To change the size of a data set, one can resample it into a coarser or a finer grid, or pad it at the boundaries. Padding is accomplished by placing the data into a larger volume and filling the empty regions with values. Resampling requires probing the volume (that is, computing interpolated data values at a location from voxel neighbors). Although the commonly used trilinear interpolation technique is easy to implement, it is not always the best choice for resampling, because it can introduce visual artifacts. If the quality of resampling is crucial for the application, more complex interpolation functions are required, such as piecewise cubic polynomials. Fortunately, such operations are easily performed with the help of publicly available toolkits. For example, the Teem toolkit includes a great variety of data-processing tools accessible directly from the command line, exposing the functionality of the underlying libraries without having to write any code (Kindlmann 2003). Examples of using Teem for volume data processing are included on the book's CD and Web site. Advanced data processing can also be performed on the GPU, for example, to create high-quality images (Hadwiger et al. 2001).

Local illumination techniques and multidimensional transfer functions use gradient information during rendering. Most implementations use central differences to obtain the gradient vector at each voxel. The method of central differences approximates the gradient as the difference of data values of two voxel neighbors along a coordinate axis, divided by the physical distance. For example, the following formula computes the *x* component of the gradient vector at voxel location $\vec{p}_{(i,j,k)}$:

Equation 2 Gradient Computation Using Central Differences

$$g_x\left(\vec{p}_{(i,j,k)}\right) = \frac{v\left(\vec{p}_{(i+1,j,k)}\right) - v\left(\vec{p}_{(i-1,j,k)}\right)}{2h},$$

where *h* is the distance between the voxels along the *x* axis. To obtain the gradient at data boundaries, the volume is padded by repeating boundary values. Visual artifacts caused by central differences are similar to those resulting from resampling with trilinear interpolation. If visual quality is of concern, more complex derivative functions are needed, such as the ones that Teem provides. Depending on the texture format used, the computed gradients may need to be quantized, scaled, and biased to fit into the available range of values.