Visualisation Course 2008, UG4/M.Sc.

Assignment 1 – Adding Elements to the Knee visualisation

This assignment is the first of two assessed practical exercises for the visualisation module. The aim of the practical is to produce an enhanced visualisation of the human knee dataset (`vw_knee.slc`) used in practical handout 2. This handout will guide your through various aspects of the design of this visualisation. Together with the material presented in lectures and previous practical assignments you are required to come up with an improved visualisation of this dataset.

A surface for the skin

Using the same pipeline as for the bone (practical handout 2), add an additional surface with an approximate density for skin. Start with a value of 50 density units, but this value is not the best so a little experimentation is required. Choose an appropriate colour for skin. It may be difficult to see both the bone and skin surface at the same time in the same window. Try adding an extra `renderWindow` and/or renderer object to see the two visualisations side by side to determine a suitable method of visualisation.

Part 1 - Seeing through the skin

Handout 2 asked you to set the colour of the surface by calling a method in the `vtkProperty` object. As well as having a property for colour, the object also has a property for opacity (transparency). Try modifying the opacity property to make the skin surface semi-transparent. The surface actually has two `vtkProperty` objects associated with it – one for the front faces of the polygons and one for the back-faces. The colour and transparency of front and back faces of the skin surface can hence be set independently, a useful technique for giving a greater sense of '3D depth' yet allowing the viewer to see through the skin. Try experimenting with different combinations of colour and opacity to improve your visualisation.

Part 2 - Cutting a hole in the skin

Transparency can be useful for seeing through surfaces, but it is often difficult to see details in a semi-transparent object. The alternative is to use an opaque surface but cut a circular hole in it in an appropriate place by clipping it with an implicit surface. To do this, create an implicit sphere using the `vtkSphere` object. Polygons can be cut by the sphere using the `vtkClipPolyData` object.

Hint : set the following attributes in `vtkClipPolyData` :

```
GenerateClippedOutputOn
GenerateClipScalarsOn
SetValue 0
```

Positioning the sphere

Now that the sphere has been created, the next problem is to position it in the right place and make it the correct size. This is a bit tricky since the sphere is invisible unless it intersects a part of the already created bone geometry. There are several ways of making the sphere visible for easy positioning, your submitted code should include one of them (either in a separate file, or commented out in the final version).

Part 3 – Final Visualisation

For your final visualisation you may choose to combine opacity, clipping (as described above) and other visualisation features presented in lectures. You may wish to use one or more windows and/or renderers or additional interactive techniques. You are required to use what you have learned to provide an enhanced visualisation of the available data. This part of the assignment is designed to be open-ended. Your resulting visualisation does not need to be overly complex (which may confuse the user) but needs to provide a good solution to the visualisation problem for this data.

Submission

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

- **Part 1 (20%):** An **MPEG animation** showing a semi-transparent skin surface varying in opacity while the model is slowing rotating (approx 30 seconds). You may use either the mpeg_encode or xvidcap tool for this task but the animation must be automated in VTK.
- **Part 2 (20%):**
  An **image (JPEG)** showing the knee joint through a hole cut in the skin surface.
- **Part 3 (20%):**
  An brief **MPEG animation** (< 1 minute) of an example interaction with your completed knee visualisaton
  - *you may wish to investigate other interaction methods available within VTK. Remember that xvidcap can be used to capture one or multiple windows.*
- **Code (30 %) :** A commented TCL script(s), Java or C++ code(s) all three parts, including an option for positioning the sphere. The TCL script should be usable in the form `vtk scriptname.tcl` and read the data file `vw_knee.slc` from the current directory. 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 (10%)** 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 (< 150 words) desciption of what your visualisation for part 3 shows and how you achieve this.

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):

```
UG4  :      submit cs4 vis-4 cw1 your_directory_name
MSc  :      submit msc vis-5 cw1 your_directory_name
```

Submission deadline for this part is **Wednesday 27h February at 5pm.**
The usual lateness penalties will apply.

This part carries 50% of the final practical mark for the course, and for a rough guide should take about 8 hrs.

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.

Notes

Note if you run out of space in you home area to store all the MPEG frames, you can use temporary spaces on the filesystem such as /tmp.
You can also use your personal machine/laptop to do the practical. If you do I would appreciate it if your movies can be played with the supplied `gmplayer` player on DICE. There is a windows version of mpeg_encode available from Berkeley. Please test them prior to submission.
Remember to leave enough time to test your program on the DICE machines.

Taku Komura 25/01/08 (based on previous exercise by Gordon Watson)