Title
Feature-Enhanced Visualization of Multidimensional, Multivariate Volume Data Using Non-photorealistic Rendering Techniques

Authors
Lum, Eric
Ma, Kwan-Liu

Publication Date
2002

Peer reviewed

Feature-Enhanced Visualization of Multidimensional, Multivariate Volume Data Using Non-photorealistic Rendering Techniques

Aleksander Stompel Eric B. Lum Kwan-Liu Ma
University of California at Davis
{astompel,lume,ma}@cs.ucdavis.edu

Abstract

This paper presents a set of feature enhancement techniques coupled with hardware-accelerated non-photorealistic rendering for generating more perceptually effective visualization of multidimensional, multivariate volume data, such as those obtained from typical computational fluid dynamics simulations. For time-invariant data, one or more variables are used to either highlight important features in another variable, or add contextural information to the visualization. For time-varying data, rendering of each time step also takes into account the values at neighboring time steps to reinforce the perception of the changing features in the data over time. With hardware-accelerated rendering, interactive visualization becomes possible leading to increased explorability and comprehension of the data.

Keywords: animation, hardware-accelerated rendering, multidimensional data, multivariate data, non-photorealistic rendering, scientific visualization, streamlines, stroke based rendering, vector field, visual perception, volume rendering

Notes to reviewers: To help evaluate this work, all images should be viewed in color and the reviewers are encouraged to watch the animations that can be downloaded from: http://www.cs.ucdavis.edu/~ma/PG02/

1 Introduction

Direct volume rendering is a powerful 3D visualization technique which turns discrete sampled data obtained from numerical simulations or 3D images scanners into continuous imagery of the physical phenomena or structures under study. However, software volume rendering was too slow to make it a practical solution for most applications. As a result, since it was introduced in more than ten years ago [3, 25, 13], significant efforts have been made to accelerate the rendering calculations by using software optimizations [14, 12], parallel computers [19, 11], and hardware support [2, 24]. Recently, interactive volume rendering has become increasingly accessible to more people due to the hardware texturing support in most commodity graphics cards.

An active area of current research in volume visualization is the development of techniques for generating more perceptually effective visualizations. The efforts so far are mainly concerned with visualizing volume data of a single scalar or vector field by using non-photorealistic rendering (NPR) techniques [5, 8, 16, 17]. Little work has been done for the visualization of multidimensional and multivariate data such as time-varying data and multiple scalar and vector fields data, for example, commonly obtained from computational fluid dynamics (CFD) simulations.

Previous work in time-varying data visualization has mainly focused on data encoding [26, 20, 15], feature tracking [1, 27], and rendering efficiency [18]. The multivariate volume data visualization problems has begun to receive a lot of attention [7, 28, 31, 23, 4] but with an emphasis more on either color volume visualization or simultaneous visualization of multi-modality data.

A rather unexplored direction in multidimensional and multivariate volume data visualization is the use of the additional dimensions and variables available to generate more expressive visualization of specific aspects of the volume data set. The resulting enhancements either add contextual information to the visualization or help draw attention to selected features in the data. In this paper, we show how such enhancements can be integrated into non-photorealistic rendering of features in both temporal and spatial domains of the data. To demonstrate our enhanced NPR techniques, three time-varying data sets each of which consists of both scalar field and vector field data are used.

2 Non-Photorealistic Rendering

Before introducing the multivariate and temporal non-photorealistic enhancement techniques in Section 3, we first describe the implementation of our hardware texture volume renderer, followed by a discussion of the basic NPR techniques that have previously been applied to volume rendering [5, 17] and their implementation using a commodity graphics card such as an *Nvidia GeForce 3*. These NPR techniques including silhouette, warm to cool shading, gradient enhancement, and depth-based enhancement can be effective in enhancing spatial structures or can be used to de-emphasizing certain aspects of a visualization so that attention is drawn to other properties, such as the temporal characteristics of the data.

2.1 Hardware Texture Volume Rendering

Direct volume rendering can be accomplished by drawing a set of view-aligned polygons that sample a 3-D texture containing the volumetric data [30]. Using pixel textures allows a texel value to store coordinates into a second texture. Because scalar and diffuse lighting information can be encoded into a texel, transfer function and shading can be changed through the variation of a single texture [22]. Kniss et al. [10] describe how multi-dimensional transfer functions can be interactively specified and rendered with traditional lighting by using multi-textured/multi-pass rendering.

Our hardware-accelerated volume renderer makes extensive use of the multi-texturing capabilities of modern graphics cards but with the goal of producing non-photorealistic volume renderings. Multi-texturing allows several textures to be combined on a single polygon during the rendering process. By utilizing several separate volumetric textures that store scalar data value, gradient magnitude, and gradient direction, and combining them with properly adjusted color palettes, the non-photorealistic rendering techniques of interest can be done in hardware.

We utilize four texture units for each rendering pass, with two passes for every view aligned polygon. The first pass renders the shaded volume, while the second pass contains silhouette and specular contributions. For both rendering passes the four texture units are assigned to the same paletted texture data. The difference between the two passes occurs in the adjustment of the palettes used for each texture.

Each of the four texture units are assigned a different texture. The first texture consists of the original scalar values stored in a paletted texture with 8-bit precision. The second one stores the normalized gradient direction of each voxel. These values are encoded in an 8-bit paletted texture based on which of 240 direction vectors from the sides of a subdivided icosahedron each direction is most closely approximates [30]. The gradient direction information is used for lighting and silhouette operations. A third texture contains gradient magnitudes and is used for enhancing surfaces. The fourth is a 1-D texture for manipulating color and opacity based on the spatial properties of voxels.

2.2 Silhouette

Silhouette edge rendering is a widely used technique in non-photorealistic rendering. It can be used alone to provide a compact line-based representation or can be combined with other rendering techniques as a means to disambiguate depth relationships. Figure 1 shows visualization of turbulent vortex flow without and with silhouette enhancement.

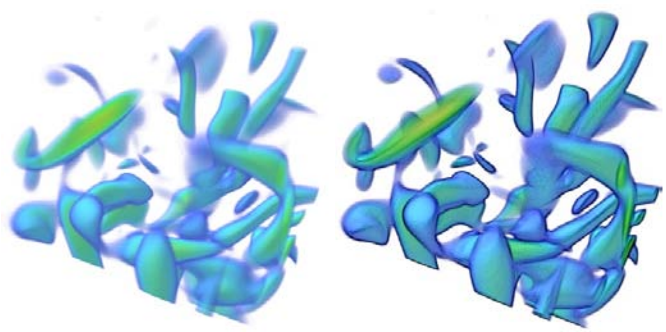


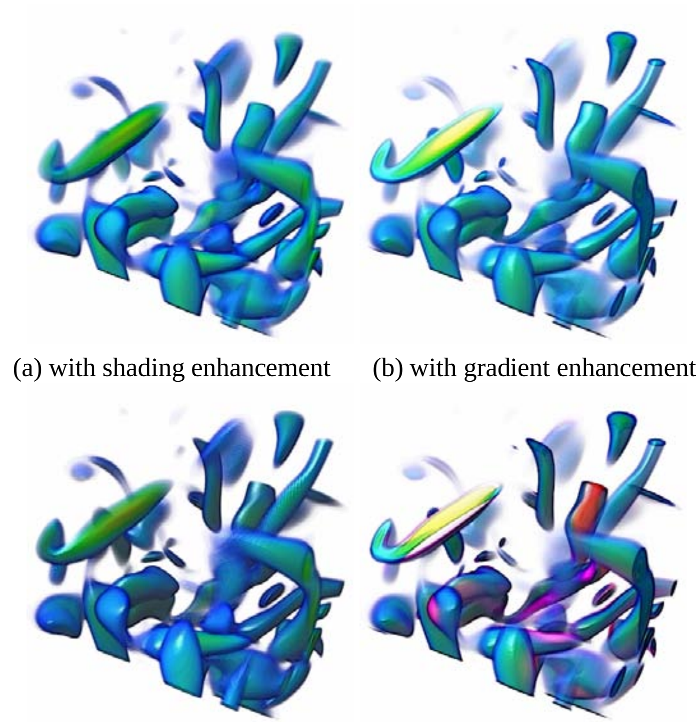
Figure 1. Enhanced visualizations by adding silhouette. Left: direct volume rendering. Right: with silhouette.

Silhouette edge rendering techniques can be broadly classified into object and image space algorithms. In object-space approaches dark lines are drawn in regions where surfaces have orientations perpendicular to the view direction. We accomplish this with multi-texturing, modulating texels with black where gradient is perpendicular to the view direction within a tolerance that can be adjusted by the user. Specifically, the silhouette rendering is implemented using a second texture pass that applies the silhouette and specular contribution. For this pass we use the scalar value texture in the first texture unit, and the gradient direction texture in the second. The desired combined texture has an opacity that depends on both the opacity of the voxel from the transfer function as well as degree to which that voxel's gradient is perpendicular to the viewing direction. For this pass, we therefore assign each palette entry in the gradient direction texture with an opacity that is the highest when the transformed gradient vector represented by that index is perpendicular to the viewing direction. Finally, the color for each entry is then assigned to be the desired color of the silhouette, typically black. This approach has the limitation that line thickness can be non-uniform, particularly when surfaces have orientations that are near-perpendicular to the view direction for regions that span a large screen area.

For a software implementation, we have used an image-based silhouette edge rendering method, which is a post-process of adding silhouette edges by applying image processing operations to the rendered volume. Screen-space approaches typically apply an edge detection mask to the z-buffer to detect silhouettes, however, in a volume rendering context z-value is not well defined. We have found that using the average z-value can produce reasonable results.

2.3 Shading

In order to make silhouette edges and specular highlights more visible, Gooch et al. [6] describe a warm to cool shading model that uses variation in color temperature to indicate lighting. By varying color hue rather than intensity to indicate illuminations, the technique reserves dark colors for silhouette edges, and bright saturated colors for highlights. Similarly we have implemented this technique by modulating voxel colors specified by the transfer function with a shading color that varies from warm to cool. By varying the saturation and intensity of the shading, the user is able to control the extent temperature is varied and the degree traditional intensity based illumination is present. Figure 2 (a) shows adding shading enhancement to the visualization displayed in Figure 1.



(a) with shading enhancement (b) with gradient enhancement

(c) with depth enhancement (d) temporal domain enhancement

Figure 2. Enhanced visualizations using NPR.

In hardware, we compute tone based lighting using the gradient direction texture in the second texture unit. Each index in this texture contains an index into a sampling of a normalized vector space. Each time a spatial viewing parameter is changed, the gradient direction palette is modified. For each index the dot product is calculated between the transformed vector represented by each index, and the light direction. Once the dot product is calculated, the color tone for that product is looked up in a tone shading colormap. The entry found in the colormap is finally stored in that palette for that texture index. The second texture unit is setup to modulate the color from the scalar data in the first texture unit. The second texture unit does not affect the alpha channel, since lighting does not influence opacity.

The user is able to interactively specify the colors used in tone shading by selecting a set of key color entries that are linearly interpolated across the colormap. By manipulating and shifting these key colors across colormap lighting parameters can be selected suited for the data set being visualized.

2.4 Gradient Based Enhancement

Gradients have been used for the enhancement of surfaces in volume rendering applications [13]. Since the transition between features in a volume tend to have the highest gradient magnitude, the enhancement of the opacity in these regions can help to clarify surfaces. In hardware, we have implemented surface enhancement with a texture unit that is assigned to a gradient magnitude texture. The user has the ability of specifying an arbitrary gradient opacity map that modulates the rendered texel. In cases where one wants to visualize structures that exist between two materials of similar scalar value, enhancing regions with the highest gradient would emphasize the wrong features. By specifying an arbitrary gradient map, one can reduce opacity in these high gradient regions.

It is helpful to allow for specifying separate gradient enhancing functions for the specular and silhouette rendering pass. Specular lighting and silhouettes are usually associated with surfaces and are less meaningful when applied to the solid semi-transparent regions often associated with direct volume rendering. Thus a map can be set such that specular and silhouette rendering only occurs on the surfaces. Figure 2 (b) shows the result of adding gradient enhancement.

2.5 Depth Based Enhancement

Another important visual cues is depth. For example, color can be manipulated based on distance from the viewer to improve depth perception. Aerial perspective has been used by painters to convey depth through the variation of color hue and value. Typically warmer hues are used for the foreground and become cooler in the background. In addition, color values tend to become lighter and less intense with distance. This can be implemented in hardware using a one dimensional texture that modulates the color of the rendered volume along some direction. The depth cues provided by the variation in color are evident in Figure 2 (c) where the warmer colored foreground tubes appear closer. This can be contrasted with Figure 2 (a) where the spatial relationship between vortices is less clear.

3 Enhancement Techniques for Multidimensional and Multivariate Volume Data

Almost all data sets obtained from numerical modeling of physical phenomena or chemical process are multidimensional in nature and record multiple scalar and vector properties at each data point. In our study, we have experimented with several different approaches with the goal of generating more rich, expressive visualizations. The first approach is to present more than one property in a single visualization. In some disciplines like medical imaging, it is often advantageous to make simultaneous visualization of data sets from different modalities like CT, MRI, and PET. In CFD, it is a common practice to plot, for example, streamlines and isosurfaces in a single image. In our study, we have investigated how to realize this capability using NPR to improve the perceptual effectiveness and thus the clarity of the resulting pictures. Often, one property is shown to provide some contextual information of the visualization.

The other basic approach to multidimensional and multivariate feature enhancement is to use one or more properties of the data in rendering of another property of the data. There is thus a large number of possible combinations that can be used to achieve various types or levels of enhancement. That is why all techniques presented here must be hardware-acceleratable to maintain the needed interactivity for the user to freely explore different combinations for specific feature enhancements. Since this interactive data exploration technique is new to scientists, we expect them to find some surprising results which might lead to new discoveries.

3.1 Stroke Based Rendering

Rather than drawing lines, tubes or ribbons to illustrate a vector field like in conventional flow visualization, we have experimented with stroke based rendering, an automatic approach to creating non-photorealistic imagery by placing discrete elements called strokes. Many stroke-based rendering algorithms and styles have been proposed but they have been mainly introduced for artistic rendering. For illustrating vector field, it is required each stroke gives an indication of direction. Color and transparency can be used to add other information about the flow field. Most importantly, the appearance of the strokes should depend if the strokes are there to provide supplemental information or they are the main features of interest. Using NPR allows us to more easily reflect this difference in the resulting visualization.

A key task in stroke based rendering is seed points selection. Much work has been done [29, 9, 21] on providing aesthetically pleasing streamlines through careful selection of seed points. Typically the emphasis is on producing a visually uniform density of streamlines in the final image. Our approach is to select seeds in a way such that the final distribution of field lines has density proportional to magnitude of the underlying field selected by the scientist.

3.2 Spatial Domain Enhancement

In our study so far, spatial domain enhancement is mainly concerned with enhanced vector field visualization, in particular, by superimposing the visualization of other properties in the data. Stroke based rendering is used for depicting both vector direction and magnitude. The ability for the scientist to interactively control the length and appearance of streamlines allow more intuitive exploration of the vector field. When the lines become dense and are displayed along with other properties, using NPR can help tremendously on clarifying the structure of field lines and their spatial relationship with other displayed features.

Figure 3 demonstrates simultaneous vortex tube visualization and streamline visualization with a pencil drawing style. The direction of the streamline is indicated with its fading tail. The left-most image with shorter strokes shows direction information better while the middle image with longer strokes reveals structural information better. By coloring the streamlines, more information about the flow field is provided. In the right-most image, the streamlines are colored according to velocity magnitude. However, choosing the saturation and brightness levels for the lines must be done carefully since densed, intertwined color lines generally do not present structural information well. Note that the vortices are shaded gray to provide a context for the field line visualization.

Figure 4 displays simultaneous visualizations of vorticity field and velocity field. Rendering of vorticity magnitude shows the strength of the pair of wake vortices while rendering streamlines as pencil-drawing-like strokes reveals local flow direction. Comprehension of the velocity field can be greatly improved by interactively changing the density of the strokes as well as the viewing angle. The direction of each stroke gives an indication of the local flow direction.

3.3 Temporal Domain Enhancement

The general practice in visualizing time-varying volume data is to create an animation by rendering individual time steps independent of other time steps. The enhancement techniques we have described so far can be applied to such

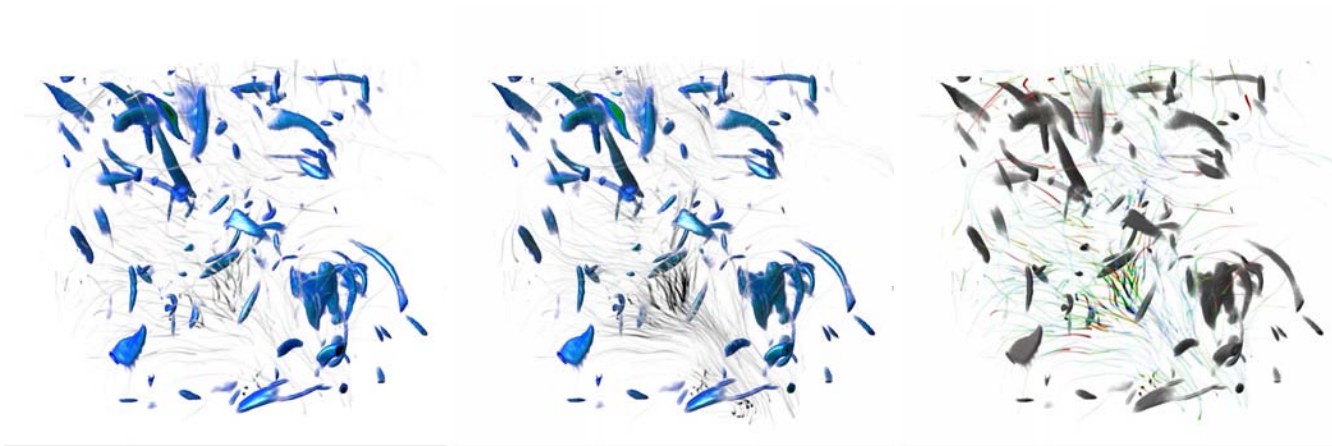


Figure 3. Simultaneous visualization using volume rendering of vorticity field and stroke-based rendering of velocity field. Left: pencil drawing style. Middle: long strokes. Right: color strokes.

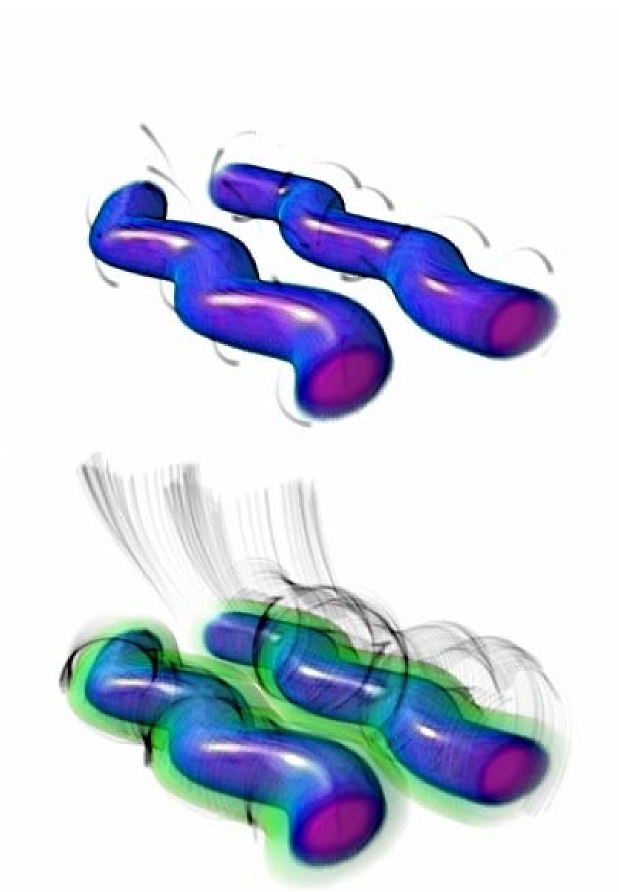
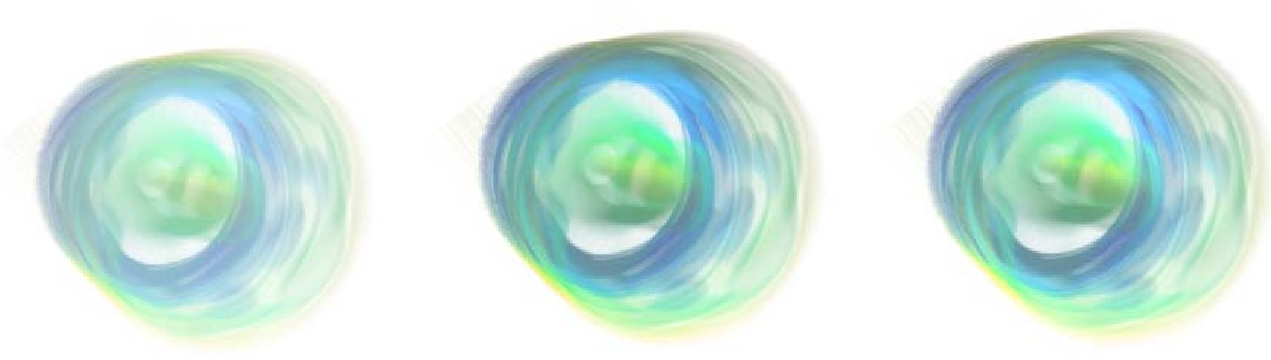


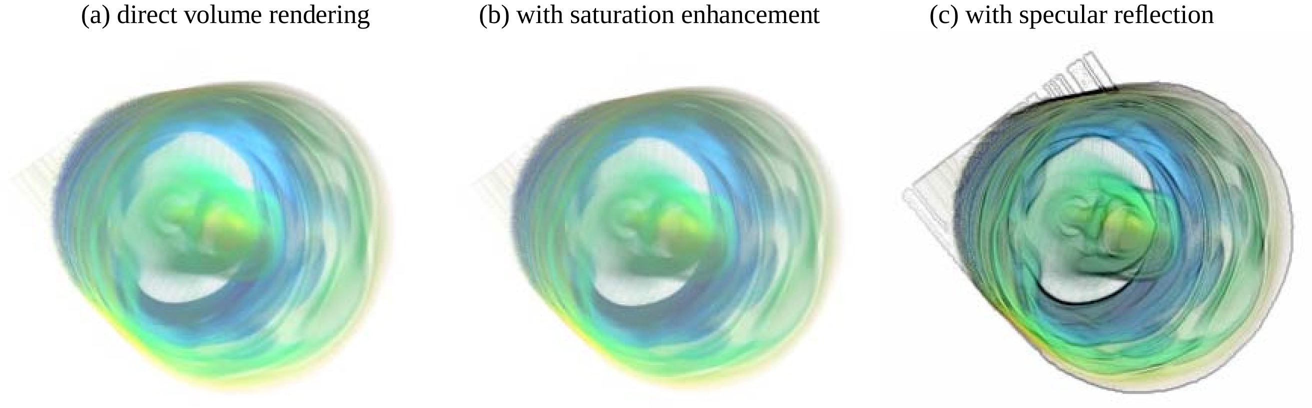
Figure 4. Simultaneous visualization of wake vortices. Top: using spares, short strokes. Bottom: using denser, longer strokes.

a rendering process to enhance the content of each frame. Care must be taken since artifacts might be introduced into the animation as a result of adding spatial enhancement using NPR. The temporal domain enhancement we introduce here is free of this artifact problem since we achieve enhancement by using information from immediately neighboring time steps, which ensures coherence from one time step to the next.

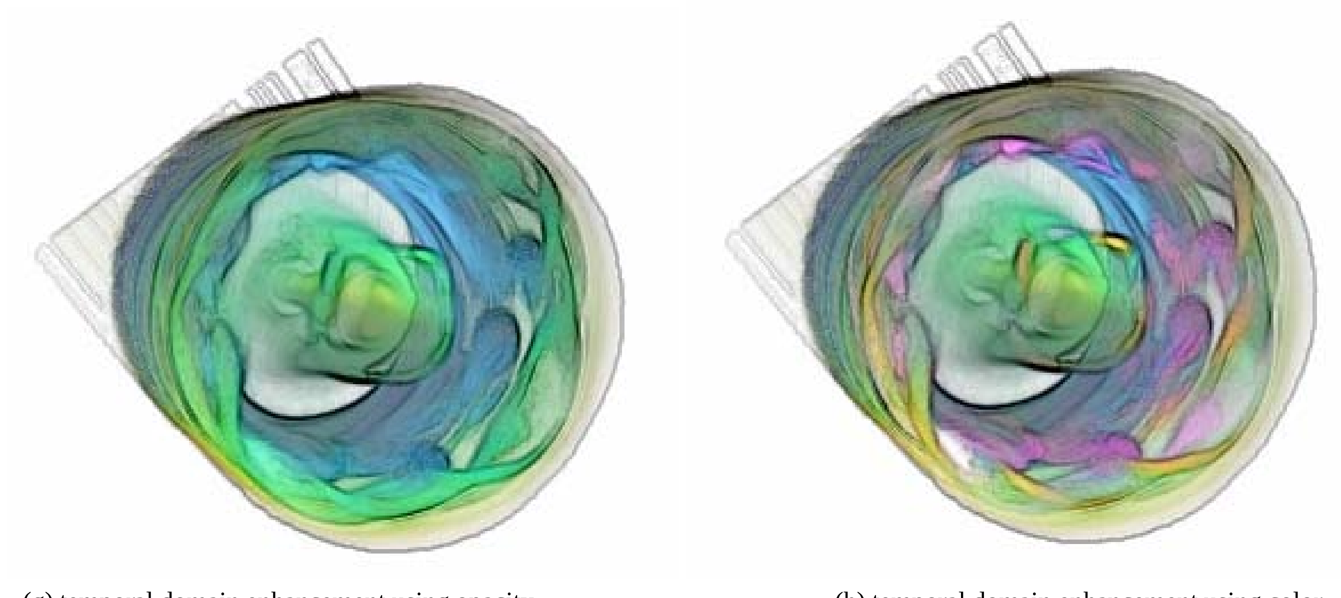
The basic approach is to use the data at previous time step and next time step for the rendering of current time step. The example presented here shows an enhancement of features varying faster over time. Such features can be identified by computing the gradient values in the time dimension and enhanced by modifying shading according to the gradient values. The resulting visualization can draw the viewers' attention to the fast varying features. A similar enhancement can be applied to slowly varying features, changes in direction, changes at a particular rate, etc. Figure 5 (g) and (h) show such an enhancement while (a)-(f) illustrating the results of adding individual NPR techniques. In (g), opacity is adjusted according to the changing rate so the faster changing features become more opaque. In (h), color is also adjusted so that redish parts are the fast varying features. Finally, Figure 6 show selected frames from an animation created using such an enhancement to drawing more attention to the fast varying features. Figure 2 (d) shows the result of applying the same enhancement highlighting the fast varying parts in a still image. Even though the changed color and opacity mapping is only applied to selected features, the resulting visualization would not become misleading if the scientist has a full (and interactive) control of the enhancement.



(a) direct volume rendering (b) with saturation enhancement (c) with specular reflection



(d) with gradient enhancement (e) with depth enhancement (f) with silhouette



(g) temporal domain enhancement using opacity (h) temporal domain enhancement using color

Figure 5. A complete example using a time-varying data set obtained from a CFD simulation.

Figure 6. Selected frames from an animation with temporal-domain enhancement. Time progresses from top to bottom, left to right.

4 Conclusions

Increasingly, NPR will be used in scientific visualization to make more perceptually-effective illustrations. Our study demonstrates that additional enhancements can be integrated into conventional NPR techniques to increase clarity and information level of the resulting visualization in both spatial and temporal domains.

It is important to emphasize that the interactivity is the key to more expressive and understandable visualization. That is, scientists must be provided with interactive control of rendering and visualization parameters to receive immediate visual feedbacks of varying the parameters. Only in this way, the enhancements made will be meaningful and acceptable by the scientists.

For real-world applications, we must also address the large data problem. The cost of always plotting streamlines at the data resolution can be prohibitively high. One feasible approach is to precompute streamlines and store them in an hierarchical fashion such that the visualization can be progressively refined.

Our study so far has only used regular-grid data sets. Many CFD simulations use irregular grid which presents new challenges to our approach which relies on hardware acceleration. Similarly, a viable solution is to precompute and store as many streamlines as possible for interactive visualization. Our future work will thus focus on the development of a hierarchical streamline organization for more efficient storage and retrieval.

Acknowledgments

This work has been sponsored by the NSF PECASE, NSF LSSDSV, and DOE SciDAC. The authors are grateful to Dr. GuoWei He at ICASE/NASA LaRC, Dr. Charlie Zheng at Kansas State University, and Dr. Pierre Lallemand at ASCI of CNRS in France for providing the test data sets.

References

[1] D. C. Banks and B. A. Singer. A predictor-corrector technique for visualizing unsteady flow. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):151–163, 1995.

[2] B. Cabral, N. Cam, and J. Foran. Accelerated volume rendering and tomographic reconstruction using texture mapping hardware. In *Proceedings Volume Visualization Workshop*, pages 91–98, October 1994.

[3] R. A. Drebin, L. Carpenter, and P. Hanrahan. Volume rendering. In *Proceedings of SIGGRAPH '88*, pages 65–74, 1988.

[4] D. Ebert, C. Morris, P. Rheingans, and T. Yoo. Designing effective transfer functions for volume rendering from photographic volumes. *IEEE Transactions in Visualization and Computer Graphics*, 8(2), April-June 2002.

[5] D. Ebert and P. Rheingans. Volume illustration: Non-photorealistic rendering of volume models. In *Proceedings of IEEE Visualization 2000 Conference*, pages 195–202, October 2000.

[6] A. Gooch, B. Gooch, P. Shirley, and E. Cohen. A non-photorealistic lighting model for automatic technical illustration. In *SIGGRAPH '98 Conference Proceedings*, pages 447–452, July 1998.

[7] P. Hastreiter and T. Ertl. Integrated registration and visualization of medical image data. In *Proceedings of Computer Graphics International '98*, pages 78–85, 1998.

[8] V. Interrante. Illustrating surface shape in volume data via principal direction-driven 3d line integral convolution. In *SIGGRAPH '97 Conference Proceedings*, pages 109–116, August 1997.

[9] B. Jobard and W. Lefer. Creating evenly-spaced streamlines of arbitrary density. In W. Lefer and M. Grave, editors, *Visualization in Scientific Computing '97. Proceedings of the Eurographics Workshop in Boulogne-sur-Mer, France*, pages 43–56, Wien, New York, 1997. Springer Verlag.

[10] J. Kniss, G. Kindlmann, and C. Hansen. Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets. In *Proceedings of Visualization 2001 Conference*, 2001.

[11] P. Lacroute. Real-Time Volume Rendering on Shared Memory Multiprocessors Using the Shear-Warp Factorization. In *Proceedings of Parallel Rendering Symposium*, pages 15–22, 1995.

[12] P. Lacroute and M. Levoy. Fast volume rendering using a shea-warp factorization of the viewing transformation. In *SIGGRAPH '94 Conference Proceedings*, pages 451–458. ACM SIGGRAPH, 1994.

[13] M. Levoy. Display of Surfaces from Volume Data. *IEEE Computer Graphics and Applications*, pages 29–37, May 1988.

[14] M. Levoy. Efficient Ray Tracing of Volume Data. *ACM Transactions on Graphics*, 9(3):245–261, July 1990.

[15] E. Lum, K.-L. Ma, and J. Clyne. Texture hardware assisted rendering of time-varying volume data. In *Proceedings of Visualization 2001 Conference*, October 2001.

[16] E. B. Lum and K.-L. Ma. Nonphotorealistic rendering using watercolor inspired textures and illumination. In *Proceedings of Pacific Graphics 2001*, October 2001.

[17] E. B. Lum and K.-L. Ma. Hardware-accelerated parallel non-photorealistic volume rendering. In *Proceedings of the International Symposium on Nonphotorealistic Rendering and Animation*, June 2002.

[18] K.-L. Ma and D. Camp. High performance visualization of time-varying volume data over a wide-area network. In *CDROM Proceedings of Supercomputing 2000 Conference*, November 2000.

[19] K.-L. Ma, J. S. Painter, C. Hansen, and M. Krogh. A Data Distributed Parallel Algorithm for Ray-Traced Volume Rendering. In *Proceedings of Parallel Rendering Symposium*, pages 15–22, 1993. San Jose, October 25-26.

[20] K.-L. Ma and H.-W. Shen. Compression and accelered rendering of time-varying volume datasets. In *Proceedings of the Workshop on Computer Graphics and Virtual Reality, 2000 International Computer Symposium*, December 6-8 2000.

[21] X. Mao, Y. Hatanaka, H. H., and A. Imamiya. Image-guided streamline placement on curvilinear grid surfaces. In *Proceedings of Visualization'98 Conference*, pages 135–142, 1998.

[22] M. Meissner, U. Hoffmann, and W. Strasser. Enabling classication and shading for 3d texture mapping based volume rending using opengl and extensions. In *IEEE Visualization '99 Conference Proceedings*, 1999.

[23] S. Muraki, T. Nakai, and Y. Kita. Basic research for coloring multichannel mri data. In *Proceedings of Visualization 2000 Conference*, 2000.

[24] H. Pfister, J. Hardenbergh, J. Knittel, H. Lauer, and L. Seiler. The volumepro real-time ray-casting system. In *Proceedings of SIGGRAPH '99*, pages 251–260, 1999.

[25] P. Sabella. A rendering algorithm for visualizing 3d scalar fields. In *Proceedings of SIGGRAPH '88*, pages 51–58, 1988.

[26] H.-W. Shen, L.-J. Chiang, and K.-L. Ma. A Fast Volume Rendering Algorithm for Time-Varying Fields Using a Time-Space Partitioning (TSP) Tree. In *Proceedings of the IEEE Visualization '99 Conference*, pages 371–278, 1999.

[27] D. Silver and X. Wang. Tracking and visualizing 3d turbulent features. *IEEE Transactions on Visualization and Computer Graphics*, 3(2), June 1997.

[28] I. Takanashi, E. Lum, K.-L. Ma, J. Meyer, B. Hamann, and A. Olson. Segmentation and 3d visualization of high-resolution human brain cryosections. In *Proceedings of Visualization and Data Analysis 2002 Conference*, January 2002.

[29] G. Turk and D. Banks. Image-guided streamline placement. *Computer Graphics*, 30(Annual Conference Series):453–460, 1996.

[30] A. Van Gelder and U. Hoffman. Direct volume rendering with shading via three-dimension textures. In *ACM Symposium on Volume Visualizatrion '96 Conference Proceedings*, 1996.

[31] B. Wilson, E. B. Lum, and K.-L. Ma. Interactive multi-volume visualization. In *Proceedings of ICCS 2002 Workshop on Computer Graphics and Geometric Modeling*, April 2002.