Volumetric Normal Mapping in Rendering of Multivariate Volume Data

S. Seipel[1,2]

[1]Uppsala universitet, Sweden
[2]Högskolan i Gävle, Sweden

Abstract

The work presented in this paper introduces volumetric normal maps for producing visual structure in direct volume rendering (DVR) of 3D data for the purpose of visualizing multiple attributes in a 3D volume. We use volumetric normal maps to represent normal vector glyphs that are subsequently applied to warp the gradients in the primary volume data. This method is intended to visualize some secondary attribute in DVR. We demonstrate that our method can render visual structures in DVR without the need of explicit surface reconstruction and texturing.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Display algorithms

1. Introduction

During the past decade rendering of multivariate 3D scalar data has become a field of increasing interest in visualization research. In the engineering sciences, multi-valued volumetric data are computed in e.g. computational fluid dynamics [SLM02] as well as there are application examples in the field of meteorology [KHGR02]. Along with the technical advancement of medical imaging modalities and with the development of new diagnostic procedures, multi-attribute 3D scalar data is becoming more wide-spread in medical diagnosis. Typical applications are in the field of cancer diagnosis where under administration of contrast agents image enhancement in magnetic resonance imaging (MRI) is observed over time. The specific change of signal intensity registered over a period is an important criterion the differentiation of malignant from benign tissue [MBM*05]. Simultaneous assessment of multiple attributes in a certain anatomic region is also an essential issue in cardiac diagnostics, where functional parameters (like blood flow) are assessed in the context of organs [OKG*06]. Anatomical context is hereby usually represented by some physical tissue property (e.g. densities) as it is acquired using specific 3D imaging modalities. Human anatomy as represented by this primary data serves as a frame of reference for the diagnostic assessment. By means of other imaging modalities such contrast magnetic resonance imaging (MRI) or position emission tomography (PET), to mention a few, some other functional property can be captured and associated with the anatomical structure. In contrast to simulation calculations in the engineering and natural sciences, where multiple variables can be evaluated on the same data grid resolution, in medical functional imaging, often the resolutions of different modalities can be different. A functional variable, like blood circulation, brain activity or metabolism is often represented with a much lower resolution compared to the anatomical reference structures.

2. Related work

Visualizing 3D volumetric multivariate data in context on a flat 2D computer screen poses some intricate problems, to which a number of different approaches have been suggested. Spatial decomposition is a means to break down the three-dimensional context onto several 2D views; it can be extended to visualize different attributes in associated views. Linking views together, the user can re-establish spatial context by interactively exploring and browsing through the data set [AM07]. A similar technique that decomposes 3D data volumes into slices was presented by Oeltze et al. [OKG*06]. In order to explore functional attributes, the authors propose a height field rendering upon selected 2D slices of the original volume to reveal some quantitative secondary parameter associated with the primary volume data.

Another branch of research related to multi-variate direct volume rendering is the field of multi-dimensional transfer function design. In volume rendering, transfer functions define how data is mapped upon colour and opacity of the voxels as they are composited onto screen. First and second order derivatives of the same data can be used for visual mapping such as to allow for better distinguishing different anatomical structures [Lev88, KD98]. Multidimensional transfer functions can generally be used to also map multiple, completely independent variables in a data set to colour and opacity. However, to evaluate the entire value range of multiple attributes in a 3D volume one needs to interactively modify the multidimensional transfer function. An alternative method to immediately visualize multiple attributes with visual cues is by mapping multiple variables independently upon the colour components of the underlying colour model. In the field of breast cancer diagnosis this approach has been adopted by Mehnert et al., who visualized parameters derived from contrast image sequences upon saturation and intensity [MBM*05]. When screen resolution can be sacrificed, different attributes of some sample in the dataset can be expressed on-screen by utilizing symbolic information (e.g. graphical glyphs) or colour icons [Lev91].These visualizations are in particular suitable for categorical data rather than for quantitative data. The work presented in [IFP97] used structured textures mapped upon transparent iso-surfaces to visualize 3D data volumes. Generally, this approach can be used to map additional information associated with the volume upon the object's iso-surface [Tay02].

3. Volumetric texture mapping

Conventional texture mapping requires at first instance extracting an explicit surface from the implicit model, upon which 2D textures can be mapped. Existing iso-surface reconstruction methods such as e.g. Marching Cubes [LC87] can do this for the mesh co-ordinates, however, there is no general method that delivers stable and well behaved texture coordinate parameterizations for varying iso-surface levels that would maintain undistorted appearances of the texture pattern. This is in particular important, if the texture exhibits a systematic or regular pattern. Also, for cases where transparent direct volume rendering is intended, the previously mentioned technique is not applicable. A consequent step is therefore to apply 3D textures when visualizing multivariate volumetric data. But how should these textures look?

3.1. Volumetric normal glyphs

Following conventional practice in volume rendering, objects are made visible by deriving some geometric property (local surface gradient, reflectance, opacity etc.) from scalar data in the 3D field and by applying some heuristic illumination and compositing model for rendering. In this process colour is usually a property that is predetermined as a result of object classification; obviously visual discrimination of e.g. bones and vessels is supported if they have different colours. Strictly speaking, the object class can be considered as a secondary nominal attribute per voxel. In order to load additional information from any tertiary attribute from the data set into the visualization, we propose manipulating the approximated normal vectors used for shading calculations during rendering. Any such normal vector manipulation should be performed in a systematic way so as to become apparent as an intentional visual cue rather than a sporadic artefact. The perturbation of the normal vector in direct volume rendering of objects can easily be controlled by some secondary parameter (e.g. blood flow, metabolic activity). Its systematic structure is according to our proposal based on a 3D normal map that contains volumetric normal vector glyphs; i.e. typical patterns of normal vector distributions give rise to some apparent visual pattern when applied to rendering of the volume data. Figure 1 (left) demonstrates an example of a volumetric normal map where the components of the normal vector glyphs are mapped to RGB colour during rendering for illustration purposes. This 3D normal map has a resolution of 128^3 voxels and it contains 16 spherical normal glyphs with a diameter of approximately 32 voxels. These spherical normal glyphs are rasterized into the 3D normal map at random positions and all voxels inside a spherical glyph contain a normal vector definition.

3.2. Normal mapping and blending

During rendering this normal map is used in an axis-aligned manner, but repeated periodically along all axes depending on the size of the original data volume and the desired rendering quality for the glyphs.

$$N = N_O + \beta N_G \tag{1}$$

Equation (1) denotes the simplest way in which the normal vector N_O from the original voxel model can be blended with the sampled glyph normal N_G from the normal map. Parameter β represents a tertiary attribute from the multivariate data set at the particular voxel location; it controls the weight of the glyph normal.

We developed a volume renderer based on fragment programs, that implements the volumetric normal mapping and blending according to (1). Using this volume renderer we visualized a teapot applying the normal map shown in Figure 1. Hereby, the modulating parameter β was chosen to be constant 1.0 to make the effect of the normal perturbation equally apparent accross the entire teapot. Evidently, the teapot shows clear bumps which can be both convex and concave, depending on which hemisphere of the glyph its surface intersects with. Volumetric normal mapping according to (1) is of limited use because the value of parameter β only can control how evidently the spherical normal glyph will be rendered; whereas the size of the bumps and their convexity is depending on where the object to be rendered intersects with the spherical normal glyphs.

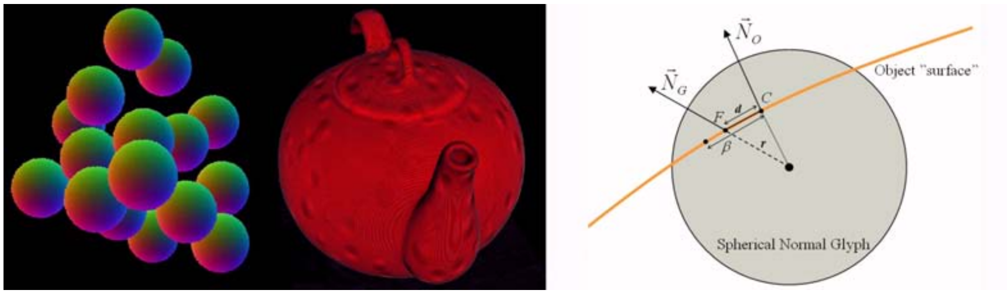

Figure 1: *A volumetric normal texture containing spherical normal glyphs (left). The same volumetric normal map applied in a high opacity volume rendering of a teapot (middle). Geometric relationship between the object's surface and a normal glyph in the 3D normal texture (right).*

3.3. Controlling apparent size of normal glyphs

In order to control the size and the convexity of the bumps we need to determine at fragment level, at what distance from the bump center the current fragment is positioned in order to eventually deny normal mapping to limit the size of the apparent bump. The required distance d from center C of the bump as seen from the observer for any fragment F in question is not available at fragment level. Therefore, we extend the 3D normal map by adding to each normal entry of a spherical glyph its distance r to the center of the sphere. This distance is stored as the alpha component of the RGBA texture. Utilizing symmetry of the spherical glyph, we can then within the fragment program determine the fragments actual distance from the bump's center as it would appear along the surface of the voxel object. If this distance exceeds some threshold defined by some secondary variable β, volumetric normal mapping will be denied and only the surface normal from the object will contribute to shading.

$$d = r sin(\alpha) \tag{2}$$

where

$$\alpha = cos^{-1}(\vec{N_O} \cdot \vec{N_G}) \tag{3}$$

Figure 1 (right) illustrates the geometric relationships between the normal vector from the object's surface N_O and the glyph's normal vector N_G as well as the relation between a thought object boundary surface cutting through a spherical normal glyph. Distance d can be calculated effortlessly inside the fragment program using (2) and (3).

Volumetric normal mapping can be suppressed by comparison of the fragment distance to the bump's center with some parameter β to control maximum bump size. Controlling the convecity of the bumps is done by evaluating the angle between object surface and the spherical glyph normal.

4. Results and Discussion

We applied our direct volume renderer for multivariate data initially to synthetic data sets. The first data set is a 256^3 sized volume into which we rasterized a sphere with a radius of approximately 120 voxels as the first parameter. As the secondary attribute we rendered 4 minor spheres with a radius of 60 voxels. Figure 2 shows the results of the rendering where we chose a high opacity surface rendering mode on the first attribute (i.e. no transparency). The picture shows the difference between simple normal blending (left) and the size and convexity controlled method (middle). In places where an attribute is found in the secondary attribute, the small volumetric normal glyphs become apparent as bumps on the surface of the object. We combined the same secondary data field containing 4 minor spheres with a real CT data volume. Figure 2 (right) shows how the result of the volumetric normal mapping appears in context of a real data set. It illustrates that our method is generally suitable to visualize additional attributes by introducing systematic volumetric normal vector perturbations. An advantage of our method compared to conventional 2D bump mapping or texture mapping is, that it can be implemented in a direct volume renderer. It does therefore not require reconstruction of explicit surface models. What Figure 2 does not show is that volumetric normal glyphs become generally evident also in low opacity rendering conditions. We consider this a valuable feature; when users interactively explore opacity transfer functions the secondary attribute will nevertheless be visible. The same holds true when apparent object surfaces grow or shrink due to interactive manipulation of segmentation thresholds. As in glyph based information visualization, the method presented in this paper is not suitable to visualize attributes on a continuous scale. However, for nominal attributes or ordinal attributes on a limited scale, volumetric normal mapping provides a new alternative to the use of colour or 2D textures. From a theoretical perspective, size control of the normal glyphs according to equation (2) is valid only if the surface of the primary object is almost pla-

Figure 2: *Volumetric bump mapping illustrated in a fully synthetic data set (left and middle). The same secondary attribute field is rendered in context with a medical data set (right). Spherical glyphs are visible only at the left forehead.*

nar with respect to the area of intersection with the spherical normal glyph. Since the normal glyphs are intended to introduce high-frequency visual features in comparison to the object upon which they are rendered, however, we consider this condition to be true for the intended applications. More work needs to be carried out that investigates how well differently, more complexly shaped volumetric normal glyphs can be used to convey information.

5. Acknowledgements

3D normal glyphs and 3D normal mapping were implemented into a shader-based volume renderer by Cezary Bloch as part of his master's thesis project [Blo06]. The medical data set was kindly provided by Tomasz Luksza.

References

[AM07] AKIBA H., MA K.-L.: An interactive interface for visualizing time-varying multivariate volume data. In *Proceedings of EuroVis07: Joint Eurographics - IEEE VGTC Symposium on Visualization* (2007).

[Blo06] BLOCH C.: *Rendering of Multivariate Volume Data*. Master's thesis, Uppsala University, Sweden, 2006.

[IFP97] INTERRANTE V., FUCHS H., PIZER S. M.: Conveying the 3d shape of smoothly curving transparent surfaces via texture. *IEEE Transactions on Visualization and Computer Graphics 3*, 2 (1997), 98–117.

[KD98] KINDLMANN G., DURKIN J. W.: Semi-automatic generation of transfer functions for direct volume rendering. In *IEEE Symposium on Volume Visualization* (1998), pp. 79–86.

[KHGR02] KNISS J., HANSEN C., GRENIER M., ROBINSON T.: Volume rendering multivariate data to visualize meteorological simulations: a case study. In *VISSYM '02: Proceedings of the symposium on Data Visualisation 2002* (Aire-la-Ville, Switzerland, Switzerland, 2002), Eurographics Association, pp. 189–ff.

[LC87] LORENSEN W. E., CLINE H. E.: Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, July 1987), vol. 21, ACM Press, pp. 163–169.

[Lev88] LEVOY M.: Display of surfaces from volume data. *IEEE Computer Graphics and Applications 8*, 3 (1988), 29–37.

[Lev91] LEVKOWITZ H.: Color icons: merging color and texture perception for integrated visualization of multiple parameters. In *VIS '91: Proceedings of the 2nd conference on Visualization '91* (Los Alamitos, CA, USA, 1991), IEEE Computer Society Press, pp. 164–170.

[MBM*05] MEHNERT A., BENGTSSON E., MCMAHON K., KENNEDY D., WILSON S., CROZIER S.: Visualisation of the pattern of contrast enhancement in dynamic breast mri. In *Proceedings of WDIC 2005* (2005), APRS, pp. 67–72.

[OKG*06] OELTZE S., KUSS A., GROTHUES F., HENNEMUTH A., PREIM B.: Integrated visualization of morphologic and perfusion data for the analysis of coronary artery disease. In *Proceedings of EuroVis06: Joint Eurographics - IEEE VGTC Symposium on Visualization* (2006), pp. 131–138.

[SLM02] STOMPEL A., LUM E. B., MA K.-L.: Feature-enhanced visualization of multidimensional, multivariate volume data using non-photorealistic rendering techniques. In *Proceedings of Pacific Graphics 2002* (Date October 09–11 2002), IEEE, pp. 1–8.

[Tay02] TAYLOR R.: Visualizing multiple scalar fields on the same surface. *IEEE Computer Graphics and Applications 22*, 2 (March-April 2002), 6–10.