

SIGGRAPH 97

24TH INTERNATIONAL CONFERENCE

ON COMPUTER GRAPHICS

AND INTERACTIVE TECHNIQUES

C O U R S E N O T E S



Texture Synthesis with Line Integral Convolution

ORGANIZER

Kwan-Liu Ma

Institute for Computer Applications in Science
and Engineering

LECTURERS

Brian Cabral

Silicon Graphics, Inc.

Hans-Christian Hege

Detlev Stalling

Konrad-Zuse-Zentrum für Informationstechnik

Victoria L. Interrante

Kwan-Liu Ma

Institute for Computer Applications in Science
and Engineering

Conference 3-8 August 1997
Exhibition 5-7 August 1997



Los Angeles Convention Center
Los Angeles, California USA

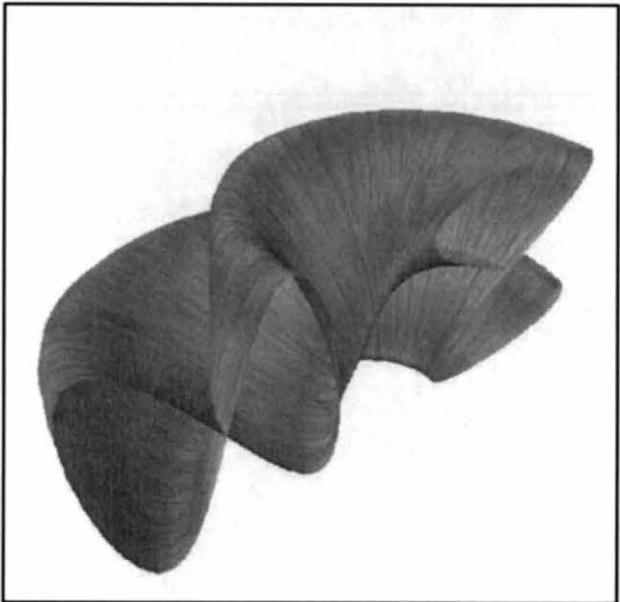
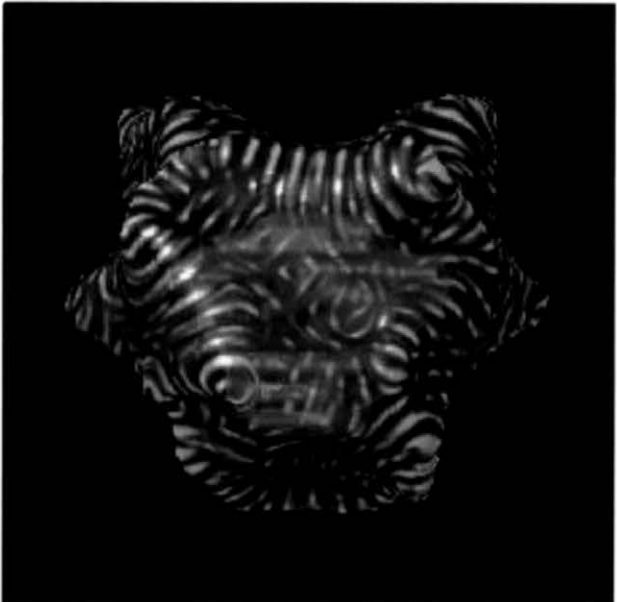
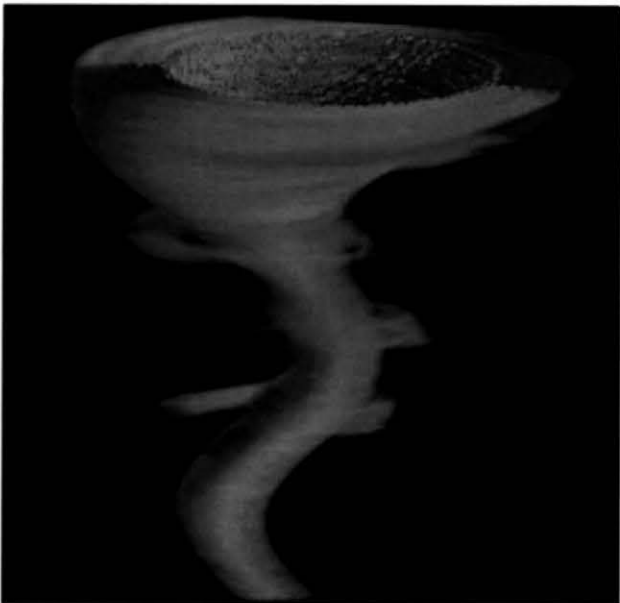
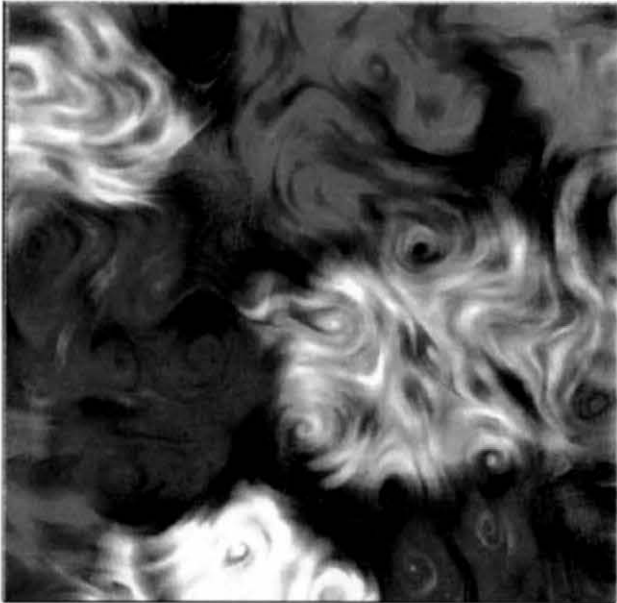
Texture Synthesis with Line Integral Convolution

Course Notes for SIGGRAPH '97

Organizer

Kwan-Liu Ma

Institute for Computer Applications in Science and Engineering



COURSE ORGANIZER



Kwan-Liu Ma
ICASE
NASA Langley Research Center
Hampton, Virginia 23681-0001
kma@icase.edu

LECTURERS



Brian Cabral
Silicon Graphics, Inc.
2011 N. Shoreline Blvd.
Mountain View, CA 94039-7311
cabral@sgi.com



Hans-Christian Hege
ZIB
Takustrasse 7
D-14195 Berlin-Dahlem, Germany
hege@zib.de



Victoria Interrante
ICASE
NASA Langley Research Center
Hampton, Virginia 23681-0001
interran@icase.edu



Detlev Stalling
ZIB
Takustrasse 7
D-14195 Berlin-Dahlem, Germany
stalling@zib.de

BIOGRAPHIES

Brian Cabral is a manager of visualization toolkit Core Rendering software within the Advanced Graphics Division at Silicon Graphics Computer Systems. His current technical focus is on Cosmo3D based toolkits for visualizing large geometric databases and scientific and engineering data. Previously, Brian, developed algorithms and techniques in the areas of Medical visualization and image processing, scientific visualization, and physical based shading and lighting. Prior to working at SGI, Brian, was a lead engineer and researcher at Lawrence Livermore National Laboratory where he worked a variety of visualization algorithms, tools and systems. He received a BS in computer science from California State University Stanislaus and a MS in computer science from University of California, Davis. His interests include differential geometry, image processing, shading techniques, computational geometry and signal processing. When he's not toiling over a large equation or chunk of code Brian enjoys the good life of California.

Hans-Christian Hege is head of the Scientific Visualization Department at Konrad-Zuse-Zentrum Berlin (ZIB) – a research institute of the State of Berlin operating in the field of algorithmic mathematics. He studied physics at Free University Berlin. From 1984-1989 he has been research assistant at the physics department there, working in the areas quantum field theory, statistical physics and numerical physics. He is a co-founder of mental images and Gesellschaft für Digital Simulation (GDS) that started 1986 to produce computer animations and rendering software. From 1986 to 1989 he worked as researcher at mental images and managing director at GDS. Since 1989 he is with ZIB, first as scientific consultant and then as director of the Visualization and Parallel Computing department. His research interests are in computer graphics, scientific visualization, biomedical computing, and physics. He is member of ACM, IEEE, and the steering committee 'Computer Graphics' of GI.

Victoria Interrante is a staff scientist at the Institute for Computer Applications in Science and Engineering (ICASE), a center of research in applied mathematics, numerical analysis, fluid dynamics and computer science operated by the Universities Space Research Association at the NASA Langley Research Center. She received a PhD in computer science from the University of North Carolina at Chapel Hill in 1996, where she completed a dissertation under the direction of Drs. Henry Fuchs and Stephen Pizer on the design of perceptually-inspired artistic techniques for improving the comprehensibility of layered transparent surfaces. Her current research focuses on the application of insights from perceptual psychophysics, art and illustration to the design of more effective techniques for visualizing data, the study of shape and depth perception, the use of texture to convey 3D shape and flow information.

Kwan-Liu Ma received his PhD degree in computer science from the University of Utah. He is a staff scientist at the Institute of Computer Applications in Science and Engineering and an adjunct assistant professor of computer science at Old Dominion University, where he teaches scientific visualization. He is a co-chair of the 1997 Symposium on Parallel Rendering, and also in the program committee of Visualization and Mathematics '97. His research interests include computer graphics, scientific visualization and high performance computing. In particular, he has developed several parallel volume rendering algorithms, feature extraction algorithms, and interactive visualization techniques. Dr. Ma is a member of the IEEE, ACM, SIGGRAPH and Phi Kappa Phi.

Detlev Stalling studied physics at the University of Osnabrück and the Free University of Berlin, where he received his MS degree in 1993. In 1992 he spent some months at the European Center for Particle Physics (CERN). After study he joined the Konrad-Zuse-Zentrum Berlin (ZIB) as a research scientist in the scientific visualization department. His research interests include accurate visualization methods, geometric modeling, and image processing. Currently he is working on his PhD thesis about visual methods for analyzing vector fields.

Table of Contents

Prolegomenon - <i>Kwan-Liu Ma</i>	1
I. Introduction to Line Integral Convolution - <i>Brian Cabral and Casey Leedom</i>	5
1 Introduction	5
2 Background, History and Motivation	5
3 The Basic Mathematics	7
4 An Algorithm for LIC	7
4.1 The Variable Step Euler's Method	8
4.2 Approximating the Kernel Integral	9
4.3 Vector Field Singularities	10
5 Computing the Output Image	11
5.2 Filter Kernel and Convolution Behavior	11
5.3 Periodic Motion Filtering	11
5.4 Image Aliasing	13
5.5 Advection Error and Aliasing	14
6 Future Directions	15
6.1 Another Anti-aliasing Scheme	15
6.2 Topological Structure of LIC	15
6.3 Higher Order LIC	15
References	16
II. LIC: Acceleration, Animation, and Zoom - <i>Hans-Christian Hege and Detlev Stalling</i>	17
1 Introduction	17
2 Continuous Formulation of LIC	19
2.1 Vector Fields and Field Lines	19

2.2 Line convolution	21
3 Fast and Flexible Generation of LIC Images	22
3.1 Computation of Convolution Lines	22
3.1.1 Field Interpolation	22
3.1.2 Field Integration	23
3.1.3 Prepresentation of Field Lines	25
3.1.4 Stepping along Field Lines	26
3.2 Computation of LIC Integrals	27
3.2.1 Exploiting Redundancies	27
3.2.2 Putting it Together: A Fast LIC Algorithm	27
3.2.3 Seed Point Selection	29
3.2.4 Adaptive Field Line Length	31
4 Resolution Independence, Zooming and All That	32
5 Animating LIC Textures	33
5.1 Intensity Correlations	35
5.2 Frame Blending	36
5.3 Space Dependent Velocity	37
5.4 Temporal Coherence	38
6 Parallelization	39
6.1 Parallelization of the Original LIC Algorithm	39
6.1.1 Coarse-Grain Parallelization	39
6.1.2 Fine-Grain Parallelization	39
6.2 Parallelization of the fastLIC Algorithm	41
6.2.1 Parallelization in Image Space	41
6.2.2 Update Scheme	43
6.2.3 Data Collection	45
6.2.4 Results	45
7 Conclusions and Outlook	46
References	48

III. LIC on Surfaces - *Detlev Stalling* 51

1 Introduction	51
2 Tangent Curves on Surfaces	51
3 LIC in Parameter Space	53
3.1 Parametric Surfaces	53
3.2 The Algorithm	54
3.3 Compensating Texture Distortions	54
4 LIC in Physical Space	55
4.1 Local Texture Approach	57
4.2 Ray-Surface Intersection Approach	57
4.3 Solid Noise	59
4.4 Field Line Integration on Triangular Surfaces	59
4.5 Triangle Packing	60

5 Extensions and Future Directions	61
References	62

IV. Visualizing Surface Shape and 3D Flow Using Volume LIC - *Victoria Interrante* **65**

1 Introduction	65
2 Conveying 3D Shape with Line	65
2.1 Defining Principal Directions	65
2.2 Defining the Input to LIC	66
2.3 Advecting the Particles and the Empty Space Between Them	67
2.4 Applying the Output Texture	67
2.5 Preliminary Results	68
2.6 Refining the Texture Definition	68
2.6.1 Stroke Length	69
2.6.2 Stroke Width	70
2.6.3 Stroke Color	71
2.6.4 Limiting Texture Computations to a Region of Interest in the Volume	71
2.7 Directions for Future Work	71
2.7.1 Depicting Multiple Transparent Layers	71
2.7.2 Non-photorealistic Rendering	72
2.8 Conclusions	73
3 Conveying 3D Flow with Volume LIC	73
3.1 Background and Motivation	73
3.2 Clarifying the Display of Densely Clustered Elements	74
3.2.1 Differentiating the Distinct Texture Elements	75
3.2.2 Enhancing Depth Discontinuities	75
3.2.3 Facilitating Local Depth Order Judgments	76
3.2.4 Emphasizing Selected Regions of Interest	77
3.3 Conclusions	78
References	79

V. Other Extensions to LIC - *Kwan-Liu Ma and Han-Wei Shen* **81**

1 Introduction	81
2 Visualizing Multiple Surface Layers	82
2.1 Bi-Variate Volume Rendering	82
3 Dye Advection for Flow Visualization	83
3.1 Dye Injection and Smearing	84
3.2 Fast Searching Algorithm	85
3.3 Convolution Kernel	86
3.4 Advection Length	87
3.5 Dye Rendering	89

4 Oriented Line Integral Convolution	91
5 Multi-frequency LIC	94
6 Visualizing Time-Varying Data	97
6.1 UFLIC	97
6.2 Visualization Examples	98
References	99
Final Remarks - Kwan-Liu Ma	103

Prolegomenon

We have seen striking, realistic images made with synthetic textures for 3d graphics, visualization and special effects. The use of synthesized textures has recently gained increasing popularity because more elaborated, powerful techniques are being developed and may be applied to application problems in an economical manner. The Line Integral Convolution (LIC) method introduced by Brian Cabral and Casey Leedom [2] is a particularly powerful and elegant texture synthesis algorithm for visualizing *motion* and *shape*. LIC transforms vector data into lively visualization by distorting an input texture image according to the local vector field.

Cabral and Leedom's creation has inspired many visualization researchers. Forssell [3, 4] extends LIC to vector fields on the curvilinear grids used in computational fluid dynamics simulations. Stalling and Hege develop a fast and resolution independent LIC algorithm that is an order of magnitude faster than the original implementation [13]. Both Cabral/Leedom [1] and Stalling/Zockler/Hege [14] also investigate how to parallelize the LIC calculations to achieve real-time visualization. Kiu and Banks show how the use of multi-frequency noise texture images may enhance a user's perception of the features in the vector flow [8]. Wegenkittl, Gröller and Purgathofer add orientation information to still LIC images [16].

Shen, Johnson and Ma show how to integrate the LIC algorithm into direct volume rendering to construct multiple semi-transparent surfaces and to simulate the use of dye advection in experimental flow visualization [12]. Okada and Kao develop an enhancement technique especially for highlighting flow separation and reattachment boundaries in flow fields defined in 3D curvilinear multi-block grids [10]. To apply LIC to arbitrary 3D polygonal surfaces, Battke, Stalling and Hege come out with an efficient local texture algorithm [5], while Mao, et al. use a view-dependent approach to generate more accurate results [9].

More recently, Interrante experiments with the idea of using the LIC method to illustrate the shape of arbitrary surfaces [6]. She also investigates perceptual issues in visualizing three-dimensional flow data [7]. Wegenkittl and Gröller design an approximated version of their oriented LIC algorithm to allow very fast visualization [15]. Shen and Kao are currently developing a new algorithm for handling time-varying data in a more physically correct manner [11].

The continuous, intensive effort by various researchers has made LIC now practical for many applications. Therefore, we believe a course on the LIC method is appropriate and will greatly benefit the users of 3d graphics and visualization technology. This course provides a comprehensive presentation of the LIC techniques. You will learn about the core algorithm and why it works. You will also learn how to implement the algorithms efficiently and how certain key parameters control the intended

visual effects. You will be exposed to various application problems through many visual examples including videos and live demonstrations. At the end of the course, you should be able to apply LIC to your application problems by modifying the computer source programs provided by the lecturers.

The lectures contain three parts: the basic algorithm, the optimized algorithms and the extended algorithms. Brian Cabral will talk about the history and motivation of LIC and give an overview of the basic algorithm. This lecture will provide you with the needed background to follow the material covered in the rest of the course.

Next, Hans-Christian Hege and Detlev Stalling will talk about their optimized LIC algorithms and how the LIC calculations may be parallelized to achieve interactive vector field visualization. They will also describe how to perform LIC on arbitrary surfaces by converting the surfaces into triangular meshes.

Then, Victoria Interrante will describe how LIC can be applied to visualization of surface shape to generate non-photorealistic scientific illustrations. She will also address some perception issues with using the LIC method.

Finally, Kwan-Liu Ma will talk about combining LIC and direct volume rendering to visualize vector and scalar volume data in a single visualization, and how to enhance local features in flow-field data by using colors. He will also describe some useful extensions of LIC including the idea of using multi-resolution texture input introduced by Kiu and Banks, the oriented LIC techniques invented by Wegenkittl, Gröller and Goldsteiner, as well as the techniques developed by Shen and Kao for visualizing time-varying data.

We have prepared an implementation of the LIC algorithm and installed in the Creative Applications Laboratory set up by the conference. In this way you can try out some of the ideas introduced in the lectures to grasp the key concepts more clearly. As the results of applying LIC can vary significantly when the input parameters and texture change, we believe this hand-on experience will be highly rewarding to you. The source code is also made available on-line.

Besides the authors/lecturers, David Banks, E. Gröller, Ming-Hoe Kiu, Casey Leedom, and Han-Wei Shen have helped put these notes together. We would especially like to thank Viki Dennis, Barb Helfer, Scott Senften and Stephen Spencer of SIGGRAPH for working closely with us to make possible our course and notes.

Kwan-Liu Ma
April 26, 1997

References

- [1] CABRAL, B., AND LEEDOM, C. Highly Parallel Vector Visualization Using Line Integral Convolution. In *in Proceedings of the 7th SIAM Conference on Parallel Processing for Scientific Computing* (February 1995).
- [2] CALBRAL, B., AND LEEDOM, L. Imaging Vector Fields Using Line Integral Convolution. In *in Proceedings of SIGGRAPH '93 Conference* (1993).

-
- [3] FORSSEL, L. K. Line Integral convolution. In *in Proceedings of the Visualization '94 Conference* (1994).
- [4] FORSSEL, L. K., AND S., C. Using Line Integral Convolution for Flow Visualization: Curvilinear Grids, Variable Speed Animation, and Unsteady Flows. *IEEE Transactions on Visualization and Computer Graphics* 1, 2 (June 1995).
- [5] H., B., STALLING, D., AND HEGE, H. Fast Line Integral Convolution for Arbitrary Surfaces in 3D. In *Visualization and Mathematics* (1997), H. Hege and P. K., Eds., Springer-Verlag.
- [6] INTERRANTE, V. Illustrating Surface Shape in Volume Data via Principal Direction-Driven 3D Line Integral Convolution. In *in Proceedings of SIGGRAPH '97 (to appear)* (1997).
- [7] INTERRANTE, V. Visualizing 3D Flow with Volume LIC: issues and insights. Tech. Rep. ICASE Report No. ?? (to appear), Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, March 1997.
- [8] KIU, M., AND BANKS, D. Multi-Frequency Noise for LIC. In *in Proceedings of the Visualization '96 Conference* (1996).
- [9] MAO, X., KIKUKAWA, M., FJUITA, N., AND IMAMIYA, A. Line Integral convolution for Arbitrary 3D Surfaces through Solid Texturing. In *in Proceedings of Eighth Eurographics Workshop on Visualization in Scientific Computing (to appear)* (1997).
- [10] OKADA, A., AND KAO, D. Enhanced line integral convolution with flow feature detection. In *in Proceedings of IS&T/SPIE Symposium on Electronic Imaging Science and Technology* (1997).
- [11] SHEN, H., March 1997. Personal Communication.
- [12] SHEN, H., JOHNSON, C., AND MA, K. Visualizing Vector Fields Using Line Integral Convolution and Dye Advection. In *in Proceedings of 1996 Symposium on Volume Visualization* (1996).
- [13] STALLING, D., AND HEGE, H. Fast and Resolution Independent Line Integral Convolution. In *in Proceedings of SIGGRAPH '95 Conference* (1995).
- [14] STALLING, D., ZOCKLER, M., AND H., H. Parallel Line Integral Convolution . In *in Proceedings of the 1st Eurographics Workshop on Parallel Graphics and Visualisation* (September 1996).
- [15] WEGENKITTL, R., AND GRÖLLER, E. Fast Oriented Line Integral Convolution for Vector Field Visualization via the Internet. Tech. Rep. TR-186-2-97-06, The Institute of Computer Graphics, Vienna University of Technology, 1997.
- [16] WEGENKITTL, R., GRÖLLER, E., AND PURGATHOFER, W. Animating Flowfields: Rendering of Oriented Line Integral Convolution. Tech. Rep. TR-186-2-96-23, The Institute of Computer Graphics, Vienna University of Technology, 1996. (to appear in *Computer Animation '97*).