3D skeletonization module

1. Background

A binary 3D object can be represented by its skeleton, which consists of the spines of the object parts. A 3D skeleton may consist of 3D curves (spines), isolated 3D points and, possibly, 3D surfaces. More information on 3D skeletons can be found in [1]. The generation of a skeleton is realized by applying an iterative process which erodes the object layer by layer until only the object spines remain, which form the skeleton of the 3D object. This iterative process is called *thinning*. A thinning algorithm contains a set of voxel deleting conditions, which enable it to erode the object iteratively. The thinning process should support the following properties, so that the thinning result can be characterized as a skeleton of the 3D binary object:

- *Geometry preservation* is a major concern of thinning algorithms. For example, an object like "b" should not be converted into an object like "o". To preserve the geometry of the original image, a thinning algorithm must contain certain geometry preserving conditions.
- *Topology preservation* is the second major concern of thinning algorithms. For example, an object like "o" should not be converted into an object like "c".
- Finally, a skeleton of an object should be, ideally, as thin as possible (one voxel wide) and should represent the object through its spine or *medial axis*.

A skeleton of an object can have a variety of applications in 3D image processing, such as:

- *Automatic path generation*. 3D volume visualization of objects, especially in medical applications, can be beneficial from the patient's side. Such a process can implement virtual endoscopy, where the navigation of a virtual camera can be guided by a path inside an organ. The skeleton of the organ can serve as such an automatically generated path.
- *Shape matching, identification, classification*. As skeletons retain only vital information about the shape of an object, the object modeling is simplified.
- *Length measurement*. By representing an object through its medial axis length measurement can be simplified in 3D space.

- *Compression*. A 3D skeleton may need substantial less information for its description than the entire 3D object itself.

2. 3D Skeletonization Module Overview

3D Skeletonization is a module of EIKONA3D software package. It conforms to the module expansion capability of EIKONA3D and offers a thinning function available for the user. The function is called *3D Pattern Thinning* function which based on a pattern-based thinning algorithm and extracts the skeleton of a single volume. The module can be applied only on *binary* volumes. A binary volume contains only two intensity values, a *foreground* voxel value (255), and a *background* voxel value (0). A simple way to create a binary volume is by *thresholding* (a basic operation of EIKONA3D). This module is designed to work for all versions of EIKONA3D.

3. Installation

The module comes in a single DLL (Dynamic Link Library) file called `3dthin.dll`. In order to install the module you only have to copy the file in the installation directory of EIKONA3D (where the `eikona3d.exe` file resides in) and restart EIKONA3D to recognize the new module.

4. Use of 3D Skeletonization Module

After launching the EIKONA3D package, the skeletonization module will create a new submenu under the name *3D Skeletonization,* as illustrated in Figure 1.

Figure 1: *3D Skeletonization* menu function in EIKONA3D.

The skeletonization module operates on a single input volume that EIKONA3D handles. The output is a volume containing the generated skeleton of the object(s) comprising the input volume. It is noted again that both the input and the output volumes of 3D Skeletonization should be binary volumes. Even if the user selects a grayscale volume, it is converted inside the module to a binary one by thresholding it with a voxel intensity threshold equal to 1. This means that every voxel whose value is greater than or equal to the threshold takes a value of 255.

It is worth noting that not every 3D object can be thinned to generate a well formed 3D skeleton. A *well-composed* object is likely to be thinned to a better result. 3D binary object preprocessing by 3D interpolation and/or filtering can help producing a satisfactory result.

5. 3D Skeletonization operations

5.1 3D Pattern Thinning

When the user selects the 3D Pattern Thinning operation a dialog box appears permitting the user to select the input volume. The input dialog filters only the grayscale volumes already loaded in EIKONA3D. After the selection of an input volume, a second dialog box appears permitting the user to select the output volume. The output volume can be placed either on an existing buffer or on a new one. A dialog will appear displaying the progress of the thinning operation as illustrated in Figure 2.

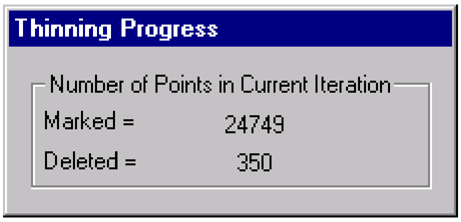

Figure 2: Dialog which displays the thinning progress.

The algorithm works iteratively and it is time-consuming. In each iteration, a number of object voxels is marked in order to be tested against a set of deleting conditions. The box *Marked* in the dialog shows the number of voxels currently marked for checking. If an object voxel satisfies the deleting conditions, it is deleted. The box *Deleted* in the dialog shows the number of object voxels deleted so far from the set of the marked object voxels.

5.2 Thinning Example

Figure 3 illustrates the skeleton extracted using the 3D pattern thinning operation on a pulmonary tree. The 3D data were obtained from CT slices.

Figure 3: a) 3D rendering of pulmonary airway tree. b) The skeleton produced by the 3D pattern thinning algorithm.

6. Performance

The performance of 3D Pattern Thinning depends on the number of object points included in the volume and the object shape itself. A well-shaped object could be more easily thinned and take less time than a non-smooth object. Generally speaking, the thinning process using this operation is a time consuming operation for even medium sized volumes (having frame size 128×128 or 256×256 pixels). For example, the thinning process on the volume displayed in Figure 3 (size 128×128×128) took about 5 minutes to complete on a single Pentium II class CPU at 400 MHz or above. The math morphology thinning runs significantly faster because of the type of operations it performs.