

Guest Editors' Introduction: Special Section on IEEE Visualization

Jarke J. van Wijk, *Member, IEEE*, Robert J. Moorhead II, *Senior Member, IEEE*, and
Greg Turk, *Member, IEEE*

THIS issue contains extended versions of six of the strongest papers taken from the IEEE Visualization 2003 Conference. This year we have followed a slightly different procedure than in previous years. We intended to give prospective authors more time to prepare an extended version, while simultaneously keeping a short time between the conference and the release of this issue. Therefore, in June 2003, just after the selection of papers for the conference, we started to select the most interesting papers. We, paper cochairs of IEEE Visualization 2003, started from the opinions of the reviewers, followed by several rounds of rereading and discussion. The authors of six outstanding papers were invited to submit an extended and revised version of their original work. Each revised paper was thoroughly reviewed by several experts before it was recommended for acceptance. We are grateful to the reviewers for their thorough and quick reviews, and to David Ebert, Editor-in-Chief of *TVCG*, for his strong support, wise advice, and always astonishingly quick responses.

One of the most fascinating aspects of visualization is the width of the field. We selected the papers on quality, but we were delighted to find out that, in the end, our selection covers a wide variety of topics, approaches, and methods. Novel results on feature extraction and simplification are given, but also novel interpolation results; the handling of various standard types of data, including scalar volume data, terrain data, and vector volume data, is discussed, but also a new representation is presented; sophisticated mathematics, including group theory and Morse theory, is used, but also the capabilities of modern graphics hardware are exploited to the limit.

The paper by David Banks, Stephen Linton, and Paul Stockmeyer is based on a paper that was awarded the IEEE Visualization 2003 Conference Best Paper Award. The particular strength of this paper is that it lays a rigorous mathematical foundation below an important class of visualization algorithms. The authors consider the classic Marching Cubes algorithm and its variations, which play a

central role in many applications. A well-known result is that, if we mark each node of a cube as either red or blue, in total 14 or 15 different cases can be distinguished. For more complex situations, such as higher-dimensional cubes, the number of cases explodes. Counting by hand is not possible anymore and the results of various researchers vary. Banks et al. denote this class of algorithms as *substotope algorithms* and present an elegant taxonomy, based on the number of dimensions, the number of colors, the shape, and the symmetry used. Computational Group Theory is used to count the number of cases for different choices of dimensions, colors, and shapes. The results confirm previous results, explain the differences, and, furthermore, the number of cases for several new algorithms are predicted. One of the additions of this new version is that the authors show how Pólya counting theory can be used to produce a closed-form upper bound on the case counts.

The paper by Peer-Timo Bremer, Herbert Edelsbrunner, Bernd Hamann, and Valerio Pascucci is another example of a great result based on a sound mathematical basis. The problem addressed is how to derive a hierarchy of simplified representations of functions defined over two-dimensional domains. Simplification of height-fields is a key example, but the method is more generally applicable. They argue that the topological features of the function should be taken into account during the simplification process. Based on Morse theory, they show how such a hierarchical representation can be generated by progressively cancelling pairs of critical points and adapting the mesh accordingly. The overall strategy is simple and sound, but is not easy to convert into a full-proof algorithm. The authors describe in detail how to do this and show various applications.

In visualization, we sometimes need less, and sometimes more, detail. The paper by Christian Rössl, Frank Zeilfelder, Günther Nürnberger, and Hans-Peter Seidel discusses interpolation of volume samples given on a regular grid, an important case, for instance, for many medical applications. The standard workhorse for this is trilinear interpolation, which does not give a smooth reconstruction, whereas a triquadratic or tricubic interpolation or approximation is often prohibitively costly. The authors present an alternative solution, based on piecewise trivariate quadratic splines. Each cube is subdivided into 24 tetrahedra, in total 65 coefficients have to be calculated. Their solution has a number of pleasant properties, including a smooth reconstruction, efficient computation, evaluation, and visualization of the model. In particular, along a ray through the volume, the splines are univariate, piecewise quadratics.

- J. van Wijk is with the Department of Mathematics and Computer Science, Technische Universiteit Eindhoven, PO Box 513, 5600 MB Eindhoven, The Netherlands. E-mail: vanwijk@win.tue.nl.
- R.J. Moorhead II is with the ERC, Mississippi State University, PO Box 9267, Mississippi State, MS 39762. E-mail: rjm@erc.msstate.edu.
- G. Turk is with the College of Computing, Georgia Institute of Technology, 801 Atlantic Dr., Atlanta, GA 30332-0280. E-mail: turk@cc.gatech.edu.

For information on obtaining reprints of this article, please send e-mail to: tvccg@computer.org.

Several examples are given that convincingly show that this method is superior to trilinear interpolation, at a limited cost.

Regular grids are a standard representation in many applications, but they can be improved upon. Huamin Qu and Arie Kaufman present in their paper an attractive alternative: the O-buffer. This is a generic and versatile framework for sample-based graphics, where a much higher quality is achieved than with conventional representations at a low cost. As with all great ideas, the concept is simple. A regular 2D or 3D grid is used as a basis, but, in each cell (pixel or voxel), not only the value of a sample is stored, but also an offset to indicate the position of the sample in the cell. Typically, a one byte offset is used for 2D O-buffers such that the position of the sample in a 16×16 subgrid is recorded. A large number of variations on this concept, including various hierarchical ones, are presented, and the authors show how this representation can be used for many applications, including image-based rendering and volume rendering. A particular strong feature is that this framework easily allows for the combination of different types of graphics primitives in the same scene.

Efficient processing of low-level representations is an important aspect of visualization; another direction is the extraction of features. Aaron Lefohn, Joe Kniss, Charles Hansen, and Ross Whitaker present in their paper a new method for this. The method is based on deformable isosurfaces, implemented with level-set methods. This class of methods already has shown its great potential, but the high computational cost limits its usefulness. The new method presented achieves an amazing factor of 10 to 15 increase in processing speed by using the graphics processor (GPU). This is not an easy path. The core of the level-set algorithm is to incrementally deform a surface, based on local properties such as a function value or various derivatives. This surface is represented as an isosurface and it is moved by modifying the values of a set of points close to it, the so-called narrow band. The architecture of current GPUs does not lend itself easily for this, but the authors demonstrate how, via an ingenious method, texture memory can be turned into a multi-dimensional virtual memory system. As a result, the user is able to interactively view and steer the level-surface as it evolves, which dramatically widens the applicability of this class of methods.

Han-Wei Shen, Guo-Shi Li, and Udepta Bordoloi present in their paper another highly innovative application of GPUs. The visualization of 3D vector fields is a challenging problem which plays an important role in many disciplines. Since the complexity of these data cannot be captured in a single still image, animation, and real-time interaction are vital to gain insight. The authors present a novel framework that offers both high speed and great flexibility. The key idea here is to use a 3D texture to store information on streamlines. For instance, per voxel, the identity and a parameter value are stored. The authors show how this can be done fast using the GPU. Next, visualizations are produced by using the stored information as texture coordinates. By varying the mapping and the texture used, a wide variety of visualization techniques can

be emulated in real-time, varying from stream tubes to dense, texture-based representations.

So far, we have pointed out the wide variation in the papers presented. However, more important is what they share. We hope that the reader will agree that the papers presented here have a high quality, a wide applicability, and that they will form an inspiration for the research in our fascinating field.

Jarke van Wijk
Robert Moorhead
Greg Turk
Guest Editors



Jarke J. van Wijk received the MSc degree in industrial design engineering in 1982 and the PhD degree in computer science in 1986, both with honors. He worked from 1988 to 1998 at the Netherlands Energy Research Foundation ECN in The Netherlands, where he was engaged in research on flow visualization and computational steering. In 1998, he joined the Technische Universiteit Eindhoven and, in 2001, he was appointed a full professor in visualization. His main research interests are information visualization and flow visualization, both with a focus on the development of new visual representations. He has (co)authored more than 70 papers on visualization and computer graphics, including 13 IEEE Vis and 6 IEEE InfoVis papers. He is a member of the IEEE.



Robert J. Moorhead II received the PhD degree in electrical and computer engineering and the MSEE degree from North Carolina State University in 1985 and 1982, respectively. He received the BSEE degree summa cum laude and with research honors from Geneva College in 1980. He is the director of the Visualization, Analysis, and Imaging Lab in the ERC GeoResources Institute and a professor of Electrical and Computer Engineering at Mississippi State University. He previously worked as a research staff member in the Imaging Technologies Department at the IBM T.J. Watson Research Center from 1985 to 1988. He has authored more than 95 papers or book chapters on visualization, image processing, and computer communications. He was the lead conference cochair for the IEEE Visualization '97 Conference, the chair of the IEEE Computer Society's Technical Committee on Visualization and Graphics in 1999 and 2000, and a papers cochair for the IEEE Visualization 2002 and 2003 Conferences. He is a senior member of the IEEE.



Greg Turk received the PhD degree in computer science in 1992 from the University of North Carolina (UNC) at Chapel Hill. He was a postdoctoral researcher at Stanford University for two years, followed by two years as a research scientist at UNC Chapel Hill. He is currently an associate professor at the Georgia Institute of Technology, where he is a member of the College of Computing and the Graphics, Visualization, and Usability Center. His research interests include computer graphics, scientific visualization and computer vision. He is a member of the IEEE.