Exploring Vector Fields with Distribution-based Streamline Analysis

Kewei Lu[*] The Ohio State University
Abon Chaudhuri[†] The Ohio State University
Teng-Yok Lee[‡] The Ohio State University
Alexander G. Suttmiller[§] The Ohio State University
Han-Wei Shen[¶] The Ohio State University
Pak Chung Wong[‖] Pacific Northwest National Laboratory

ABSTRACT

Streamline-based techniques are designed based on the idea that properties of streamlines are indicative of features in the underlying field. In this paper, we show that statistical distributions of measurements along the trajectory of a streamline can be used as a robust and effective descriptor to measure the similarity between streamlines. With the distribution-based approach, we present a framework for interactive exploration of 3D vector fields with streamline query and clustering. We demonstrate the utility of our framework with simulation data sets of varying nature and size.

1 INTRODUCTION

Visualization and exploration of vector fields plays an important role in understanding the data generated from simulations in many science and engineering disciplines. Among the numerous methods in practice, streamline based techniques continue to be one of the most popular approaches. However, when dealing with very large vector fields, a direct visualization of all the densely computed streamlines is seldom useful due to visual clutter and occlusion. Moreover, the user is often interested in looking at some specific flow features based on the goal of exploration. Hence, it is important to be able to classify the streamlines based on features, and display only those ones that are relevant to the user's objective.

We propose a novel idea that uses the distribution of feature measures over a streamline to be the descriptor, and uses this descriptor to measure the similarity between streamlines. The distribution-based streamline distance metric has a major property compared with some existing point location-based distance metrics such as closest point distance [1] and Hausdorff distance [2]: the distributions of certain geometric properties on a streamline such as curvature are invariant to translation and rotation. Also, compared with point location-based metrics, our method has lower time complexity, because point location-based methods need to iterate through all the points on the two streamlines, the time complexity of our method depends on the number of bins of histograms instead.

2 PROPOSED METHOD

The main idea of the proposed method is illustrated in Figure 1. First, we seeded the field densely and generated a large number of streamlines, so that no flow feature is missed. Then, Given a user-specified measure, statistical distributions can be used to describe its values and spread along the trajectory of a streamline. Assuming k measures such as $m_1, \ldots, m_k$ are selected for analysis, we compute each of them at every sample point s_p along a streamline, resulting in a k-tuple measure $\{m_1^p, m_2^p, \ldots, m_k^p\}$ for each point. We compute a distribution separately from each measure and then represent it by a histogram. Since the property of a streamline can vary from points to points, simply keeping a single 1D histogram may not be sufficient because the order of the measured values along the streamline can be essential for describing its shape and for comparing with other streamlines. For example, Figure 2(a) and 2(b) show two different streamlines color coded by curvature, one is more turbulent at the beginning and the other is turbulent at the beginning and the end. However, these two streamlines are undistinguishable by their 1D histograms since they are very similar, as shown in Figure 2(c) and 2(d).

[*]e-mail:luke@cse.ohio-state.edu
[†]e-mail:chaudhua@cse.ohio-state.edu
[‡]e-mail:leeten@cse.ohio-state.edu
[§]e-mail:suttmill@cse.ohio-state.edu
[¶]e-mail:hwshen@cse.ohio-state.edu
[‖]e-mail:pak.wong@pnnl.gov

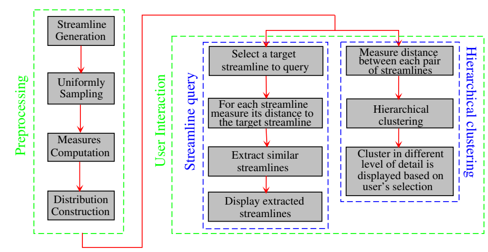
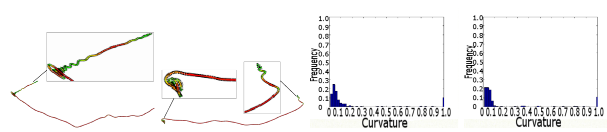


Figure 1: The major steps for our framework

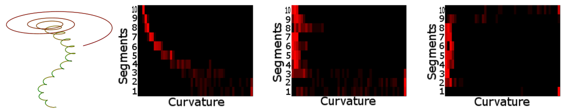


(a) (b) (c) 1D Histogram (d) 1D Histogram

Figure 2: Limitation of the 1D histogram representation.

To resolve the ambiguity, we use 2D histograms instead of the simple 1D histograms to represent a streamline. Given a streamline, first it is divided into M segments in such a way that each segment contains the same number of sample points. For each segment, we have a histogram and these M histograms are combined together to form a 2D histogram. Also we create the same number of segments for every streamline to facilitate easy segment-wise comparison between two streamlines. The goal of dividing a streamline into multiple segments is to preserve the order of feature measured along a streamline, albeit loosely, while still being able to use histograms to describe a streamline. Having more segments implies a higher storage cost but can encode richer information. On the other hand, It may produce less meaningful distributions because fewer points are collected within each segment. Observing this trade-off, we allow the user to decide the number of segments based on whether the ordering information is important in the analysis. Since the 2D histograms can be constructed fast, user can change the number of segments during exploration and get response in a few seconds.

An example of a 2D histogram constructed in this way is shown in Figure 3(b). Figure 3(a) shows the streamline. Based on the 2D histogram representation of the streamline, we can understand some underlying features about the streamline, such as the represented streamline has high curvature with high variation at the beginning, but it gradually decreases to low curvature with low variation. Since the 2D histogram incorporates the order information to some extent, it can better differentiate streamlines than simply using 1D histograms. The 2D histograms for the streamlines shown in Figure 2(a) and 2(b) are shown in Figure 3(c) and 3(d). Based on the two 2D histograms, it is easy to distinguish these two streamlines. With the histogram computed from each streamline, we can compute the streamline similarity based on the distance between the histograms. In the context of our problem, considering cross-bin relationship is important, so we use the Earth Mover's Distance [3] to compute the distance between two histograms.



(a) (b) 2D Histogram (c) 2D Histogram (d) 2D Histogram

Figure 3: Example for 2D histogram. (b) is for streamline in (a). (c),(d) are corresponding to the streamlines in Figure 2(a) and 2(b)

3 Applications

Our distribution-based similarity measurement approach can be used in many different applications, we describe two visualization applications: similar streamline query and hierarchical clustering.

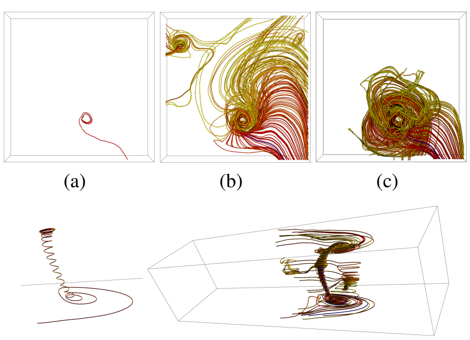
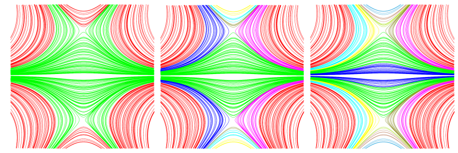


Figure 4: Query results for Isabel and Plume. (a),(d): two target streamlines. (b),(e): query results for (a) and (d) based on our method. (c): query results for (a) based on Hausdorff distance

To explore the underlying vector field, users can request to display only streamlines that have a similar shape to that of the target streamline. To support this, we can perform a similarity query where all streamlines are compared with the target streamline based on our distribution-based distance metric. Curvature and torsion are two metrics to characterize space curves. To compute the distance between two given streamlines, we first compute the distance between their curvature histograms and the distance between their torsion histograms. Then we sum these two quantities to get the final distance. Figure 4 shows the query results. Compared with our method, Hausdorff distance metric extract the streamlines which are closed to the target streamline but not necessary the most similar streamlines as shown in Figure 4(c).



(a) (b) (c)

Figure 5: Hierarchical clustering results based on our method.(a),(b) and (c) have 2, 7 and 10 clusters respectively.

Another way to explore flow fields is through hierarchical streamline clustering. We use a bottom up agglomerative clustering method in our implementation. Users can show clusters at any level to highlight the features in the flow field at different levels of detail. In Figure 5, we shows the hierarchical clustering results for a simple 2D vector field.

4 Performance

Table 1: Timings for Streamline Query

Data	# lines	Timing(in seconds)		
		Histogram Const.	Ours	d_h
Isabel	2000	0.92	0.062	314.108
Plume	2000	1.232	0.031	339.567

In Table 1, we summarize the timing of streamline query for Hurricane Isabel(Figure 4(a)) and Plume(Figure 4(d)). The average length of streamlines is around 2000. We compare the performance of our distribution-based method with the Hausdorff distance d_h. It can be seen that our method outperformed by several times in terms of speed. This is important for the users to get a rapid feedback for their queries. And for hierarchical streamline clustering, we are able to build the similarity matrix fast by using our method.

5 Conclusion And Future Work

Our framework can be extended along several directions. First, we will extend our framework to handle time-varying vector data. Secondly, besides the three geometric measures, curvature, curl and torsion, additional measures including domain-specific physical quantities can be included to explore 3D vector fields. We believe that combining different feature measures can produce more robust query and clustering results, and reveal different features in the vector field.

Acknowledgements

This work was supported in part by NSF grant IIS-1017635, US Department of Energy DOE-SC0005036, Battelle Contract No. 137365, and Department of Energy SciDAC grant DE-FC02-06ER25779, program manager Lucy Nowell.

References

[1] I. Corouge, S. Gouttard, and G. Gerig. Towards a shape model of white matter fiber bundles using diffusion tensor mri. In *Biomedical Imaging: Nano to Macro, 2004. IEEE International Symposium on*, pages 344 – 347 Vol. 1, april 2004.

[2] C. Rossl and H. Theisel. Streamline embedding for 3d vector field exploration. *Visualization and Computer Graphics, IEEE Transactions on*, 18(3):407 –420, march 2012.

[3] Y. Rubner, C. Tomasi, and L. Guibas. A metric for distributions with applications to image databases. In *Computer Vision, 1998. Sixth International Conference on*, pages 59 –66, jan 1998.