Visualization Viewpoints

Top Scientific Visualization Research Problems

Scientific visualization as currently understood and practiced is still a relatively new discipline. As a result, we visualization researchers are not necessarily accustomed to undertaking the sorts of self-examinations that other scientists routinely undergo in relation to their work. Yet if we are to create a disciplinary culture focused on matters of real scientific importance and committed to real progress, it is essential that we ask ourselves hard questions on an ongoing basis. What are the most important research issues facing us? What underlying assumptions need to be challenged and perhaps abandoned? What practices need to be reviewed? In this article, I attempt to start a discussion of these issues by proposing a list of top research problems and issues in scientific visualization.

"Study the science of art and the art of science"—Leonardo da Vinci

Scientists and mathematicians have a long tradition of creating lists of important unsolved problems both to focus the field's attention and to provide a forum for discussion. Perhaps the most famous list of unsolved problems is David Hilbert's list of 23 problems that he proposed in 1900 (http://mathworld.wolfram.com/HilbertsProblems/html). More recently, Stephen Smale proposed a list of 18 outstanding problems in mathematics.[1] These lists include important problems whose pursuit has been crucial to the development of the field. Such lists continue to be created in many areas of science and mathematics and help to motivate future research (see http://www.geocities.com/ednitou for a Web site with links to many such lists).

Because computer science is such a new discipline and computer graphics even newer, it wasn't until 1966 that Ivan Sutherland created the first list of unsolved problems in computer graphics.[2] Jim Blinn and Martin Newell created a second list in 1977 (see http://www.siggraph.org/publications/newsletter/v33n1/columns/conf.html). Additional lists were created by Paul Heckbert in 1987 and by Jim Blinn in 1998.[3,4] (Interestingly, the majority of these list makers, including myself, were either professors—Sutherland, Newell—at or graduated from—Newell, Blinn—the University of Utah.

The field of scientific visualization is newer still, "launched" only in 1987 by the National Science Foundation report, *Visualization in Scientific Computing*.[5]

(Here I mean the discipline employing computational means, not simply visualizing science, which is as old as science itself.) In 1994, Larry Rosenblum edited a special issue of *IEEE Computer Graphics and Applications* on research issues in scientific visualization, focusing on recent advances and emerging topics in visualization (vol. 14, no. 2, Mar./Apr. 1994). However, not until 1999 did Bill Hibbard create his list of top 10 visualization problems.[6] Hibbard organized his list within the broad categories of

- visual quality,
- integration,
- information,
- interactions, and
- abstractions.

As such, it addressed many problems and issues facing the discipline. While I believe Hibbard's list to be an important beginning—and while I also note that some of the items in Sutherland's original 1966 list are still unsolved and reappear on my list—I also believe it is important to refresh these lists regularly and to add new viewpoints with regard to visualization research problems and issues. In this way, we may motivate visualization researchers to think either about new problems or about persistent problems in a new way.

Top scientific visualization research problems

I have been assembling my own list of the most important issues facing researchers in scientific visualization, a list that represents my personal view of the field. In the last year or so, I have been presenting my ideas and updating my list based partly on the feedback I've obtained (see Figure 1). My strongest concern in creating and presenting this list is not to impose my own ideas on the field but rather to start a discussion about important research issues within scientific visualization. I recently participated in an IEEE Visualization 2004 panel proposal called, "Can We Determine the Top Unresolved Problems of Visualization?" and note views among the authors varied widely.[7] This is as it should be. The most important thing is that researchers formulate positions and that their positions be disseminated and discussed. To be clear, the items on my list are not all "problems"; some are possible "directions"

Chris Johnson

University of Utah
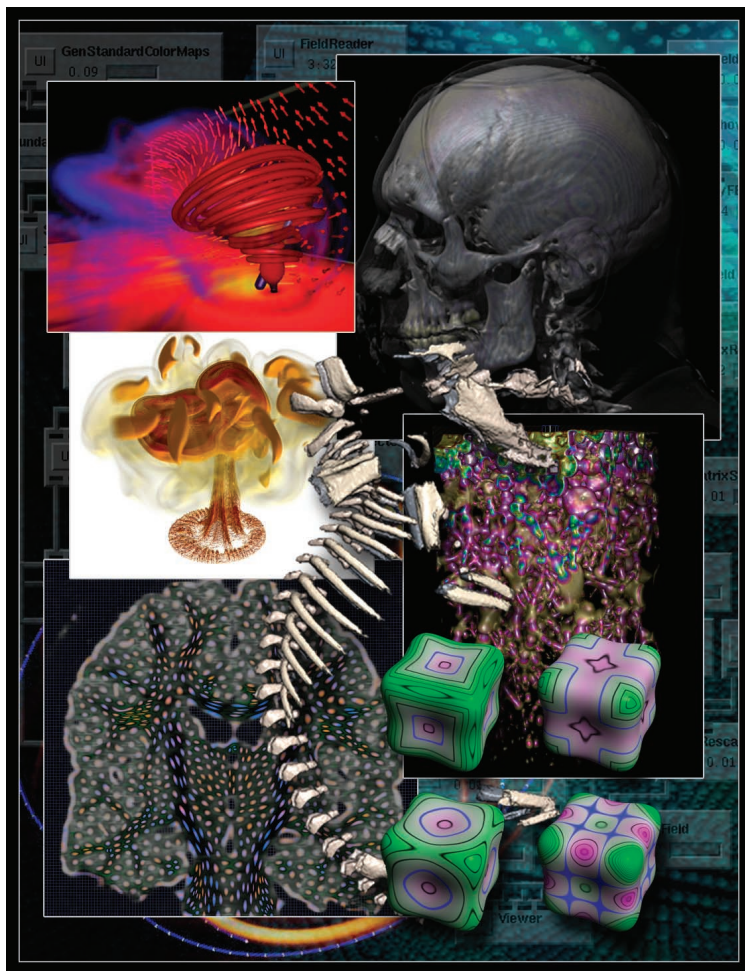
1 Visualization research results from the University of Utah's Scientific Computing and Imaging Institute (http://www.sci.utah.edu).

(such as items 8 and 10) and some pertain to "best practices" (such as items 1 and 2). However, I think they all are important to think about and deserve inclusion. I note that my list is not ranked; however, I am sure that readers will easily identify their most important issues—and tell me if I neglected to include their favorites in my final list.

Here, then, is my list of the top problems and issues in visualization research.

1. Think about the science

Too often, creators of visualization technology do not spend enough (or indeed any) time endeavoring to understand the underlying science they are trying to represent, just as application scientists sometimes create crude visualizations without understanding the algorithms and science of visualization. To establish mutually beneficial peer relationships with application scientists and to create effective visual representations, visualization scientists need to spend more time understanding the underlying science, engineering, and medical applications. The benefits of working directly with application scientists are enormous, and yet all too often visualization scientists "hide" from end users. There is no substitute for working side-by-side with end users to create better techniques and tools for solving challenging scientific problems.[8]

2. Quantify effectiveness

In 1993, during his keynote address at the IEEE Visualization 93 Conference, Fred Brooks said that "scientific visualization is not yet a discipline, although it is emerging as one. Too often we still have an ad hoc technique and rules of thumb." The majority of papers in visualization involve new techniques for characterizing scalar, vector, or tensor fields. However, the new techniques are rarely compared with previous techniques, and their effectiveness is seldom quantified by user studies. Fortunately, the importance of user studies in visualization research is now being recognized (see the excellent article by Robert Kosara and his colleagues for this magazine: "Thoughts on User Studies: Why, How, and When"[9]). Unfortunately, it is also too rarely the case that the effectiveness of new methods is quantified within the scientific visualization literature. If we wish visualization to become a scientific enquiry, visualization scientists need to understand and use the scientific method, employing its steps:

- Observation and description of a phenomenon or group of phenomena.
- Formulation of a hypothesis to explain the phenomena.
- Use of the hypothesis to predict the existence of other phenomena or to predict quantitatively the results of new observations.
- Evaluation of the proposed methods and quantification of the effectiveness of their techniques.

3. Represent error and uncertainty

When was the last time you saw an isosurface with "error bars" or streamlines with "standard deviations" or volume visualizations with representations of confidence intervals? With few exceptions, visualization research has ignored the visual representation of errors and uncertainty for 3D visualizations. However, if you look at highly peer-reviewed science and engineering journals, you will see that the majority of 2D graphs represent error or uncertainty within the experimental or simulated data.

Why the difference? Clearly, if it is important to represent error and uncertainty in 2D, it is equally important to represent error and uncertainty in 2D and 3D visualizations. It is also often important to quantify error and uncertainty within new computer graphics techniques (see my previous Visualization Viewpoint article in the Sept./Oct. 2003 issue of *IEEE Computer Graphics and Applications* for further discussion of this subject[10]).

4. Perceptual issues

Research on the human visual system is vast, yet visualization researchers rarely study or apply what is known about the visual system when designing visualization techniques. The computer graphics and information visualization communities may be ahead in this regard, but there is still much to be gained by all groups in studying the biophysics and psychophysics of the visual system.[11]

5. Efficiently utilizing novel hardware architectures

Hardware and systems software goes well beyond the ever-present pain of platform and compiler portability, device driver versions, and other mundane implementation issues. Modern visualization researchers have unprecedented graphics power available on inexpensive desktop workstations, but effectively utilizing that power can be a challenge. Visualization algorithms should strive for utmost efficiency for targeted platforms. Examples of algorithms that achieve such efficiency include RTRT, the Real Time Ray Tracer, which leverages the CPU, and algorithms that leverage graphics hardware (GPUs).

Research into the most appropriate mapping between special-purpose hardware, such as GPUs, and general-purpose processors is an area that will see much advancement in the near future. How can we ensure the integrity of images when using low-cost graphics hardware that is often designed for games? How can we harness the ever-increasing power available on GPUs with relatively limited increases in CPU bus bandwidth? How can we creatively use new graphics hardware features for new visualization algorithms and efficiently utilize cost-effective clusters for interactive visualization?

Lately there has been an explosion of papers and codes written on taking advantage of GPUs. In order to make GPUs more effective, we need to make them easier to program so that we can readily integrate them into visualization and graphics algorithms and tools.

6. Human–computer interaction

Effective human–computer interaction was on Sutherland's 1966 list.[2] HCI continues to be one of the top research and development goals for both visualization and computer graphics. I cannot begin to address the importance of effective interaction, much less details about how to achieve it in such a short article, especially given that HCI is a field unto itself. A starting place might be Ben Shneiderman's visual-information-seeking mantra: "Overview first, zoom and filter, then details-on-demand."[12] Two recent papers by Andries van Dam and his colleagues discuss the overall progress in interaction and provide comments on research challenges.[13,14]

7. Global/local visualization (details within context)

Currently, most graphical techniques emphasize either a global or local perspective when visualizing vector or scalar field data, yet ideally one wishes for simultaneous access to both perspectives. The global perspective is required for navigation and development of an overall gestalt, while a local perspective is required for detailed information extraction. Most visualization methods display either global variations, as is the case with line integral convolution and other dense vector field visualization methods, or local variations, as occurs in the use of streamlines.

When one uses a global operation, such as drawing a vector at every cell, it is impossible to navigate due to the visual occlusion of the many vectors. However, local methods such as the vector rake, which avoid this occlusion by limiting the sampling rate, fail to give an overall view of the underlying field. Furthermore, it is easy to miss interesting phenomena during such exploration. What we lack are a multitude of methods that convey details within an overall context.

8. Integrated problem-solving environments (PSEs)

Visualization is now most often seen as a postprocessing step in the scientific computing pipeline (geometric modeling → simulation → visualization). However, scientists now require more from visualization than a set of results and a tidy showcase in which to display them. The 1987 National Science Foundation Visualization in Scientific Computing workshop report poses the problem in these terms:

> Scientists not only want to analyze data that results from super-computations; they also want to interpret what is happening to the data during super-computations. Researchers want to steer calculations in close-to-real-time; they want to be able to change parameters, resolution or representation, and see the effects. They want to drive the scientific discovery process; they want to interact with their data.
>
> The most common mode of visualization today at national supercomputer centers is batch. Batch processing defines a sequential process: compute, generate images and plots, and then record on paper, videotape or film.
>
> Interactive visual computing is a process whereby scientists communicating with data by manipulating its visual representation during processing. The more sophisticated process of navigation allows scientists to steer, or dynamically modify computations while they are occurring. These processes are invaluable tools for scientific discovery.[5]

Although these thoughts were reported more than 15 years ago, they express a very simple and still current idea: scientists want more interaction (see item 6 on the list) between modeling, simulation, and visualization than is currently made possible by most scientific computing codes. The scientific investigation process relies heavily on answers to a range of "what if?" questions. Integrated PSEs that tightly couple interactive visualization techniques with geometric modeling and simulation techniques allow these questions to be answered more efficiently and effectively and thus help to guide the investigation as it occurs. Such PSEs can also provide computational steering and more interactive modes of investigation. Integration also requires that we develop improved tools and techniques for managing visualizations. Similarly, while VTK, the Visualization Toolkit, is a great first step, integration requires further research in visualization software architecture.

9. Multifield visualization

Computational field problems such as computational fluid dynamics (CFD), electromagnetic field simulation,

and weather modeling—essentially any problems whose physics can be modeled effectively by ordinary and/or partial differential equations—constitute the majority of computational science and engineering simulations. The output of such a simulation might be a single field variable (such as pressure or velocity) or, more commonly, a combination of fields involving a number of scalar fields, vector fields, or tensor fields. Similarly, data collected experimentally is often multifield in nature (and from multiple sources). Scientific visualization researchers have concentrated on effective ways to visualize large-scale computational and experimental fields. However, much current and previous visualization research has focused on methods and techniques for visualizing a single field variable (such as the extraction of a single scalar field variable as an isosurface). While single-variable visualization can satisfy the needs of the user, it is clear that it would also be useful to be able to effectively visualize multiple fields simultaneously and to visualize interaction between multiple fields.

10. Integrating scientific and information visualization

The amount of information available to scientists from large-scale simulations, experiments, and data collection is unprecedented. In many instances, the abundance and variety of information can be overwhelming. The traditional method for analyzing and understanding the output from large-scale simulations and experiments has been scientific visualization. However, an increasing amount of scientific information collected today has high dimensionality and is not well suited to treatment by traditional scientific visualization methods. To handle high-dimensional information, so-called information visualization techniques have emerged. There is now a growing community of information visualization scientists.

Curiously, the information visualization and scientific visualization communities have evolved separately and, for the large part, do not interact (see the May/June 2003 Visualization Viewpoints article).[15] As such, a significant gap has developed in analyzing large-scale scientific data that has both scientific and information characteristics. The time has come to break down the artificial barriers that currently exist between information and scientific visualization communities and work together to solve important problems. A simple example where scientific and information visualization techniques could have an immediate positive benefit to the application scientist is in analyzing, understanding, and representing error and uncertainty in complex 3D simulations (see item 3 and my earlier Visualization Viewpoint article[10]).

11. Feature detection

Analysis of complex, large-scale, multidimensional data is recognized as an important component in many areas, including computational fluid dynamics, computational combustion, and computational mechanics. Modern high-performance computers have speeds measured in teraflops and produce simulation data set sizes measured in gigabytes to terabytes and even petabytes. With such large-scale data, locating and representing important features within the data is essential. Feature detection involves both application-specific and application-independent aspects. Application-specific features, such as finding and tracking vortices within a CFD data set, require a close understanding of the underlying science (see item 1). Application-independent aspects are necessary to compare and track evolution of features efficiently and robustly. While there has been some work in categorizing features, there is still work to be done. General feature tracking is still an unsolved problem.

12. Time-dependent visualization

Currently, most interactive visualization techniques involve static data. The predominant method for visualizing time-dependent data is first to select a viewing angle, then to render time steps offline and play the visualization back as a video. While this often works adequately for presentational purposes, the lack of ability to engage in interactive engagement and exploration undermines the effectiveness and relevancy of investigative visualization. While there are a few recent examples of interactive time-dependent visualization techniques, there could be considerable improvement in this area.[16]

13. Scalable, distributed, and grid-based visualization

The available graphics throughput in PC graphics cards continues to grow. At the same time, other powerful graphics facilities are becoming available as part of grid-based computing systems. Missing are effective ways to tap into the available graphics capabilities of these distributed graphics systems to create scalable visualizations. It is clear that we need innovation on all fronts: hardware for plugging in multiple graphics PC cards, software to efficiently coordinate distributed visualization resources, and scalable algorithms to effectively take advantage of such distributed visualization resources.

14. Visual abstractions

Hibbard is entirely correct to emphasize the importance and necessity of effective visual abstractions. As he says, we need to

> … define effective abstractions for the visualization and user interaction process. Examples include the relational and field data models for information being visualized, the data flow model of the visualization process, mathematical models of human visual perception, mathematical models of interaction, models of users and their tasks, and more general models of computing and distributed computing. Effective abstractions have generality where it is needed but also make limiting assumptions that permit efficient and usable implementations. In addition to its practical consequences, this is a foundational problem for visualization.[6]

While this item could easily be absorbed within other items on my list, in particular items 10 and 15, it is so important that it deserves its own bullet.

15. Theory of visualization

As I inferred in item 14, we must engage what Hibbard calls "foundational problem[s]" in our field. Finally, therefore, I think it is time that we start to explore and articulate a foundation and theory of visualization. A theory of visualization is certainly nascent, though some would say it would be premature to concentrate on its development. It may indeed be too soon to *finish* such a theory, but I am convinced that there is no time like the present to start. In computer graphics, efforts to develop a theory of rendering have proceeded on two fronts. One concentrates on the physics of light transport, leading to equations that describe how light passes through a medium and reflects from an object. The other concentrates on the human visual system and the brain's interpretation of an image that is received. The first approach lends itself to well-known and to novel algorithms for solving (or estimating) complicated equations. Its basic primitives are light sources, geometry, and reflectance functions.

The second effort requires anticipating how a human will respond to a rendered image, which is a much more subtle enterprise (see item 4). In the same way, a theory of visualization could be comprised of two distinct aspects: one depends only on the underlying data, while the other concentrates on the human response to imagery. A data-driven theory of visualization serves as a preparatory step for rendering. An abstract data set is somehow converted into lights, geometry, and reflectance; after that point, rendering is performed as a postprocess. Another theoretical item that deserves attention is the need for techniques that handle complex and dynamic geometry. While there are many techniques for performing operations on regular volumetric grids, the development of techniques for unstructured data has lagged behind. Visualization techniques for handling large, dynamic, unstructured grids are essentially nonexistent.

As da Vinci understood the need for practitioners to study their own practices, whether the art of science or the science of art, so too did he comprehend the need to theorize those practices in order to understand them and hence to strengthen them. I can't help but agree with him when he says, "He who loves practice without theory is like the sailor who boards ship without a rudder and compass and never knows where he may cast."

Conclusion

Researchers in scientific visualization will determine the futures not only of their own field, but of the many scientific fields to which they contribute. We can best take advantage of our position by ensuring that our discipline is as rigorous and productive as any other science even as we vigorously pursue technological innovation. In this way, we may motivate visualization researchers to think either about new problems or about persistent problems in a new way. ∎

References

1. S. Smale, "Mathematical Problems for the Next Century," *Mathematical Intelligencer*, vol. 20, no. 2, 1998, pp. 7-15.
2. I.E. Sutherland, "Ten Unsolved Problems in Computer Graphics," *Datamation*, vol. 12, no. 5, May 1966, pp. 22-27.
3. P. Heckbert, "Ten Unsolved Problems in Rendering," *Workshop on Rendering Algorithms and Systems, Graphics Interface '87*, 1987.
4. J. Blinn, "Transcription of Keynote Address at Siggraph 98," *Computer Graphics*, vol. 33, no. 1, Feb. 1999, pp. 43-47.
5. *Visualization in Scientific Computing*, B.H. McCormick, T.A. DeFanti, and M.D. Brown, eds., ACM Press, 1987.
6. B. Hibbard, "Top Ten Visualization Problems," *Proc. ACM Siggraph*, vol. 33, no. 2, ACM Press, 1999, pp. 21-22; http://www.siggraph.org/publications/newsletter/v33n2/columns/hibbard.html).
7. T.M. Rhyne et al., "Panel on Can We Determine the Top Unresolved Problems of Visualization?" *IEEE Visualization 2004*, to be published.
8. F.P. Brooks Jr., "The Computer Scientist as a Toolsmith II," *Comm. ACM*, vol. 39, no. 3, Mar. 1996, pp. 61-68.
9. R. Kosara et al., "Thoughts on User Studies: Why, How, and When," *IEEE Computer Graphics and Applications*, vol. 23, no. 4, July/Aug. 2003, pp. 20-25.
10. C.R. Johnson and A.R. Sanderson, "A Next Step: Visualizing Errors and Uncertainty," *IEEE Computer Graphics and Applications*, vol. 23, no. 5, Sept./Oct. 2003, pp. 6-10.
11. C. Ware, *Information Visualization: Perception for Design*, Morgan Kaufmann, 2000.
12. B. Shneiderman, *Information Visualization: Dynamic Queries, Starfield Displays, and LifeLines*, white paper; http://www.cs.umd.edu/hcil/members/bshneiderman/ivwp.htm.
13. A. van Dam et al., "Immersive VR for Scientific Visualization: A Progress Report," *IEEE Computer Graphics and Applications*, vol. 20, no. 6, Nov./Dec., 2000, pp. 26-52.
14. A. van Dam, D.H. Laidlaw, and R.M. Simpson, "Experiments in Immersive Virtual Reality for Scientific Visualization," *Computers and Graphics*, vol. 26, no. 4, Aug. 2002, pp. 535-555.
15. T. M. Rhyne, "Does the Difference between Information and Scientific Visualization Really Matter," *IEEE Computer Graphics and Applications*, vol. 23, no. 2, May/June 2003, pp. 6-8.
16. C. Hansen and C.R. Johnson, *The Visualization Handbook*, Academic Press, 2004, to be published.

Readers may contact Chris Johnson at the Scientific Computing and Imaging Inst., Univ. of Utah, 50 S. Central Campus Dr., Rm. 3490, Salt Lake City, UT 84112, crj@sci.utah.edu.

Readers may contact Theresa-Marie Rhyne by email at tmrhyne@ncsu.edu.