Multi-Granularity Noise for Curvilinear Grid LIC

Xiaoyang Mao, Lichan Hong †, Arie Kaufman ‡, Noboru Fujita, Makoto Kikukawa and Atsumi Imamiya
Yamanashi University, Japan
mao@esi.yamanashi.ac.jp
† Bell Laboratories, Lucent Technologies, USA
lhong@research.bell-labs.com
‡ State University of New York at StonyBrook, USA
ari@cs.sunysb.edu

Abstract

A major problem of the existing curvilinear grid Line Integral Convolution (LIC) algorithm is that the resulting LIC textures may be distorted after being mapped onto the parametric surfaces, since a curvilinear grid usually consists of cells of different sizes. This paper proposes a way for solving the problem through using multi-granularity noise as the input image for LIC. A stochastic sampling technique called Poisson ellipse sampling is employed to resample the computational space of a curvilinear grid into a set of randomly distributed points. From this set of points, we are able to reconstruct a noise image with its local noise granularity being adapted to the physical space cell size of the grid.

Keywords: scientific visualization, vector field visualization, texture synthesis, texture mapping.

Introduction

Line Integral Convolution (LIC) [2, 3] has been attracting large attention as a powerful texture-based vector field visualization technique. Texture based approaches are particularly useful in visualizing large and complex vector fields, where traditional techniques such as arrow plotting and particle advection may either result in cluttering images or fail to capture some important features of flows due to inadequate sampling of vector fields. The first attempt to employ texture synthesis technique for vector field visualization was made by van Wijk [7]. He proposed a way to generate various textures (called *spot noise*) through convolving an input white noise image with randomly distributed 2D filter kernels (called *spot*). By choosing the shape of filter kernels to be elliptical and aligning the major axes of the ellipses with the directions of vectors, textures effectively depicting the global appearance of flows can be obtained. Instead of 2D filter kernels, LIC convolves a white noise image with one dimensional filter kernels defined along the local streamlines of a vector field, so that highly curved complex flows can also be effectively visualized.

The originally LIC algorithm proposed by Cabral and Leedom [2] can be used only for regular 2D Cartesian grids. For many applications, however, it is important to visualize the flow over or near a 3D surface. Forssell and Cohen [3] successed in extending LIC for visualizing the flow on a 3D parametric surface represented as a slice of a 3D curvilinear grid. Their algorithm is realized as the two-way mapping between the computational space and the physical space of a curvilinear grid. First, the vectors in the 3D physical space are mapped into the computational space, which is a regular 2D Cartesian grid of unit cell size. Then, the original LIC algorithm is performed and the resulting 2D LIC image is mapped back onto the 3D parametric surface in the physical space. A major advantage of this approach is that by making use of texture mapping hardware, it is possible to render the resulting LIC texture mapped 3D surface in real time. However, as a curvilinear grid usually consists of cells of drastically different sizes, the resulting LIC texture may be largely distorted after being mapped to the 3D parametric surfaces. Such texture distortion is particularly undesirable for the purpose of visualization because a distorted LIC texture may even cause some wrong conclusions about the characteristic of a flow.

Forssell and Cohen [3] suggested to vary the length of convolution kernels in computational space according to the local cell sizes in the physical space. This usually can not resolve the problem completely, because not only the filter kernels but also the noise granularity of the input image is stretched or condensed with the mapping. In other words, in addition to the sizes of filter kernels, it is also necessary to adapt the local noise granularity of input image according to the local cell sizes in the physical space.

This paper presents a new technique for generating multi-granularity noise. A stochastic sampling technique called *Poisson ellipse sampling* is employed to resample the computational space of a curvilinear grid with a set of randomly distributed points. Each point is associated with a binary noise value and an elliptical region whose size is decided based on the physical space cell size at that position. Then a multi-granularity noise can be constructed from these sample points for generating 3D LIC texture without distortion in the physical space. In the next section, we first briefly review the curvilinear grid LIC algorithm and show an example of the distorted LIC texture. Some related work is also discussed there. Then, we introduce the Poisson ellipse sampling and explain how to generate the LIC image with the multi-granularity noise. Before giving the conclusion, we show some experimental results of the presented technique.

Background

Given a 2D vector field represented as a regular Cartesian grid, the original LIC algorithm proposed by Cabral and Leedom [2] takes a white noise image of the same size as its input image. The output image, which is also of the same size as the grid, is generated by convolving the input image at each pixel with a 1D filter kernel defined on the local streamline passing through the corresponding cell in the grid. A curvilinear grid can be considered to be obtained by non-linearly transforming a regular grid of unit cell size so as to fill a volume or warp around an object of complex shape while keeping the grid topology. Usually, the regular grid defining the logical organization of the grid is called the *computational space*, and the warped structure is called the *physical space* of the curvilinear grid. To visualize the flow over a 3D parametric surface represented as a slice of a 3D curvilinear grid, the technique developed by Forssell and Cohen [3] makes use of the above 2D LIC algorithm by first mapping the 3D physical space velocity vectors on a slice into its computational space, which is a 2D regular Cartesian grid. Denoting the physical space and computational space coordinates of a point as $P_p = (x, y, z)$ and $P_c = (\xi, \eta, \zeta)$, respectively, then the computational space velocity vectors can be obtained by multiplying the physical space velocity vectors by the inverse Jacobian matrix:

$$\begin{bmatrix} \frac{\partial \xi}{\partial t} \\ \frac{\partial \eta}{\partial t} \\ \frac{\partial \zeta}{\partial t} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial x}{\partial t} \\ \frac{\partial y}{\partial t} \\ \frac{\partial z}{\partial t} \end{bmatrix}$$

Figures 2, 3 and 4 show the grid over the surface of a spiked blunt body, the velocity vectors on the surface, and the velocity vectors after being mapped to the computational space, respectively.

Now an LIC image can be calculated in the computational space by dropping the $\partial\zeta/\partial t$ component and using the 2D vector $[\partial\xi/\partial t, \partial\eta/\partial t]^T$. Figure 5 is the LIC image computed with the vector field of Figure 4. Finally, the flow over the parametric surface can be visualized by mapping the resulting 2D LIC texture onto the 3D surface. Figure 6 is obtained by mapping the texture in Figure 5 onto the parametric surface in Figure 2.

The texture shown in Figure 6 has obviously been largely condensed at the spike, where the grid is of high density, and has been stretched on the body due to the large cell size. Such a distortion even causes the illusion that the flow over the body is faster than that around the spike. Moreover, the directional blurring effect of LIC has been partially counteracted by the aliasing defect and can hardly been seen around the tip of spike.

To rectify such a distortion, Leeuw and van Wijk [5] mentioned the possibility of using rectilinear white noise input images in extending their enhanced spot noise technique for curvilinear grid. Obviously this approach is applicable only to those grids where the cells lying in the same row or column are all of the same size at least in one dimension. Kiu and Banks [4] proposed a method for generating multi-frequency noise as the sum of masked images obtained by applying low-pass filters of increasing widths to the original high frequency white noise. It seems possible to add some modifications to their algorithm and generate a multi-frequency noise by choosing the width of the low-pass filter at each pixel according to the local cell size in the physical space. However, high frequency components have usually been filtered out in such a multi-frequency noise, and therefore the directional blurring effect in the resulting LIC image may become less obvious compared with the images obtained with white noise bitmap. Our new technique to be presented in the following section adaptively changes the local noise granularity of input image while preserving its high frequency components.

Poisson ellipse sampling

Poisson ellipse sampling is a simplified version of the *Poisson sphere/ellipsoid sampling* used for volume rendering 3D curvilinear grids through splatting [6]. With the original Poisson sphere/ellipsoid sampling, the physical space of a 3D curvilinear grid is resampled with a set of randomly distributed points. Each of them has a surrounding spherical or ellipsoidal region which inhibits the existence of other sample points into that region. The sizes of spheres and ellipsoids are designed to match the local cell sizes of the grid. Ellipsoids are used to resample the grid with different rates for each of the three dimensions.

As shown in Figure 1, the idea presented in this paper is to resample the 2D computational space instead of the 3D physical space, for a slice of a curvilinear grid in a similar way to obtain a set of randomly distributed points, each of which is associated with a circular or elliptical region. As a disc is also an ellipse whose two axes are identical, here we treat it as an ellipse and call the sampling method *Poisson ellipse sampling.* The input image of LIC is reconstructed from this set of sample points by assigning a binary noise value to each ellipse. Obviously, the local noise granularities are decided by the sizes of these ellipses. Therefore we can obtain a multi-granularity noise simply by adjusting the sizes of ellipses to be inversely proportional to the local cell sizes in the physical space. In other words, we use a small ellipse for a sample point if it corresponds to a position of large cell size in the physical space and use a large ellipse if the position is of small cell size. Assuming the Jacobian matrix for a point in physical space is

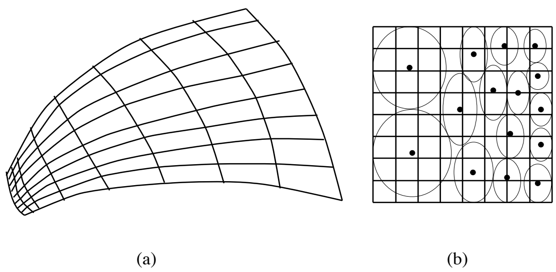

Figure 1: *Poisson ellipse sampling: (a) The physical space of a curvilinear grid. (b) Resampling of the computational space.*

$$\mathcal{J} = \left[\begin{array}{ccc} \boldsymbol{J}_\xi & \boldsymbol{J}_\eta & \boldsymbol{J}_\zeta \end{array} \right] = \left[\begin{array}{ccc} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{array} \right]$$

then the local cell sizes along dimension ξ, η and ζ can be measured by $|J_\xi|, |J_\eta|$ and $|J_\zeta|$, respectively. Suppose we are visualizing the flow on a slice which is orthogonal to ζ dimension in the computational space, then the sizes of the two axes of the ellipse can be calculated with $|J_\xi|$ and $|J_\eta|$ in the following way:

$$e_\xi = a_\xi + \frac{maxsize_\xi - |J_\xi|}{maxsize_\xi - minsize_\xi} b_\xi$$

$$e_\eta = a_\eta + \frac{maxsize_\eta - |J_\eta|}{maxsize_\eta - minsize_\eta} b_\eta$$

Here $maxsize_\xi, minsize_\xi$ are the largest and smallest cell sizes in dimension ξ and $maxsize_\eta, minsize_\eta$ are the largest and smallest cell size in dimension η. $a_\xi, b_\xi, a_\eta, b\eta$ are user-specified float values which control the range of granularity, that is, e_ξ takes the value between a_ξ and $a_\xi + b_\xi$, and e_η takes the value between a_η and $a_\eta + b_\eta$. For example, a_ξ and a_η can be chosen as 1.0. b_ξ and b_η are the half of the grid resolution in dimension ξ and η, respectively.

Currently, a stratified dart throwing algorithm is used for resampling the computational space. That is, for each cell in the computational space, we generate a random points inside the cell and check if it is already enclosed in the ellipse of an existing sample point. If not, calculate the Jacobian matrix for this point by interpolating the Jacobian matrices at the four corner points of the cell, and decide an ellipse as well as a binary noise value for the point. This process is repeated until either the generated points have reached some expected number, or no point has been generated for a certain period. Here is the pseudo code of the resampling algorithms:

```
CHECKED ← 0;
for  each cell in the computational space
  repeat
    Generate a random point p(ξ,η);  /*ξ,η ∈ [0,1]*/
    if P  is not covered by existing ellipse extents
      begin
        CHECKED ← 0;
        Interplate J
           from the four corner points of the cell;
        Calculate e_ξ, e_η;
        Generate a binay noise value;
        Append the point to the sample points list;
      end
    else
      CHECKED ← CHECKED + 1;
  until CHECKED >= NUM_TO_STOP;
```

Differing from Poisson sphere/ellipsoid sampling, in which spheres and ellipsoids should overlap with each other to prevent leaving gaps during volume rendering [6], the ellipses should not overlap with each other in Poisson ellipse sampling for retaining the high frequency in reconstructed noise. This is realized by using an extent which is twice the size of the ellipse in point-ellipse enclosing test at the fifth line of the above algorithm. As shown in Figure 1, the resampled computational space actually has many regions not beeing covered by any ellipses. Precisely, we need to generate the weighted Voronoi diagram by using each ellipse as the power diagram for each sample point [1] and construct the input noise by assigning a binary value to each of the subregions in the Voronoi diagram. As explained in the next section, here we assign the noise value to each ellipse, and for the region which is not covered by any ellipse, the noise value at the closest sample point will be used. We found that such an approximation is enough to generate LIC images of reasonable quality from the perspective of solving the texture distortion problem.

LIC on multi-granularity noise

Instead of reconstructing a white noise image and using it as the input for LIC, we take the sample points and their binary noise values as the input directly and reconstruct the noise values during the execution of LIC only at those points required for convolution. The local streamline in the computational space is calculated in the same way as in [2] with the vectors mapped from the physical space. To decide the input noise value at a point, we search the list of sample points to see if the point is covered by the ellipse of a sample point. If such a sample point is found, then its noise value is used. Otherwise, the noise value of the closest sample point is used.

Results

The Poisson ellipse sampling technique together with the LIC algorithm on multi-granularity noise has been implemented on the commercial visualization software AVS [1]. The graphics workstation used was SGI Indy with R5000 150MHZ CPU and 64 megabytes memory running an Iris 6.2 operating system. In this section we will show some results of applying the new technique to two datasets obtained from CFD simulations.

The first dataset is the same one as that used in the second setion. This is the numerical simulation result of the supersonic flow attacking a spiked blunt body [8]. The size of the original grid is $80 \times 30 \times 80$ and the second slice from the body surface is used here (See Figure 2). As shown in Figure 7, the resampling process results in a set of points sparsely distributed at spike (left area in Figure 7) and densely distributed on the blunt body (right area in Figure 7). The multi-granularity noise reconstructed from these sample points is shown in Figure 8. Figure 9 is the resulting 2D LIC image and the final texture mapped image is shown in Figure 10. Compared with the LIC texture generated with a constant granularity white noise (See Figure 6), here the texture distortion due to the difference of cell sizes has been compensated and the flow near the tip of spike is also clearly visualized.

The second example is the *Post* dataset from NASA (See Figure 11). Figure 12 is the texture mapped LIC image generated with constant granularity noise. The resampling of the computational space is shown in Figure 13. The distribution of sample points is getting sparse gradually from the top to the bottom area, as the cell sizes are getting smaller toward the center in the physical space. Figure 14 is the LIC image generated with the multi-granularity noise reconstructed from the sample points shown in Figure 13. Figure 15 is obtained by texture mapping the LIC image in Figure 14 back to the physical space. Compared with Figure 12, we can see in Figure 15 the feature size of the LIC texture around the center area becomes larger and closer to those of outer regions.

Concluding remarks

A new technique for generating distortion free LIC textures for 3D curvilinear grids has been presented. By employing an adaptive stochastic sampling technique called Poisson ellipse sampling, we are able to adjust the noise granularity continuously

[1] AVS is a trademark of Advanced Visual Systems Inc.

over the input image according to the local cell sizes of the curvilinear grids.

As we mentioned previously, a distorted LIC image can give a user a wrong impression about a vector field. From this point of view, the LIC image generated with the multi-granularity noise is superior to that generated with the constant granularity noise. However, as low sampling rate is used for the regions of small cell sizes, detailed information at these regions may be lost in an LIC image generated with the multi-granularity noise. Conversely, an LIC image generated with the constant granularity white noise preserves all the information and we can reveal the details of these regions by zooming into them. In case a curvilinear grid consists of cells of drastically different sizes, it might be difficult to remedy the texture distortion completely because the range of sampling rates in the computational space can not exceed the resolution of the grid. One possible solution is to resample the vector field into higher resolution, though this is not a desirable approach from the perspective of time/space efficiency.

A future research direction is to improve the Poisson ellipse resampling algorithm. The current dart throwing algorithm usually leaves many regions uncovered by any ellipses and we need a method to generate a more densely packed distribution. Another direction is to combine the multi-granularity noise with other texture-based visualization tools. For example, it is easy to add some minor modifications to the spot noise algorithm [7], so that the multi-granularity noise can be used for generating distortion free spot noise textures on curvilinear grids. The proposed method for generating multi-granularity noise may also be used for synthesizing other kinds of distortion-free textures on parametric surfaces.

Acknowledgements

The authors are deeply grateful to Kozo Fujii from Japan Institute of Space and Astronautical Science for providing the test datasets. Special thanks to Issei Fujishiro from Ochanomizu University for his helpful comments. This work was partially supported by Telecommunications Advancement Organization of JAPAN, the National Science Fundation grant MIP9527694 and Office of Naval Research Grant Nooo149710402.

References

[1] F. Aurenhammer (1991). "Voronoi Diagrams - A Survey of a Fundamental Geometric Data Structure." *ACM Computing Surveys*, Vol. 23, No. 3, pp. 345-405.

[2] B. Cabral and C. Leedom (1993). "Imaging Vector Field Using Line Integral Convolution." *Proceedings of SIGGRAPH'93*, pp. 263-270.

[3] L.K. Forssell and S.D. Cohen (1995). "Using Line Integral Convolution for Flow Visualization: Curvilinear grids, Variable-speed Animation and Unsteady Flows" *IEEE Transaction on Visualization and Computer Graphics*, Vol.1, No.2, pp. 133-141.

[4] M. H. Kiu and D. C. Banks (1996). "Multi-Frequency Noise for LIC." *Proceedings of Visualization'96*, pp. 121-126.

[5] W.C. Leeuw and J. J. van Wijk (1995). "Enhanced Spot Noise for Vector Field Visualization." *Proceedings of Visualization'95*, PP. 233-239.

[6] X. Mao, L. Hong and A. Kaufman (1995). "Splatting of Curvilinear Volumes." *Proceedings of Visualization'95*, pp. 61-68.

[7] J.J. van Wijk (1991). "Spot Noise: Texture Synthesis for Data Visualization". *Computer Graphics*, Vol. 24, No.4, pp. 309-318.

[8] M. Yamauchi, K. Fujii and F. Higashino (1994). "Numerical Investigation of Supersonic Flows around a Spiked Blunt-Body" *Journal of Spacecraft and Rockets*, Vol. 32, No.1, pp.32-42.

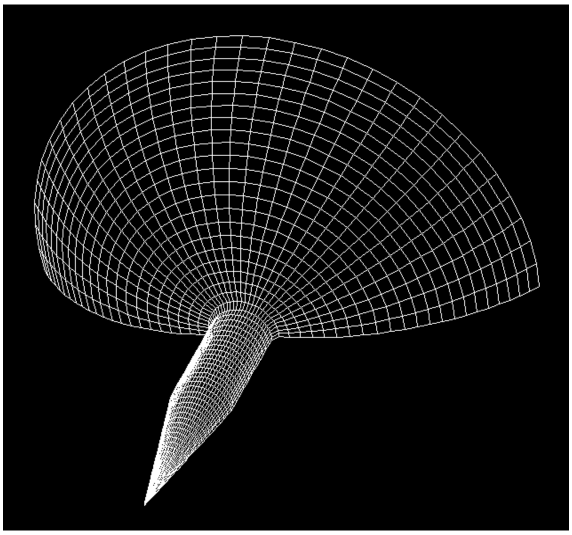

Figure 2: *Curvilinear grid over a spiked blunt-body.*

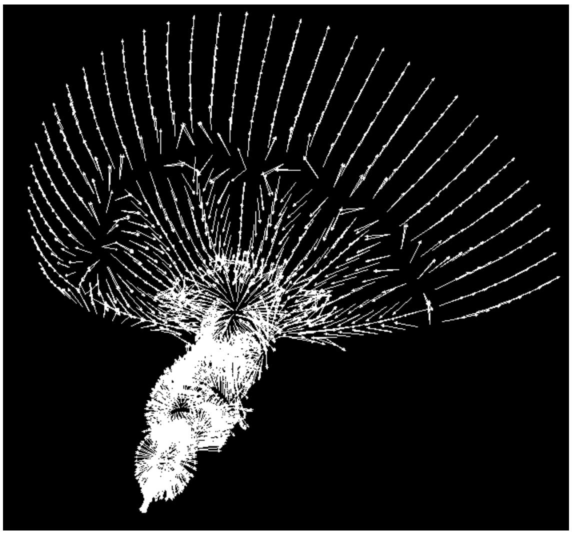

Figure 3: *Velocity vectors in the physical space.*

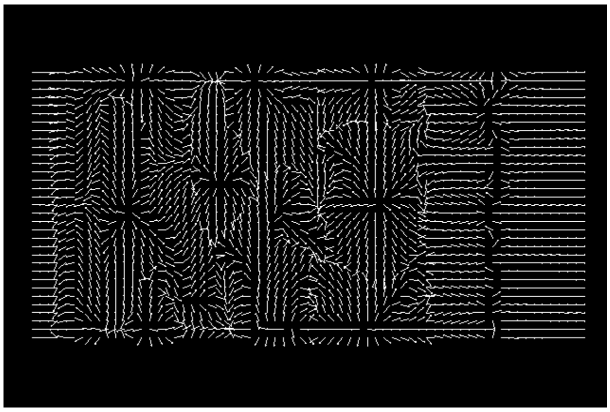

Figure 4: *Velocity vectors in the computational space.*

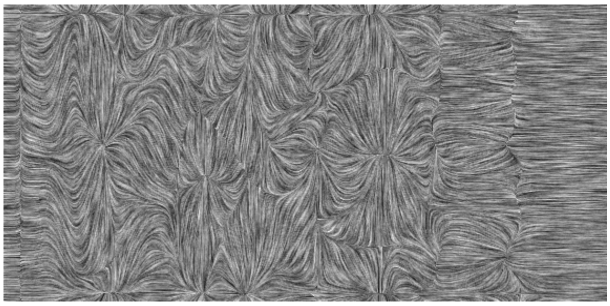

Figure 5: *LIC image calculated with a white noise image and the 2D vector field in Figure 4.*

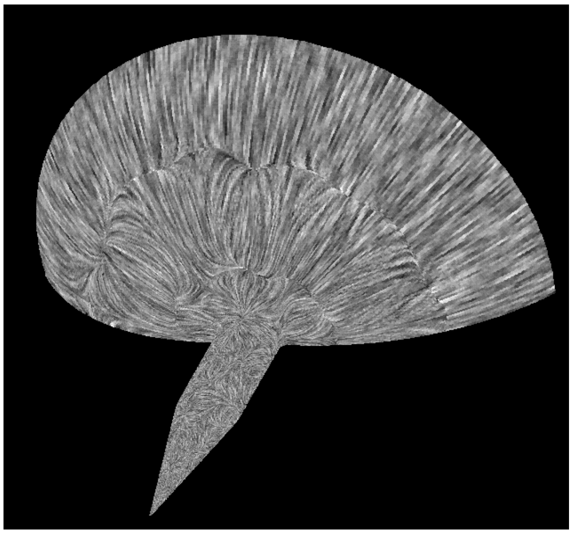

Figure 6: *Texture mapping of the LIC image in Figure 5.*

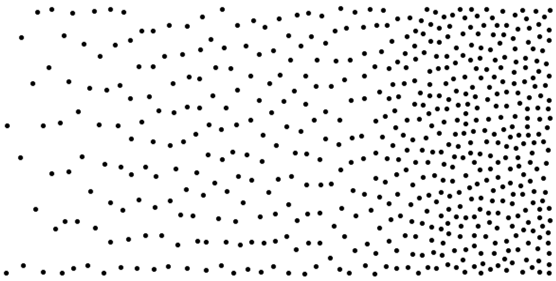

Figure 7: *Resampling of the computational space of the spiked blunt-body grid.*

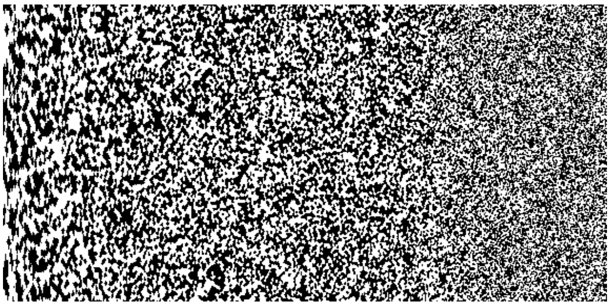

Figure 8: *Multi-granularity noise reconstructed from the sample points shown in Figure 7.*

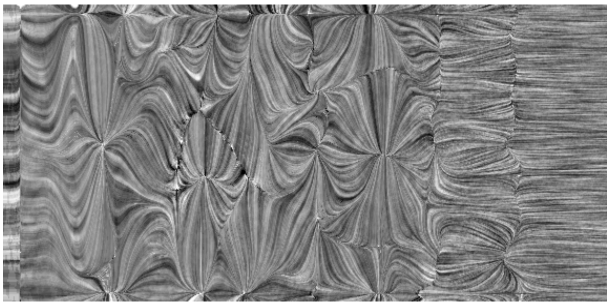

Figure 9: *LIC image with the multi-granularity noise in Figure 8.*

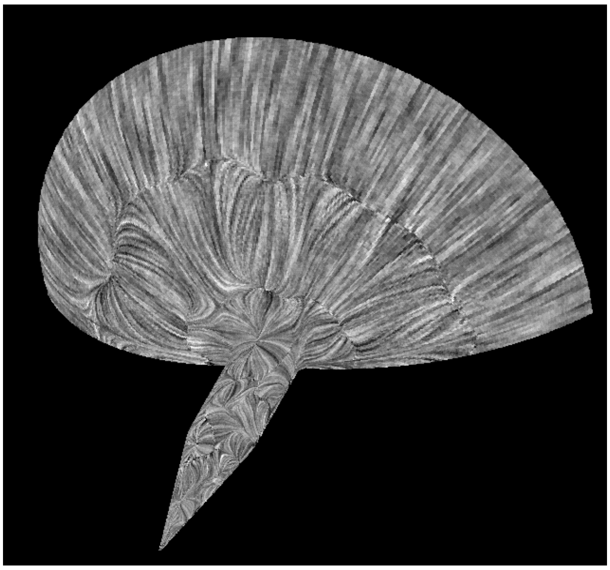

Figure 10: *Texture mapping of the LIC image in Figure 9.*

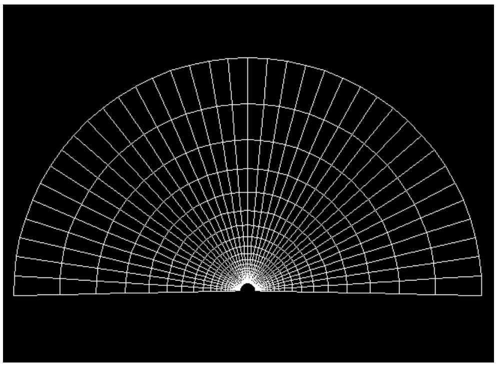

Figure 11: *Curvilinear grid structure of Post dataset.*

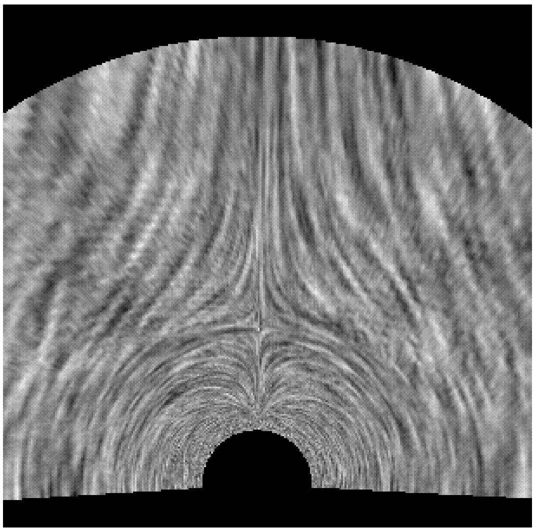

Figure 12: *Texture mapped LIC image with constant granularity noise.*

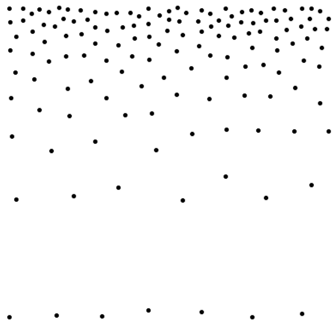

Figure 13: *Resampling of the computational space.*

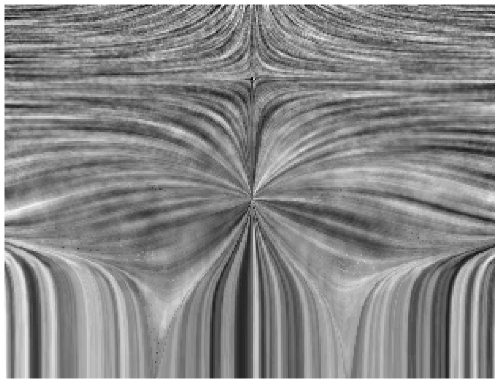

Figure 14: *LIC image with the multi-granularity noise.*

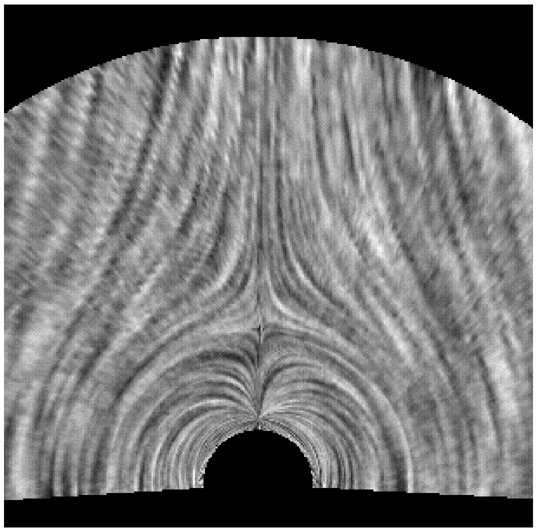

Figure 15: *Texture mapping of the LIC image in Figure 14.*