International Institute of Information Technology Bangalore

Basic Computational Topology

SM-402

Fiber Surfaces: Generalizing Isosurfaces to Bivariate Data

Hamish Carr, Zhao Geng, Julien Tierny, Amit Chattopadhyay, Aaron Knoll

Submission by:

Arjun Subhedar (IMT2021069)

Ritik Kumar Gupta (IMT2021098)

ज्ञानमुत्तमम्

Abstract

A new method for examining bivariate fields has been developed using fiber surfaces. These surfaces are constructed from sets of fibers and can be extracted using an efficient algorithm based on Marching Cubes. Geometric primitives can also be used to construct fiber surfaces interactively. This approach allows for the generation of parameterized families of fiber surfaces with respect to arbitrary polygons. By using fiber surfaces, features can be captured geometrically for quantitative analysis in bivariate fields, which were previously only analyzed visually and qualitatively using multi-dimensional transfer functions in volume rendering.

1 Background

Contours and Isosurfaces - Contours and isosurfaces can be defined as the inverse image of an isovalue in a scalar field f: $R^3 \rightarrow$ R. This separates the domain into pieces, which is useful for many datasets, where the isosurface is often a closed surface representing a boundary.

Previous research has explored various techniques related to the paper's contributions, such as the utilization of Marching Cubes for the extraction of isosurfaces and separating surfaces, approaches for visualizing multiple fields simultaneously and the use of multi-dimensional transfer functions for direct volume rendering.

Extracting Isosurfaces using Marching Cubes - The Marching Cubes algorithm divides the space into a grid of cubes with known data values at intersections, then extracts a surface in each cube based on a given isovalue in four stages -

1. Classification: Data values at each corner of the cube are compared to the isovalue and classified as "black" or "white".
2. Triangle Topology: The eight bits are converted to a single-byte integer and used to retrieve triangle topology from a look-up table.
3. Vertex Interpolation: Linear interpolation is applied to each triangle vertex along the edge to determine the exact location.
4. Normal Vectors: Normal vectors are constructed using flat normals of the faces or by estimating the gradient vector at the vertex using central differencing and interpolation.

Multifield Visualization - Multifields displayed as multidimensional histograms, mesh continuity is important, linear features related to topology, further ongoing research, but these methods are complex and not fully developed.

Direct Volume Rendering - Direct volume rendering is slow for complex data and produces only an image, not a geometric surface suitable for modeling and simulation.

2 Fiber Surfaces

In our study, we have considered functions, $f{:}R^3 \rightarrow R^2$, where each point in the 3-dimensional domain is associated with two values (h_1, h_2). A fiber is thus defined by a point $h = (h_1, h_2)$, and the fiber itself can be found by intersecting of isosurfaces of h_1 in f_1 and h_2 in f_2 (where f_1 and f_2 represent the two output variables).

The task at hand is to discover a way to create distinct separating surfaces for bivariate volumes by generalizing the concept of contours. To achieve this, we utilize fibers to construct surfaces. The continuity of fibers and contours proves to be beneficial. This property enables a continuous variation of fiber sets in the domain, resulting in the creation of a surface or multiple surfaces as they sweep through the domain.
Properties of Fiber Surfaces are similar to that of general univariate Isosurfaces.

1. Each fiber surface component is a continuous surface (single connected component of a fiber surface).

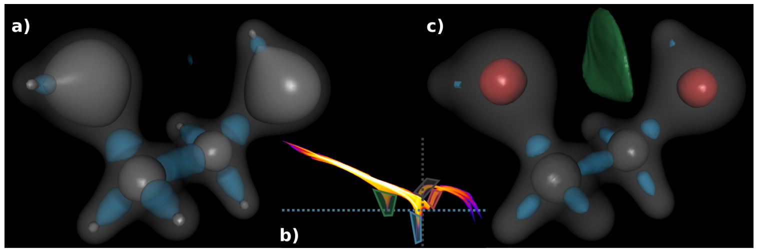
2. If the path P separates $\operatorname{Ran} f$ into regions, then the corresponding fiber surface $f^{-1}(P)$ is a separating surface in $\operatorname{Dom} f$.

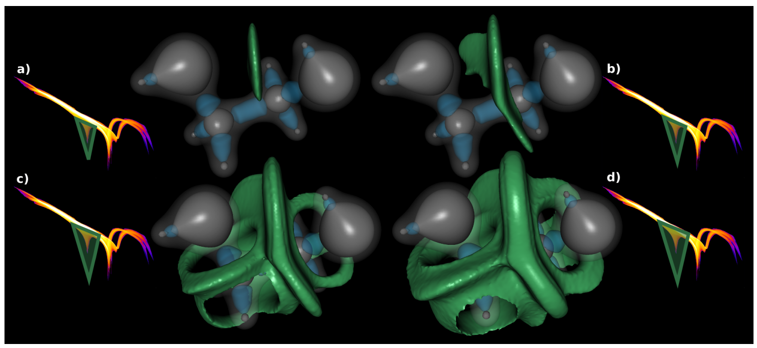
The simplest fiber surfaces are obtained by lines in range. For a line in the range of the function, we get a surface in the domain. And if this line is separating the region into two, then the corresponding fiber surface that we get is also separating.
General case of fiber surfaces can be associated with arbitrary curves instead of simple lines, which can be approximated to polygons. We assume a separating polygon: i.e. a closed loop of range line segments, which we call the fiber surface control polygon or FSCP. While traveling around the Fiber Surface Component Plot (FSCP), the continuity of the bivariate volume function f ensures that fibers may either separate into components or join together. However, the fibers themselves undergo continuous deformation into each other, resulting in a set of continuous surfaces that are swept out in the domain.
Fiber surfaces have a property similar to isosurfaces, which separates an inside from an outside. The continuity of f implies that each point on a path belongs to a fiber, which is defined by a point in the range. If a set disconnects the range of f, then it must also disconnect the domain of f. Mesh vertices whose fibers are inside the FSCP are also inside the fiber surface, and vertices whose fibers are outside the FSCP are also outside the fiber surface.

3 Algorithm

The paper's algorithm computes the fiber surface of a function f over a polygon P in its range $\operatorname{Ran}(f)$ by utilizing the Marching Cubes algorithm to generate a polygonal mesh. The threshold value for fiber surfaces is the polygon itself. The algorithm classifies each vertex in the mesh based on its function value relative to the polygon, and then computes a set of triangles that intersect the isosurface defined by the polygon. For each intersected cell edge (u, v), the algorithm computes the intersection point w of the line segment $f(u), f(v)$ and the polygon P. The parameter t on $f(u), f(v)$ for w is then found, and the vertex e is interpolated using the formula $e = u + t(v - u)$. The normal vector at e is also computed using interpolation (not done in our implementation as this step is required for shading while visualising the surface). By iterating over all intersected cell edges in the mesh and computing the corresponding vertices and normals of the approximating triangles, the algorithm generates a polygonal mesh that approximates the fiber surface.

4 Examples





This example demonstrates the use of bivariate data for extracting fiber surfaces from electron density and reduced gradient data of an ethanediol molecule. These functions help chemists identify regions of atomic influence and molecular interactions.
Other examples cited by the authors were those in the fields of cosmology(two output values were concentration of matter and concentration of dark matter) and combustion (two field values were temperature and temperature gradient magnitude).

5 Conclusion

The authors have developed a method to generalize isosurfaces to bivariate data for identifying regions corresponding to features in multi-dimensional transfer functions. The method is easy to implement and can benefit from well-known optimizations for Marching Cubes. The authors expect the method to be parallelizable and plan to explore trivariate functions in the future. They also intend to develop constructs similar to fiber surfaces, which may be more useful analytically than visually.

References

[1] https://www.sciencedirect.com/topics/computer-science/marching-cube-algorithm

[2] http://paulbourke.net/geometry/polygonise/

[3] https://en.wikipedia.org/wiki/Marching_cubes