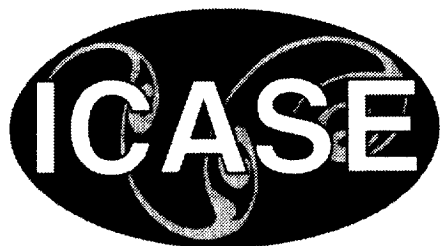NASA/CR-1998-208437
ICASE Report No. 98-26

Recent Advances in Visualizing 3D Flow with LIC

Victoria Interrante
ICASE, Hampton, Virginia

Chester Grosch
Old Dominion University, Norfolk, Virginia
and
ICASE, Hampton, Virginia

Institute for Computer Applications in Science and Engineering
NASA Langley Research Center
Hampton, VA

Operated by Universities Space Research Association

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

July 1998

RECENT ADVANCES IN VISUALIZING 3D FLOW WITH LIC[*]

VICTORIA INTERRANTE[†]   CHESTER GROSCH[‡]

Abstract. Line Integral Convolution (LIC), introduced by Cabral and Leedom in 1993, is an elegant and versatile technique for representing directional information via patterns of correlation in a texture. Although most commonly used to depict 2D flow, or flow over a surface in 3D, LIC methods can equivalently be used to portray 3D flow through a volume. However, the popularity of LIC as a device for illustrating 3D flow has historically been limited both by the computational expense of generating and rendering such a 3D texture and by the difficulties inherent in clearly and effectively conveying the directional information embodied in the volumetric output textures that are produced. In an earlier paper, we briefly discussed some of the factors that may underlie the perceptual difficulties that we can encounter with dense 3D displays and outlined several strategies for more effectively visualizing 3D flow with volume LIC. In this article, we review in more detail techniques for selectively emphasizing critical regions of interest in a flow and for facilitating the accurate perception of the 3D depth and orientation of overlapping streamlines, and we demonstrate new methods for efficiently incorporating an indication of orientation into a flow representation and for conveying additional information about related scalar quantities such as temperature or vorticity over a flow via subtle, continuous line width and color variations.

Key words. 3D flow visualization, line integral convolution, volume rendering, depth representation.

Subject classification. Computer Science

1. Surface LIC: a brief review. For the purposes of flow visualization, line integral convolution is typically applied to an input texture containing uncorrelated intensities (for example a white noise image) to produce an output texture in which the intensities are locally correlated in the direction indicated by the flow. The value at each point in the output texture is computed as a weighted sum of the values in the input texture that lie at evenly-spaced intervals along a smoothly continuous streamline defined by the vector field, originating at the specified point and extending for a predetermined distance in both directions. Stalling and Hege [3] showed that the computational efficiency of the basic LIC algorithm can be enhanced if the output texture values are computed in an incremental fashion along each streamline rather than separately in scan-line order, and that the continuity and clarity of the output texture can be improved when a highly accurate streamline integration technique, such as an adaptive fourth-order Runge-Kutta method is used. Forsell [4] showed how the basic LIC algorithm could be extended to enable the depiction of flow across a curvilinear surface in $\Re^3$, by generating the output texture in parameter space and then mapping it onto the surface in 3D, and Battke et al. [5] and Mao et al. [6] introduced methods for using LIC in conjunction with a solid 3D input texture and a tangential vector field to show flow across arbitrary

---

[†] Institute for Computer Applications in Science and Engineering, Mail Stop 403, NASA Langley Research Center, Hampton, VA 23681-2199 (email: interran@icase.edu).

[‡] Old Dominion University and Institute for Computer Applications in Science and Engineering, Mail Stop 403, NASA Langley Research Center, Hampton, VA 23681-2199 (email: chet@icase.edu).

surfaces, generating the texture values directly across the triangles of the surface in 3D [5], or explicitly computing the image of the texture at the locations of the visible surface points in a particular view [6].

2. Other Flow Visualization Techniques. LIC represents of course only one of many possible methods for visualizing a 3D flow. Our intention in this paper is not to suggest that 3D LIC is superior to other 3D flow visualization methods (which have been very nicely reviewed elsewhere [7]) but merely to suggest that, perhaps contrary to popular expectation, 3D LIC can be a viable complementary technique for effectively conveying local and global information about a 3D flow. It is important to stress, however, that the generation and rendering of 3D LIC textures is fairly computationally expensive, and that these methods, although aesthetic, are best suited for applications in which interactivity is not required.

3. Volume LIC. When line integral convolution is applied to a solid noise texture using a 3D vector field, the result is a solid 3D output texture, such as shown in figure 1, in which the values of the voxels are everywhere locally correlated according to the directions of the 3D flow. The mechanics of the computation are straightforward, but how can such data be visualized? It is hard to clearly appreciate the 3D flow information represented by this solid texture: within any individual 2D slice, the image of the flow will be incomplete and it can be difficult to mentally reconstruct an accurate perception of the 3D flow from a series of 2D slices viewed sequentially; it is often problematic to define an appropriate set of surfaces across which the 3D flow information can meaningfully be shown; and the inner details of the 3D texture are completely lost when the data is imaged as a set of partial opacity values via direct volume rendering. These obstacles notwithstanding, we believe that if appropriately defined and rendered, a 3D LIC texture has considerable potential to provide a full, immediate and intuitive impression of the global and local characteristics of a 3D flow. The challenge we face is to determine how to achieve such a representation. Shen et al. [2] have suggested complementing a volume-rendered LIC texture with simulated dye advection; here we look at other options.

FIG. 1. *Left: a solid 3D texture of random noise. Right: the texture after line integral convolution.*

4. Region of Interest Specification. In certain cases, it is possible to use a scalar function such as temperature or vorticity, to identify, a priori, specific critical regions in a flow volume, within which (or across which) the depiction of flow information is especially important, and it is sometimes possible to clarify the presentation of the data by isolating and selectively emphasizing the flow information in these particular regions. When LIC is used in conjunction with a Region of Interest (ROI) thus defined, we have found that substantially better results can be achieved if the ROI mask is applied as preprocess to the input texture, before the line integral convolution, rather than as a postprocess to the output afterwards. In the

first case, the visible portion of the flow texture will be completely determined by the scalar region of interest masking function, the boundary of which will not in general follow the direction of the flow. In the second case, in which the ROI function is applied before LIC, the apparent region of interest segmentation will be guided by the flow itself, with the result that the effective boundaries of the ROI will be everywhere aligned with the direction of the flow. Figures 2 and 3 illustrate the difference in these two approaches.
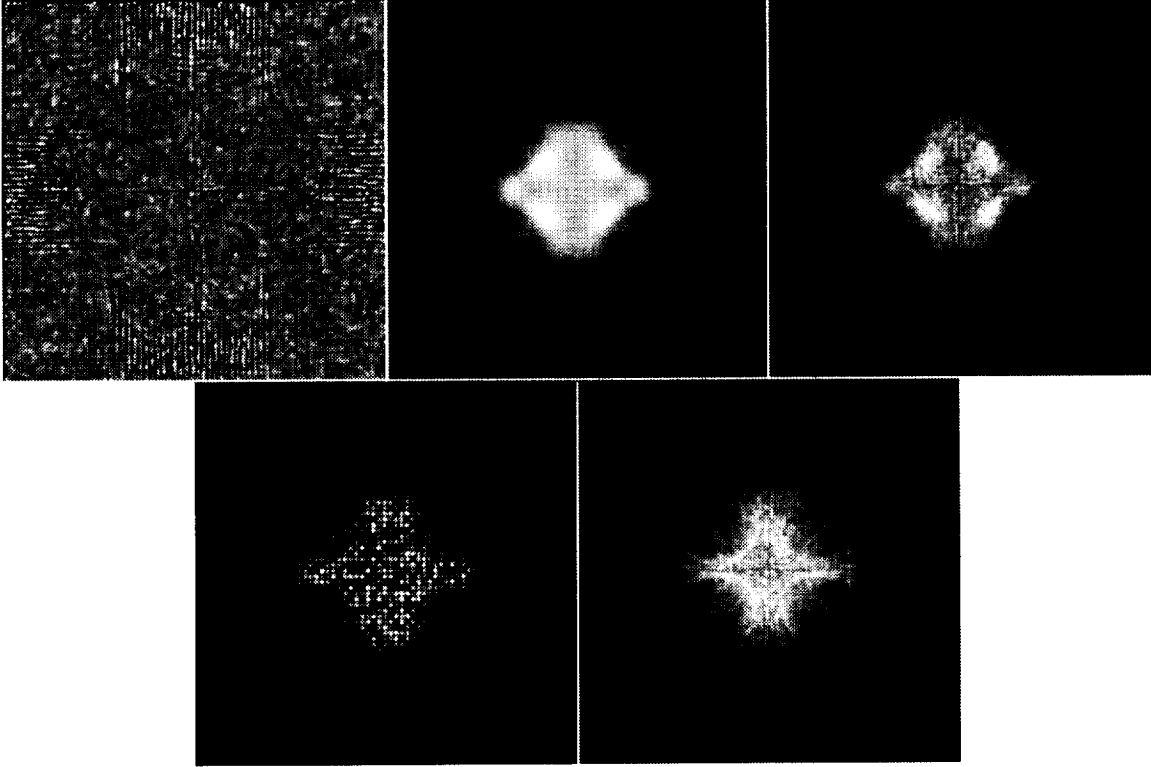

FIG. 2. *Top left: a 2D slice from a solid 3D texture generated from a 3D vector field and a solid noise input using volume LIC; top center: the corresponding slice through a region-of-interest mask defined as a function of velocity magnitude; top right: the masked LIC texture. Bottom left: a masked input texture for LIC; bottom right: the result after applying LIC to this input.*
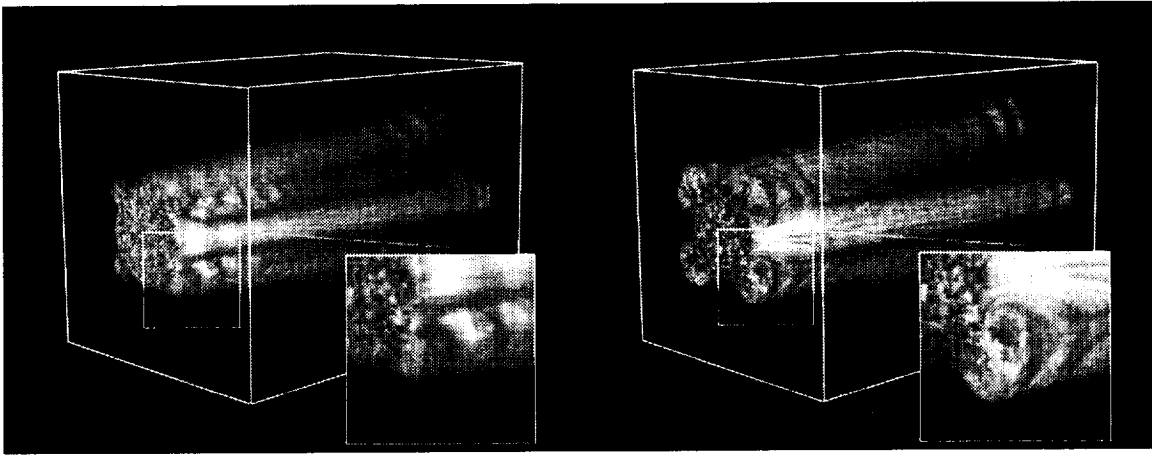

FIG. 3. *A 3D view of the volume-rendered solid textures. Left: results when the region-of-interest segmentation is applied as a postprocess after LIC. Right: results when region-of-interest masking is applied to the input texture before performing the LIC. Differences are particularly evident in the area of the volume just behind the tabs in the jet, where the four pairs of distinct but short-lived, counter-rotating vortices are formed.*

3

The data used to generate these images was obtained from a numerical simulation of effect of tabs on jets. The goal of the flow research was to investigate the extent to which one might be able to reduce the amount of noise produced by a jet engine by adding tabs to generate vortices that could facilitate the mixing of the hot, supersonic flow with the colder, subsonic coflow. The flow data was generated on a nearly rectangular grid, of resolution 101x101x148; the resolution of the input and output textures was defined to be twice as large (202x202x296), so that the details of the flow could be more easily seen. The difference between figures 2 and 3 is especially apparent in the vicinity of the four pairs of counter-rotating vortices that were induced by the addition of the tabs.

5. Sparse Input Textures. When line integral convolution is applied to a solid noise texture, even one that has been masked by a region of interest function, the resulting 3D output texture is difficult to visualize as anything other than a solid object, perhaps with fuzzy boundaries. By applying the LIC instead over an input texture consisting of a sparse set of distributed points [8], taking care to advect the empty space along with the full, it is possible to instead produce a solid texture of the kind shown in figure 4, that in effect represents a scan converted collection of aesthetically placed, densely clustered streamlines. Best results are achieved with this approach when the opaque points in the input texture, rather than being laid out on a regular grid or distributed by a purely random or jittered sampling method, are instead specified according to an approximate Poisson-disk distribution, in which the spots are both prevented from falling too close to each other and at the same time will be found in approximately equivalent numbers on any arbitrary plane through the volume [8]. Figure 5 illustrates the differences between these various point distribution methods. The input texture used to generate the volume shown in figure 4 was premultiplied by the same region of interest mask used in figures 2 and 3 above; constraining the streamlines to originate in more rapidly moving regions of the flow but allowing them to then extend where they may.
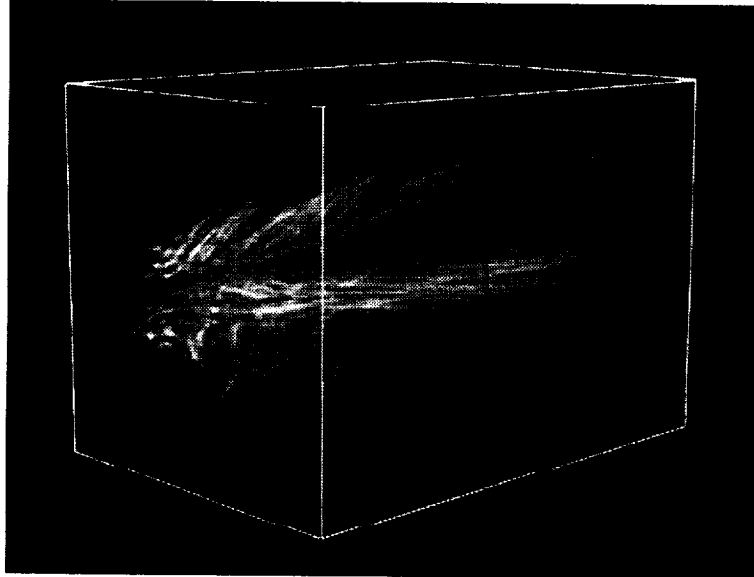

FIG. 4. *When line integral convolution is performed in 3D over an input texture of distributed points, the resulting output texture resembles a collection of scan-converted streamlines. The texture in this image has been volume rendered, and shaded according to the direction of the flow, as suggested by Stalling et al. [9], but it nevertheless remains difficult to adequately appreciate the 3D character of the flow or the relative spatial orientations of the individual lines in this image.*
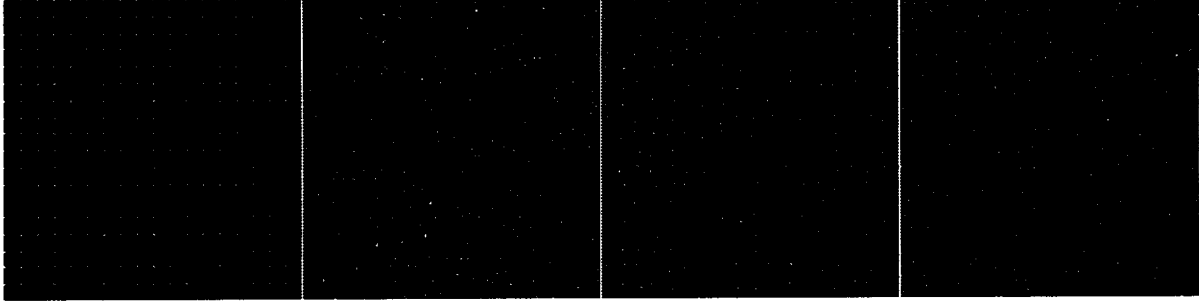
4

FIG. 5. *An illustration of four point distribution methods. From left to right: uniformly distributed points, randomly distributed points, a jittered point distribution (where $\Delta_{max} = 0.25$ of the inter-element spacing), and an approximate Poisson disk distribution. When $\Delta_{max} < 1.0$ in the jittered point distribution, there are guaranteed to be rows and columns in the image on which no sample point can fall. However, as Δ_{max} approaches 1.0, the minimum guaranteed separation between adjacent sample points declines to zero.*

LIC can also be used in conjunction with a sparse input texture to more effectively visualize a 3D flow in the vicinity of a surface of interest. In figure 6 below, a ridge strength function has been applied over the velocity magnitude volume shown earlier, to define a boundary surface between the regions of relatively faster and relatively slower flow. An advantage of this approach, particularly for a pulsing flow such as this one, is that it facilitates the coherent segmentation of the data according to the relationships between the values, while allowing the precise definition of fast and slow to remain fluid.
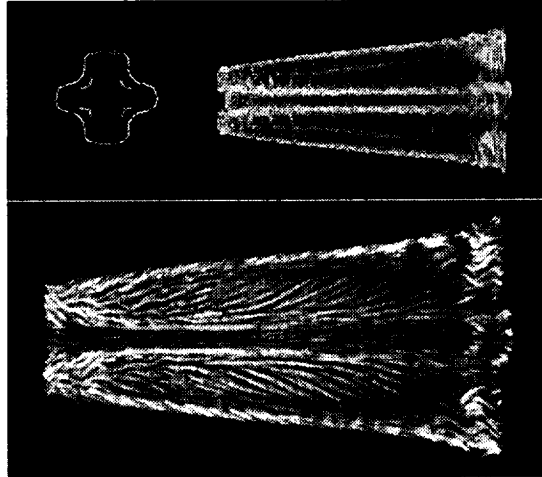

FIG. 6. *LIC is used here to show 3D flow in the vicinity of a surface of interest. Top: ridges of velocity magnitude define a thin surface between the faster and slower moving regions of the flow. Bottom: the result when 3D LIC is applied, with a short filter kernel, to an input texture consisting of evenly distributed points among the voxels intersected by this surface.*

Eliminating the need to project the flow directions onto the surface decreases the likelihood of generating a misleading impression of the flow, a problem discussed by Max et al. in some detail [10]. Because all calculations are performed in 3D, the tufts in the output texture are able to accurately reflect the local 3D orientation of the flow in the immediate vicinity of the surface of interest. We expect that the effectiveness of this sort of visualization would be considerably enhanced by stereo viewing.

6. Clarifying the Depth Relations within the Flow. We alluded earlier to the problem of effectively conveying the 3D shape and relative depth relations among the similarly directed, densely clustered streamlines traced by LIC. At first glance this appears to be a problem of simple differentiation: like-colored lines, existing at different depths but projected onto adjacent pixels in a particular view, will

5

necessarily appear to coalesce into an indistinguishable clump. Phrased in this fashion, the problem points to a seemingly obvious solution: to differentiate the individual lines by rendering them in different colors. Unfortunately, as can be seen in figure 7, the introduction of color variations alone does little to improve the clarity of the depth order relations among the overlapping lines. Figure 8 shows why we should expect this to be the case, and suggests a better solution.
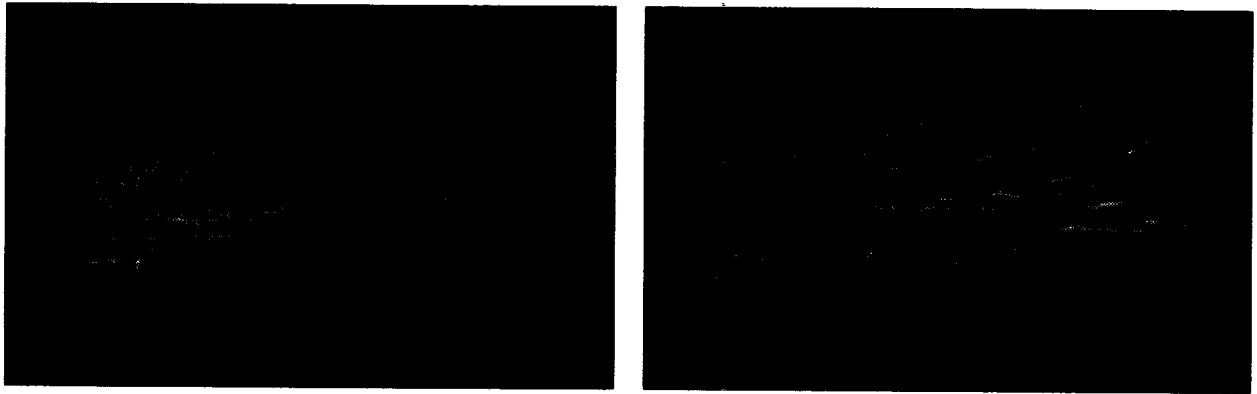

FIG. 7. *The introduction of color variations alone does little to clarify the depiction of overlapping streamlines.*

In ordinary binocular vision, depth discontinuities generally coincide with the presence of inter-ocularly unpaired regions in the views from each eye [11]. Artists and illustrators have long exploited this correspondence by using gaps to indicate the passing of one object behind another; examples of this technique can be found as early as in the Paleolithic paintings within the caves of Lascaux. One of the first haloed line drawing algorithms for computer graphics was proposed by Appel et al. in 1979 [12].
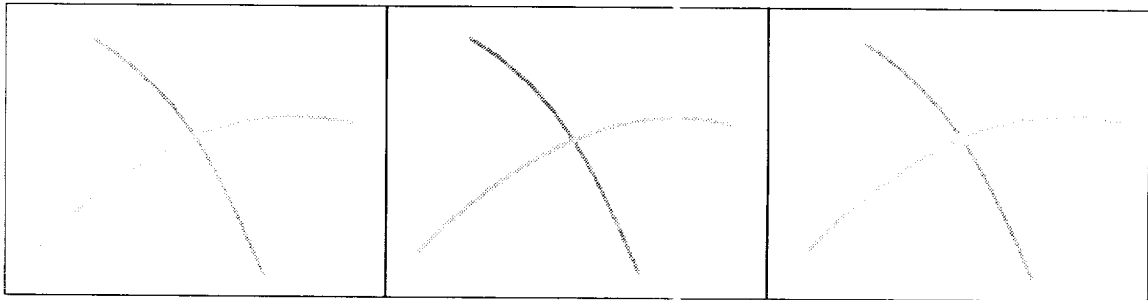

FIG. 8. *Two overlapping lines of roughly equivalent luminance but differing hue. Depth order relations are explicitly emphasized in the leftmost and rightmost depictions through the introduction of the subtle gaps flanking the foremost line.*

A simple modification to the basic LIC algorithm allows the efficient computation of a matched pair of textures, as shown in figure 9, that can be used to generated images like the one in figure 10, in which the depth discontinuities are highlighted by gaps. We automatically define a subtle and smoothly continuous 3D visibility-impeding halo region that fully encloses each streamline in the original 3D texture by performing the LIC simultaneously over two input textures containing identically located spots of concentric sizes. Because the streamline tracing need only be done once for the pair of volumes, the overhead associated with the creation of the halos is kept to a minimum. Halos are implemented, during raycasting volume rendering, by decreasing the contribution to the final image of any voxel encountered after a halo has been entered and subsequently exited by an amount proportional to the largest previously encountered halo opacity.
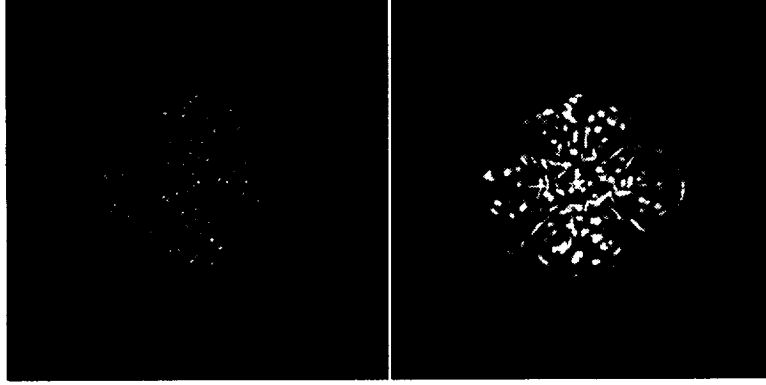
FIG. 9. *Slices through a pair of 3D LIC textures representing a set of streamlines (left) and an enclosing set of 3D halos (right).*
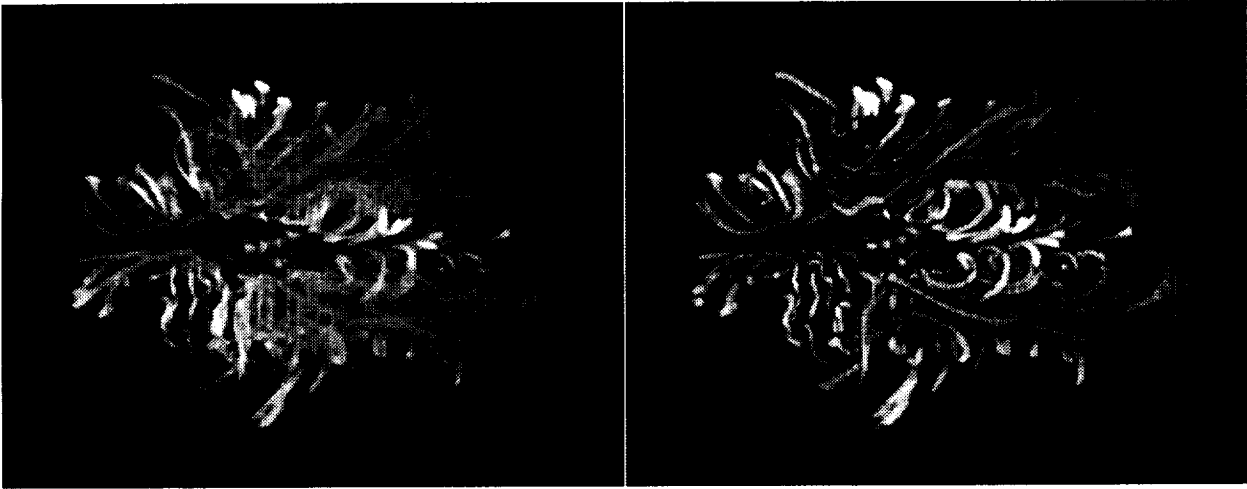

FIG. 10. *A side-by-side comparison illustrating the effectiveness of visibility-impeding halos for conveying depth discontinuity information and for facilitating appreciation of the depth extent in the flow.*

Specifically, the volume renderer takes as input two volume textures (streamlines and halos), and rays are traced through each. Color and opacity are accumulated, at first, in the ordinary manner along the rays through the volume of streamlines. As tracing proceeds, simultaneously, through the volume of halos, entries and exits into the halo regions are recorded, and the maximum value encountered in each halo region is noted. When a halo exit is detected, the value of the accumulated opacity along the ray is incremented by an amount proportional to the maximum density encountered in the previously traversed halo region. It is because each line will necessarily be everywhere surrounded by its own halo that it is important to allow the voxels lying between the entrance and exit points of the first-encountered halo to be rendered in the normal fashion. It should be noted that this particular implementation assumes a black background, and will fail to indicate the existence of depth discontinuities between lines whose halos overlap in 3-space, even if the lines themselves do not actually intersect.

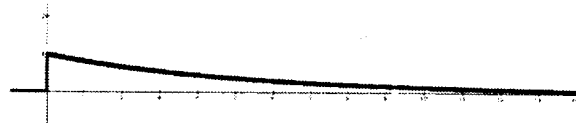**7. Indicating Directional Information via oriented fast-LIC .** Wegenkittl et al. [13] recently showed how basic LIC could be used in conjunction with an asymmetric, triangular shaped filter kernel and a sparse 2D input texture to produce images in which information about the forward/backward direction of the flow is locally conveyed through variation in the intensity of the rendered streamlets. The images

produced by this method are very compelling, and inspired us to see if we could come up with a simple modification to Stalling and Hege's fast-LIC algorithm that would allow the efficient computation of 3D LIC textures that conveyed directional information in a similar manner. The fast-LIC method, which we are using for 3D texture generation, gains significant efficiency through the use of incremental calculations for the computation of the integrated intensity along a streamline, but as formulated it appears to require the use of a box filter.

$$I'_0 = \sum_{i=0}^{flength} v_i; \quad I'_1 = \sum_{i=1}^{flength+1} v_i; \quad I'_1 = I'_0 - v_0 + v_{flength+1}$$

We quickly realized that we would be able to achieve both the oriented effect of OLIC and the computational advantage of fast-LIC if instead of a box filter or a triangular filter we employed an asymmetric filter of the form

$$I'_0 = \sum_{i=0}^{flength} v_i \, c^i, \quad c < 1$$

where *flength* represents the length of the filter kernel (or the number of weighted samples, in the forward direction along the streamline from each point, that are combined to determine the value at the corresponding point in the output texture), v_i represents the value of the input texture that lies under the ith sample point, and c is a constant that influences the rate of the decrease in the weight of the samples that lie farther along the streamline.
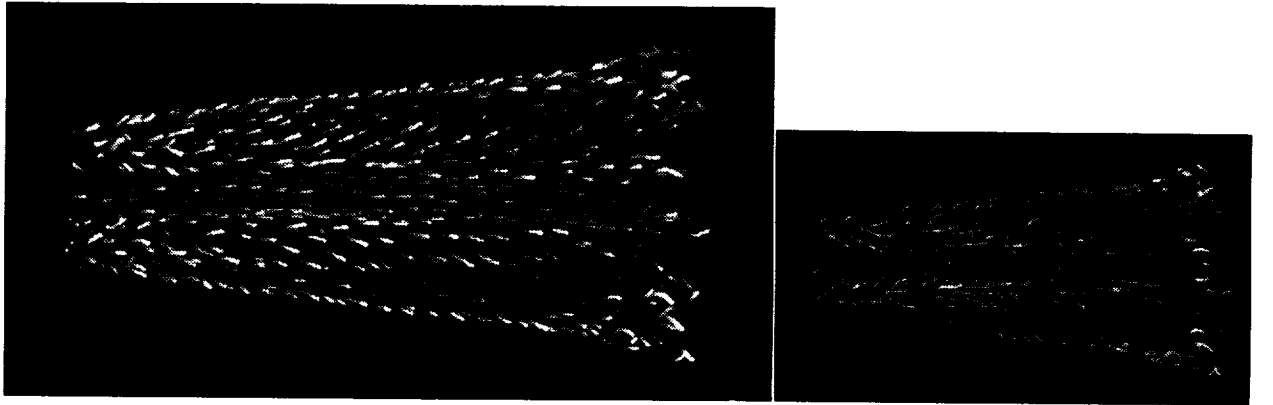

FIG. 11. *Left: Tapered streamlines convey the forward/backward direction of the 3D flow. This depiction of orientation information was inspired by the OLIC method previously introduced by Wegenkittl et al. [13]. Right: the same streamlines without tapering.*

8

With this new filter definition, one can continue to use incremental calculations to derive the integrated output texture values at subsequent voxels along a streamline:

$$I_1' = (I_0' - v_0)/\mathbf{c} + v_{flength+1}\mathbf{c}^{flength}$$

$$I_{-1}' = (I_0' - v_{flength}\mathbf{c}^{flength})\mathbf{c} + v_{-1}$$

The equations as written reflect the fact that we successively shift new values into an array v of fixed bounds, with the result that at each step the current origin remains denoted v_0 and the intensity of the next point denoted I_1' or I_{-1}', depending on the direction of travel along the streamline.

8. Conveying Additional Information over the flow through local variations in width and color. When 3D LIC is used in conjunction with volume rendering, several devices can be easily employed to allow the intuitive display of additional scalar variables over the 3D flow texture. As we demonstrated in earlier related work [8], line width can be easily varied, at the time of rendering, if the LIC has been performed in parallel over an ordered series of input volumes (containing spots of increasing sizes), to generate a correspondingly ordered series of output LIC textures. The composite image can be adaptively generated from the precomputed LIC textures by allowing the values in a registered scalar field to define
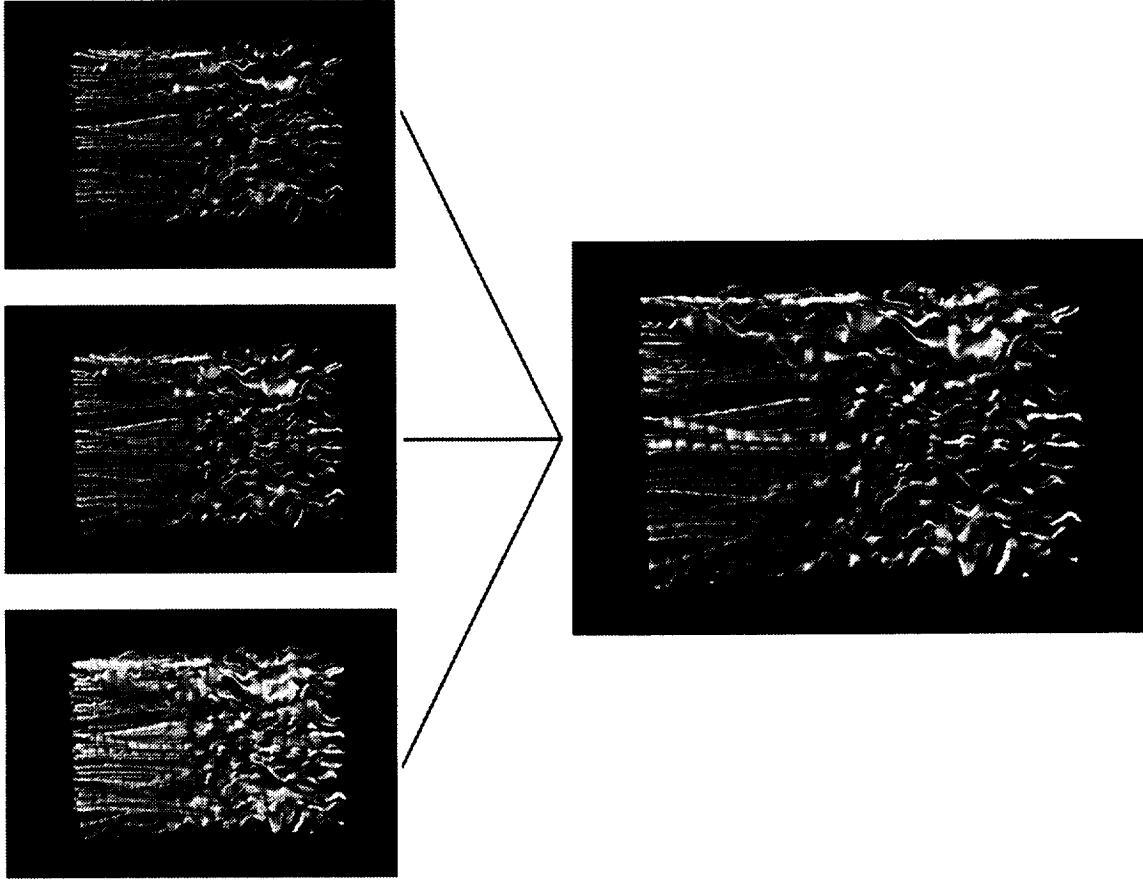

FIG. 12. *A composite texture of lines of varying widths is assembled from a concentric series of LIC textures according to the values given by a third function- in this case, velocity magnitude. Although only three images are shown on the left, six volumes in all were needed to generate the image on the right, three for the streamlines and three more for the halos.*

the index of the texture volume into which the texture lookup is performed at any particular point. In the image of figure 12, the regions of relatively slower velocity in the flow near the upper and lower boundaries are indicated by the increased widths of the streamlines in those regions. The ability to elegantly represent lines of varying width is one of the advantages of a volume representation. The numerically simulated flow data depicted in figure 12 was generated on a nearly rectangular grid at a resolution of 101x101x151; the LIC texture was computed from this data at a resolution of 202x202x302. Of interest in this simulation is the possible role of friction in the breakdown in the coherence of the lateral oscillations in the flow.

Color remains perhaps the most powerful means of non-intrusively conveying additional information over a flow. The use of a volume rendering approach makes it easy to vary the color definition across any predefined 3D texture according to the values in an accompanying scalar field. In figure 13, color is used to indicate total vorticity magnitude across the flow; the striations in the color scheme highlight the pressure waves that are being propagated along the flow in the axial direction. In figure 14, two different color washes are illustrated over a flow through a rectangular aperture. On the left, color is used to indicate temperature, showing the effects of friction across the boundary layers along the top and bottom of the aperture, and on the right, saturation is used to indicate the magnitude of the streamwise vorticity, effectively highlighting the more turbulent regions of the flow.
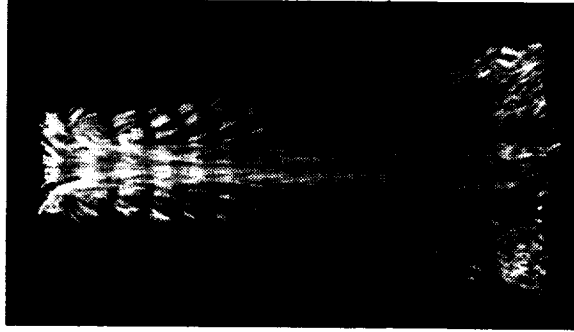

FIG. 13. *The color of the lines in this image are defined, using an approximate heated-object color scale, according to the local values of vorticity magnitude across the volume. This depiction highlights the propagation of the periodic pressure waves along the axial direction of the flow.*
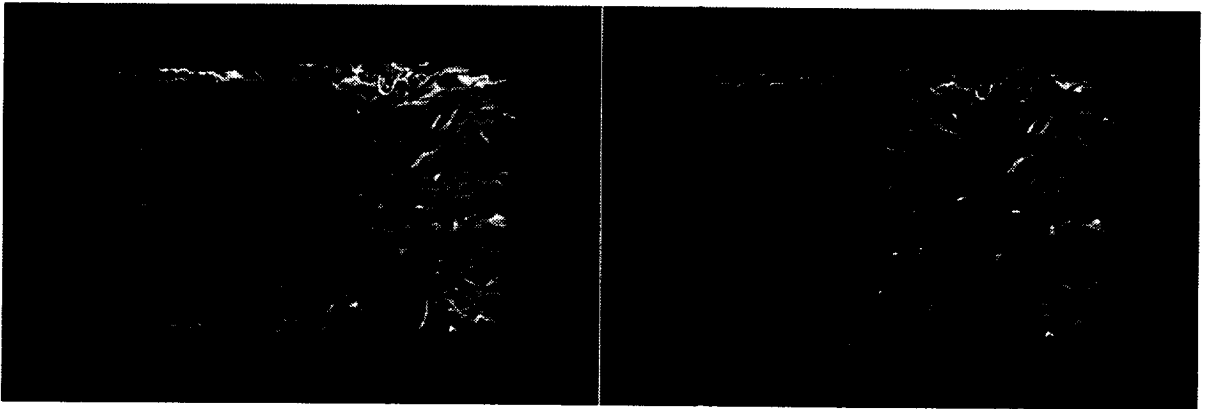

FIG. 14. *Two different color wash images generated from the same flow data. On the left, color varies from red to yellow with increasing temperature; on the right, saturation increases from grey to purple according to the magnitude of the streamwise vorticity.*

9. Conclusions and future work. Although 3D LIC has not been commonly used for the visualization of 3D flow, we believe that it is a viable tool for this purpose – particularly in those instances where high image quality is desired and the ability to generate images at interactive frame rates is not required. We have demonstrated several new strategies for effectively illustrating 3D flow with volume LIC, including the use of visibility-impeding halos to emphasize the discontinuity in depth between overlapping lines and the use of an asymmetric filter kernel in combination with fast-LIC for the efficient computation of 3D flow textures that reveal directional information. We are continuing to investigate methods for more efficiently generating smooth, cyclic animations of 3D LIC textures along streamlines in steady flow data, and are actively working on an extension of our 3D LIC algorithm to the visualization of pathlines in 3D unsteady flow data.

10. Acknowledgments. We are grateful to Marc Levoy for providing the original volume rendering platform upon which this work was built, and to Kwan-Liu Ma, Tom Crockett, David Banks, Hans-Christian Hege and many others for insightful comments on this work.

REFERENCES

[1] B. CABRAL and C. LEEDOM. *Imaging Vector Fields Using Line Integral Convolution*, Computer Graphics Proceedings, Annual Conference Series, 1993, pp. 263-269.

[2] H.-W. SHEN, C. R. JOHNSON and K.-L. MA. *Visualizing Vector Fields Using Line Integral Convolution and Dye Advection*, 1996 Symposium on Volume Visualization, pp. 63-70.

[3] D. STALLING and H.-C. HEGE. *Fast and Resolution-Independent Line Integral Convolution*, Computer Graphics Proceedings, Annual Conference Series, 1995, pp. 249-256.

[4] L. FORSELL. *Visualizing Flow Over Curvilinear Grid Surfaces Using Line Integral Convolution*, Proceedings of Visualization 94, pp. 240-247.

[5] H. BATTKE, D. STALLING and H.-C. HEGE. *Fast Line Integral Convolution for Arbitrary Surfaces in 3D*, in *Visualization and Mathematics*, H.-C. Hege and K. Polthier, eds., Springer-Verlag, 1997, pp. 181-195.

[6] X. MAO, M. KIKUKAWA, N. FUJITA and A. IMAMIYA. *Line Integral Convolution for Arbitrary 3D Surfaces Through Solid Texturing*, Proc. Eighth Eurographics Workshop on Visualization in Scientific Computing, April 1997.

[7] F. POST and J. VAN WIJK. *Visual Representation of Vector Fields: Recent Developments and Research Directions*, in *Scientific Visualization: Advances and Challenges*, L. Rosenblum et al., eds., IEEE Computer Society and Academic Press, pp. 367-390, 1994.

[8] V. INTERRANTE. *Illustrating Surface Shape in Volume Data via Principal Direction-Driven 3D Line Integral Convolution*, Computer Graphics Proceedings, Annual Conference Series, 1997, pp. 109-116.

[9] D. STALLING, M. ZÖCKLER and H.-C. HEGE. *Fast Display of Illuminated Field Lines*, IEEE Transactions on Visualization and Computer Graphics, 3(2), pp. 118-128, April/June 1997.

[10] N. MAX, R. CRAWFIS and C. GRANT. *Visualizing 3D Velocity Fields Near Contour Surfaces*, Proc. Visualization 94, pp. 248-255.

[11] S. SHIMOJO and K. NAKAYAMA. *Real World Occlusion Constraints and Binocular Rivalry*, Vision Research, 30(1), pp. 69-80, 1990.

[12] A. APPEL, F. J. ROHLF and A. STEIN. *The Haloed Line Effect for Hidden Line Elimination,* Proceedings of SIGGRAPH 79, pp. 151-157.

[13] R. WEGENKITTL, E. GRÖLLER and W. PURGATHOFER. *Animating Flowfields: Rendering of Oriented Line Integral Convolution,* Proc. Computer Animation 97, June 1997, pp. 15-21.

REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE July 1998	3. REPORT TYPE AND DATES COVERED Contractor Report

4. TITLE AND SUBTITLE

Recent Advances in Visualizing 3D Flow with LIC

5. FUNDING NUMBERS

C NAS1-97046
WU 505-90-52-01

6. AUTHOR(S)

Victoria Interrante
Chester Grosch

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Institute for Computer Applications in Science and Engineering
Mail Stop 403, NASA Langley Research Center
Hampton, VA 23681-2199

8. PERFORMING ORGANIZATION REPORT NUMBER

ICASE Report No. 98-26

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Langley Research Center
Hampton, VA 23681-2199

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

NASA/CR-1998-208437
ICASE Report No. 98-26

11. SUPPLEMENTARY NOTES

Langley Technical Monitor: Dennis M. Bushnell
Final Report
To appear in IEEE Computer Graphics and Applications

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified Unlimited
Subject Category 60, 61
Distribution: Nonstandard
Availability: NASA-CASI (301)621-0390

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Line Integral Convolution (LIC), introduced by Cabral and Leedom in 1993, is an elegant and versatile technique for representing directional information via patterns of correlation in a texture. Although most commonly used to depict 2D flow, or flow over a surface in 3D, LIC methods can eq 1ivalently be used to portray 3D flow through a volume. However, the popularity of LIC as a device for illustrating 3D flow has historically been limited both by the computational expense of generating and rendering such a 3D text ire and by the difficulties inherent in clearly and effectively conveying the directional information embodied in the volumetric output textures that are produced. In an earlier paper, we briefly discussed some of the factors that may underlie the perceptual difficulties that we can encounter with dense 3D displays and outlined several strategies for more effectively visualizing 3D flow with volume LIC. In this article, we review in more detail techniques for selectively emphasizing critical regions of interest in a flow and for facilitating the accurate perception of the 3D depth and orientation of overlapping streamlines, and we demonstrate new methods for efficiently incorporating an indication of orientation into a flow representation and for conveying additional information about related scalar quantities such as temperature or vorticity over a flow via subtle, continuous line width and color variations.

14. SUBJECT TERMS 3D flow visualization; line integral convolution; volume rendering; depth representation	15. NUMBER OF PAGES 17
	16. PRICE CODE A03

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT

NSN 7540-01-280-5500