Surface Rendering

The Surface Rendering module is the file `surfrend.dll`. If this DLL exists in the directory of EIKONA3D when the program starts, a sub-menu called *Surface Rendering* is added under the *Modules* menu. The menu options of this sub-menu provide the necessary tools for three-dimensional surface reconstruction using triangulation and surface rendering. The source data for surface reconstruction is required in the form of a volume (image sequence) that has been manually processed using the *Color Contour Follow* option (and probably the *Manual Frame Alignment* option) under the *Operations* menu of EIKONA3D (or any other externally created image sequence with similar content). The manual tracking of the closed contours of the object surface(s) in each slice (frame) of the volume, gives a different color to the contours of each object of interest (see the description of *Build Surface Script* menu option below for more details on color selection restrictions), whereas the background remains black. If alignment between successive frames is needed, it can be performed on the source image sequence or on the processed image sequence that contains the colored contours. Usually, the source image sequence contains more information that can be of help in the alignment and it is preferred for performing the alignment procedure. In this case, the contour following is performed directly on the aligned image sequence. The following menu options then use the contours to construct the object surface(s) via triangulation (or even just a 3D representation of the contours only), write it to an industry-standard AutoCAD script file, and produce surface renderings, which can be optionally saved to disk.

***Build Surface Script*:** Through this menu item, the user is first prompted to specify the input color volume (which is supposed to contain the frames of the image sequence with the colored contours that will be connected to form the object surface) through the *Select Input Volume* dialog box. Then, he is prompted to specify the name of the output script file through the *Select Output Script File* dialog box and, finally, the interlayer distance (distance between two successive layers in proportion to the 2-D resolution of the frames) through the *Interlayer Distance* dialog box. Then, the volume is processed to create an AutoCAD script file (the default extension is `.scr`). During this processing, the user is prompted through the *Specify Object Connections* dialog to specify possible connections between different objects (differently colored)

that cannot be guessed automatically, thus providing a way to reconstruct complex objects (see next paragraph for more details). The produced output script file can subsequently be used by AutoCAD or other third-party applications that support this format, or be directly displayed using the *Interactive Rendering* menu option.

Care should be taken in the formation of the input volume containing the color contours in order to avoid unexpected results. A few rules should be followed. First of all, the background pixels (those that do not belong to any contour) in each frame should be black (R=G=B=0). If a contour (in any frame) does not contain any other internal contours of different objects, it makes no difference if it is filled with the same color as the contour or with the background color, since the algorithm tracks and uses only the external boundary of each connected colored shape. For the same reason, the width of a contour makes no difference to the final result. Also for the same reason, care should be taken so that a contour not filled with the contour color does not have any discontinuities (holes), in which case the tracking will continue to the internal boundary of the contour and represent it as a petal-like shape. Another important issue is the coloring of the contours. The group of contours (in two or more successive frames) belonging to the same object should have exactly the same color, so that the algorithm can automatically connect them to form the surface of the object. Furthermore, the color of an object should be different than the color of any other object, in order to avoid unexpected connections between different objects. We define a 3D object as single, if it has only one contour in each volume frame. In order to reconstruct a complex object, it should be separated in two or more single objects. A complex object contains one or more branches. In the case of a branch, we have a body that splits in two or more branches (the body can be formed by one or more successive contours). In such a case, we should form a single object (with respect to coloring) that contains the body and one of the branches and each of the rest branches should form a new single object (this holds recursively for the case of a branch that splits in other branches). As the algorithm cannot automatically connect the branches on the body, the user should specify a connection (in the *Specify Object Connections* dialog) for each branch that forms a new object. Specifically, if the branch occurs in the direction of increasing *z* (frame index), it should be specified that the lower chain (that one in the frame with the lower index) of each branch that forms a new single object (selected as object A in the *Specify Object Connections* dialog) is to be connected on the object that contains the body (selected as object B in the *Specify*

Object Connections dialog). If the branch occurs in the direction of decreasing *z* (frame index), it should be specified that the upper chain (that one in the frame with the higher index) of each branch that forms a new single object (selected as object A in the *Specify Object Connections* dialog) is to be connected on the object that contains the body (selected as object B in the *Specify Object Connections* dialog). The selection of an object in the *Specify Object Connections* dialog is leaded by the object colors, as they are found in the input volume. In the output surface script, each branch that forms a new single object remains as a separate object, including also the part of the surface that connects it on the object that contains the body. If the surface script is rendered, in order to view a complex object as one object, it suffices to give all its component objects the same characteristics (color or material, depending on the application).

***Build Lines Script*:** This menu item is similar to the previous one, with the difference that it produces a 3D representation of the contours without connecting them to produce a surface. Thus, it can help for previewing the orientation in the 3D space of the contours that exist in the frames of a volume suitable to be processed with the *Build Surface Script* procedure.

***Read Lines Script*:** This menu enables the conversion of a lines script file back to a volume. First, it asks for the name of the input script file through the *Select Script File to Open* dialog box, then asks for the output volume through the *Select Output Volume* dialog box, and finally asks for the interlayer distance (the same interlayer distance used when creating the lines script must be given for normal results).

***Interactive Rendering*:** This menu item enables the visualization of a previously created script file (containing a surface/lines representation) through surface rendering. The user needs only to specify the directory and the name of the script file from the displayed *Open* dialog box. In the rendering window the user can rotate the 3D object by dragging the mouse in any orientation. Clicking once with the right mouse button in the window resets the viewing angles. The *Display Coordinates* option is also available for the content of this window. The following menu options (*Set Object Colors*, *Take Snapshot*, *Make Movie*), which are enabled during a surface rendering operation, provide further useful operations related to the content of the

surface rendering window. Examples of interactive rendering can be seen in Figure 1 and Figure 2.

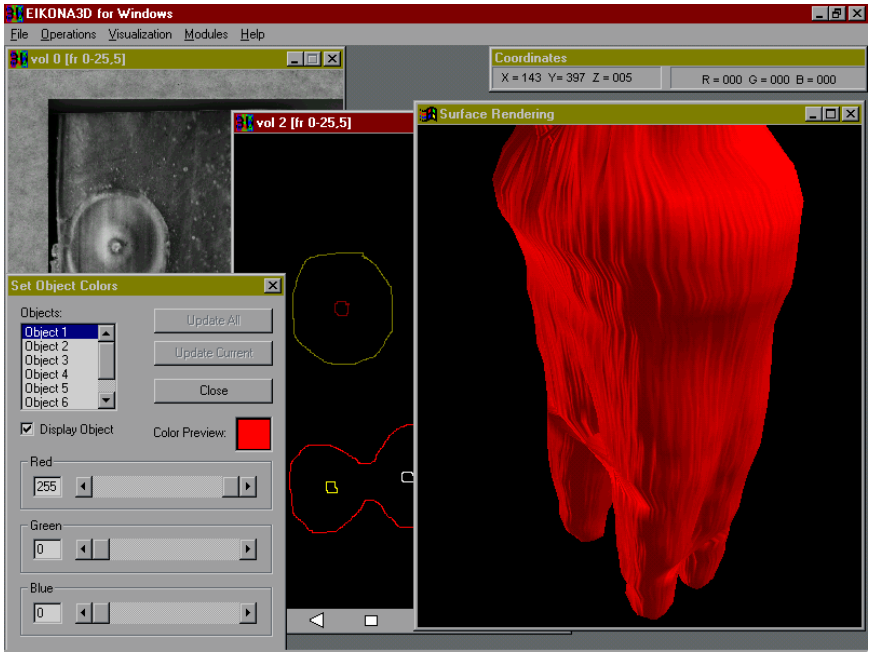

Figure 1: An example of interactive surface rendering of a 3D surface representation of a tooth, produced from an aligned sequence of frames containing contours. A view of outer surface is shown.

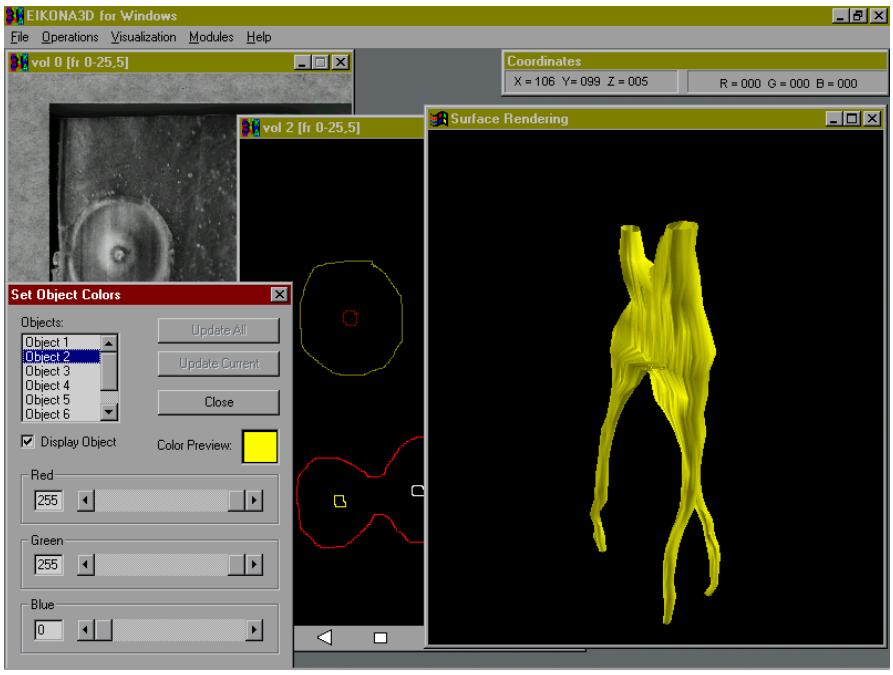

Figure 2: An example of interactive surface rendering of a 3D surface representation of a tooth, produced from an aligned sequence of frames containing contours. A view of root canals is shown.

***Set Object Colors*:** This menu item is enabled during an interactive rendering procedure of a script file. When it is selected, it displays the *Set Objects Color* dialog, which enables the user to set the RGB values for each identified object through the corresponding sliders or edit boxes, or hide/unhide any object through the *Display Object* toggle. Any changes are reflected to the rendering window through the *Update All* and *Update Current* buttons. The dialog is closed through the *Close* button and can be recalled many times while the rendering procedure is active. It should be noted that the script files do not encode the color of the objects. Thus, the colors assigned to the objects when an interactive rendering procedure is initialized are not the same as in the input volume from which the script file was built, but they result from a default color separation scheme. Through the *Set Objects Color* dialog the user can assign his preferred colors to the objects.

***Take Snapshot*:** This menu item is enabled during an interactive rendering procedure of a script file. It enables the transfer of the current content of the rendering window to the first frame of a volume selected through the *Select Output Volume* dialog box. By this way it can be saved to disk with the appropriate *File* menu options.

***Make Movie*:** This menu item is enabled during an interactive rendering procedure of a script file. It enables the production of a movie displaying a 360 degrees rotation of the surface around the vertical axis. The angle difference between two successive frames is inversely analogous to the specified number of movie frames. The user first specifies the number of movie frames through the *Make Movie* dialog, and then selects the output volume where the movie frames are to be written into through the *Select Output Volume* dialog box. Then, the movie can be saved to disk as a normal volume with the appropriate *File* menu options.