1

Visualization Paradigms

Chandrajit L. Bajaj
University of Texas, Austin

ABSTRACT

A wide variety of techniques have been developed for the visualization of scalar, vector and tensor field data. They range from volume visualization, to isocontouring, from vector field streamlines or scalar, vector and tensor topology, to function on surfaces . This multiplicity of approaches responds to the requirements emerging from an even wider range of application areas such as computational fluid dynamics, chemical transport, fracture mechanics, new material development, electromagnetic scattering/absorption, neuro-surgery, orthopedics, drug design. In this chapter I present a brief overview of the visualization paradigms currently used in several of the above application areas. A major objective is to provide a roadmap that encompasses the majority of the currently available methods to allow each potential user/developer to select the techniques suitable for his purpose.

1.1 Introduction

Typically, informative visualizations are based on the combined use of multiple techniques. For example figure 1.1 shows the combined use of isocontouring, volume rendering and slicing to highlight and compare the internal 3D structure of three different vorticity fields. For a detailed description of each of the approaches we make reference to subsequent chapters in this book and previously published technical papers and books [Bow95, Cle93, HU94, KK93, NHM97, REE^{+}94, Wat92].

DATA VISUALIZATION TECHNIQUES, Edited by C. Bajaj

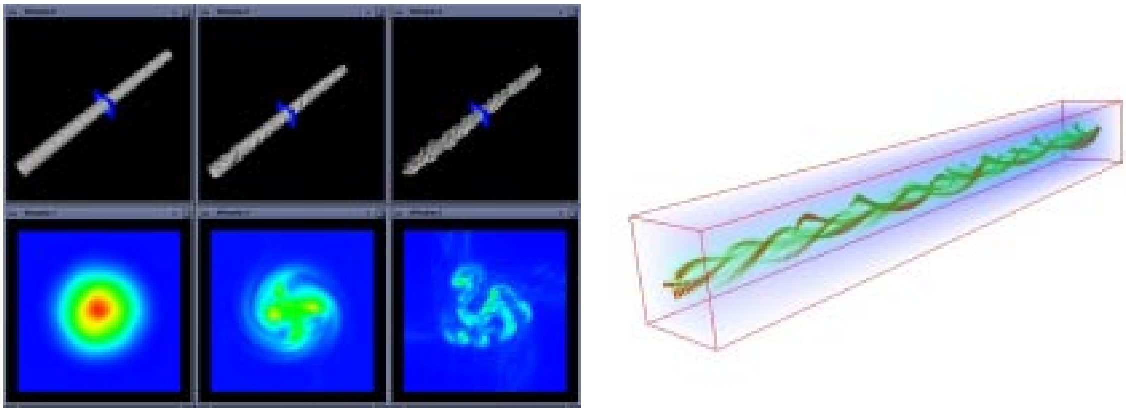

Figure 1.1 The combined display of isocontours, slicing and volume rendering used to highlight the 3D structure of vorticity fields.

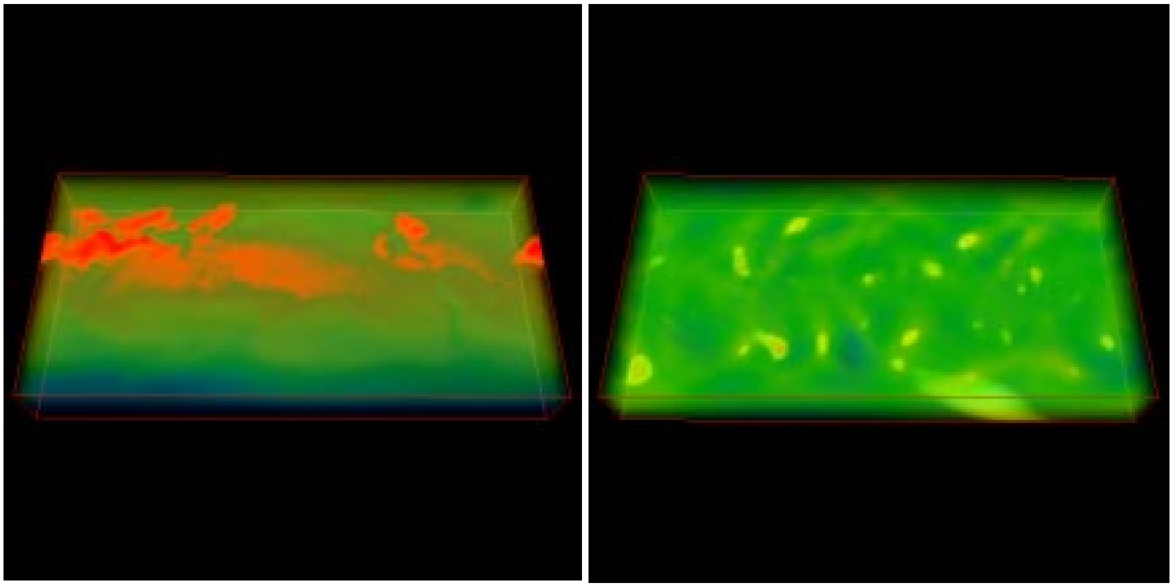

Figure 1.2 Two volume renderings showing snapshots of wind speed in a global climate model.

1.2 Volume Rendering

Volume rendering is a projection technique that produces image displays of three-dimensional volumetric data (see fig. 1.2). Its main characteristic is the production of view-dependent snapshots of volumetric data, rather than the extraction of geometric information such as isocontouring.

Chapter 2 surveys alternate volume rendering algorithms reported in the literature. Two main classes of approaches that have been developed differ mainly on the order of projection of the volume cells. Secondary distinctions arise from the differences in color accumulation and composition techniques to produce the final image.

Forward projection techniques traverse the volume (object space approach) projecting and display each volumetric cell or voxel. This approach takes advantage of graphics hardware acceleration by selecting appropriate drawing primitives to approximate the voxel image.

Backward projection techniques traverse the image (image space approach) and cast through the data volume, one light ray per pixel, accumulating color intensities along the ray to determine the final pixel color.

Cell projection and splatting are both forward projection techniques. In cell projection, the cells of the data volume are traversed and their images computed by subdivision into a polygonal approximation. In splatting, the samples of the volume are traversed and their contribution to the final image is computed by convolution with a reconstruction kernel. Cell projection technique can be optimized by taking advantage of the spatial coherence of the volume cells both in the case of regular grids and in the case of unstructured meshes. Splatting has been shown to be a fast technique for hardware assisted scalar volume visualization, and was extended to vector fields (see details in Chapter 5) Additional splatting techniques are developed for texture based visualization of velocity fields in the vicinity of contour surfaces (see details in Chapter 6)

Backward projection methods are accelerated by exploiting the coherence between adjacent rays. This idea has been implemented in a number of approaches using: (i) adaptive sampling along the rays depending on the "importance" of different regions(ii) templating the paths of parallel rays through regular grids,(iii) bounding with simple polyhedra significant regions that give the main contribution to the output image, or (iv) maintaining the front of propagating rays through irregular grids. The high computational cost of volume rendering in the spatial domain can sometimes be replaced by an asymptotically faster computation in the frequency domain [Lev92, Mal93, TL93].

1.3 Isocontouring

Isocontouring is the extraction of constant valued curves and surfaces from 2d and 3d scalar fields. Interactive display and quantitative interrogation of isocontours helps in determining the overall structure of a scalar field (see fig. 1.3) and its evolution over time (see fig. 1.4).

Chapter 3 surveys the most commonly used isocontouring algorithms along with recent improvements that permit rapid evaluation of multiple isocontour queries, in an interactive environment. Traditional isocontouring techniques examine each cell of a mesh to test for intersection with the isocontour of interest. Accelerated isocontouring can be achieved by preprocessing the input scalar field both in its domain (the geometry of the input mesh) and in its range (the values of the sampled scalar field).

One the one hand, one takes advantage of the known adjacency information of mesh cells (domain space optimization). Given a single cell c on an isocontour component one can trace the entire isocontour component from c, by propagating from cell to cell using inter-cell adjacency. This reduces the search for isocontour components from a search in the entire input mesh to a search in a much smaller subset called the *seed set*. A *seed set* is a subset of the mesh cells which has at least one cell on each connected component of each isocontour. From this typically very small seed set of mesh cells one searches for starting cells for each component of the desired isocontour and then applies contour propagation through cell adjacencies.

On the other hand, one independently optimizes the search for isocontours exploiting the simplicity of the range of the scalar field (range space optimization). The values of the field are scalars that in range space form an interval. Within each cell of the mesh (or of mesh cells of the seed set) the scalar field usually has a small continuous variation that can be represented in range space as a (small) subinterval. The isocontour computation is hence reduced in range space to the search for all the segments that intersect the currently selected isovalue w. This search can be optimally performed using well known interval tree or segment tree data structures.

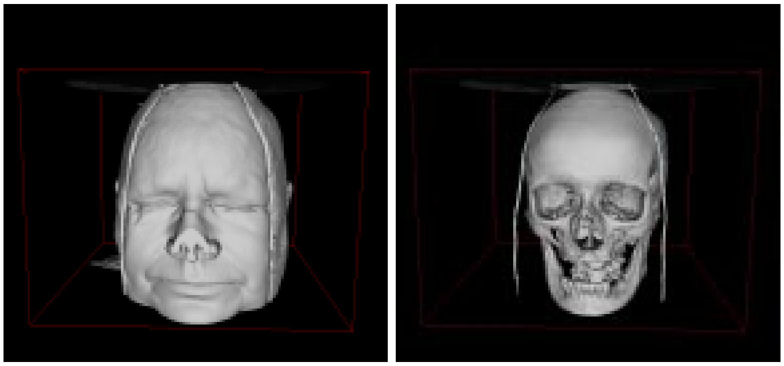

Figure 1.3 Skin and bone head models extracted as two different isocontours from the same volumetric MRI data of the Visible Human female.

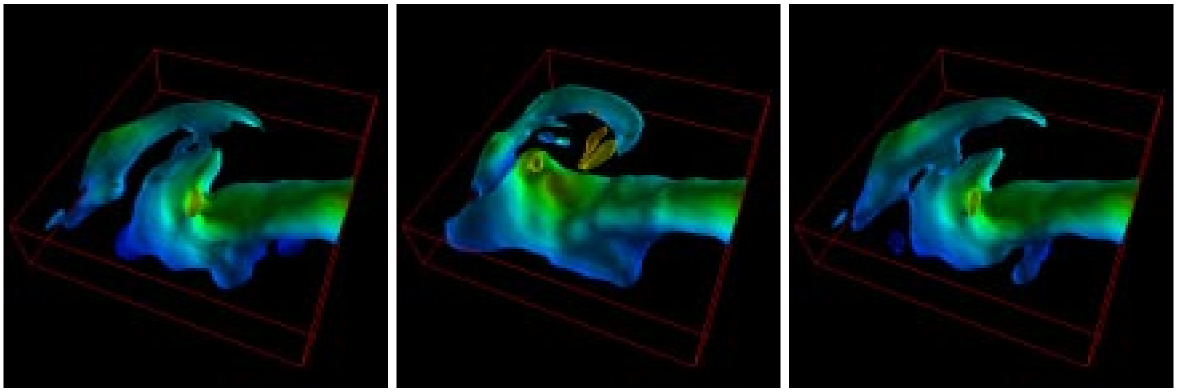

Figure 1.4 Three isocontours of wind speed that show the time evolution of air dynamics in a global climate model.

1.4 Flow Visualization

Visualization of vector fields is generally more complicated than visualizing scalar fields due to the increased amount of information inherent in vector data. Clearly vector data can be contracted to scalar quantities, for example by computation of vector magnitude, scalar product with a given vector, or magnitude of vorticity. In this case, scalar visualization techniques such as isosurfaces and volume rendering can be applied. Additional approaches to visualization of vector fields include iconography, particle tracking, and qualitative global flow visualization techniques. Chapter 5 reviews flow visualization techniques while Chapter 6 describes more in detail the approaches designed to take advantage of currently existing graphics hardware to increase performance. For additional detail, refer to the papers cited in these two chapters.

Particle tracking or advection techniques are based on following the trajectory of a theoretically massless particle in a flow. In its simplest form, the path traversed by a particle in a steady flow is called a *streamline*. If the flow is unsteady, or time-varying, the path followed by a particle over time is called a *path line*. A curve resulting from a number of particles emitted at regular or irregular intervals from a single source is called a *streak line*. Numerical techniques commonly used for evaluating the above equation include Euler and Runge-Kutta methods. In the case of incompressible flow, a single stream-function in 2D can be constructed such that the contours of the stream-function are streamlines of the vector field. In 3D, a pair of dual stream functions is required, and streamlines will occur as the intersections of isocontours of the two functions [KM92].

Particle tracking techniques may also be extended by grouping multiple particles together to form a *stream ribbon*, *stream surface*, *stream tube* or *flow volume*. Global techniques such as Line Integral Convolution present a qualitative view of the vector field which presents intuitively meaningful visualizations for the user. Flow "probes" may be placed at user-specified or computed locations to reveal local properties of the flow field such as direction, speed, divergence, vorticity, etc. Properties are mapped to a geometric representation called an *icon*. The complexity of the icon increases with the amount of information that it is designed to represent. Representing curl (vorticity), which is itself a vector field with additional physical meaning, can be achieved by a cylindrical icon with candystriping to indicate both the direction and magnitude of vorticity.

1.5 Quantification

In the quest for interrogative visualization [Baj88], in which the user can not only see the data, but navigate and query for increased understanding, the ability to quantify and perform volumetric measurements is vital. Another challenge to visualization is to give quantitative information concerning time dependent studies and time-varying structures (e.g. flow). In the study of paralysis, researchers are constructing models of spinal cords and regions of damaged cord from histological samples. Figure 1.5 (left) is an example of a histological slice of an injured rat spine. In Figure 1.5 (right), the damaged region has been reconstructed as a surface, and is visualized along with orthogonal slices of the 3D histological specimen. Traditionally, spinal damage has been

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase® Smarter legal research.