3D Visualization of Volume Data

Yaseen Umar Khan

Abstract

Investigate the use of MATLAB for navigating and visualizing an assortment of different datasets in 3-dimensional (3D) space using Volume Rendering techniques. We used existing MATLAB functions such as contour slices, slice planes and isosurfaces for visualization of scalar data. In this project I have performed Plotting of simulated data for wind in 3-D space. 3-D visualization of the earth topography also helps us to get a 3-D perception of the earth's shape Used the MRI images data to generate a cool 3-D visualization of the face and a slice of the skull.

Introduction

In scientific visualization and computer graphics, volume rendering is a set of techniques used to display a 2-dimensional (2D) projection of a 3D discretely sampled data set. A typical 3D dataset is a group of 2D slice images acquired by a computed tomography (CT), magnetic resonance imaging (MRI), or ultrasound (US) scanner. Usually these are acquired in a regular pattern (e.g., one slice every millimeter) and usually have many image pixels arranged in a regular pattern. This is an example of a regular volumetric grid, with each volume element, or voxel, represented by a single value that is obtained by sampling the immediate area surrounding the voxel.

Typical scalar volume data is composed of a 3D array of data and three coordinate arrays of the same dimensions. The coordinate arrays specify the x-, y-, and z-coordinates for each data point. The units of the coordinates depend on the type of data. For example, flow data might have coordinate units of inches and data units of psi. Many existing MATLAB functions are useful for visualizing scalar data:

- Slice planes provide a way to explore the distribution of data values within the volume by mapping values to colors. You can orient slice planes at arbitrary angles, as well as use nonplanar slices. You can specify data used to color isosurfaces, enabling you to display different information in color and surface shape.
- Contour slices are contour plots drawn at specific coordinates within the volume. Contour plots enable you to see where in a given plane the data values are equal.
- Isosurfaces are surfaces constructed by using points of equal value as the vertices of graphic objects.

To render a 2D projection of the 3D data set, one first needs to define a camera in space relative to the volume. Also, one needs to define the opacity and color of every voxel. For example, a volume may be viewed by extracting isosurfaces from the volume and rendering them or by rendering the volume directly as a block of data. The latter is a computationally intensive task but can be performed in several ways. In this Project, I will be exploring an assortment of MATALB functions for navigating and visualizing discrete data in 3D space

Methods

Volume rendering is a set of techniques used to display a 2D projection of a 3D discretely sampled data obtained from various sources such as CT scan, MRI or US. We used existing MATLAB functions such as contour slices, slice planes and *isosurfaces* for visualization of scalar data. To get an idea about how to visualize vector volume data, we plotted wind flow using preexisting functions such as *coneplot*, to plot the velocity vectors magnitude and direction, *streamline,* to show flow of velocity vector field and *streamtube* function to plot velocity vector with normalized divergence of the vector field at each point. To illustrate Topographic data, we used contour plot by plotting points that have zero altitude to visualize the outline of the continents and using elevation data as custom color map. Shades of green and blue were used to visualize altitude data and depth below sea level respectively. Then we used texture mapping to map the topography to a spherical surface. The function *surface()* was used for this purpose. Similarly, to visualize 3D MRI data we use *contourslice* to display contour plot of a volume slice and then after smoothing the data we use *isosurface* to calculate *isodata* and then used patch to display this 3D MRI data. To improve the quality of the *isosurfaces* we used *isonormals* function resulting in a beautiful 3D display of face with a sliced-away top of the head. Finally, by combining *isocap* with the *isosurfaces* and using colormap supplied with the data we could display isocap at other data values to visualize a slice of the brain.

Code

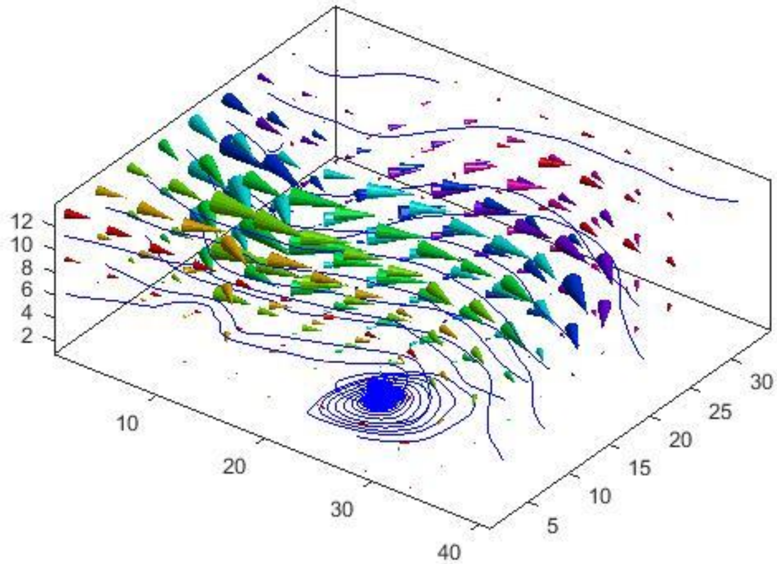
https://github.com/yaseenumar/3-D-Visualization-of-Volume-Data-

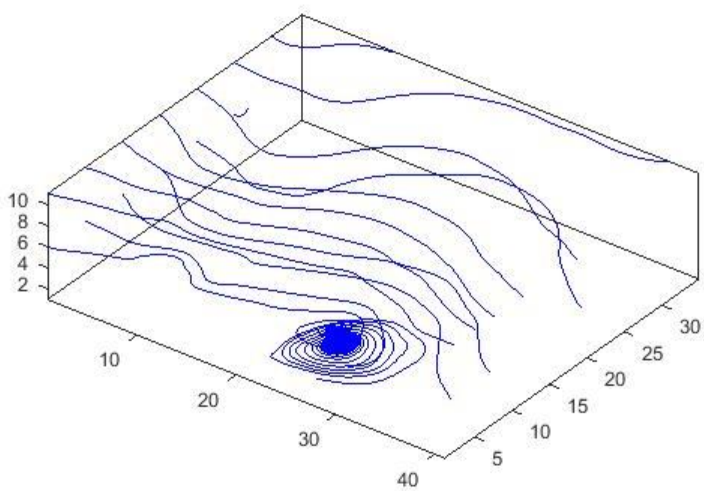
Results

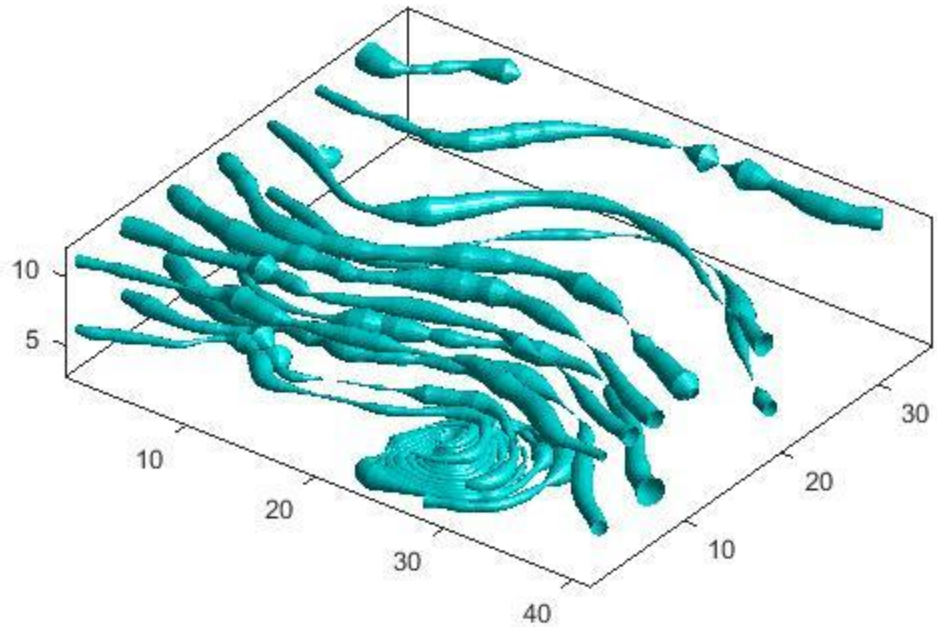
Plotting simulated data for wind in 3-dimensional space helps in visualization of the wind flow velocity vector and allows us to observe how the velocity of the wind changes in 3D space. Similarly, the 3-dimensional visualization of the earth topography also helps us to get a 3D perception of the earth's shape. Finally, using isosurfaces and isocap functions the MRI images data was used to generate a cool 3D visualization of the skull(See Appendix for Result Images).

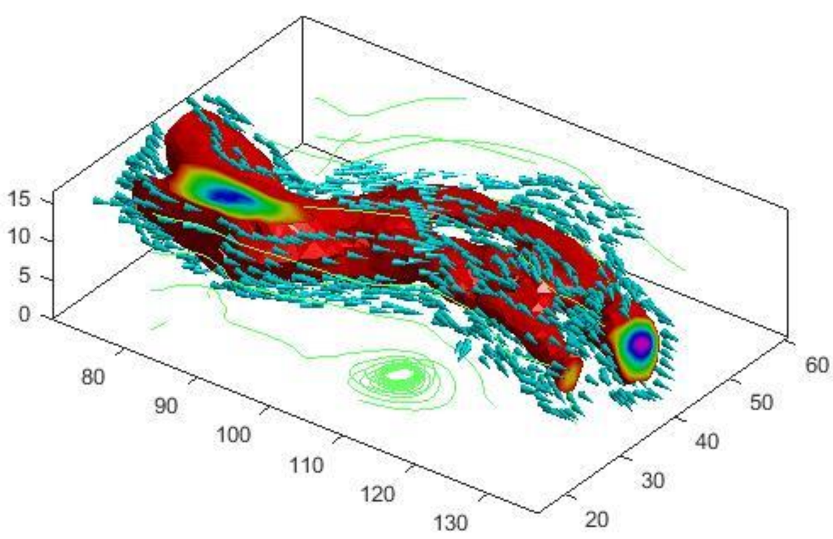
APPENDIX

1. Visualizing Vector Volume Data

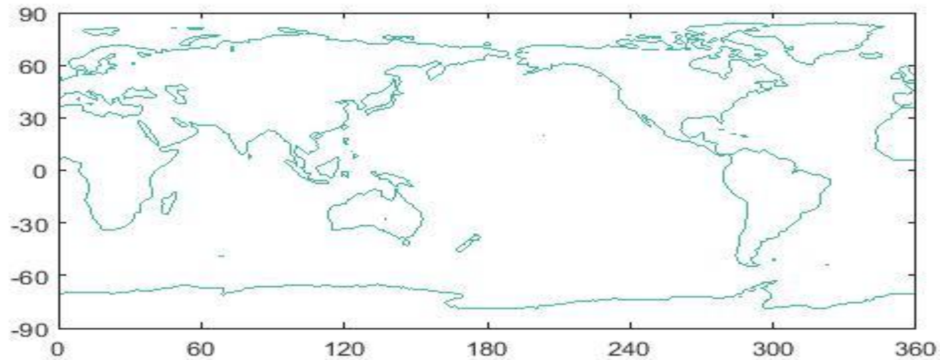


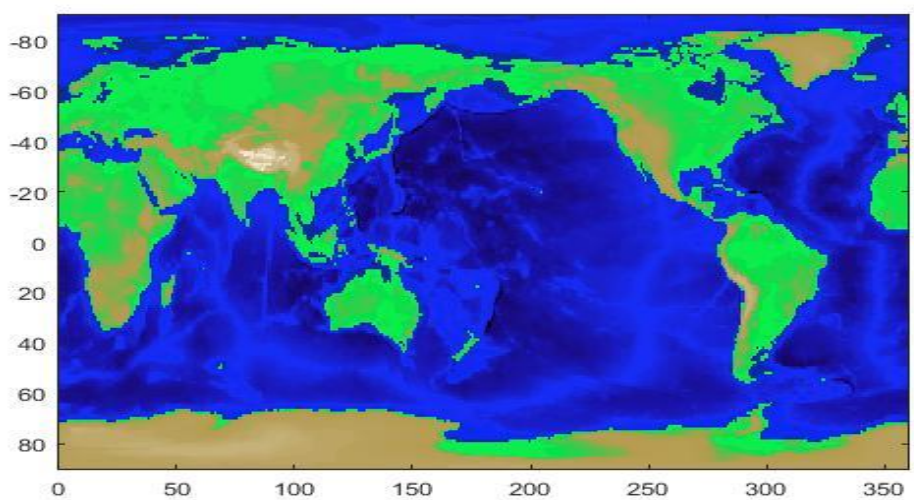



10
5
10
20
30
40
10
20
30

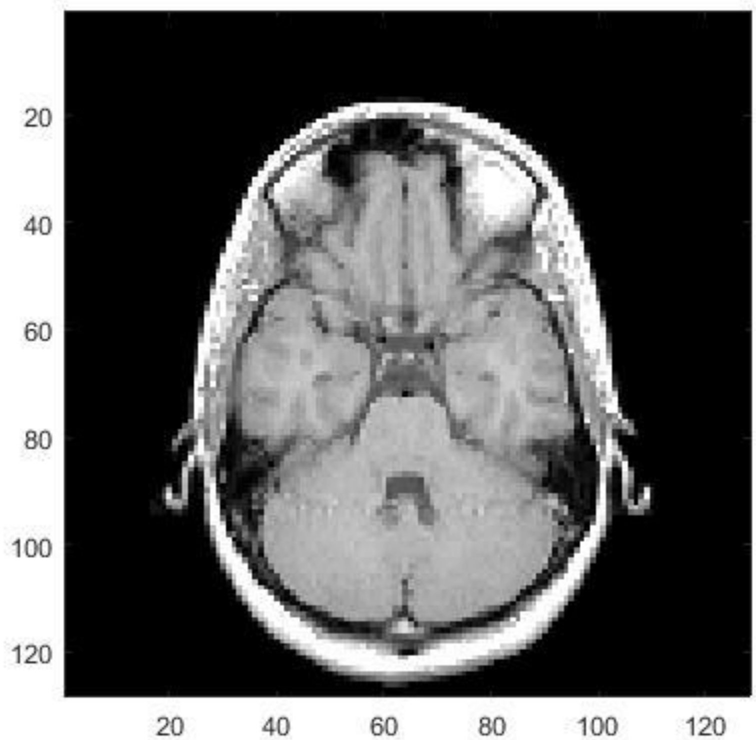

15
10
5
0
80
90
100
110
120
130
20
30
40
50
60

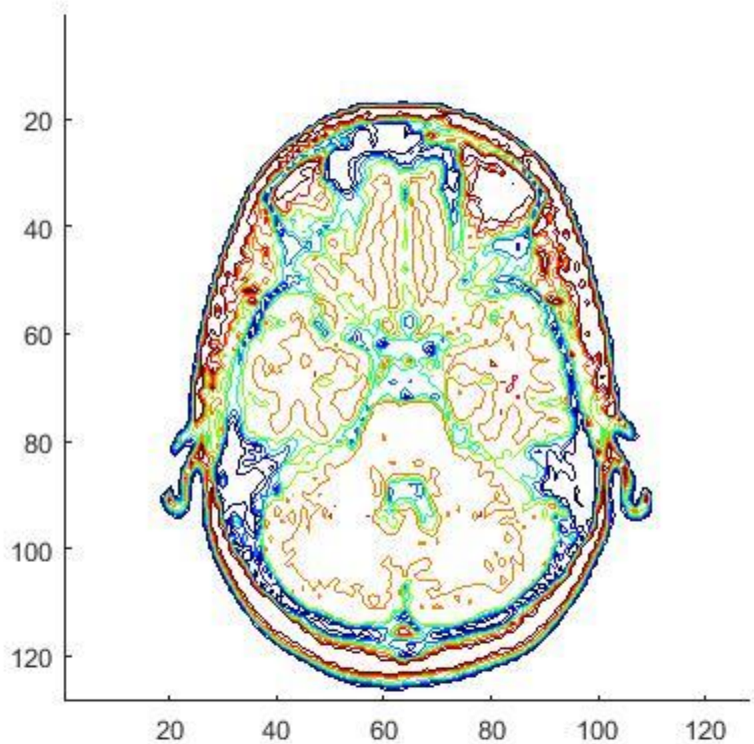
2. Displaying Topographic Data

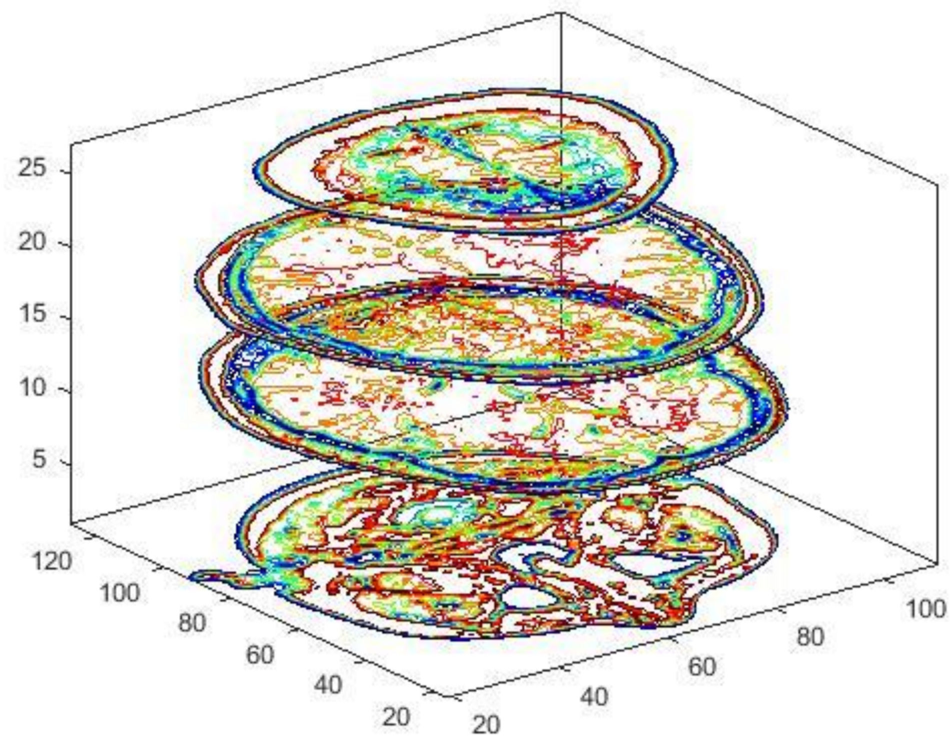


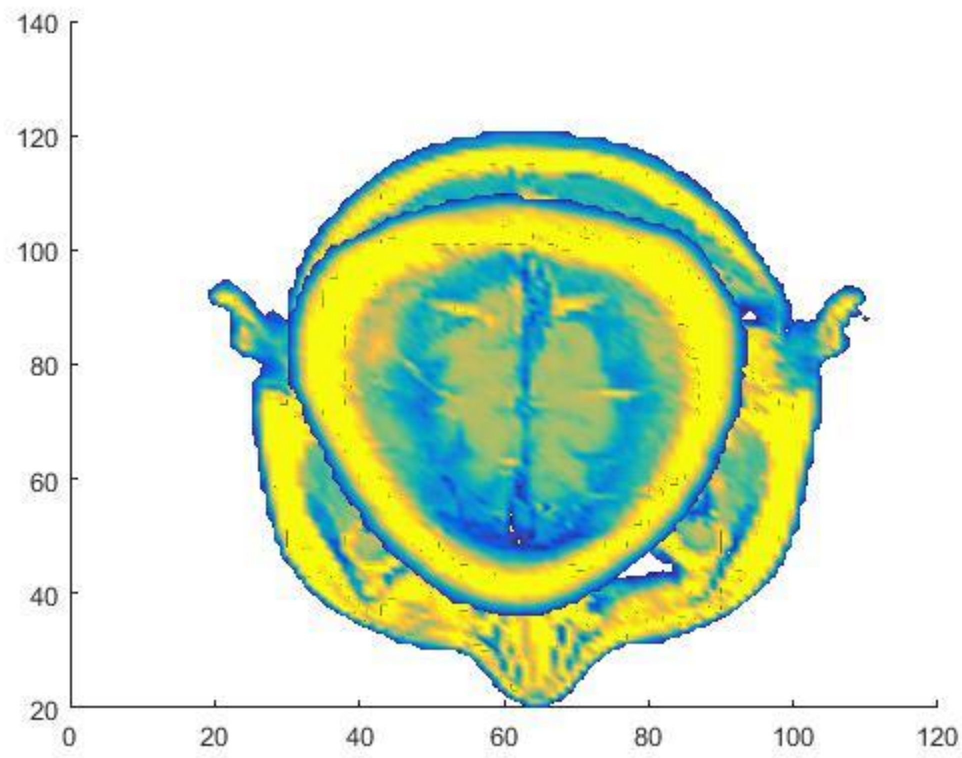


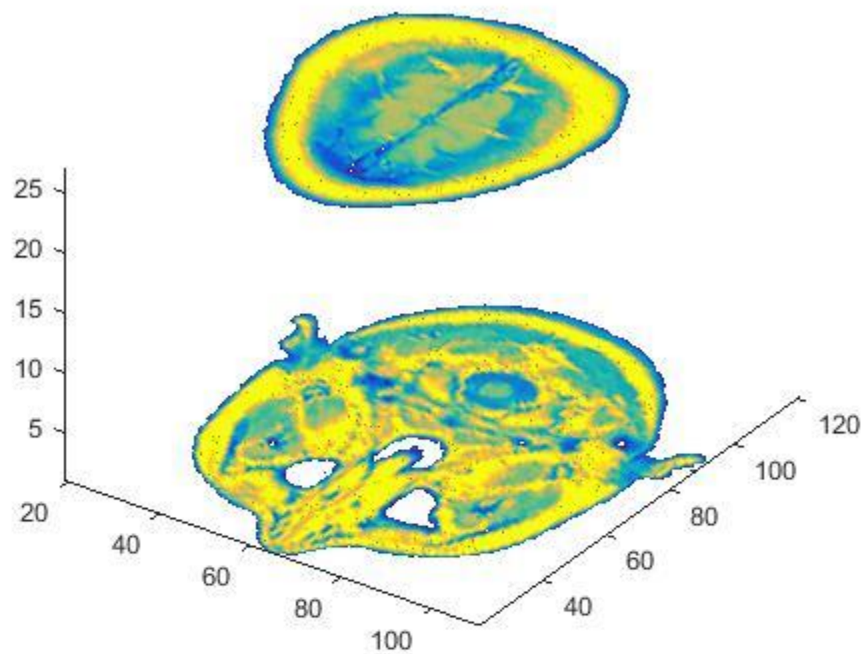
3. Displaying 3-dimensional Medical Images

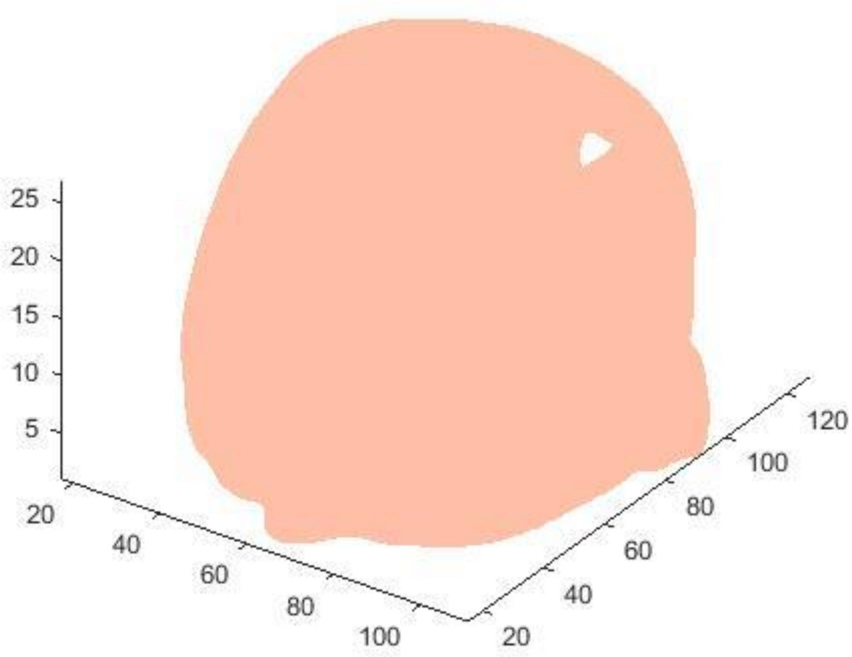


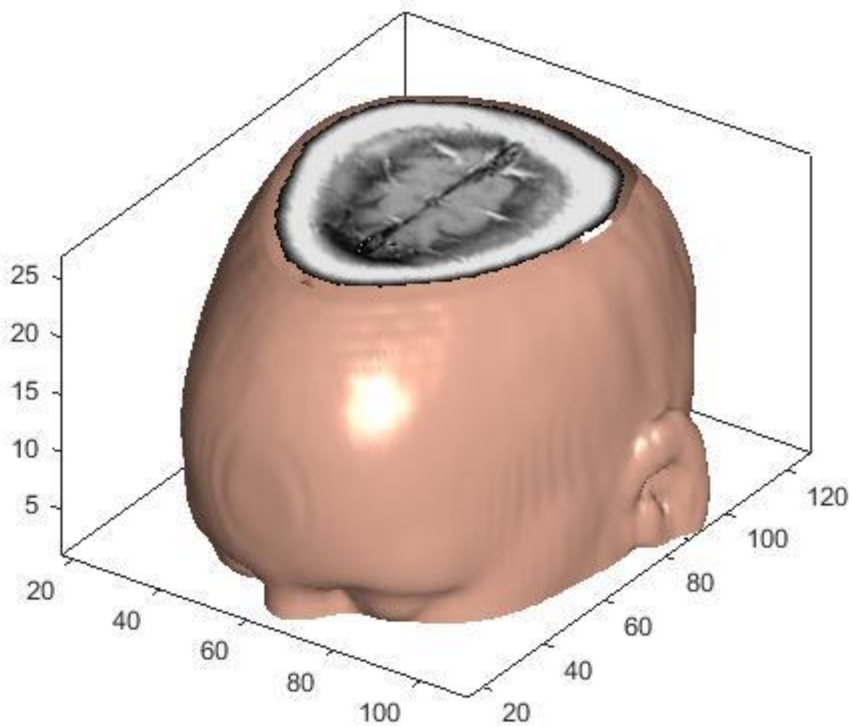


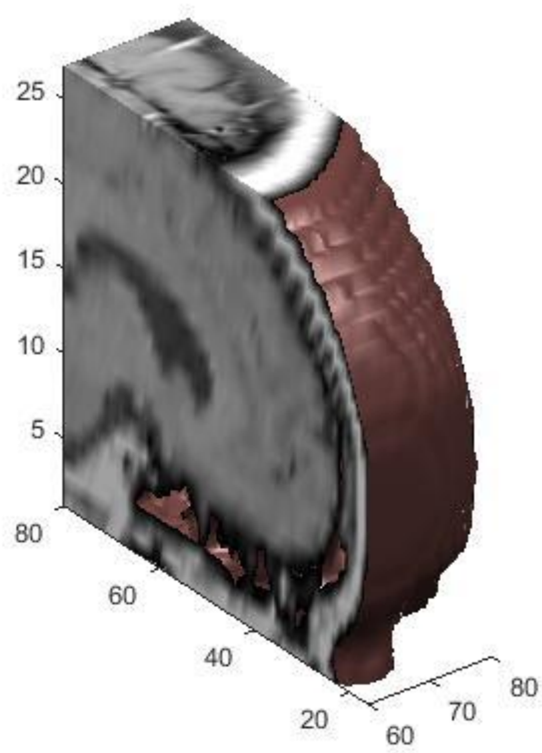

25
20
15
10
5
120
100
80
60
40
20
20
40
60
80
100


140
120
100
80
60
40
20
0
20
40
60
80
100
120


25
20
15
10
5
20
40
60
80
100
40
60
80
100
120


25
20
15
10
5
20
40
60
80
100
20
40
60
80
100
120


25
20
15
10
5
20
40
60
80
100
20
40
60
80
100
120


25
20
15
10
5
80
60
40
20
60
70
80