SVVR Assignment 2

Sander Nugteren (6042023)   Merel de Groot (6103677)

November 17, 2014

1 Visualization Pipeline

The assignment is to create a visualization pipeline with VTK. The source will be a dataset consisting of images from a CT scanner. These 2D images combined form a 3D image of a body. Using VTK we will create a pipeline that takes the dataset as input and creates a 3D visualization of the body. The order of the sections will correspond to the steps of the pipeline.

1.1 Source

The data consists of 94 images created by a CT scanner. Each file is 131,072 bytes and the images are 256x256 or 512x512 pixels in size. Since 131,072 devided by 512x512 is only a half, we know that this is not possible. Deviding 131,072 by 256x256 however, gives us 2 bytes, which is a reasonable size of data per pixel. The collection of binary images is read by using VTKImageReader2.

1.2 Filter: Contour

Using the iso-surface contour visualization.
- value range for the contour value: 0 to 2^{16}
- render an isosurface contour of the dataset, add an interactor so that you can interact with the resulting visualization
- allow the initial contour value to be passed from the commandline so that you can pass a different value each time you run your program ¿ effect?

1.3 Mapper

1.4 Actor: Colours

Default: object in a dark blue on a black background
Three ways to change the colour of the object:

- Tell the contour filter stage to not compute scalar values and then set a colour in the actor

- Tell the mapper stage to ignore scalar values and then set a colour in the actor
- Tell the mapper what the actual scalar range is.

Next, change the background colour and increase the size of the render window as well.

1.5 Data Spacing

Configure "data spacing" in source. By default voxels are isotropic, but we need ??

1.6 Renderer: Setting the Scene

The renderer will create a camera and configure it so that the visualization will be visible in the render window. However, in most cases this will be suboptimal; the initial camera angle will not provide a useful view on your visualization. In that case it will be necessary to manipulate the camera. VTK uses two models to manipulate the camera:
- Camera is focused at a focal point (your visualization) and moves around this focal point using the Elevation, Roll and Azimuth methods.
- The movement of the camera is centered at the position of the camera and the orientation of the camera is controlled using the Yaw, Roll and Pitch methods.
Determine a better camera angle and add an outline to your visualization that shows a bounding box to represent the extent of the input data.

2 Results

Explore all contour values in the dataset.