Transfer Functions for Direct Volume Rendering

Gordon Kindlmann

gk@cs.utah.edu

http://www.cs.utah.edu/~gk

Scientific Computing and Imaging Institute
School of Computing
University of Utah

Contributions:
Many, as noted
Outline

1. Transfer Functions: what and why

2. Review of current methods

3. Ideas for future work
Introduction

Transfer functions make volume data visible by mapping data values to optical properties.

slices:

volume data:

volume rendering:
Transfer Functions (TFs)

Simple (usual) case: Map data value $f$ to color and opacity

Motivation

Shading, Compositing...

Human Tooth CT
Optical Properties

Anything that can be composited with a standard graphics operator (“over”)

- Opacity: “opacity functions”
  - Most important
- Color
  - Can help distinguish features
- Emittance
  - Why don’t we use this more often?
- Phong parameters ($k_a$, $k_d$, $k_s$)
- Index of refraction
Alas…

Setting transfer functions is difficult, unintuitive, and slow
TFs as feature detection

Where’s the edge?

\[ v = f(x) \]

“here’s the edge!”

Result is set of edge pixels:
TFs as feature detection

We are looking in the data value domain, not the spatial domain

\[ v = f(x) \]

\[ \alpha(v) \]

\[ \alpha(v) \]

“here’s the edge!”
TFs as feature detection

$v = f(x)$

"here’s the edge!"

Domain of the transfer function does not include position

$v_0$

"here’s the edge!"
Goals

• Make good renderings easier to come by
• Make space of TFs less confusing
• Remove excess “flexibility”
• Provide one or more of:
  • Information
  • Guidance
  • Semi-automation
  • Automation
Outline

1. Transfer Functions: what and why

2. Review of current methods

3. Ideas for future work
Current Methods

Organization

1. Trial and Error (manual)
2. Spatial Feature Detection
3. Image-Centric
4. Data-Centric
5. Others
1. Trial and Error

1. Manually edit graph of transfer function
2. Enforces learning by experience
3. Get better with practice
4. Can make terrific images

William Schroeder, Lisa Sobierajski Avila, and Ken Martin; Transfer Function Bake-off Vis ’00
1. Trial and Error (manual)
2. Spatial Feature Detection
3. Image-Centric
4. Data-Centric
5. Others
2. Spatial Feature Detection

Transform TF specification to feature detection in the spatial domain

- extremely flexible
- different parameter space
- not exactly transfer functions …

1. Fang, Biddlecome, Tuceryan (Vis ‘98) “Image-based Transfer Function Design…”

2. Rheingans, Ebert (Vis ’00, TVCG July ’01) “Volume Illustration: Non-photorealistic…”

3. Hladůvka, Gröller (VisSym ’01) “Salient Representation of Volume Data”
### Traditional Volume Rendering Pipeline

1. **Volume values** $f_i(x_i)$
2. **Shading**
3. **Voxel colors** $c_i(x_i)$
4. **Classification**
5. **Voxel opacities** $\alpha(x_i)$
6. **Shaded, segmented volume** $[c_i(x_i), \alpha(x_i)]$
7. **Resampling and compositing** (raycasting, splatting, etc.)
8. **Image pixels** $C_\lambda(u_i)$

### Volume Illustration Rendering Pipeline

1. **Volume values** $f_i(x_i)$
2. **Transfer function**
3. **Volume Illustration**
   - **Color modification**
   - **Opacity modification**
4. **Final volume sample** $[c_i(x_i), \alpha(x_i)]$
5. **Image pixels** $C_\lambda(u_i)$

---

### Feature Enhancement

- **Boundary, silhouette enhancement**
- **Depth and Orientation Cues**
- **Halos, depth cueing**

*Thanks to Penny Rheingans and David Ebert*
Volume Illustration

3. Spatial Features

Original TF  Boundaries (gradient)
Volume Illustration

3. Spatial Features

Silhouettes
Halos

Blurs distinction between transfer functions and feature detection
Organization

1. Trial and Error (manual)
2. Spatial Feature Detection
3. Image-Centric
4. Data-Centric
5. Others
3. Image-centric

Specify TFs via the resulting renderings

- **Genetic Algorithms** (“Generation of Transfer Functions with Stochastic Search Techniques”, He, Hong, et al.: Vis ’96)
- **Design Galleries** (Marks, Andalman, Beardsley, et al.: SIGGRAPH ’97; Pfister: Transfer Function Bake-off Vis ’00)
- **Thumbnail Graphs + Spreadsheets** (“A Graph Based Interface…”, Patten, Ma: Graphics Interface ’98; “Image Graphs…”, Ma: Vis ’99; Spreadsheets for Vis: Vis ’00, TVCG July ’01)
Genetic Algorithms

Initial stochastic search; refinement can be user driven or automated (“fitness functions”)

“Generation of Transfer Functions with Stochastic Search Techniques”, He, Hong, et al.: Vis ’96
Effective method for general class of "parameter tweaking" problems

- Provide convenient GUI to whole parameter space ("what’s possible?")
- Sampling parameter space: dispersion
- Organize output images: arrangement

Inputs: Transfer Functions

Outputs: Images

Organize Images for easy browsing
Design Galleries

VoIDG
(software available)

Marks, Andalman, Beardsley, et al.: SIGGRAPH ’97; Pfister: Transfer Function Bake-off Vis ’00
3. Image-Centric

Thumbnail Graphs, Spreadsheets

Exploration guided by logically connected visual history or spreadsheet

“A Graph Based Interface for Representing Volume Visualization Results”, Patten, Ma: Graphics Interface ‘98

“Visualization Exploration and Encapsulation via a Spreadsheet-Like Interface”, Jankun-Kelly, Ma: TVCG July 2001
Organization

1. Trial and Error (manual)
2. Spatial Feature Detection
3. Image-Centric
4. Data-Centric
5. Others
4. Data-centric

Specify TF by analyzing volume data itself

1. Salient Isovalues:
   - **Contour Spectrum** (Bajaj, Pascucci, Schikore: Vis ’97)
   - **Statistical Signatures** (“Salient Iso-Surface Detection Through Model-Independent Statistical Signatures”, Tenginaki, Lee, Machiraju: Vis ’01)
   - **Other computational methods** (“Fast Detection of Meaningful Isosurfaces for Volume Data Visualization”, Pekar, Wiemker, Hempel: Vis ’01)

2. “Semi-Automatic Generation of Transfer Functions for Direct Volume Rendering”
   (Kindlmann, Durkin: VolVis ’98; Kindlmann MS Thesis ’99; Transfer Function Bake-Off Panel: Vis ’00)
Salient Isovalues

What are the “best” isovalues for extracting the main structures in a volume dataset?

Contour Spectrum (Bajaj, Pascucci, Schikore: Vis ’97; Transfer Function Bake-Off: Vis ’00)

• Efficient computation of isosurface metrics
  • Area, enclosed volume, gradient surface integral, etc.
• Efficient connected-component topological analysis
• Interface itself concisely summarizes data
Contour Spectrum
Statistical Signatures

- Localized $k$-order central moments
- At each position $P$ in volume, compute ...
  - $LM$: mean over local window $W$
  - $m_k$: local higher order moment (LHOM)

\[
m_k = \frac{1}{2} \sum_{w} (x - LM)^k, (\forall x \in W)\]

Example: $m_3$

(Thanks to Shiva Tenginaki, Jinho Lee, Raghu Machiraju)
Boundary Model

- Small window
- Boundary if $|C_1 - C_2| > 0$
- Binomial distribution of materials
- Extrema and zero-crossings of moments and cummulants are influenced by presence of boundaries
Moments + Cummulants

\[ m_2 \quad m_3 \quad m_4 \]

Skew

Kurtosis
Scatterplots

skew vs. value
Other Computational Methods

“Fast Detection of Meaningful Isosurfaces for Volume Data Visualization”, Pekar et al.: Vis ‘01

Integral of gradient magnitude over isosurface
• High for isovalues of strong boundaries
• Can be computed with divergence theorem: Integral of vector field over surface is same as integral of divergence in the interior
• Application of classical vector calc
• Rapid computation with Laplacian-weighted histograms
Other Computational Methods

Pekar et al. “Fast Detection of Meaningful Isosurfaces for Volume Data Visualization”, Vis ‘01
MEAN gradient combined with the opacity transfer function

Pekar et al. “Fast Detection of Meaningful Isosurfaces for Volume Data Visualization”, Vis ‘01
“Semi-Automatic …”

Reasoning:

- TFs are volume-position invariant
- Histograms “project out” position
- Interested in boundaries between materials
- Boundaries characterized by derivatives

→ Make 3D histograms of value, 1\textsuperscript{st}, 2\textsuperscript{nd} deriv.

By (1) inspecting and (2) algorithmically analyzing histogram volume, we can create transfer functions
Derivative relationships

Edges at maximum of 1\textsuperscript{st} derivative or zero-crossing of 2\textsuperscript{nd}
(1) Scatterplots

- Project histogram volume to 2D scatterplots
  - Visual summary
  - Interpreted for TF guidance
  - No reliance on boundary model at this stage
(2) Analysis

Volume Graphics Distance Map

\[(x, y, z)\]

\[d\]

3D position

New Distance Map

\[d\] Signed distance to boundary

0 \[V\] 255

data value
(2) New Distance Maps

- Supports 2D distance map: $d(v,g)$; $g =$ gradient magnitude
- Produced automatically from histogram volume via boundary model
Opacity function:
\[ \alpha(v) = b(d(v)) \]
\[ \alpha(v,g) = b(d(v,g)) \]

- Opacity function:
  \[ \alpha(v) = b(d(v)) \]
  \[ \alpha(v,g) = b(d(v,g)) \]

Automatically generated from histogram volume

\text{distance function: } d(v)

\text{data value: } v

\text{Created by user}

\text{boundary emphasis function: } b(x)

\text{opacity: } a

\text{“distance”: } x

\text{Whole process}

\text{Created by user}

\text{data value: } v

\text{distance function: } d(v)

\text{opacity function: } \alpha(v)

\text{boundary emphasis function: } b(x)

\text{opacity: } a

\text{Automatically generated from histogram volume}

\text{“distance”: } x

\text{data value: } v

\text{opacity function: } \alpha(v)
Results: CT Head

CT head slice

\[ f \rightarrow f' \rightarrow f'' \]

\[ d(v) \]

\[ V \rightarrow X \rightarrow \alpha \]
Results: CT Head

\[ \alpha = b(x) \]

\[ \alpha(v) \]

\[ \alpha = b(x) \]

\[ \alpha(v) \]
Results: Tooth

Boundary emphasis function simple to set

\( \alpha(v) = b(d(v)) \)
Tooth: 2D transfer function

Detected 4 distinct boundaries between 4 materials

White regions in colormapped 2D distance function plot are boundary centers

- Pulp
- Background
- Dentine
- Enamel

Color transfer function
2D Opacity Functions

Mostly accurate isolation of all material boundaries
Organization

1. Trial and Error (manual)
2. Spatial Feature Detection
3. Image-Centric
4. Data-Centric
5. Others
5. Other methods

- New domains: curvature
- New kinds of interaction
Curvature

“Curvature-Based Transfer Functions for Direct Volume Rendering”, Hladůvka, König, Gröller: SCCG ’00

• Uses 2D space of $K_1$ and $K_2$: principal curvatures of isosurface at a given point
• Graphically indicates aspects of local shape
• Specification is simple
Different Interaction

“Interactive Volume Rendering Using Multi-Dimensional Transfer Functions and Direct Manipulation Widgets” Kniss, Kindlmann, Hansen: Vis ’01

- Make things opaque by pointing at them
- Uses 3D transfer functions (value, 1\textsuperscript{st}, 2\textsuperscript{nd} derivative)
- “Paint” into the transfer function domain
Motivation

3D Transfer Function

3D transfer functions allow
• easier boundary selection
• accurate boundary visualization
Outline

1. Transfer Functions: what and why
2. Current Methods
3. Ideas for future work
Different domains, ranges

- **Time-varying data** ("A Study of Transfer Function Generation for Time-Varying Volume Data", Jankun-Kelly, Ma: Volume Graphics ’01)
- Multi-dimensional TFs expressive and powerful
  - Leverage current techniques for ease of use
- 2D opacity functions: let’s use them!
  - Marc Levoy’s 1988 CG+A Paper

Ranges: Emitance, textures, what else?
Other directions

- Variations on the histogram volume:
  - Different quantities, assumptions, models, analysis?
- Histograms/scatterplots entirely loose spatial information
  - Any way to keep some of it?
  - Can TFs have volume position in domain?
Image-centric methods have a certain appeal
- Any way to steer and constrain them more effectively?
- Image-space analysis of TF fitness?

What kinds of tools do we really want?
- Analytical vs. expressive; simplifying vs. honest?
- What is the proper role for human experimentation?
Questions?