
DTU

DTU

02507 Project work within Image Analysis and Computer Graphics

Virtual Cochlea Unfolding

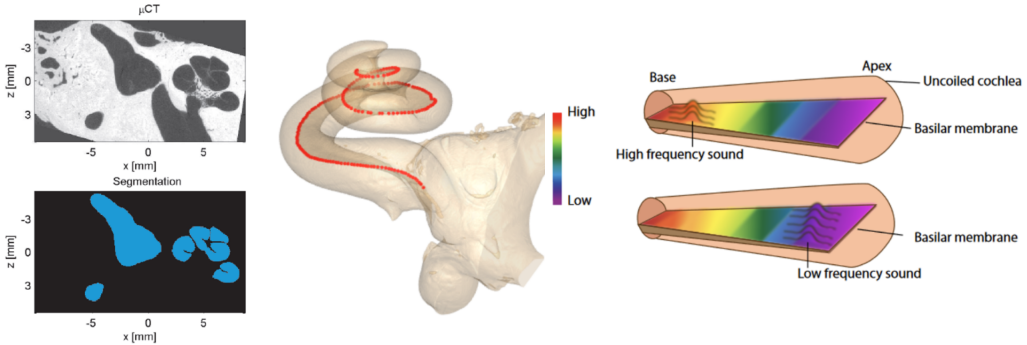
Authors

Clara Mejlhede Lorenzen - s180350
Aleksandar Lukic - s194066
Freja Christine Slot Hansen - s204493
Victoria Charlotte Ipsen - s184048
Caroline Ankjær Andersen - s175201

Supervisor

Hans Martin Kjer

Motivation

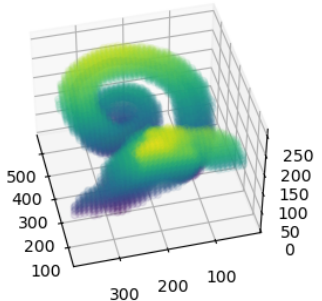


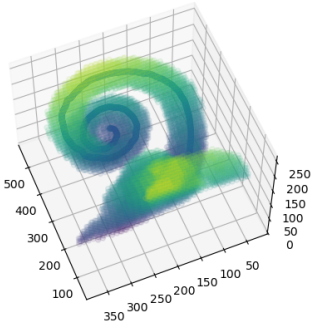
The illustration of the project data is taken from the project description.

Our approach

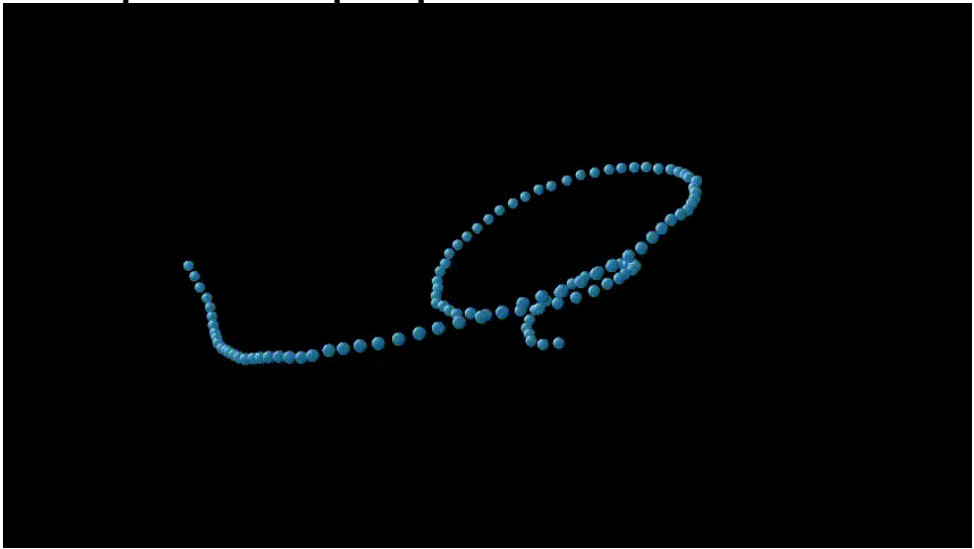
1. Data transformation
2. Interpolation of spiral data points
3. Placement of intersecting planes
4. Extracting data from the 3D data set
5. Visualising slices in a new NIfTI file
6. Mapping frequencies onto unfolded cochlea

1. Data transformation

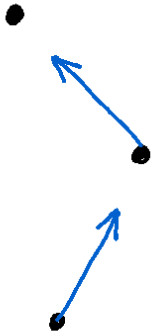


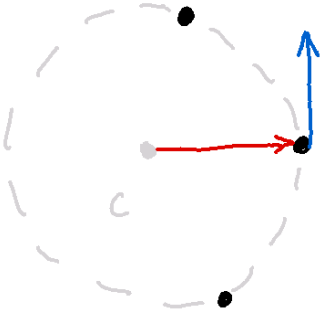


2. Interpolation of spiral points



2. Interpolation of spiral data points

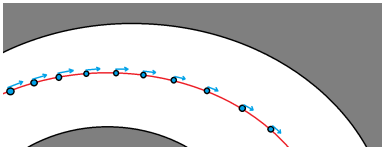


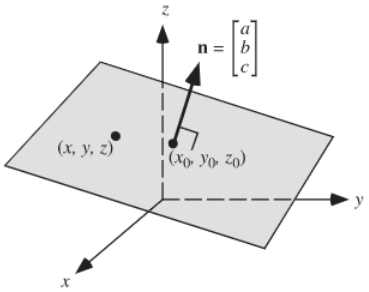


3. Placement of intersecting planes

- Traversing the cochlea following the centred spiral points
- Slices along the direction of the cochlear curvature
- The planes are orthogonal with the normal vectors

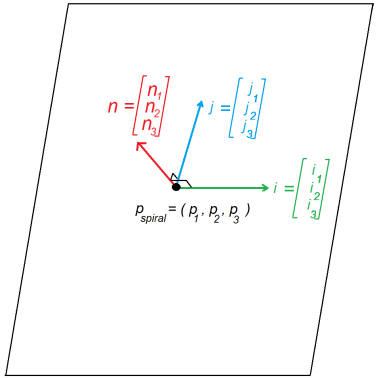
Cochlear curvature





3. Placement of intersecting planes

- To prevent torsion, we fix the slices to the z-axis, when finding $\hat{i}$.
- Any spiral point can be given as: $P_{spiral} = \begin{pmatrix} p_1 & p_2 & p_3 \end{pmatrix}$
- $\hat{i} = n \times \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T$ normalised
- $\hat{j} = n \times \hat{i}$ normalised
- With the spiral point as centerpoint and the two normalised vectors, we can find all points in the plane.



3. Placement of intersecting planes

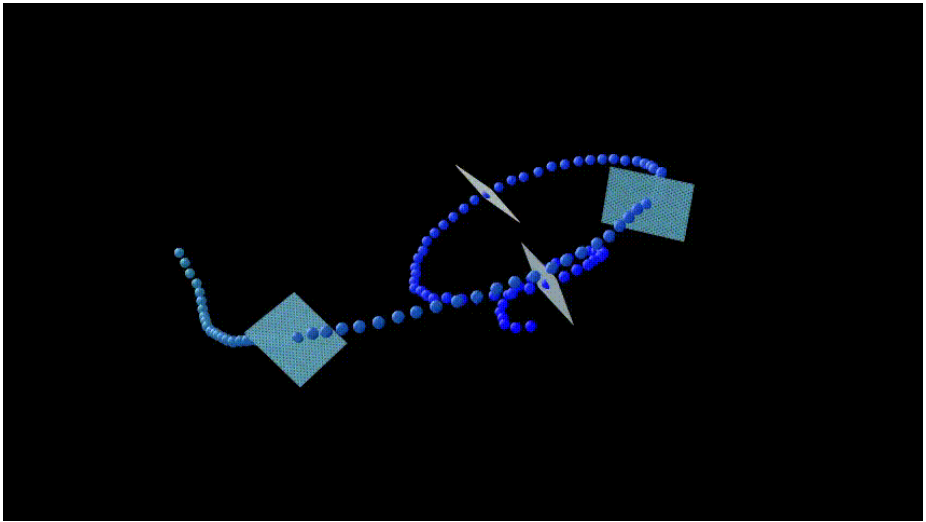
The following parametric equation is given for any plane from which a slice can be extracted:

$$P_{plane} = P_{spiral} + u\,\hat{i} + v\,\hat{j} \tag{1}$$

where:

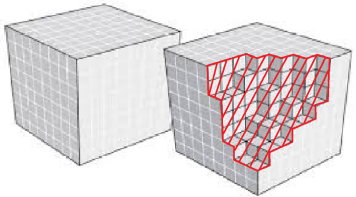
- P_{plane} - calculated point in the 3D plane
- P_{spiral} - A given spiral point, acting as centerpoint of the slice
- u, v - free parameters to navigate the plane axes
- $\hat{i} = n \times (0, 0, 1)^T$ normalised
- $\hat{j} = n \times \hat{i}$ normalised

3. Placement of intersecting planes

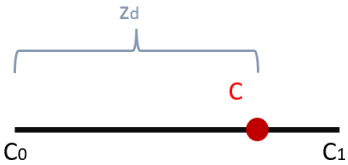


3. Placement of intersecting planes

- Using parametric equations to extract voxels.
- 3D-voxel space is made of whole number coordinates. However, the plane results in positions within the voxel themselves!
- Picking out voxels from the 3D-image, has to minimise data loss and redundancy.
- Achievable via interpolation from 3D to 2D-space.



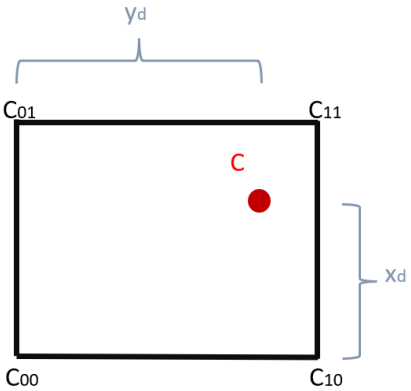
4. Extracting data from the 3D data set



Linear interpolation - 1D

$$c = c_0(1 - z_d) + c_1\ z_d$$

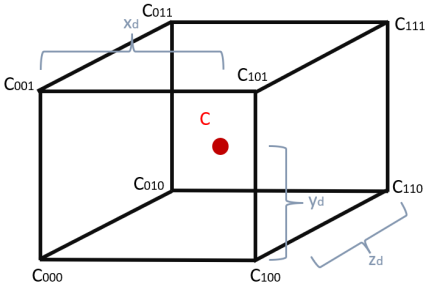
4. Extracting data from the 3D data set



Bilinear interpolation - 2D

$$c_0 = c_{00}(1 - y_d) + c_{10}\, y_d$$
$$c_1 = c_{01}(1 - y_d) + c_{11}\, y_d$$

4. Extracting data from the 3D data set



Trilinear interpolation - 3D

$$c_{00} = c_{000}(1 - x_d) + c_{100}\ x_d$$
$$c_{01} = c_{001}(1 - x_d) + c_{101}\ x_d$$
$$c_{10} = c_{010}(1 - x_d) + c_{110}\ x_d$$
$$c_{11} = c_{011}(1 - x_d) + c_{111}\ x_d$$

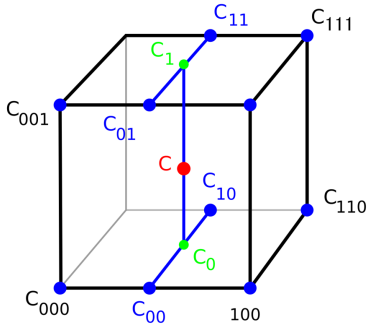
4. Extracting data from the 3D data set

$$c_{00} = c_{000}(1 - x_d) + c_{100}\ x_d$$
$$c_{01} = c_{001}(1 - x_d) + c_{101}\ x_d$$
$$c_{10} = c_{010}(1 - x_d) + c_{110}\ x_d$$
$$c_{11} = c_{011}(1 - x_d) + c_{111}\ x_d$$

$$c_0 = c_{00}(1 - y_d) + c_{10}\ y_d$$
$$c_1 = c_{01}(1 - y_d) + c_{11}\ y_d$$

$$c = c_0(1 - z_d) + c_1\ z_d$$

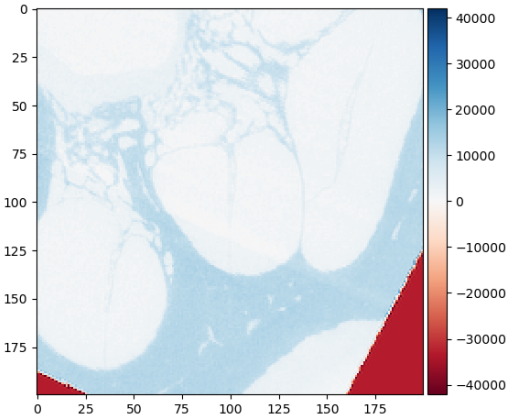
Trilinear interpolation - 3D



$$c \approx l(b(c_{000}, c_{010}, c_{100}, c_{110}), b(c_{001}, c_{011}, c_{101}, c_{111}))$$

5. Visualising slices in a new NIfTI file

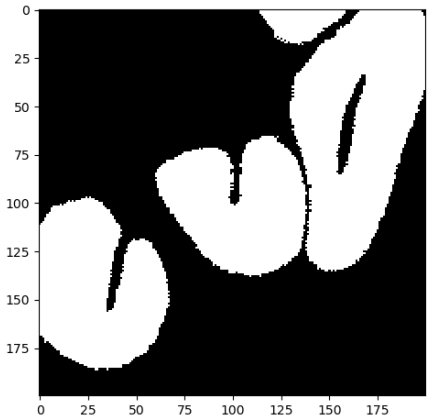
Irrelevant data is removed via BLOB analysis:



The raw data

5. Visualising slices in a new NIfTI file

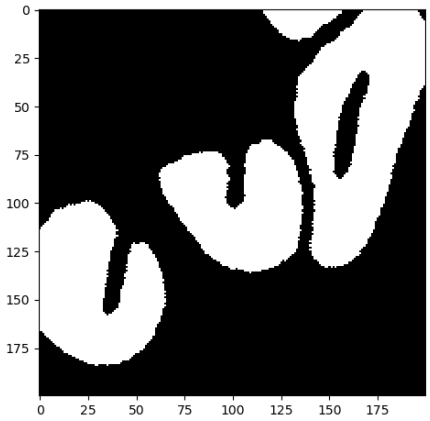
Irrelevant data is removed via BLOB analysis:



The given segmentation

5. Visualising slices in a new NIfTI file

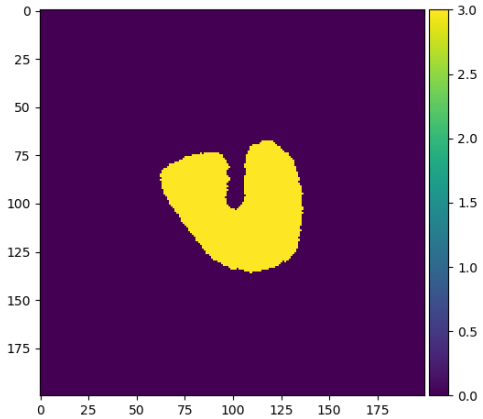
Irrelevant data is removed via BLOB analysis:



Eroded with disk-size 2 for better separation

5. Visualising slices in a new NIfTI file

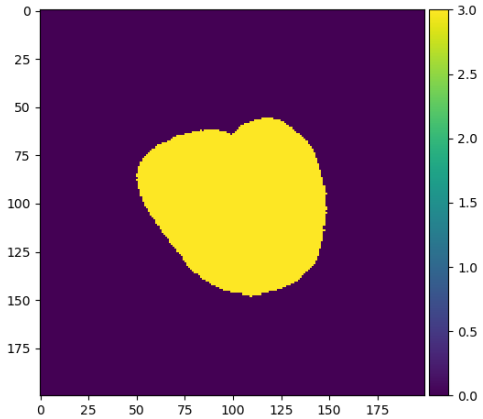
Irrelevant data is removed via BLOB analysis:



Only the BLOB with centroid closest to the image center is kept

5. Visualising slices in a new NIfTI file

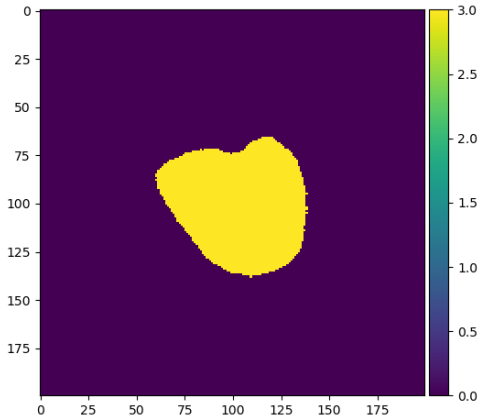
Irrelevant data is removed via BLOB analysis:



Dilation to remove gaps

5. Visualising slices in a new NIfTI file

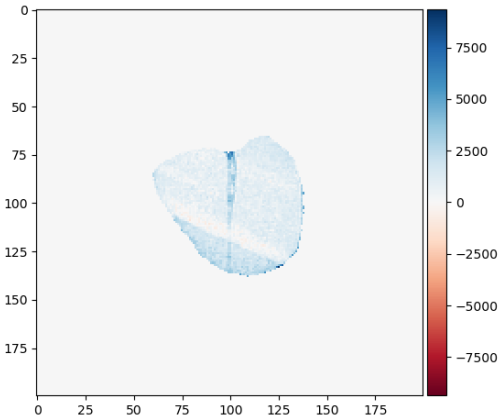
Irrelevant data is removed via BLOB analysis:



Erosion to get down to original size

5. Visualising slices in a new NIfTI file

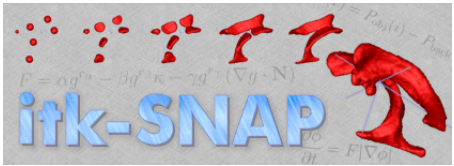
Irrelevant data is removed via BLOB analysis:



All datapoints outside the final BLOB are disregarded

5. Visualising slices in a new NIfTI file

See the result in ITK-SNAP



6. Mapping frequencies onto unfolded cochlea

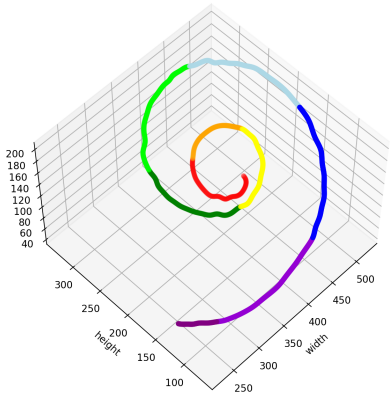
Greenwood function:

$$f = \int_a^b \Delta f_{cb} = A(10^{ax} - K) \tag{2}$$

where:

- f - characteristic frequency of the sound in hertz
- A - scaling constant
- a - slope of the straight-line portion of the frequency-position curve
- x - fractional length along the cochlear spiral measured from the apical end of the cochlea to the region of interest. $0 < x < 1$.
- K - constant of integration, determined by the lower frequency audible limit in the species.

6. Mapping frequencies onto unfolded cochlea

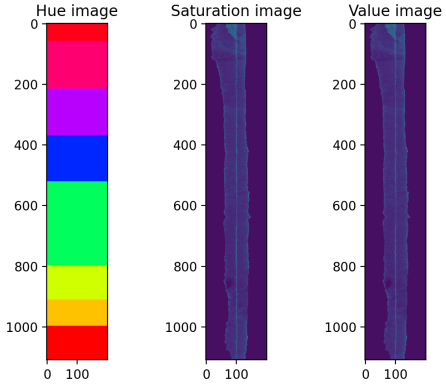


3D representation of the cochlea colored by frequencies

- 20-125Hz
- 125-250Hz
- 250-500Hz
- 500-1000Hz
- 1000-2000Hz
- 2000-4000Hz
- 4000-8000Hz
- 8000-16000Hz
- 16000-20677Hz

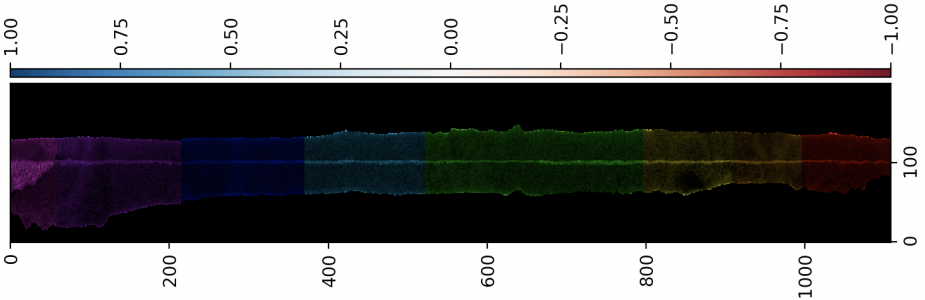
Spans inspired by Li, H., Helpard, L., Ekeroot, J. et al.

6. Mapping frequencies onto unfolded cochlea



HSV color planes of the unfolded cochlea

6. Mapping frequencies onto unfolded cochlea



Unfolded cochlea colored by frequency