur_side 2

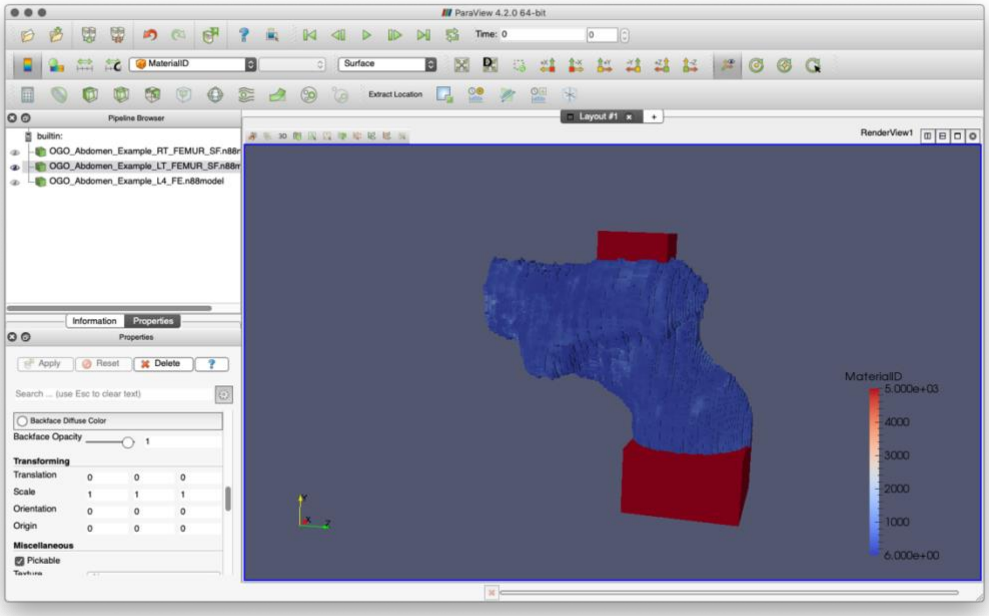
OUTPUT: *_RT_FEMUR_SF.n88model*



2. Left proximal femur sideways fall model:

$ python ogo_sideways_fall_fe.py /Path/To/OGO_Example_Data/OGO_Abdomen_Example_IC_K2HPO4.nii /Path/To/OGO_Example_Data/OGO_Abdomen_Example_PERI_CORR.nii --mask_threshold 2 --femur_side 1

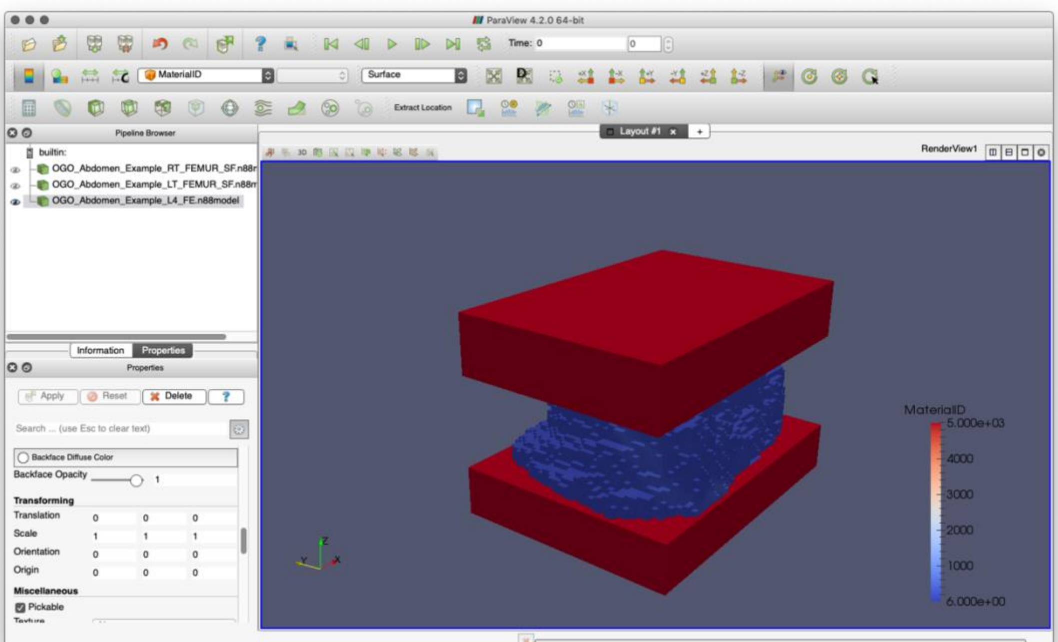
OUTPUT: *_LT_FEMUR_SF.n88model*



3. L4 vertebral compression model

$ python ogo_l4_compression_fe.py
/Path/To/OGO_Example_Data/OGO_Abdomen_Example_IC_K2HPO4.nii
/Path/To/OGO_Example_Data/OGO_Abdomen_Example_PERI_CORR.nii --mask_threshold 7

OUTPUT: * *_L4_FE.n88model*



Step 5: Solve FE models
All FE models can be solved in FAIM (> v8.0) using standard solving methods.

Step 6: FE model post processing
The Pistoia criterion failure load script from FAIM can be used to estimate the FE model failure load. A critical strain value should be 0.011 and a critical volume value of 7.0% should be used to estimate the failure load.

Step 7: Tabulate the BMD and FE Pistoia failure load values in the directory
Typically, multiple models are all solved at one time in large batches. The following scripts will tabulate either the integral BMD or FE estimate failure loads of all the models in the specified directory.

1. Tabulate all integral BMD values for the right proximal femur:
$ python ogo_bmd_tabulate.py --bone 1 /Path/To/Results/Directory

2. Tabulate all integral BMD values for the left proximal femur:
$ python ogo_bmd_tabulate.py --bone 2 /Path/To/Results/Directory

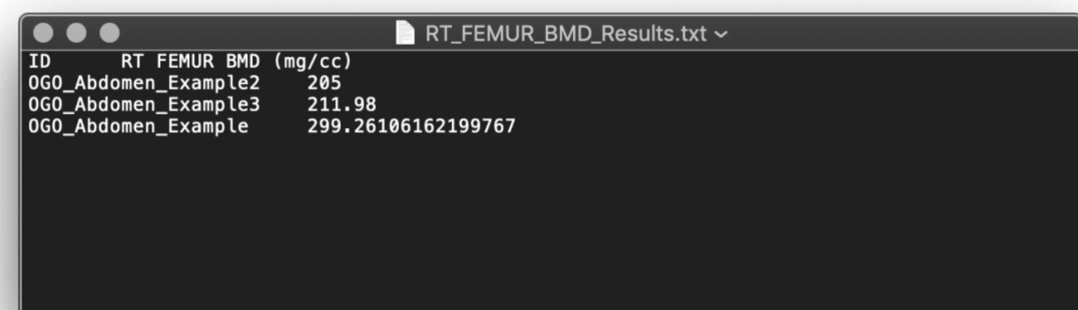
3. Tabulate all integral BMD values for the L4 vertebra:
$ python ogo_bmd_tabulate.py --bone 4 /Path/To/Results/Directory

4. Tabulate all FE estimate failure loads for the right proximal femur:
$ python ogo_fe_pistoia_tabulate.py --model 1 /Path/To/Results/Directory

5. Tabulate all FE estimate failure loads for the left proximal femur:
$ python ogo_fe_pistoia_tabulate.py --model 3 /Path/To/Results/Directory

6. Tabulate all FE estimated failure loads for the L4 vertebra:
$ python ogo_fe_pistoia_tabulate.py --model 6 /Path/To/Results/Directory

OUTPUT: *RT_FEMUR_BMD_Results.txt; LT_FEMUR_BMD_Results.txt;* OR *L4_SPINE_BMD_Results.txt*




```
ID      RT FEMUR BMD (mg/cc)
OGO_Abdomen_Example2    205
OGO_Abdomen_Example3    211.98
OGO_Abdomen_Example     299.26106162199767
```

Additional QCT density calibration methods

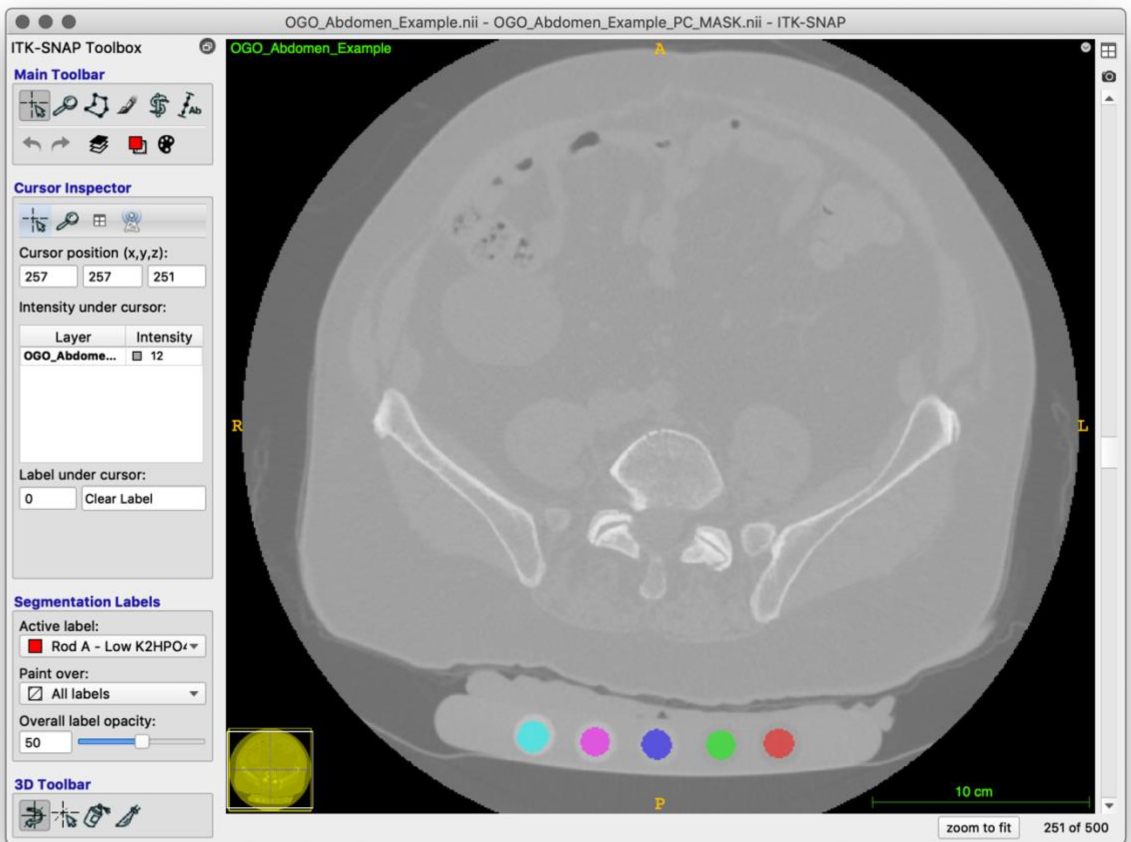
Density calibration phantoms are the gold standard to perform QCT analysis but are generally limited to a research-based setting. Additional scripts are included to create a density calibrated image for some additional calibration phantoms including:

- Mindways Model 3 CT phantom
- Mindways Model 3 CT phantom asynchronous calibration
- B-MAS200 phantom

Adding other density calibration phantoms is relatively trivial and can be done by modifying one of these mentioned scripts.

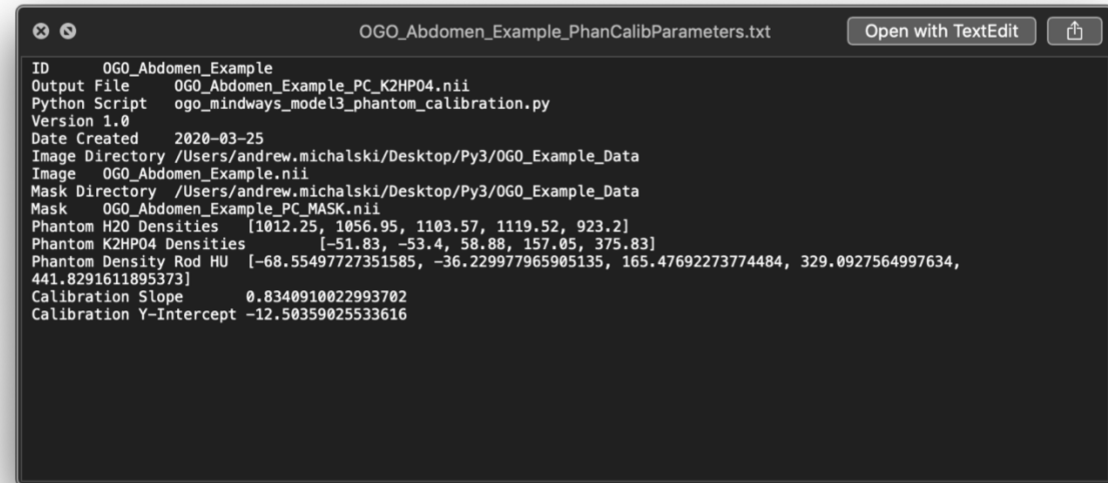
1. Mindways Model 3 CT phantom

The calibration mask image is generated using ITK-SNAP and the provided labels file (*Mindways_Model3_Phantom-Calibration_ITKSNAP_Labels.txt*). Place regions on the calibration rods as depicted below and save file as **_PC_MASK.nii*.



Generate the density calibrated image using the script *ogo_mindways_model3_phantom_calibration.py*.
$ python ogo_mindways_model3_phantom_calibration.py
/Path/To/OGO_Example_Data/OGO_Abdomen_Example.nii
/Path/To/OGO_Example_Data/OGO_Abdomen_Example_PC_MASK.nii

OUTPUTS:
1. **_PhanCalibParameters.txt* – Text file containing the density calibration parameters


OGO_Abdomen_Example_PhanCalibParameters.txt

```
ID      OGO_Abdomen_Example
Output File     OGO_Abdomen_Example_PC_K2HPO4.nii
Python Script   ogo_mindways_model3_phantom_calibration.py
Version 1.0
Date Created    2020-03-25
Image Directory /Users/andrew.michalski/Desktop/Py3/OGO_Example_Data
Image   OGO_Abdomen_Example.nii
Mask Directory  /Users/andrew.michalski/Desktop/Py3/OGO_Example_Data
Mask    OGO_Abdomen_Example_PC_MASK.nii
Phantom H2O Densities   [1012.25, 1056.95, 1103.57, 1119.52, 923.2]
Phantom K2HPO4 Densities        [-51.83, -53.4, 58.88, 157.05, 375.83]
Phantom Density Rod HU  [-68.55497727351585, -36.229977965905135, 165.47692273774484, 329.0927564997634,
441.8291611895373]
Calibration Slope       0.8340910022993702
Calibration Y-Intercept -12.50359025533616
```

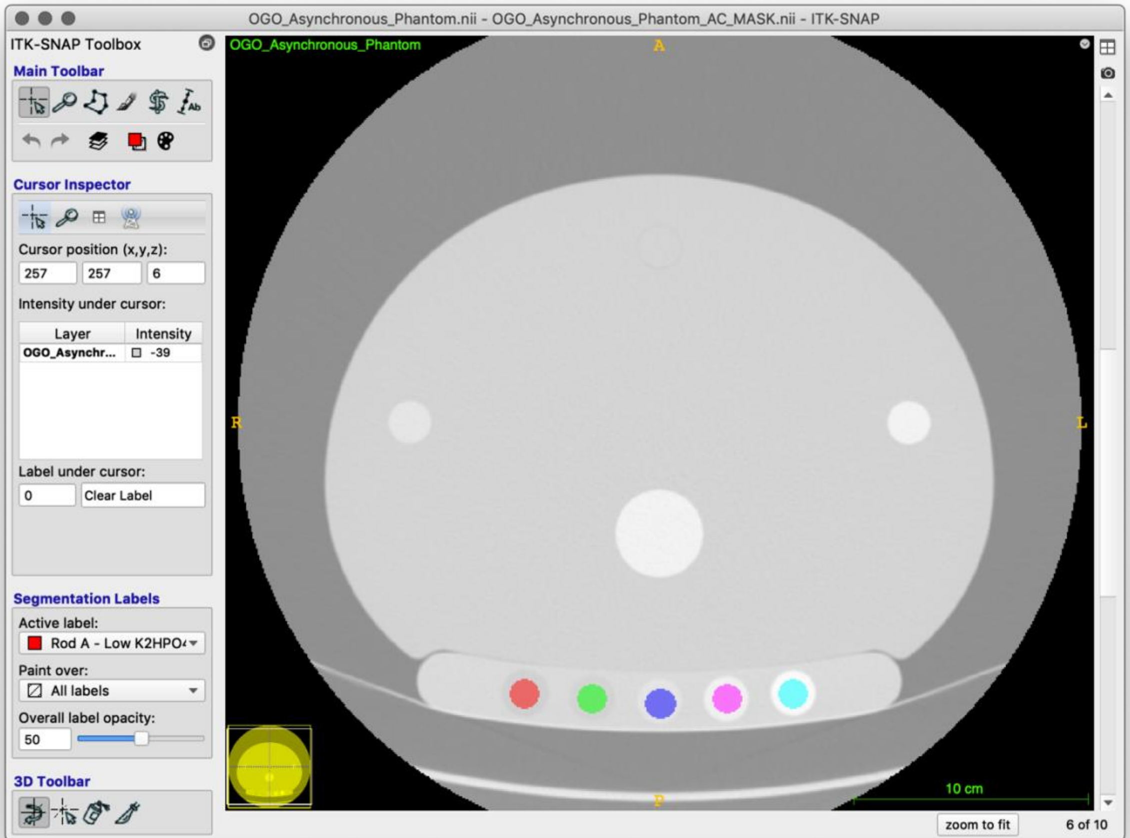
2. * *_PC_K2HPO4.nii* – Image file of the K_2HPO_4 density calibrated image.

All other analyses are the same as previously stated.

2. Mindways Model 3 Asynchronous calibration

The calibration mask image is generated using ITK-SNAP and the provided labels file (*Mindways_Model3_Phantom-Calibration_ITKSNAP_Labels.txt*). Place regions on the calibration rods as depicted below and save file as **_AC_MASK.nii*.

Generate the density calibrated image using the script *ogo_mindways_model3_asynchronous_calibration.py*.
$ python ogo_mindways_model3_asynchronous_calibration.py /Path/To/CT_Scan.nii /Path/To/AsynchronousScan.nii /Path/To/AsynchronousScan_AC_MASK.nii



OUTPUTS:

1. *_AsyncCalibParameters.txt* – Text file containing the asynchronous calibration information

```
OGO_Abdomen_Example_AsyncCalibParameters.txt
ID      OGO_Abdomen_Example
Output File     OGO_Abdomen_Example_AC_K2HP04.nii
Python Script   ogo_mindways_model3_asynchronous_calibration.py
Version 1.0
Date Created    2020-03-25
Image Directory /Users/andrew.michalski/Desktop/Py3/OGO_Example_Data
Image   OGO_Abdomen_Example.nii
Phantom Directory       /Users/andrew.michalski/Desktop/Py3/OGO_Example_Data
Phantom OGO_Asynchronous_Phantom.nii
Mask Directory  /Users/andrew.michalski/Desktop/Py3/OGO_Example_Data
Mask    OGO_Asynchronous_Phantom_AC_MASK.nii
Phantom H2O Densities   [1012.25, 1056.95, 1103.57, 1119.52, 923.2]
Phantom K2HP04 Densities        [-51.83, -53.4, 58.88, 157.05, 375.83]
Phantom Density Rod HU  [-68.47065868263473, -33.2821341956346, 192.10648518815051, 371.4762096774194,
506.91895161290324]
Calibration Slope       0.7411462992195103
Calibration Y-Intercept -1.625905421212382
```

2. *_AC_K2HPO4.nii - Image file of the K_2HPO_4 density calibrated image.