CraniumPy

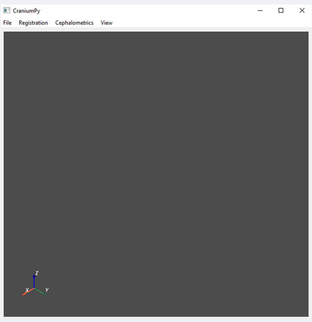
T. Abdel-Alim

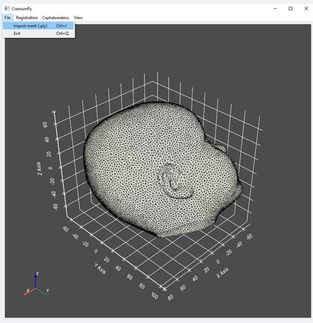
Import mesh

File > Import mesh

Default file format: .ply
Tested with .ply, .stl, and .obj

All pre-processed images are saved as .ply objects.





Mesh registration:

Registration > Enable picking (ctrl+P)

With picking enabled, hover over one of the following three landmarks and press P to toggle a red pointer

Landmarks:

a. Nasion: ctrl+N to confirm
b. Left Tragus ctrl+L to confirm
c. Right Tragus ctrl+R to confirm

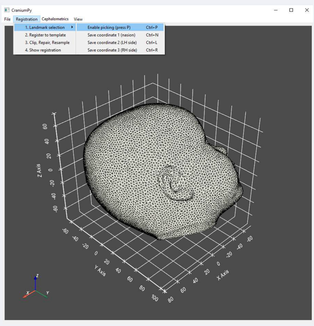
After confirmation, the red dot will turn green and the landmark coordinates will appear in the top right corner.

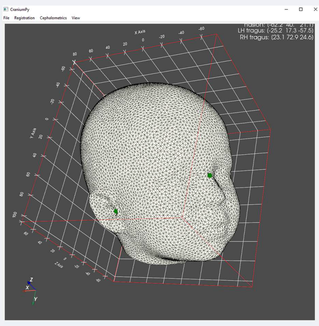
When the three landmarks have been confirmed (3 green dots) register the mesh to the template (in orange):

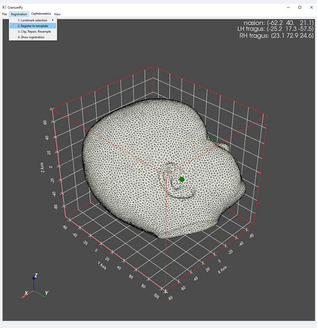
Registration > Register to template

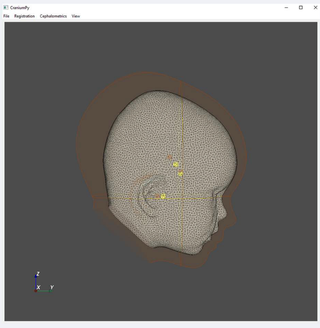
The registration uses the centroid of the three landmarks (translation) and the normal vectors (rotation).

Registered mesh saved as:
“filename_rg.ply”









Mesh clipping and optimization:

Registration > Clip, Repair, Resample

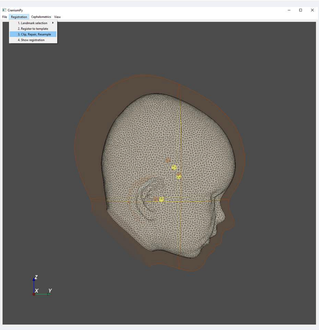
This executes the following three operations:

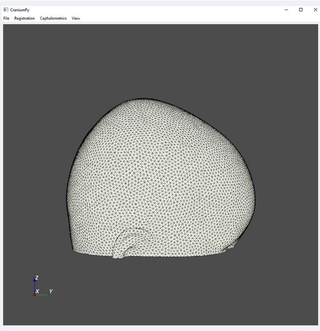
1. Clip the mesh along the nasion-tragus plane
2. Holes and mesh artifacts are patched (pymeshfix)
3. The mesh is resampled using voronoi clustering (pyacvd) default $n_{vertices}$=10.000

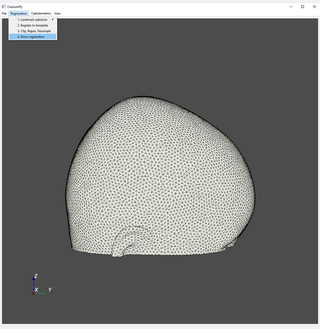
Cranial mesh (clipped) saved as:
“filename_rg_C.ply”

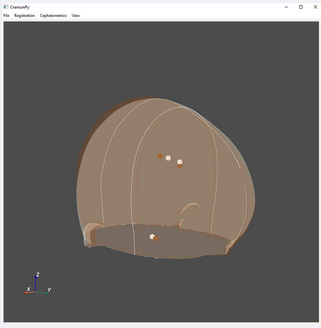
The registration of the final cranial mesh can be visualized using:

Registration > Show registration









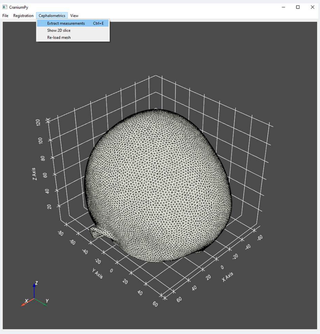
Extracting cranial measurements:

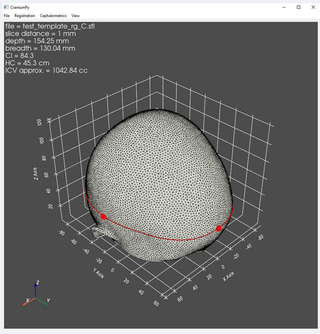
Cephalometrics > Extrct measurements

This function returns the following cephalometric measurements:

- head depth
- head breadth
- cephalic index
- head circumference
- volume

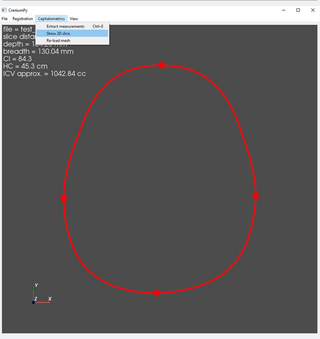
These measurements are calculated in a single transverse slice, which is the slice at which the head depth is highest.





This transverse slice can also be visualized by itself:

Cephalometrics > Show 2D slice



The cranial mesh is returned using:

Cephalometrics > Re-load mesh