Visualisation Course 2008, MSc – L11 **(ONLY)**

Assignment 2 – Visualisation of Mouse Embryo Development

This assignment is the second of two assessed practical exercises for the visualisation module. The aim of the practical is to visualise a set of 'section volumes' of data from 4 stages of mouse embryo development. This practical is again open-ended in its nature. You are supplied with the data, a description of its nature and a few suggested methods of visualisation. Your task is to produce a suitable visualisation of the data.

Data

The data you are given for this exercise consists of 4 'section volumes' of data from 4 stages of mouse embryo development. The volumes have been derived from photographs of slices. Some of the volumes have been photographed on a dark background, and some on a light background. You are also provided with a 'painted volume' of anatomical domains – i.e particular values of the scalar value correspond to particular anatomical components. There are 9 anatomical domains present in the data. The numbers corresponding to domains are a bit bizarre; this is due to the historical limitations of 8-bit displays.

Anatomy	**Numbers**
Primitive Streak	64-95
Surface Ectoderm	96-127
Gut	128-159
Future brain	160-191
Future spinal cord	192-223
Left carotid artery	224
Right carotid artery	225
Optic vesicle	226
Outer embryo wall	227

The data is available as a set of SLC format files from the course home page at
http://www.inf.ed.ac.uk/teaching/courses/vis/mousedata.tgz
After uncompressing the archive using gunzip, you will find `ts10.slc ts12.slc, ts11.slc, ts13.slc, ts14.slc, ts14_painted.slc`. The sectional data files are called ts10 – ts14.slc. The painted anatomy file is called `ts14_painted.slc`. The files are named after the Theiler stages of the mouse embryo, a reference on Theiler stages can be found in http://genex.hgu.mrc.ac.uk/Atlas/intro.html .

Task

Your first task is to visualise this data using volume rendering in order to show the development of the mouse between the stages. The resolution of the data is constant – i.e one voxel is always a fixed size. Since the mouse embryo grows between the stages, the later stages of development correspond to quite large files that you will need to sub-sample before they can be rendered. You can also visualise the anatomy present in the 'painted domains' file. You can do this either as a surface or as a volume, but bear in mind the limitations of contouring algorithms. The anatomy file contains domains that overlap and enclose each other, so you will need a method of displaying the outer domains that allows the inner ones to be seen. You will want to show each anatomical domain in a different colour. It is useful to combine the two visualisations to show the painted and section volumes in the same space. You can do this either with volume rendering or showing the section volume as a 2D slice through the data.

Suggestions & Examples

• **Colour & Opacity transfer functions** – If you use volume rendering, you will need to create appropriate colour and opacity transfer functions for the data. Look at the example code to get you started.
• **`vtkColorTransferFunction`** creates a new colour transfer function. The method `AddRGBPoint` adds an RGB colour at a particular scalar value. You don't need to set every value, the function will interpolate between set values.
• **`vtkPiecewiseFunction`** creates a new opacity transfer function. Use the method `AddPoint` to add an opacity value, again the function interpolates.
• **Sub-sampling -** can be done with `vtkExtractVOI` filter. Some of the volume rendering algorithms use the volume extent to determine resolution, so you may need to alter these as well.

• **Multiple renderers** – vtk 4.2+ supports multiple renderers rendering to a single window. This will allow mixed surface and volume renders.
• **Thresholding** – To display the anatomy as a surface, threshold the data above and below the anatomy value and contour the difference.
• **Slicing the data** – Can be done using `vtkProbeFilter`.

Specific examples of VTK volume rendering are provided at the course website:

http://www.inf.ed.ac.uk/teaching/courses/vis/example_code.tgz

For more information on the mouse atlas, and perhaps some inspiration for images, look at http://genex.hgu.mrc.ac.uk

Requirements & Submission

For this assignment, you are required to submit the following (with marks awarded as shown):

Visualisation Solution (60%): a working (on DICE) visualisation solution allowing the successful visualisation of the specified data set. Submit a MPEG video (<1 minutes in length) shows the features of your visualisation and 4 representative JPEG still images. JPEG still images can be captured from VTK itself or using a screen/window capture tool. Your solution should be available as an executable demo.

As with any visualisation technique the evaluation of this work will be somewhat subjective. The marking for this assignment will similarly be subjective but will reflect the level to which the available data and the additional information available from that data is presented using suitable visualisation techniques.

Code (20%): Commented TCL script(s), Java or C++ code(s) for your visualisation solution. The TCL script should be usable in the form `vtk scriptname.tcl` and read all data files from the current directory (assume file names as per web). If you use Java or C++ you **must** also submit a binary that works under DICE with instructions on how to run it.

A README file (20%): containing anything not obvious from the comments in the source and details of how to run your scripts (e.g. "for part 1 run vtk fileX.tcl"). Additionally a brief (< 500 words) description of what your visualisation methods show, an evaluation of their relative merits for the visualisation task in hand and future improvements you would make given more time. You will get marks for understanding and appreciating the problem, even if you have not managed to complete the implementation.

Submission procedure:
Place your files in a single directory, and call the informatics electronic submission script for visualisation as follows (`man submit` for further details):

MSc : `submit msc vis-5 cw2 your_directory_name`

Submission deadline for this part is **Wednesday 9th April at 5pm.** The usual lateness penalties will apply. *This part carries 50% of the final practical mark for the course.*

Remember to leave enough time to test your program on the DICE machines.

Please ensure your submission complies with the school policy on plagiarism:
http://www.inf.ed.ac.uk/teaching/plagiarism.html *A failed assignment will not ruin your career but a plagiarised one may do. Acknowledge all other sources of code you use in your submission.*

Taku Komura 29/02/08 (based on previous exercise by Gordon Watson)