Introduction to Scientific Visualization

Stefan Bruckner

Simon Fraser University / Vienna University of Technology
**Visualization – Definition**

**visualization**: to form a mental vision, image, or picture of (something not visible or present to the sight, or of an abstraction); to make visible to the mind or imagination


**tool** to enable a **user** insight into **data**

“The purpose of computing is insight, not numbers.”

[R. Hamming, 1962]
Visualization – Goals

- Visualization, …
  - … to explore
    - Nothing is known, Vis used for data exploration
  - … to analyze
    - There are hypotheses, Vis used for verification or falsification
  - … to present
    - “everything” known about the data, Vis used for communication of results
Visualization – Areas

Three major areas

- Volume Visualization
- Flow Visualization
- Information Visualization

Scientific Visualization

- 2D/3D
- nD

usually no spatial reference

inherent spatial reference
InfoVis vs. SciVis

- **N-dimensional vs. 2/3-dimensional**
  - SciVis can be N-dimensional too (time series, simulation data, …)

- **Abstract data vs. spatial data**
  - InfoVis data may also have spatial attributes (country, state, …)

- **Discrete data vs. continuous data**
  - InfoVis data may be sampled from a continuous domain
Volume data
SciVis – Examples (2)

Flow data
sketch from Leonardo Da Vinci‘s anatomical notebooks

medical illustrations by Clarice Ashworth Francone
isolines to visualize compass deviations

wind flow visualization
Meteorology

map with iso-pressure lines

weather fronts

map for pilots
Experimental Flow Investigation

- Fixation of tufts, ribbons on ...
  - aircraft in wind tunnels
  - ship hull in fluid tanks

- Introduction of smoke particles (in wind tunnel)
- Introduction of dye (in fluids)
Visualization Scenarios

- Complexity, tech. demands
- Passive visualization
- Interactive visualization
- Interactive steering
- Benefits, possibilities

Stefan Bruckner
Visualization Pipeline

Acquisition
- Data are given

Enhancement
- Data are processed

Mapping
- Data are mapped to, e.g., geometry

Rendering
- Images generated
Focus of visualization, everything is centered around the data

Driving factor (besides user) in choice and attribution of the visualization technique

Important questions

- In what domain are the data given? (data space)
- What is the type of data? (data characteristics)
- Which representation makes sense?
## Data Space vs. Data Characteristics

<table>
<thead>
<tr>
<th>data characteristics</th>
<th>1D</th>
<th>2D</th>
<th>3D</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1D</strong></td>
<td>y=f(x)</td>
<td>spatial curve x(t)</td>
<td></td>
</tr>
<tr>
<td><strong>2D</strong></td>
<td>2D flow v(x)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>3D</strong></td>
<td>scalar field d(x)</td>
<td></td>
<td>3D flow v(x)</td>
</tr>
</tbody>
</table>
Grids – General Information

Important questions

- Which data organization is optimal?
- Where do the data come from?
- Is there a neighborhood relationship?
- How is the neighborhood information stored?
- How is navigation within the data possible?
- Calculations with the data possible?
- Are the data structured?
Grid - Types

- Cartesian grid
- Curvilinear grid
- Unstructured grid
- Scattered data
Grids - Survey

- Structured grids
  - Orthogonal grids
    - Equidistant grids
      - Cartesian grids (dx=dy)
      - Regular grids (dx≠dy)
    - Rectilinear grids
  - Curvi-linear grids
- Unstructured grids
- Hybrid grids
- Miscell.

Stefan Bruckner
Volume Visualization

- **VolVis** = visualization of volume data
  - Mapping 3D → 2D

- **Volume data**
  - 3D × 1D data
  - Scalar data, 3D data space, space filling

- **User goals**
  - Gain insight in 3D data
  - Structures of special interest + context
Volume Data

- Medicine
  - CT, MRI, PET, Ultrasound

- Biology
  - Confocal microscopy, histological cuts

- Geology
  - Seismic surveys

- Material testing
  - Industrial CT
3D Data Space

- Cartesian/regular grid
  - Most common, e.g., CT/MRI scans

- Curvilinear/unstructured grid
  - Less frequently, e.g., simulation data
Concepts and Terms

- **sampled data** (measurement)
- **analytical data** (modelling)
- **voxel space** (discrete)
- **geometric surfaces** (analytic)
- **pixel space** (discrete)

Connections:
- Voxelization from voxel space to pixel space.
- Iso-surfacing from voxel space to geometric surfaces.
- Surface rendering from geometric surfaces to pixel space.
- (Direct) volume rendering from voxel space to pixel space.
Volume Rendering (1)

- Deals with the visual representation of 3D functions
- Frequently, but not exclusively, functions are scalar-valued
- Often acquired using sampling (e.g., medical domain)
Initially volumes were visualized using two-dimensional cuts

Extraction of surface geometry for isosurfaces in the volume (e.g. Marching Cubes [Lorensen and Cline 1987])

Volume rendering introduced almost simultaneously by [Levoy 1988] and [Drebin et al. 1988]
Surface vs. Volume Rendering

- Surface rendering
  - **Indirect** volume visualization
  - Intermediate representation: iso-surface
  - Pros: Less memory, fast rendering

- Volume rendering
  - **Direct** volume visualization
  - Usage of transfer functions
  - Pros: illustrate the interior, semi-transparency
Volume Ray Casting

- **Volume**: 1D value defined in 3D
  \[ f(\mathbf{x}) \in \mathbb{R}^1, \mathbf{x} \in \mathbb{R}^3 \]

- **Ray**: Half-line
  \[ \mathbf{r}(t) \in \mathbb{R}^3, t \in \mathbb{R}^1 > 0 \]

- **Intensity profile**: values along a ray
  \[ f(\mathbf{r}(t)) \in \mathbb{R}^1, t \in \mathbb{R}^1 > 0 \]

- **Image plane**: starting points of rays
Pipeline – Overview

volume rendering pipeline

data set

reconstruction

classification

shading

compositing

final image
Pipeline – Reconstruction

volume rendering pipeline

data set

reconstruction

classification

shading

compositing

final image
Reconstruction (1)

- Usually volume data sets are given as a grid of discrete samples.
- For rendering purposes, we want to treat them as continuous three-dimensional functions.
- We need to choose an appropriate reconstruction filter.
- Requirements: high-quality reconstruction, but small performance overhead.
Reconstruction (2)
Reconstruction (3)

sample point

voxel

cell
Trilinear Interpolation

- Simple extension of linear interpolation to three dimensions
- Advantage: current GPUs automatically do trilinear interpolation of 3D textures

\[ v_p = v_{000}(1 - x_p)(1 - y_p)(1 - z_p) + v_{100}x_p(1 - y_p)(1 - z_p) + v_{010}(1 - x_p)y_p(1 - z_p) + v_{001}(1 - x_p)(1 - y_p)z_p + v_{011}(1 - x_p)y_pz_p + v_{101}x_p(1 - y_p)z_p + v_{110}x_py_p(1 - z_p) + v_{111}x_py_pz_p \]
Other Reconstruction Filters

- If very high quality is required, more complex reconstruction filters may be required.
- Marschner-Lobb function is a common test signal to evaluate the quality of reconstruction filters [Marschner and Lobb 1994].
- The signal has a high amount of its energy near its Nyquist frequency.
- Makes it a very demanding test for accurate reconstruction.
Comparison of Reconstruction Filters (1)

- Marschner-Lobb test signal (analytically evaluated)
Trilinear reconstruction of Marschner-Lobb test signal
Cubic reconstruction of Marschner-Lobb test signal
B-Spline reconstruction of Marschner-Lobb test signal
Comparison of Reconstruction Filters (5)

- **Windowed sinc** reconstruction of Marschner-Lobb test signal
Comparison of Reconstruction Filters (6)

- Marschner-Lobb test signal (analytically evaluated)
Pipeline – Classification

volume rendering pipeline

data set

reconstruction

classification

shading

compositing

final image
Projecting a 3D data set onto a 2D image is problematic.

Not all information contained in the volume is relevant to the user.

Classification allows the user to extract the important parts of the data.
During Classification the user defines the appearance of the data
- Which parts are transparent?
- Which parts have which color?
During Classification the user defines the appearance of the data:
- Which parts are transparent?
- Which parts have which color?

The user defines a transfer function.
real-time update of the transfer function important
Classification Order (1)

- Classification can occur before or after reconstruction

- **Pre-interpolative**: classify all data values and then interpolate between RGBA-tuples

- **Post-interpolative**: interpolate between scalar data values and then classify the result
pre-interpolative  

post-interpolative

same transfer function, resolution, and sampling rate
Classification Order (5)

pre-interpolative  post-interpolative

same transfer function, resolution, and sampling rate
Pipeline – Shading

volume rendering pipeline

data set

reconstruction

classification

shading

compositing

final image
Shading (1)

- Make structures in volume data sets more realistic by applying an illumination model
- Shade each sample in the volume like a surface
- Any model used in real-time surface graphics suitable
- Common choice: Blinn-Phong illumination model
Local illumination, similar to surface lighting

- **Lambertian reflection**  
  light is reflected equally in all directions

- **Specular reflection**  
  light is reflected scattered around the direction of perfect reflection
Shading (3)

shaded volume rendering

unhaded volume rendering
Gradient Estimation (1)

- Normalized gradient vector of the scalar field is used to substitute for the surface normal.
- The gradient vector is the first-order derivative of the scalar field.

\[ \nabla f(x) = \begin{pmatrix} \frac{\partial f(x)}{\partial x} \\ \frac{\partial f(x)}{\partial y} \\ \frac{\partial f(x)}{\partial z} \end{pmatrix} \]

- Partial derivative in \( x \)-direction
- Partial derivative in \( y \)-direction
- Partial derivative in \( z \)-direction
We can estimate the gradient vector using finite differencing schemes, e.g. central differences:

\[ \nabla f(x, y, z) \approx \frac{1}{2h} \begin{pmatrix} f(x + h, y, z) - f(x - h, y, z) \\ f(x, y + h, z) - f(x, y - h, z) \\ f(x, y, z + h) - f(x, y, z - h) \end{pmatrix} \]

Noisy data may require more complex estimation schemes
Gradient Magnitude

- Magnitude of gradient vector can be used to measure the “surfaceness” of a point
  - Strong changes $\rightarrow$ high gradient magnitude
  - Homogenity $\rightarrow$ low gradient magnitude

- Applications
  - Use gradient magnitude to modulate opacity of sample
  - Interpolate between unshaded and shaded sample color using gradient magnitude as weight
Pipeline – Compositing

volume rendering pipeline

data set

reconstruction

classification

shading

compositing

final image
Compositing (1)

- So far, everything discussed applies to single sample points along a viewing ray.
- How to subsequent sample when traversing the ray?
- Common models:
  - Maximum Intensity Projection
  - Emission-Absorption Model
Maximum Intensity Projection (1)

- Always display the maximum value along a viewing ray
- Motivation: visualization of contrast-enhanced tomographic scans
- Parameterless rendering, very common in medical domain
Problem: loss of spatial relationships between different structures
Conventional volume rendering uses an emission-absorption model.

Scattering effects are usually ignored due to high computational complexity.

For each pixel on the image plane, a ray integral has to be solved.

For each step along a viewing ray, perform accumulate RGBA from transfer function.
Emission-Absorption Model (2)

increase

emission

decrease

absorption

in-scattering

out-scattering
Ray Integration (1)
How do we determine the radiant energy along the ray?

**Physical model:** emission and absorption, no scattering

Every point $\tilde{s}$ along the viewing ray emits additional radiant energy

$$I(s) = I(s_0) e^{-\tau(s_0,s)} + \int_{s_0}^{s} q(\tilde{s}) e^{-\tau(\tilde{s},s)} d\tilde{s}$$
Numerical Solution (1)

Eye → Image Plane → Data Set
Numerical Solution (2)

\[ \tilde{C} = \sum_{i=0}^{[T/\Delta t]} C_i \prod_{j=0}^{i-1} (1 - A_j) \]

can be computed recursively

\[ C'_i = C_i + (1 - A_i)C'_{i-1} \]

radiant energy observed at position \( i \)
radiant energy emitted at position \( i \)
absorption at position \( i \)
radiant energy observed at position \( i-1 \)
Numerical Solution (2)

**Back-to-front compositing**

\[ C_i' = C_i + (1 - A_i)C'_{i-1} \]

**Front-to-back compositing**

\[ C_i' = C'_{i+1} + (1 - A'_{i+1})C_i \]
\[ A_i' = A'_{i+1} + (1 - A'_{i+1})A_i \]

*Early ray termination:* stop the calculation when \( A_i' \approx 1 \)
Further Reading

Flow Visualization

- FlowVis = visualization of flows
  - Visualization of change information
- Flow data
  - nD×nD data, 1D²/2D²/nD² (models), 2D²/3D²
  - Vector data (nD) in nD data space
  - Steady vs. time-dependent flow
- User goals
  - Overview vs. details (with context)
Flow Data (1)

- Simulation
  - Flow space modelled with grid
  - FEM (finite elements method), CfD (computational fluid dynamics)

- Measurements
  - Optical methods + pattern recognition, e.g.: PIV (particle image velocimetry)

- Models
  - Differential equation systems \( \frac{d\mathbf{x}}{dt} \)
Flow Data (2)
Flow Data (3)

experiment

simulation
Direct Flow Visualization

- Grid of arrows to visualize flow directions
- Normalized arrows vs. scaling with velocity
- Quite effective in 2D, problematic in 3D
- Sometimes limited expressivity (temporal component missing)
Geometric Flow Visualization

- Idea: follow the flow in time (integration) to extract the path of a particle
Streamlines – Theory

- **Flow data** $\mathbf{v}$: derivative information
  - $\frac{d\mathbf{x}}{dt} = \mathbf{v}(\mathbf{x})$
  - Spatial points $\mathbf{x} \in \mathbb{R}^n$, flow vectors $\mathbf{v} \in \mathbb{R}^n$, time $t \in \mathbb{R}$

- **Streamline** $\mathbf{s}$: integration over time, also called trajectory, solution, curve
  - $\mathbf{s}(t) = \mathbf{s}_0 + \int_{0 \leq u \leq t} \mathbf{v}(\mathbf{s}(u)) \, du$
  - Seed point $\mathbf{s}_0$, integration variable $u$

- **Difficulty**: result $\mathbf{s}$ also in the integral, analytical solution usually impossible
Solve using numerical integration techniques

Assume that locally the solution is approximately linear

Euler integration

\[ s_{i+1} = s_i + dt \cdot v(s_i) \]
Follow the current flow vector \( v(s_i) \) from the current streamline point \( s_i \) for a short time \( (dt) \)
Accuracy of results is strongly dependent on step size $dt$
Streamlines – Placement (1)

Seed fill with streamlines to achieve equal density
Streamlines – Placement (2)

- Variations of distance in relation to image width

6%  
3%  
1.5%
Illuminated Streamlines

- Illuminated 3D curves improve perception
Stream Surfaces

- Natural extension of streamlines to 3D
- Surfaces which are tangential to the vector field everywhere
- Challenges related to occlusion and visual complexity
Flow Volumes

- Volumetric equivalents of streamlines, subset of a 3D flow domain is traced in time
- Can be visualized with direct volume rendering methods
Unsteady Flow

- Path line
  - Trajectory of an individual particle in the fluid flow

- Timeline
  - Joins the positions of particles released at the same instant in time

- Streak line
  - Connects particles that have passed through a certain point in space
Texture-based Flow Visualization

- Idea: exploit visual correlations to provide a dense visualization of the flow
Line Integral Convolution

- Calculation of a texture value
  - look at streamline through point
  - filter white noise along streamline

flow data

integration

streamline

convolution

white noise

results in

LIC texel

Stefan Bruckner
laminar flow

turbulent flow
Line Integral Convolution on Surfaces
Texture Advection – Unsteady Flows
Feature-based Flow Visualization

- Extract and visualize the abstract structure of a flow

- Different elements
  - Checkpoints, defined through $v(x)=0$
  - Cycles, defined through $s(x(t+T))=s(x(t))$
  - Connecting structures (separatrices, etc.)
Critical points can be classified by the Eigenvalues of the Jacobian

$R = \text{real components, } I = \text{imaginary components}$
Glyphs/Icons

- Local/topological properties
Examples (1)

- Topology of a hurricane simulation
Examples (2)

- Visualization of flow past a circular cylinder using critical points and saddle connectors
Further Reading


Scientific visualization is data-driven, but it is crucial to keep the goal of the user in mind.

**Volume visualization**
3D scalar data
- Important to provide detailed view of structures of interest

**Flow visualization**
2D/3D vector data
- Provide overview and characterize flow behavior
Thank you for your attention!

Acknowledgements
Meister Eduard Gröller
Heliwg Hauser
Christof Rezk-Salama