for a variety of tasks.

Figure 39-5 Output of the Simple Volume Renderer Using a Voxelized Model of a Familiar Object

39.4 Implementation Details

This section presents an overview of the components commonly used in texture-based volume rendering applications. The goal is to provide enough details to make it easier to understand typical implementations of volume renderers that utilize current-generation consumer graphics hardware, such as the GeForce FX family of cards.

39.4.1 Data Representation and Processing

Volumetric data sets come in a variety of sizes and types. For volume rendering, data is stored in memory in a suitable format, so it can be easily downloaded to graphics memory as textures. Usually, the volume is stored in a single 3D array. Depending on the kind of proxy geometry used, either a single 3D texture object or one to three sets of 2D texture slices are created. The developer also has to choose which available texture formats to use for rendering. For example, power-of-two-size textures are typically used to maximize rendering performance. Frequently, the data set is not in the right format and not the right size, and it may not fit into the available texture memory on the GPU. In simple implementations, data processing is performed in a separate step outside the renderer. In more complex scenarios, data processing becomes an integral part of the application, such as when data values are generated on the fly or when images are created directly from raw data.