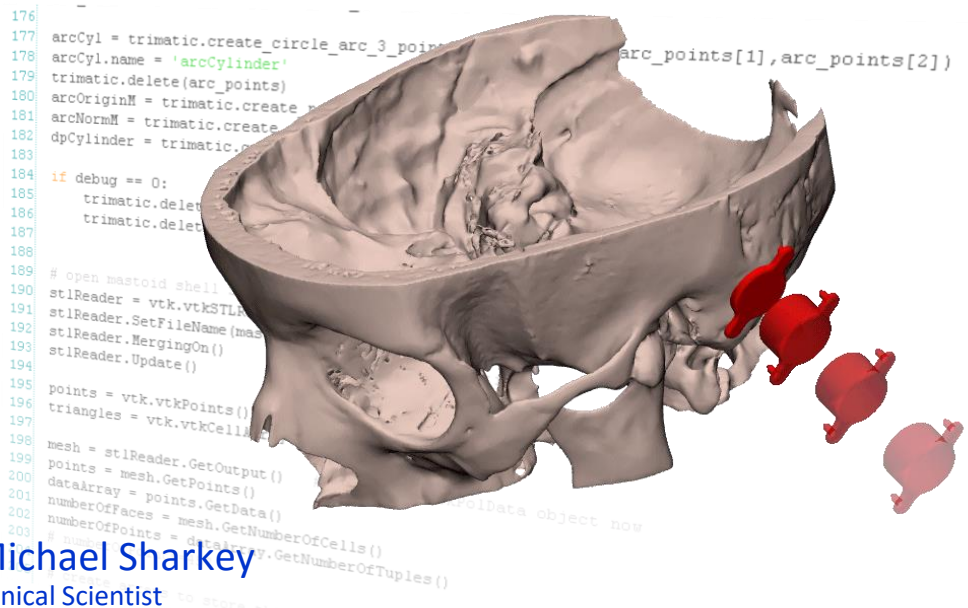
Automatically fitting devices and performing measurements of post-operative facial reconstruction through scripting



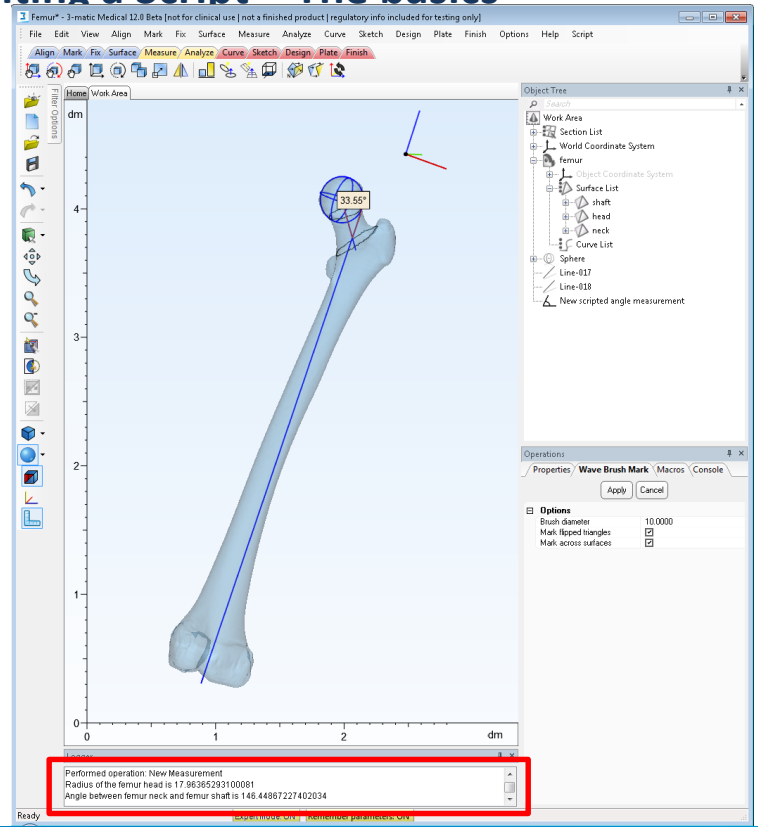
Pete Metherall & Michael Sharkey
Clinical Scientist Clinical Scientist

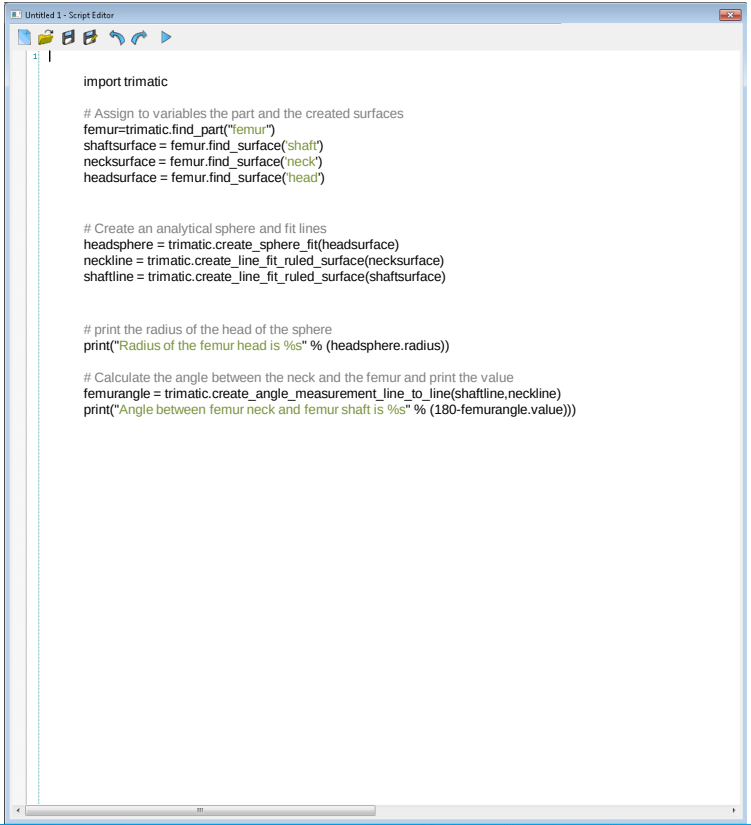
Directorate of Medical Imaging and Medical Physics,
Sheffield Teaching Hospitals, United Kingdom.

3DLAB

Sheffield Teaching Hospitals NHS
NHS Foundation Trust

Writing a Script – The basics

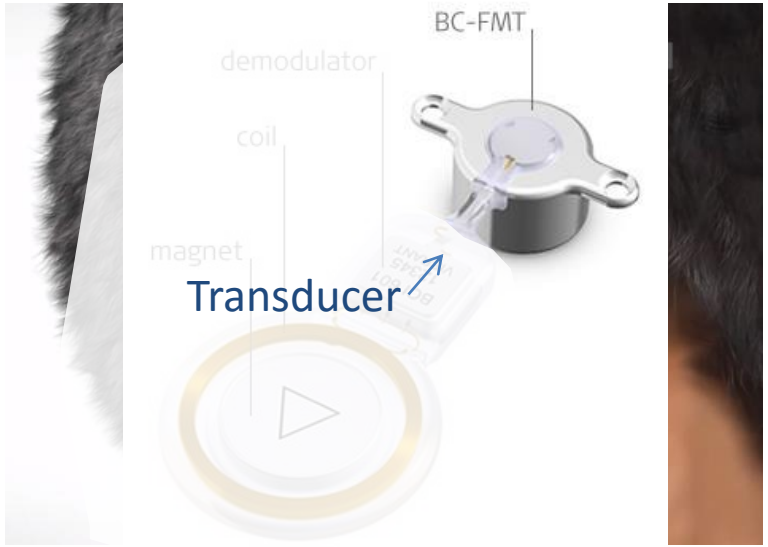




Case Study 1 - Automatically fitting a bone conduction implant

ENT Surgical Planning
Bone Conduction Implant

- Bone conduction implant that transmits sound waves via the cranial bone directly to the inner ear.
- The floating mass transducer is buried into the temporal bone.
- Requirement to ensure there is sufficient bone to accommodate the device.

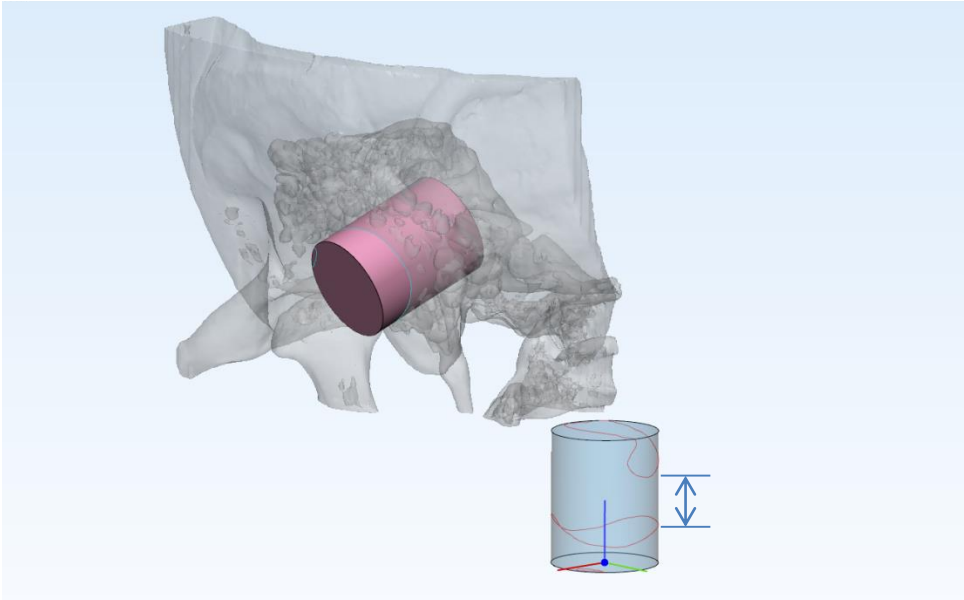


http://www.medel.com/uk/bonebridge/

Case Study 1 - Automatically fitting a bone conduction implant

Determine the direction of the implant:

↳ Move cylinder back to the origin & analyse intersection curves



Curve Analysis

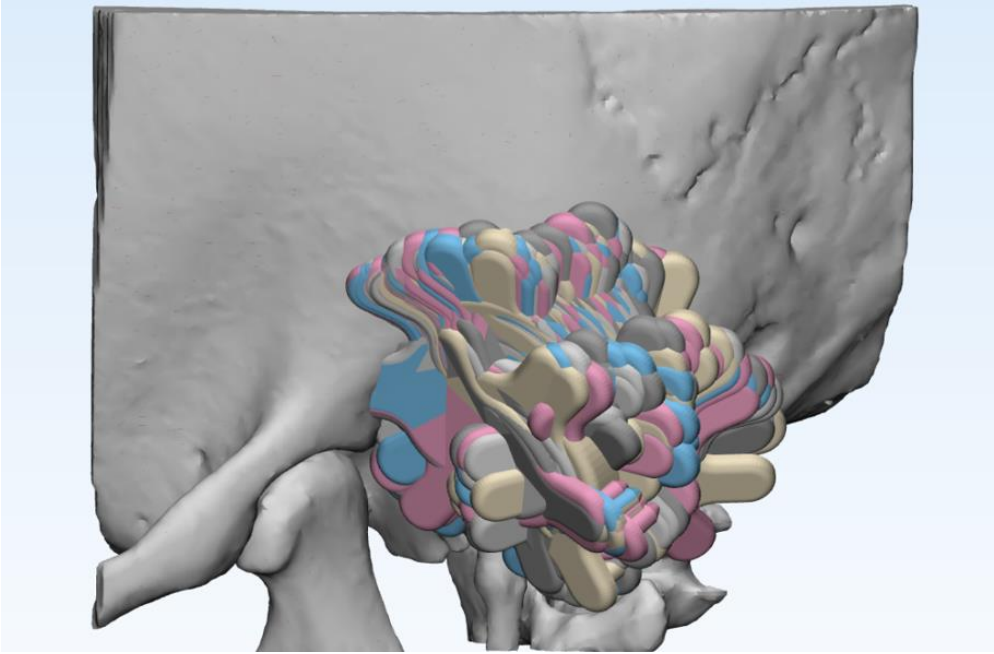
- Determine which curve is at the external bone surface (could be multiple curves)
- Measure distance from mean external height to minimum height of internal surface curve

↳ Bone thickness

Does the device fit

Case Study 1 - Automatically fitting a bone conduction implant

Full Analysis:



Device fitting

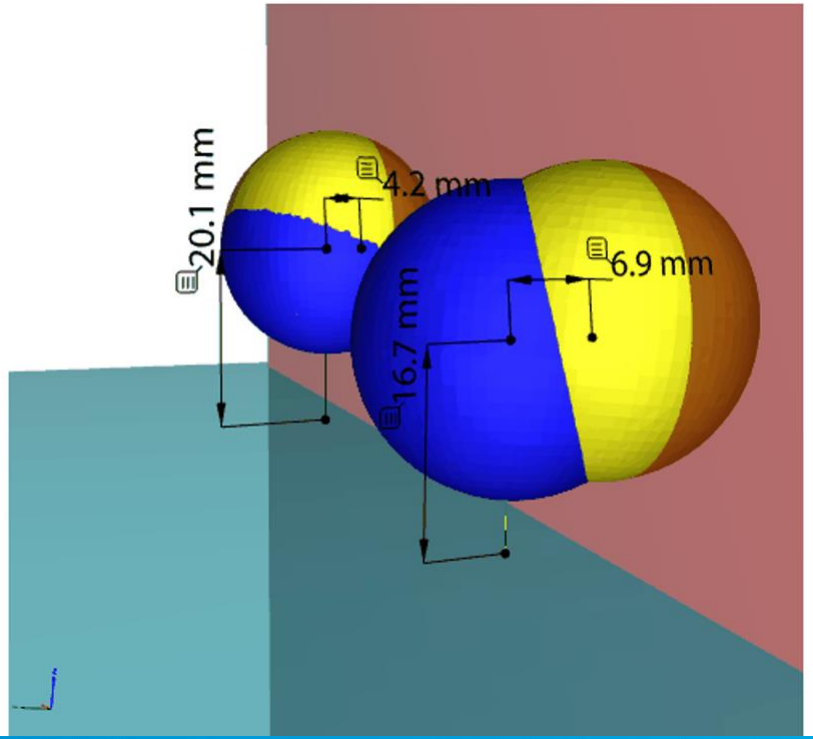
- Analysis performed at every position
- 1000 nodes ~ 2.5h hours
- Which is the best position?
- Score thickness & surface variation

Case Study 2 - Pre and post surgical measurements for enopthalmos evaluation

Maxio-facial reconstruction following trauma to the eye socket

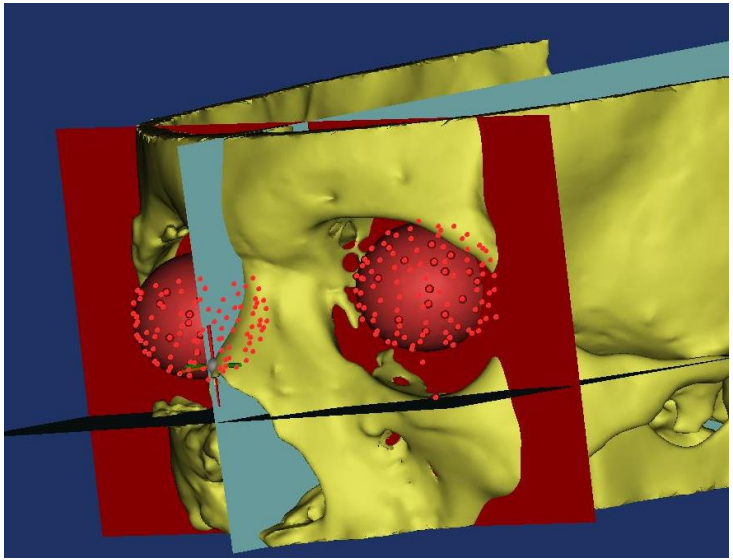
Surgical Implant inserted to re-align the displaced orbit

- Pre-surgical cone-beam CT (CBCT) scan to determine the extent of the displacement
- Post-surgical CT scan to determine the final position of the orbit
- Can we perform a CT enopthalmos evaluation and determine the pre to post displacement?



Case Study 2 - Pre and post surgical measurements for enopthalmos evaluation

Mimics – Manually define anatomical planes & export as 3D structures



```
147
148 def definePlanes():
149     '''
150     Defines points on the Frankfurter plane, the sagittal midline plane and a coronal frontal plane b
151     '''
152
153     pt_list = np.zeros((4, 3), dtype=np.float32)
154     print('Select points in the following order:')
155     print('    1) The left external auditory canal')
156     print('    2) The right external auditory canal')
157     print('    3) The inferior orbital rim on the undamaged side')
158     print('    4) The lateral orbital rim on the undamaged side')
159
160     lt_pt = mimics.analyze.indicate_point(title='Left external auditory canal', message='Please indicate point')
161     pt_list[0] = lt_pt.coordinates
162     rt_pt = mimics.analyze.indicate_point(title='Right external auditory canal', message='Please indicate point')
163     pt_list[1] = rt_pt.coordinates
164     orbital_pt = mimics.analyze.indicate_point(title='Undamaged inferior orbital rim', message='Please indicate point')
165     pt_list[2] = orbital_pt.coordinates
166     lat_pt = mimics.analyze.indicate_point(title='Undamaged lateral orbital rim', message='Please indicate point')
167     pt_list[3] = lat_pt.coordinates
168
169     return pt_list
```

Conclusions

- Scripting in Mimics and 3-matic offers great opportunities to automate processing
- Speeds up repetitive tasks
- Improves reproducibility
- Enables tasks which would be otherwise impossible