Hierarchical Watershed Ridges for Visualizing Lagrangian Coherent Structures

Mingcheng Chen, John C. Hart and Shawn C. Shadden

Abstract Lagrangian coherent structures provide insight into unsteady fluid flow, but their construction has posed many challenges. These structures can be characterized as ridges of a field, but their local definition utilizes an ambiguous eigenvector direction which can point in one of two directions, and its disambiguation can lead to noise and other problems. We overcome these issues with an application of a global ridge definition, applied using the hierarchical watershed transformation. We show results on a mathematical flow model and a simulated vascular flow dataset indicating the watershed method produces less noisy Lagrangian coherent structures.

1 Introduction

The successful visualization of a large complex scientific dataset often relies on the ability to emphasize structure hidden within it. This is particularly true of flow datasets that in their most basic form contain a velocity vector at each point in space, essentially doubling the dimensionality of the dataset, which confounds an observer's ability to perceive the data as a whole. Moreover, in unsteady flow applications, instantaneous rate of change information becomes less directly relevant to visualize, as the more salient flow information is contained by *Lagrangian* measures that intrinsically incorporate the integrated flow behavior.

A variety of analysis techniques can simplify a flow dataset by recognizing and displaying structures representing similar flow characteristics. The recent and com-

Mingcheng Chen
University of Illinois at Urbana-Champaign e-mail: mchen50@illinois.edu

John C. Hart
University of Illinois at Urbana-Champaign e-mail: jch@illinois.edu

Shawn C. Shadden
University of California, Berkeley e-mail: shadden@berkeley.edu

pelling method of Lagrangian coherent structures [19] reveals the boundaries of regions of shared characteristics for unsteady fluid flow.

Lagrangian coherent structures can be defined in a variety of different ways (e.g. [5, 20, 6]), but commonly as the ridges of a scalar field of the finite time Lyapunov exponent (FTLE) indicating the divergence of neighboring pathlines in a time-varying flow. Ridges are features typically derived from the second derivatives of the field, and so for common datasets are susceptible to noise and other issues.

A current commonly used approach to extract ridges from datasets are local,z based on a marching cubes fit of the local ridge configuration in a cell from the FTLE data at the vertices. The local definition of a ridge is based on a matrix eigenvector, which only indicates the orientation of a line, but the ambiguity created by the fact that $\mathbf{e}$ and $-\mathbf{e}$ are both equally valid eigenvectors can lead to an orientation ambiguity when detecting the ridge surface. This ambiguity can manifest as spurious false positives and other noise in the ridge surface extracted by local methods such as marching ridges often used for LCS extraction [16].

Watershed methods provide a global approach to extract topological structures from datasets. Sahner et al. [15] describe both a non-global "continuous" watershed approach that traces ridges as separatrices in the Morse structure, as well as a global "discrete" watershed transformation, and use the global watershed transformation to extract vortex and strain skeletal surfaces. We similarly propose and demonstrate watershed separatrix surface extraction for the visualization of flow structure, but for LCS instead of vortex/strain skeletal surfaces, and using a hierarchical watershed to filter out spurious details, to more clearly define the boundaries between neighboring watersheds as a global sea level rises.

These global watersheds can miss some spurious ridge features and can also produce small disjoint ridges due to noise and small field undulation. As stated in Sahner et al. [15], every watershed boundary corresponds to a height ridge or valley, but they do not necessarily coincide and furthermore ridges and valleys might exist that lack corresponding watershed boundaries.

This paper specializes the watershed approach for extracting ridges in FTLE data. We apply a region merging criteria similar to topological persistence that ranks ridges based on their configuration relative to neighboring ridges and valleys. This new filtering enabled by a global approach yields improved LCS extraction and better visualization of unsteady flow structure.

2 Lagrangian Coherent Structures

Let $\mathbf{v}(\mathbf{x},t)$ represent a time-varying velocity function. We denote the flow map $\Phi(\mathbf{x},t_0,T)$, which takes $\mathbf{x}$ to its new position at time t_0+T by integrating the velocity to trace the point along its trajectory

$$\Phi(\mathbf{x},t_0,T) = \mathbf{x} + \int_0^T \mathbf{v}(\Phi(\mathbf{x},t_0,t),t)dt. \tag{1}$$

The Cauchy-Green strain tensor is the positive definite matrix

$$\mathbf{C}(\mathbf{x},t_0,T) = \left[\frac{\partial \Phi(\mathbf{x},t_0,T)}{\partial \mathbf{x}}\right]^T \left[\frac{\partial \Phi(\mathbf{x},t_0,T)}{\partial \mathbf{x}}\right], \tag{2}$$

and measures finite-time strain of infinitesimal line elements in the fluid. The maximum separation rate is achieved when $\Delta\mathbf{x}$ is parallel to the major eigenvector of $\mathbf{C}(\mathbf{x},t_0,T)$. The finite-time Lyapunov exponent (FTLE) measures this maximum separation rate as

$$f(\mathbf{x}) = \frac{\ln \lambda_{max}(\mathbf{C}(\mathbf{x},t_0,T))}{2|T|}, \tag{3}$$

for points $\mathbf{x}$ at a given time t_0 over a given time interval T. Lagrangian coherent structures are often obtained as ridges of FTLE, but their specific definition relies on the particular definition of "ridge" that is used.

The height ridges of a scalar field f are defined as the points satisfying

$$\frac{\partial f}{\partial \mathbf{e}_1} = 0 \tag{4}$$

$$\frac{\partial^2 f}{\partial \mathbf{e}_1^2} < 0 \tag{5}$$

where f is a scalar function and $\mathbf{e}_1$ is either the minimum [2] or largest magnitude [9] eigenvector of the Hessian of f. The C-ridges of a scalar field f are defined similarly, except $\mathbf{e}_1$ is the major eigenvector of the Cauchy-Green tensor (2) [17] based on "normally hyperbolic" LCS [6]. In this work we do not target hyperbolic LCS per se, but the more generic FTLE ridge.

Lagrangian coherent structures can be revealed by a continuation method that tracks the surface from one or more seed points placed at FTLE local maxima [17]. The surface grows from these seed points by integrating a tangent plane orthogonal to the major eigenvector of the Cauchy-Green tensor. While the algorithm is shown to be quite efficient, it required at least one seed point on every LCS component, and multiple seeds on the same component could lead to duplicated surfaces.

LCS can also be revealed through a marching ridges technique [16]. Marching ridges [4] is a variant of the marching cubes isosurface technique [10] used when the orientation needed to define the isosurface is inconsistently specified. An eigenvector $\mathbf{e}$ represents an axis, without preference of $\mathbf{e}$ or $-\mathbf{e}$. Ridge surfaces formulated from eigenvectors often must choose one of these two directions $\mathbf{e}$ or $-\mathbf{e}$ for each eigenvector $\mathbf{e}$. Marching ridges strives to consistently choose eigenvector directions to define an orientable isosurface, but can fail especially when sorted eigenvectors change their order across a single cell.

LCS can also be extracted as a subset of raw features [11] satisfying

$$\det\left(\mathbf{H}^0\mathbf{g}|...|\mathbf{H}^{n-1}\mathbf{g}\right) = 0. \tag{6}$$

The matrix $\mathbf{H}$ is the Hessian of the scalar function f, but can also be interchanged with the Cauchy-Green tensor. Since (6) does not rely on eigenvectors, raw features can be extracted as an ordinary isosurface, e.g. using marching cubes, except where they may contain non-manifold self intersections. These self intersections can confound the use of raw features to find LCS, as can the numerical instability of (6).

All of these approaches rely on a local definition of ridges and LCS, which makes them susceptible to noise and other algorithm specific issues, such as surface duplication or non-orientability. A global approach would overcome these issues by defining ridges and LCS as region boundaries by growing the regions they bound instead of tracking the boundaries between regions.

3 Watershed Segmentation

In image processing, "watershed" methods have long been used for image segmentation [13]. These techniques outline the objects depicted in an image by finding ridges in the image pixel values. For LCS extraction, such global watershed methods better reduce the false positives and non-orientability of previous local approaches.

A variety of methods can be applied to a scalar field to separate it into watershed[1] ridges and regions. A region can be defined as the points that flow to the same local minimum but this can be inefficient to compute.

It is more efficient to increment a sea-level threshold value from the global minimum value to the global maximum value. When this theshold value passes a local minimum, it creates a region that grows as the threshold increases. Neighboring regions grow into each other, identifying ridges where they meet.

The Vincent-Soille (V-S) algorithm [21] runs in linear time (proportional to the number of datapoints), and classifies all of the datapoints in an dataset as either ridge or region, labeling non-ridge datapoints by the region to which they belong. The V-S algorihm first bucket sorts the datapoints, then floods each bucket of datapoints in order from least to greatest. Ridges form when a datapoint in the current bucket has neighbors belonging to two regions, but the points in the bucket often form thick regions. Hence a (linear) distance transform is applied to the buckets to compute the distance from the nearest previously defined region (using a circular queue). In the computation of this distance transform, datapoints are assigned to their closest region. If datapoints is not connected to a closest region, then it forms a new region. If a datapoint is equidistant to multiple regions, then it is classified as a ridge.

When applied to image segmentation, the watershed method typically oversegments, yielding many small regions. When applied to LCS this leads to a distracting number of insignificant ridges due to noise and subtle variation in the FTLE field, as

[1] The term watershed comes from hydrology, where it denotes a drainage basin region. Some texts that apply it to dataset analysis incorrectly use it to refer to the ridges separating these basins, and call the basins "catchment basins." To avoid confusion, we will refer to *ridges* that separate *regions.*

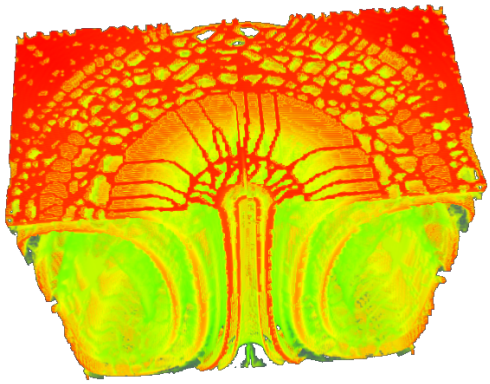


Fig. 1 Oversegmentation of the LCS of a simple convection cell flow due to variation and noise in the FTLE field.

shown in Figure 1. Hierarchical watershed methods merge similar regions to form progressively coarser segmentations and have been useful for discerning the important features in a dataset.

One method for constructing a watershed hierarchy is the waterfall transformation [1]. It constructs a graph consisting of nodes representing each ridge segment. The node's value is set to the difference between the median values of the two regions its corresponding ridge separates. Each pair of these nodes is connected with an edge if their corresponding ridge segments border the same region. Then the next level higher in this hierarchical watershed is the watershed of these ridge nodes, using the average ridge value for each node.

Alternatively, regions can be merged based on similarities in their level and/or the characteristics of the ridge separating them. To provide better control for the application of LCS extraction, we utilized this region merging approach to filter unnecessary FTLE ridges.

Our criteria to define a criterion for merging neighboring regions resembles the notion of topological persistence [3]. The persistence of a topological feature indicates how robust it is to perturbation. A well-chosen perturbation in the dataset could remove a ridge, merging the regions it separated into a single region. From the Morse theory viewpoint, this perturbation would merge a saddlepoint with the minimum of one of the regions. The persistence of a ridge is thus the difference between the lowest point on the ridge (its saddlepoint) and the larger of its two neighboring minima.

We set a persistence threshold and merge regions separated by ridges that do not meet this threshold. We accelerate this merging with a union-find data structure.

This approach is similar to scale-space hierarchical methods that smooth datasets before performing the watershed transform, using e.g. Gaussian smoothing [8]. Such techniques smooth the data with increasing filter widths to produce coarser levels of the watershed hierarchy. These smoothing operations merge neighboring regions because they cancel saddlepoint-minima pairs.

4 Polygonization

The implicit function theorem shows that the isosurface of a regular isovalue of an analytic field function is a manifold. However, the ridges arising from the FTLE are not necessarily so, and can include non-manifold junctions that require special methods for surface extraction [7, 12]. The "crease surfaces" analysis [18] for example shows that ridge surfaces consist of manifold patches that meet at non-manifold junctions where the Hessian is degenerate.

We utilize a variation of marching cubes for polygonization of the ridge surfaces. The watershed transform labels each voxel value with a region, and ridges arise in cells whose eight corner vertices (where the voxel values are evauated) lie in two or more disjoint regions. If a cell's vertices lie in only two regions, we use ordinary marching cubes to polygonize the cell.

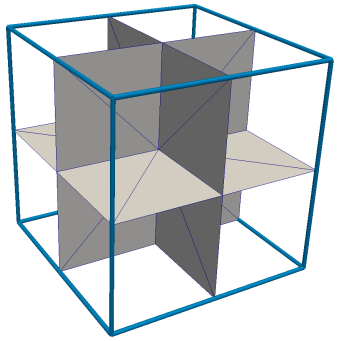


Fig. 2 Polygonization of a cell whose eight corners lie in eight different regions.

For cells that straddle three or more regions, we implement a variation of multiple material marching cubes [22]. For each of the six cell faces, we add a face center vertex and insert a pair of triangles to separate any edges whose vertices lie in separate regions. We then add a vertex at the cell center to connect these triangles. Figure 2 demonstrates the case where all eight cell corners belong to different regions. There are two cases where a vertex at the voxel face center is not needed, as shown for the front face of each example in Figure 3.

Since the cell corners indicate only the region, and not a scalar value, the vertices used to polygonize a cell are inserted at the center of edges, faces and the cell. This leads to a blocky cuberille appearance of the resulting surface as shown in Figure 4. We remove these distracting visual artifacts through a smoothing process. We implemented a constrained Laplacian smoothing through conjugate gradient minimization of the energy functional

$$E(\{\mathbf{x}_i\}) = \sum_i ||\mathbf{x}_i - \overline{\mathbf{x}}_i||^2 + \lambda ||\mathbf{x}_i - \mathbf{x}'_i||^2 \tag{7}$$

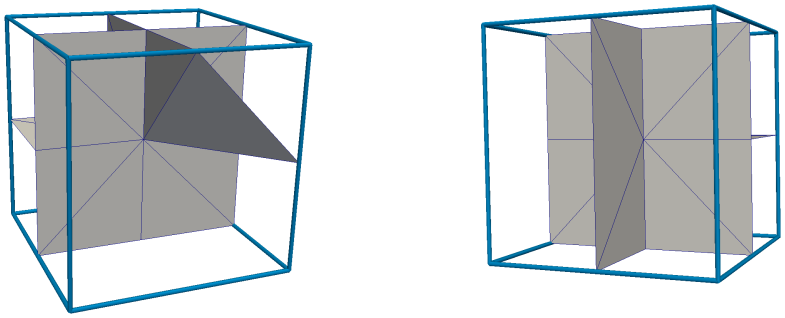


Fig. 3 Two cases where a vertex is not needed at the center of a cell face, shown for the frontmost face.

where $\bar{\mathbf{x}}_i$ is the centroid of the vertices neighboring vertex $\mathbf{x}_i$ and $\mathbf{x}'_i$ is its original position in the cuberille polygonization. The parameter λ indicates how much the original position is respected, which we set to 0.01 in our experiments.

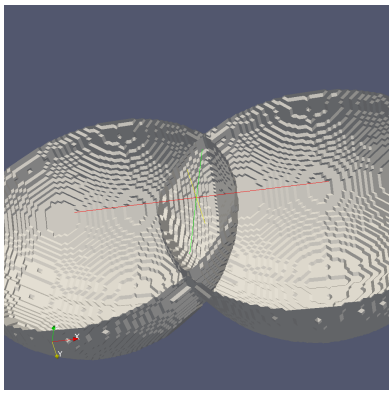


Fig. 4 Blocky artifacts created from multiple region marching cubes for voxels that only indicate region number.

The non-manifold surfaces that arise from multiple region marching cubes require special care for proper smoothing. Branch points whose neighbors may represent ridges between several different regions can confound the smoothing process, as shown in Figure 5 (left).

For each vertex, we find the maximum number of faces that share one of its edges. We limit that vertices neighbors to the ones whose edge is shared by that maximum number of faces. This process smooths non-manifold junctions well, as shown in the example of two intersecting spheres shown in Figure 5 (right).

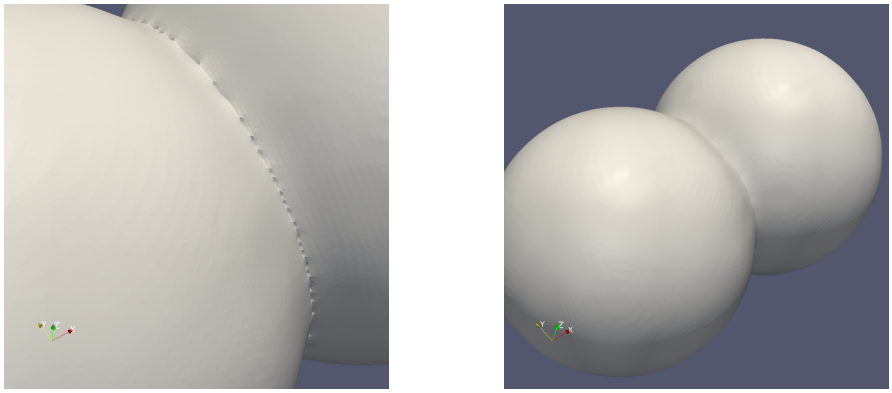


Fig. 5 Ordinary Laplacian smoothing of non-manifold surfaces creates unsmooth results (left) which are fixed by limiting the neighborhoods used for Laplacian averaging (right).

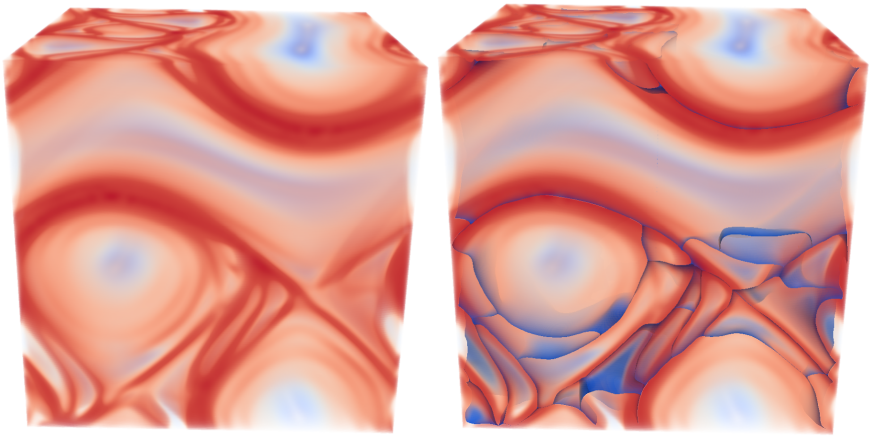
This Laplacian smoothing approach differs from the one used for multiple material marching cubes (M3C) [22]. Our approach smooths vertices even when they are shared by more than two surfaces, whereas M3C smoothing leaves such vertices stationary. Our approach also does not require additional information that M3C uses, such as which materials are adjacent to a given vertex.

5 Results

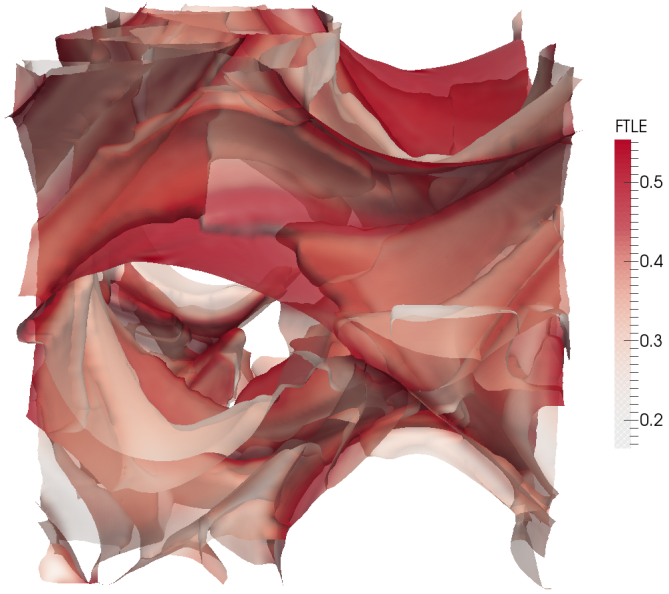
We compared the watershed approach to marching ridges on two datasets. The first is an Arnold-Beltrami-Childress (ABC) flow, shown in Figure 6. The ABC flow dataset yields an FTLE field over a 201^3 voxel array with values ranging from 0.0643 to 0.512.

Figure 7 compares the Lagrangian coherent structures extracted from the FTLE field of the ABC Flow dataset. The marching ridges example follows the recommended noise filtering steps [14], including (1) scalar thresholding (remove ridges with FTLE less than 0.3), (2) least eigenvalue thresholding (remove ”flat” ridges with eigenvalue greater than -1.0), and (3) a threshold on the size of connected components (removing disjoint components with less than 500K vertices). The displayed denoised marching ridge result consists of one connected component of 677K vertices, but even with filtering, some noise persists.

The V-S watershed approach yields 797 watershed regions at the lowest level of the watershed hierarchy, which we merge by removing low persistence ridges to 103 regions. The resulting mesh, after smoothing, consists of 1.275M faces and 622K vertices. Figure 7 also shows some smoothed stairstep artifacts that reveal some issues with the merging of watershed regions as discussed further at the end of the section.

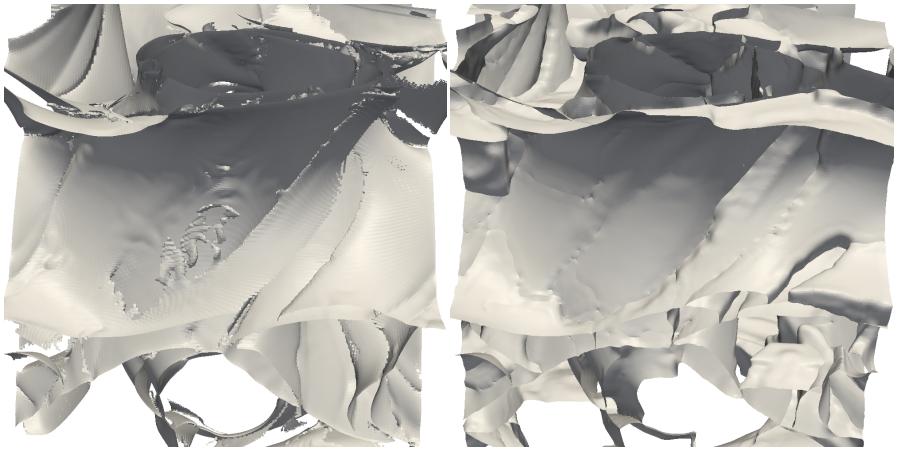


the FTLE Field with Smoothed Watershed Ridges



Smoothed Watershed Ridges with FTLE color and opacity

Fig. 6 The ABC flow, displayed as the FTLE field data (upper left) along with the embedded ridges extracted by the watershed method (upper right) and the watershed ridges themselves (lower center).



Marching Ridges Watershed

Fig. 7 Lagrangian coherent structures extracted from FTLE of the ABC Flow dataset using the marching ridges method v. the watershed method.

The second dataset we used to compare the watershed approach to marching ridges is the abdominal aortic aneurysm (AAA) dataset, shown as an FTLE field in Fig. 8. The AAA dataset is constructed from a pulsatile bloodflow simulation of a lower aorta, reconstructed as a 4.4M tetrahedral mesh. The FTLE field is a $206 \times 231 \times 261$ voxel array, ranging from 0 to 5.29812, using the value -1 indicates the outside of the aorta.

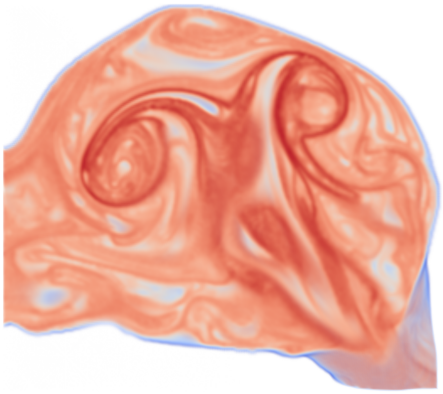
Figure 8 compares the Lagrangian coherent structures extracted from the FTLE field of the AAA dataset. The marching ridges example filtered out ridges smaller than 3.0 (which was the highest setting that prevented holes from forming in the main connected structures), set an eigenvalue threshold of zero (negative values did not improve the result) and filtered out all but the largest connected component. This yielded a mesh of 2.27M faces and 1.29M vertices. As before, the structure is evident but noise is clearly visible.

The V-S watershed algorithm yields 663 regions, which we merge into 199 regions when the height between neighboring regions differs by 0.05 or less. The resulting smoothed mesh consists of 1.80M faces and 854K faces.

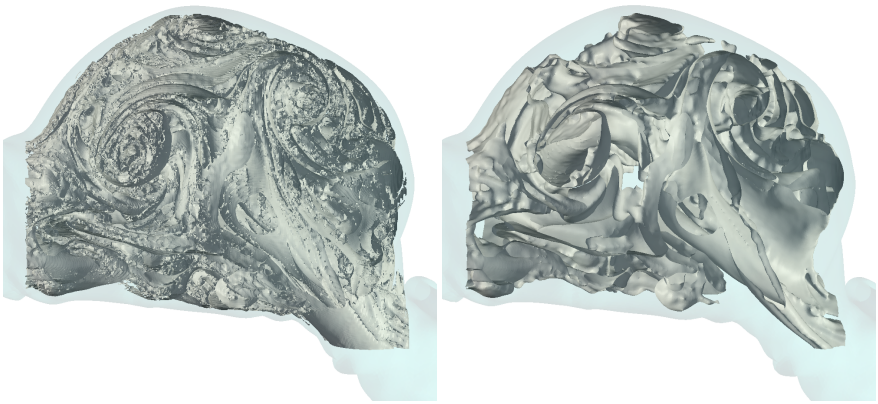
Dataset	**FTLE Resolution**	**Watershed**	**Mesh Vertices**	**Smoothing**
ABC Flow	$201 \times 201 \times 201$	34.21 s	1,225,558	36.4 s
Paitent 96	$206 \times 231 \times 261$	50.87 s	1,800,230	61.4 s

Table 1 Performance of the watershed method for LCS extraction.

The time required for the watershed transform, polygonization and smoothing for the ABC flow and the AAA data is shown in Table 1. The watershed method produces a voxel region classification on the FTLE field. The AAA FTLE field is



AAA Lower Aorta FTLE Field

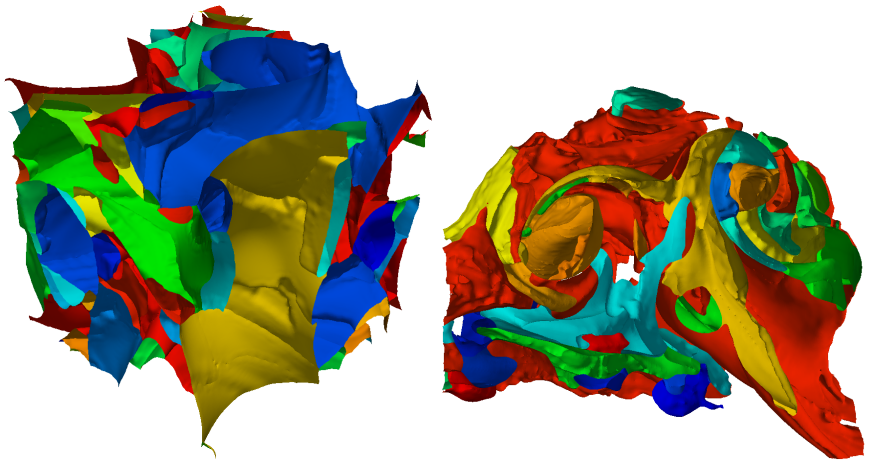


Marching Ridges Watershed

Fig. 8 Lagrangian coherent structures extracted from FTLE (top) of the AAA dataset using the marching ridges method (lower left) v. the watershed method (lower right).

53% larger than that of the ABC flow, and its watershed transform takes 49% longer to compute. These watershed voxel regions polygonized to produce the number of vertices listed which follow similar proportions, but the time of the polygonization was not a significant portion of the total time and so is not listed. The smoothing time represents a total of 20 smoothing iterations, but takes 68% longer for the larger AAA data, because the polygonized ridges are more complex and their vertices have a greater number of neighbors.

One of the most exciting aspects of the global watershed approach is that the regions can be used to coherently color the ridge surfaces, as shown in Figure 9. The choice of color can be arbitrary, but is useful to differentiate the LCS surfaces from each other as they undulate through the flow domain. Such colorings are enabled by two-sided surface shading, but are unavailable for local ridge definitions (e.g. marching ridges) that lack identification of these regions.



ABC Flow AAA

Fig. 9 Lagrangian coherent structures for the ABC flow and Patent 96, displayed using random colors assigned by regions, as extracted using our hierarchical watershed approach.

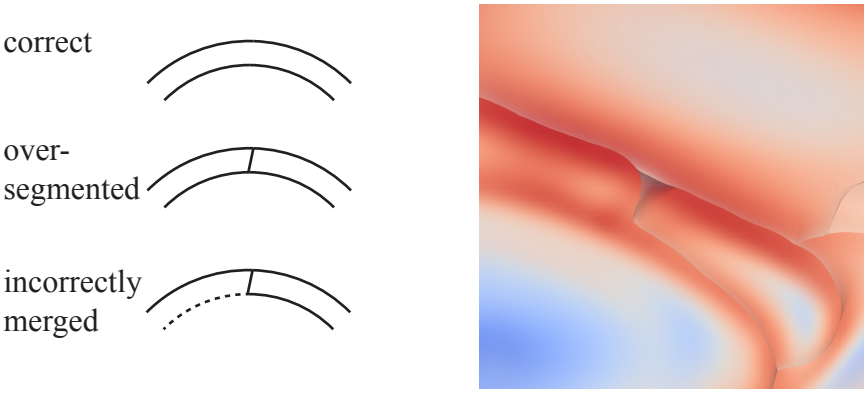


Fig. 10 Hierarchical watershed merging can sometimes merge the wrong regions.

The main drawback of the watershed approach is that the initial application of the watershed transformation (before merging) creates significant over-segmentation of the datasets, resulting in many small watersheds. These small watersheds are merged when separated by shallow ridges, but sometimes the shallowness, which we use as the persistence of the ridge, not properly eliminate some shallow ridges even though this persistence based approach works well for most spurious ridges such as are shown in Fig. 1.

As shown in Fig. 10, the ridges detected by the hierarchical watershed approach are largely at the mercy of the fluctuations of the FTLE field. In this example, such an FTLE fluctuation causes the hierarchical watershed merging to follow the wrong shorter ridge instead of keeping the correct longer ridge. This wrong shorter ridge

is indeed part of the FTLE field and in fact forms a better defined ridge according to persistence than does the correct ridge, so further work beyond the persistence measure used by the hierarchical watershed method is needed to eliminate these last few pathological cases.

6 Conclusions and Further Research

While watershed methods commonly appear in the summaries of ridge exraction methods for Lagrangian coherent structures, they are often dismissed in favor of local methods such as marching ridges. We have shown that their results are often much smoother and less noisy than such local approaches, and should be considered further.

One of the main shortcomings of the watershed approach is that it does not detect ridges that end at a minimum. Such an example might be a ridge descending from the rim to the bottom of a crater. We plan to address such cases with a combination of local and global combinations, using the local ridge definition evaluated on the current sea-level coastline front of the V-S watershed method. This combination of local and global ridge methods could yield the best of both worlds.

We have used a persistence measure as the criterion for hierarchical watershed method to merge watershed regions based on ridge shallowness. This criterion works well in many but not all cases, as illustrated in Figure 10. Further analysis and experimentation will be needed to explore new merging criteria to better preserve the important ridges for LCS visualization.

We also plan to work on high performance streaming implementations of the V-S and other watershed algorithms, updating earlier such work [13], as we as their applications to larger, out-of-core datasets.

Acknowledgements This work was supported in part by NSF Grants OCI-1047963 and OCI-1047764. We thank Siavash Ameli for discussions on LCS definitions.

References

1. Beucher, S.: Watershed, hierarchical segmentation and waterfall algorithm. In: Mathematical Morphology and Its Applications to Image Processing, *Computational Imaging and Vision*, vol. 2, pp. 69–76. Springer (1994)
2. Eberly, D.: Ridges in Image and Data Analysis. Kluwer Academic Publishers, Dordrecht (1996)
3. Edelsbrunner, H., Harer, J.: Persistent homology — a survey. In: Surveys on Discrete and Computational Geometry, *Contemporary Mathematics*, vol. 453, pp. 257–282. AMS (2006)
4. Furst, J.D., Pizer, S.M.: Marching ridges. In: SIP, pp. 22–26 (2001)
5. Haller, G.: Distinguished material surfaces and coherent structures in three-dimensional fluid flows. Physica D: Nonlinear Phenomena **149**(4), 248–277 (2001)

6. Haller, G.: A variational theory of hyperbolic Lagrangian coherent structures. Physica D **240**(7), 574–598 (2011)
7. Hege, H.C., STALLING, D., SEEBASS, M., Zockler, M.: A generalized marching cubes algorithm based on non-binary. Tech. Rep. SC-97-05, Konrad-Zuse-Zentrum (ZIB) (1997)
8. Hucko, M., Sramek, M.: Interactive segmentation of volume data using watershed hierarchies. J. of Winter School for Computer Graphics **20**(3), 217–222 (2012)
9. Lindeberg, T.: Feature detection with automatic scale selection. International journal of computer vision **30**(2), 79–116 (1998)
10. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. Proc. SIGGRAPH, Computer Graphics **21**(4), 163–169 (1987)
11. Peikert, R., Sadlo, F.: Height ridge computation and filtering for visualization. In: Visualization Symposium, 2008. PacificVIS'08. IEEE Pacific, pp. 119–126. IEEE (2008)
12. Reitinger, B., Bornik, A., Beichel, R.: Consistent mesh generation for non-binary medical datasets. In: Bildverarbeitung für die Medizin 2005, pp. 183–187. Springer (2005)
13. Roerdink, J.B., Meijster, A.: The watershed transform: Definitions, algorithms and parallelization strategies. Fundamenta Informaticae **41**(1), 187–228 (2000)
14. Sadlo, F., Peikert, R.: Efficient visualization of lagrangian coherent structures by filtered amr ridge extraction. IEEE TVCG **13**(6), 1456–1463 (2007)
15. Sahner, J., Weinkauf, T., Teuber, N., Hege, H.C.: Vortex and strain skeletons in eulerian and lagrangian frames. Visualization and Computer Graphics, IEEE Transactions on **13**(5), 980–990 (2007)
16. Schindler, B., Fuchs, R., Barp, S., Waser, J., Pobitzer, A., Carnecky, R., Matkovic, K., Peikert, R.: Lagrangian coherent structures for design analysis of revolving doors. TVCG **18**(12), 2159–2168 (2012)
17. Schindler, B., Peikert, R., Fuchs, R., Theisel, H.: Ridge concepts for the visualization of lagrangian coherent structures. In: Topological Methods in Data Analysis and Visualization II, pp. 221–235. Springer (2012)
18. Schultz, T., Theisel, H., Seidel, H.P.: Crease surfaces: From theory to extraction and application to diffusion tensor mri. Visualization and Computer Graphics, IEEE Transactions on **16**(1), 109–119 (2010)
19. Shadden, S.C.: Lagrangian coherent structures. In: R. Grigoriev (ed.) Transport and Mixing in Laminar Flows: From Microfluidics to Oceanic Currents, chap. 3, pp. 59–89. Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim, Germany (2012)
20. Shadden, S.C., Lekien, F., Marsden, J.E.: Definition and properties of Lagrangian coherent structures from finite-time Lyapunov exponents in two-dimensional aperiodic flows. Physica D: Nonlinear Phenomena **212**(3-4), 271–304 (2005)
21. Vincent, L., Soille, P.: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE transactions on pattern analysis and machine intelligence **13**(6), 583–598 (1991)
22. Wu, Z., Sullivan, J.: Multiple material Marching Cubes algorithm. International Journal of Numerical Methods in Engineering **58**(2), 189–207 (2003)