Volume Rendering with Multidimensional Peak Finding

Abstract
Peak finding provides more accurate classification for direct volume rendering by sampling directly at local maxima in a transfer function, allowing for better reproduction of high-frequency features. However, the 1D peak technique finding does not extend to higher-dimensional classification. In this work, we develop a new method for peak finding with multidimensional transfer functions. We use piecewise approximations to dynamically sample in transfer function space between world-space samples. As with unidimensional peak finding, this approach is useful for specifying transfer functions with greater precision, and for accurately rendering noisy volume data at lower sampling rates. Multidimensional peak finding produces comparable image quality with order-of-magnitude better performance, and can reproduce features omitted entirely by standard classification. With no precomputation or storage requirements, it is an attractive alternative to preintegration for multidimensional transfer functions.

1. Introduction
Direct volume rendering (DVR) is a popular technique for visualizing spatial scientific data. A strong appeal of this method is its flexibility: through choice of transfer function, a user can dynamically classify and render different phenomena in the same data set. Multidimensional transfer functions allow the user to visualize volume data more expressively, through classification of derived values such as gradient or curvature [KKH02]. For multifield volume data, multidimensional transfer functions offer insight into relations between variables and provide comparative analysis. Despite being more complex to design, multidimensional transfer functions are more powerful than 1D transfer functions and offer more control than automatic classifications.

While transfer functions offer flexibility, accurate volume rendering requires adequate sampling with respect to both the volume data and chosen transfer function. When either the volume data or transfer function possess high frequencies, high sampling rates are required to reproduce features without artifacts. The conventional solution has been to choose sufficiently smooth transfer functions; however, this limits classification. In multifield visualization, in particular, users are interested in analyzing relationships between two variables, such as regression lines. As shown in Figure 1, rendering such narrow features is costly and often inaccurate with uniform spatial sampling.

By integrating the volume and transfer function domains separately, preintegration [EKE01] reproduces high-frequency features that are omitted by standard classification. Peak finding employs a similar tactic, treating high-frequency values as discrete isovales and solving for their location along the ray for shading [KHW'09]. Classifications with Dirac impulses or sharp features are effectively rendered as isosurfaces and shaded accordingly. More significantly, noisy volume data can be classified and rendered accurately with lower sampling rate and better interactivity.

Although multidimensional preintegration has been proposed for 2D domains [Kra08], it is more costly to render, requires computation and storage of a summed area table, and would likely not extend to high-frequency 2D, or 3D and higher transfer functions. Moreover, preintegration possesses the same limitations as 1D preintegration when handling high-frequency features in the data or transfer function. Peak finding in higher dimensions has not been attempted before. Here, the challenge lies in identifying peaks in higher-dimensional transfer function space.

This paper extends peak finding to multidimensional transfer functions, and presents several novel approaches for finding and sampling at peaks. In general, we propose sampling dynamically in transfer function space, which is inexpensive compared to sampling in volume space, and allows for accurate integration across both domains. We investigate several mechanisms for doing this, using ray marching or scanline sampling on chord or spline parameterizations of the ray’s image in transfer function space. These approaches require no precomputation, and extend trivially to N-dimensional transfer functions. Though simple, multidimensional peak finding has not been examined before, and offers clear qualitative and performance benefits over conventional volume rendering with multidimensional transfer functions.

2. Related Work
Direct volume rendering was introduced by the ray caster of Levoy [Lev88]. Fast rasterization hardware made inter-
Figure 1: Volume rendering of 2-channel fluid dynamics data consisting of the forward finite-time Lyapunov exponent (FTLE) \( \Lambda^+ \); and square-root of ridge strength of the FTLE, given by minimum eigenvalue of the Hessian \( \lambda = \lambda_{\min}\{\nabla^2 \Lambda^+\} \). Multidimensional peak finding lets us quickly and accurately render features defined by regression lines in the joint histogram, in this case visualizing Lagrangian coherent structures [GLT'09] without explicit mesh extraction.

active DVR possible with slicing [CN94, CCF94]. While splatting [Wes90] is feasible, ray casting [KW03, RGW'03] has regained popularity due to its efficient implementation on current GPU hardware. Isosurface mesh extraction from structured volume data was first proposed by [LC87] and remains a common method for visualization. Direct ray casting of isosurfaces was proven on the CPU [Sra94, PSL'98] and later on the GPU [HSS'05]. Kraus [Kra05] reformulated direct volume rendering as an integration of isosurfaces, showing that irradiance can be computed without normalizing the Riemann sum over the number of samples. Multifield isosurface rendering has been used to visualize particle astrophysics data. Navratil et al. [NJB07] use marching cubes to extract separate meshes, while Linsen et al. [LLRR08] employ a particle reconstruction method to resample and render a single surface from multiple channels.

Laidlaw [Lai95] first advocated multidimensional transfer functions for improved classification of MRI data. 2D transfer functions with gradient magnitude [KKH02, Kin02] or curvature [KWTM03] can greatly improve classification flexibility, particularly for noisy scan data in biology and medicine. Kniss et al. [KPI'03] applied specially constructed Gaussian kernels to multifield volume visualization using an analytical integration method for improved visual quality. Simpler classification and blending operations, such as maximum-intensity projection, can equally be used for rendering multifield data [SR04].

Preintegration [RKE00, EKE01] integrates transfer function space using a separate Riemann sum. Irradiance on a ray segment can then be queried in a 2D lookup table. Multidimensional transfer functions can be preintegrated and rendered using a summed area table [Kra08]. This is more costly to render, requiring frustum tracing to integrate over a 2D beam footprint in transfer function space. It is also expensive to preintegrate high-resolution 2D transfer functions, and this approach would not extend to higher-dimensional classification. Peak finding [KHW'09] combines direct volume rendering with discrete scale-invariant isosurfacing by sampling directly at peaks in the transfer function. Ament et al. [AWC10] detail a more robust method for DVR integration of discrete isosurfaces that removes scale-dependency entirely. However, it is expensive (requiring 3 samples per voxel as opposed to multiple voxels between samples for peak finding) and would not extend easily to multidimensional classification due to its reliance on lookup tables.

3. Background

Direct volume rendering is a numerical integration of discrete samples blended according to an emission-absorption model approximating the radiative transport equation [Lev88]. On a ray segment, irradiance is represented continuously as:

\[
I(a,b) = \int_a^b \rho_\text{E}(f(s))\rho_\alpha(f(s))e^{-\int_a^s \rho_\alpha(f(t))dt}ds \quad (1)
\]

Here, \( \rho_\text{E} \) is the emissive term or color, \( \rho_\alpha \) is the opacity of the transfer function; \( a, b \) are the segment endpoints, and \( f(t) = f(O + tD) = f(R(t)) \) is the scalar field function evaluated in world space at \( t \) along the ray. Since the transfer
function is applied after interpolation, ρ(φ) implies postclassification. To approximate Equation 1 discretely, we employ a Riemann sum,
\[ e^{-\int_0^1 p_0(\phi(t)) \ dt} \approx \prod_{i=0}^{n} e^{-\Delta t \ p_0(f(i \Delta t))} = \prod_{i=0}^{n} (1 - \alpha_i) \] (2)
where \( \Delta t \) is the uniform sampling step, \( n = (s - a) / \Delta t \), and
\[ \alpha_i \approx 1 - e^{-\Delta t \ p_0(f(i \Delta t))} \] (3)
Discretizing the integral on \([a, b] \) yields the discrete summation for \( I \),
\[ I \approx \sum_{i=0}^{n} \hat{p}_E(i) \prod_{j=0}^{i-1} (1 - \alpha_j) \] (4)
Where \( \hat{p}_E \) is approximated at discrete points along the ray as:
\[ \hat{p}_E(i) \approx p_0(f(i \Delta t)) \hat{p}_E(f(i \Delta t)) \] (5)
Preintegration employs a separate integral in transfer function space to estimate \( \hat{p}_E \) and \( p_0 \) [KEE01], specifically the Riemann sum of irradiance between two samples \( f_0 = f(t_0) \) and \( f_1 = f(t_1) \), assuming linear spacing of \( f \) values between these points. Typically, the colors \( \hat{p}_E(i) \) are associated, i.e. integrated alongside \( \alpha_i \).
\[ \alpha_i \approx 1 - e^{-\int_0^{\Delta t} p_0((1 - \omega) f_0 + \omega f_1) d \omega} \] (6)
In order for this linear approximation to be accurate, preintegration assumes transfer function space is continuous with bounded variation (specifically, Lipschitz) along the ray. However, it is often applied in scenarios where this is not the case. Peak finding [KHW*09] assumes the transfer function is potentially discontinuous, and that at sharp peaks \( \alpha_i \) it is better approximated by the supremum:
\[ \alpha_i \approx 1 - e^{\max_{t \in [t_0, t_1]} p_0(f)} \] (7)
1D peak finding uses a scale-invariant integration [Kra05], which assumes the isosurface at that peak is always sampled with constant opacity regardless of the step size \( \Delta t \). However, scale-invariant integration is only employed where peaks exist; all other samples are integrated using postclassification. To determine if a peak exists, 1D peak finding uses a 2D lookup similar to a preintegrated table, storing the peak isovalue (or isovales) between \([f_0, f_1] \). At rendering, if a peak \( \nu \) exists in this table, one solves for the spatial location \( t \) of the isosurface such that \( f(\hat{R}(t)) = \nu \), using the secant method.

Peak finding and preintegration accomplish similar aims. Preintegration behaves nicely when both the volume and transfer function are Lipschitz and sampled adequately. Peak finding is preferable when one desires to sample the volume at a lower rate than the transfer function, in the case of extremely high-frequency transfer functions or noisy data.

Figure 2: Classification of multivariate data, and locating peaks between segments.

4. Multidimensional Peak Finding
A multidimensional transfer function is given by a \( \rho : \mathbb{R}^M \rightarrow \mathbb{R} \), which classifies a multidimensional scalar field \( f \in \mathbb{R}^M \times \mathbb{R}^N = \{ f^0, f^1, ..., f^M \}, f^i \in \mathbb{R}^N \). In theory we assume \( f \) is Lipschitz; but in practice the transfer function can be piecewise-constant (non \( C^0 \)) with sufficiently high discretization.

Integrating the data and transfer function separably is more necessary for multidimensional classification than for the univariate case. As with 1D classification, the transfer function \( \rho \) convolves the data signal, requiring sampling at the product of their frequencies. For multivariate data, that signal contains all component variables of \( f \), meaning we must sample \( \rho(t) \) at a frequency \( \nu = \nu(f^0) \times \nu(f^1) ... \nu(f^M) \times \nu(\rho) \).

Thus, multivariate postclassification is inherently higher-frequency, as illustrated in Figure 2. However, if we sample \( \rho \) and \( f \) separately, we need only sample at the maximum of those data frequencies \( \sup(\nu(f^i)) \) in world space, and separately at \( \nu(\rho) \) in transfer space.

Extending 1D peak finding to multidimensional transfer functions is nontrivial, because peaks along a world-space ray almost never occur at points in higher-dimensional transfer space, as shown in Figure 3. In most cases, therefore, one
cannot precompute a peak table and solve for an isovalue at each peak along a ray segment. When handling multidimensional transfer functions, different strategies must be used.

4.1. Separable Transfer Functions
When a multidimensional transfer function is separable into 1D transfer functions, one can use 1D peak finding with some modifications. Even when peaks are close together, it most efficient to assume one peak per ray segment and employ one isosurface solving routine (as opposed to separate solvers for separate dimensions). Instead of using the secant method, we use bisection to determine whether a root for any solvers for separate dimensions). Instead of using the secant poly one isosurface solving routine (as opposed to separate some modifications. Even when peaks are close together, it 1D transfer functions, one can use 1D peak finding with When a multidimensional transfer function is separable into each peak along a ray segment. When handling multidimensional functions, different strategies must be used. cannot precompute a peak table and solve for an isovalue at each peak along a ray segment. When handling multidimensional transfer functions, different strategies must be used.

4.2. General Multidimensional Transfer Functions
Peak finding in general multidimensional functions requires a different approach. Both enumerating peaks and solving at isosurfaces are difficult (if not intractable) for M-dimensional transfer functions. Rather than solve for a specific peak value on a segment $[f_0, f_1]$ (Figure 2), we note that local maxima can be found dynamically along the ray with sufficient sampling, and that:

- Sampling in transfer space $\rho$ is less expensive than sampling in world space $f$.
- Since $\rho$ and $f$ are compact, a contraction in $f$ yields a contraction in $\rho$.
- Finding the correct local maximum $\bar{\rho}$ on a given $[f_0, f_1]$ is more important than accurate world-space location $t_f$.

We are chiefly interested in the intersection of the image of the ray with peak manifolds in transfer function space (ridge lines in 2D). Sampling directly along the ray image is costly, as it requires an increase in the world-space sampling rate. However, it is inexpensive to sample along an approximation of that image in transfer space itself. Such an approximation can be parameterized from world-space points $f_i$ and sampled directly in transfer space $\rho$.

As with 1D peak finding, though we must still sample at the Nyquist limit in $\rho$-space, we only need to sample $f$ such that $\rho$ is monotonic on each $[f_0, f_1]$ – a less stringent requirement. We thus propose to parameterize an approximation of segment $[f_0, f_1]$ in transfer space, as illustrated in Figure 4. As $[f_0, t_1]$ contracts, the segment connecting $f_0, f_1$ contracts to approximate $\rho(f)$, as shown in Figure 4 (a) and (b). We can directly query the transfer function along these segments, identifying a local maximum and using that as our peak. In the rest of this section, we explore chord and spline parameterizations (Figure 4 (a-c), using ray marching or scanline sampling. We claim that curve approximations capture ridge line intersections, thus classified features, better than area elements (Figure 4(d), [Kra08]), and are less costly when dynamically computing maxima or integrals.

When sampling in transfer space, time complexity is linear per ray segment in the worst case, as opposed to constant for postclassification, 1D peak finding, and both 1D and multidimensional preintegration. Unlike with 1D peak finding, we must peak find on each segment, since we never know if a peak exists in $[f_0, f_1]$ until we sample there. However, due to the contractive behavior of $\rho(f)$, few samples in $\rho$ are necessary when samples in $f$ are close. Assuming $\rho$ is Lipschitz, the number of samples needed on $\rho(f)$ will be bounded, implying average-case constant time complexity per segment. In practice, sampling monotonic regions of $\rho$ incurs small cost. More samples in world-space necessitate fewer samples in transfer space, and visa-versa; we are interested in finding a good equilibrium.

4.3. Chord Parameterization
The simplest method of searching in transfer function space is to parameterize the segment between $[f_0, f_1] \in \rho \subset \mathbb{R}^M$ as a line, and sample along that chord. Like 1D peak finding and preintegration, this requires fetching front samples and storing back samples $f(t_1)$ and $f(t_0)$, respectively. We compute a constant to normalize samples over this segment:

$$L_f = ||f_1 - f_0||$$

$$d_s = \Delta s W / L_f$$

where $\Delta s$ is our sampling step in $\rho$ (pixels per sample in transfer space) and $W$ is the discretization of the transfer
function \(W=1024\) for a 1k\(^2\) texture. We then parameterize the chord as a ray \(F(s)\), where \(s \in [0, 1]\).

\[
d_f = (f_1 - f_0) \\
F(s) = f_0 + s d_f
d_f
data_0 = (f_1 - f_0)
\]

We find that better visuals and performance are achieved with relatively high-resolution transfer functions with smooth (non-pixelated) features and \(\Delta s > 1\); we use \(\Delta s = 4, W = 1024\) in Figure 1. More precise classification can be obtained with lower-resolution transfer functions and \(\gamma < 1\). For analytical transfer functions such as the one in Figure 6, we similarly set \(\Delta s\) based on the smallest desired feature size.

Through this iteration, we find the \(s_\rho\) corresponding to the maximum \(\tilde{R}(F(s))\) along the segment, then we interpolate to find the peak \(\tau\):

\[
t_\rho = t_0 + \frac{s_\rho}{t_1 - t_0}
\]

Having found our peak on this segment, we proceed to shade (Section 4.6).

4.4. Spline Parameterization

Spline interpolation is a logical improvement over piecewise linear parameterization. To accomplish this, we must maintain a stencil of four world-space samples \(f_{-1}, f_0, f_1, f_2\). We use a cubic Hermite spline formulation, as the four coefficients \(h_j(s)\) can be precomputed and efficiently accessed in a 1D texture on the GPU. As in Section 4.3, we use the chord length \(L_f\) to parameterize \(s \in [0, 1]\) and choose a suitable increment \(d_s\). Although this is an imperfect metric, arc-length parameterization would be too costly. We then parameterize the curve as a Catmull-Rom spline:

\[
F(s) = f_0 h_{00}(s) + (f_1 - f_0) h_{10}(s) + f_1 h_{01}(s) + (f_2 - f_1) h_{11}(s)
\]

Interpolating splines should improve the adherence of our approximating segments to the image \(\rho(f(t))\), providing smoother results with fewer world-space samples. However, the added cost of maintaining a 4-point stencil and evaluating the spline makes this approach impractical for most 2D transfer functions, compared to simpler chordal parameterization with more samples. The technique begins to be useful when the data itself is extremely noisy and world-space samples are farther apart, such as in the example in Figure 7.

4.5. Scanline Sampling

When \(f\) is quantized to 8-bit or lower precision and the user requires more precise classification, it is useful to employ low-resolution \((256^2)\) piecewise-constant transfer functions. To cheaply and accurately find such peaks in \(f\), we employ a scanline algorithm instead of ray parameterization. We use a digital differential analyzer (DDA), namely Bresenham’s scanline algorithm [Bre65], to scan the chord from \(f_0\) to \(f_1\) in discretized \(\mathbb{R}^M\)-space. This better guarantees that peaks features in \(\rho\) will not be missed. 2D DDA is similar to ray marching (Section 4.3), except we parameterize the distance between pixel centers, and march along either the X or Y axis, whichever is greater, incrementing the differential and terminating when we reach the endpoint on that axis. Instead of the position along the chord, we use the position along the major axis to determine \(s_\rho\) and again interpolate to find \(s_\tau\). This is illustrated in Figure 5(b).

We implemented other scanline methods such as Wu’s algorithm [Wu91] (Figure 5 (c)), which guarantees every pixel between endpoints in \(\rho\) will be scanned. However, this approach was slower than either DDA or point sampling, and did not provide better quality, likely because chordal parameterization itself is only an approximation. We also modified the DDA algorithm to rasterize slabs (Figure 5 (d)), which produces fewer artifacts but was more expensive, and can overestimate the number of peaks present. Overall, we find that ray marching on chordal parameterizations is the best solution for most transfer functions. However, DDA is useful for lower-resolution 2D transfer functions, or when the user does not wish to control \(\Delta s\).

4.6. Integrating and Shading

Since we are finding peaks in between every world-space sample, we do not need to choose a strategy for blending scale-invariant peak samples with scale-variant samples from standard (postclassified) DVR integration, as done in 1D peak finding [KHW*99]. We use our \(t_\rho\) as the root of the isosurface along the world-space ray, and use \(\tilde{R}(t_\rho)\) as the position at which to shade.

Two principal options exist for choosing the gradient of a multifield volume \(f\) when shading:
Figure 6: Comparison of various classification techniques on a close view forward (Λ+) and backward (Λ−) FTLE fields of a combustion dataset, classified using a sharp separable Gaussian 2D transfer function, evaluated analytically (sampled at a discretized resolution of 1024² in (d)). Rendering of frames (a-g) run at 33, 1.2, 35, 22, 32, 18 and 10 fps, respectively.

- Shading multiple data gradients ∇f₀, ∇f₁, etc. separately, using multiple central-differences neighbor stencils.
- Computing the gradient of ∇ρ(f), by classifying f at each point of a single central-difference stencil.

Both approaches are expensive, and are responsible for a significant share of the cost of multifield DVR regardless of whether peak finding is used or not. In our examples we opt for the first approach because ∇f tends to exhibit higher frequency than separate individual gradients.

4.7. Implementation

All presented approaches were implemented in a GPU shader ray caster written in OpenGL and GLSL. To evaluate baseline performance, this renderer is not heavily optimized; it does not employ methods for multiresolution, empty space culling or adaptive sampling. Indeed, applying such techniques to multifield DVR is nontrivial. Simple methods such as precomputing gradients or adaptive sampling could improve performance; however we have opted for the simplicity, flexibility and reproducibility. We note that performance could be greatly improved with such optimizations.

5. Results

Benchmarks were conducted on a 3.0 GHz Intel Xeon and an NVIDIA 285 GTX GPU, at 512² screen resolution unless otherwise stated. General performance can seem relatively slow since more samples are required to render high-frequency data and transfer functions than low frequency ones. However, peak finding is only modestly slower (usually 10-30%) than postclassified ray casting with the same number of world-space samples, and produces significantly better results. To achieve similar quality, postclassification frequently requires 4x-16x greater sampling, effectively making peak finding an order of magnitude faster than standard rendering. In some cases where the transfer function is particularly high frequency, such as the 4D classification example in Section 5.3, peak finding is the only way to find specified features due to numerical limitations of interpolating and sampling from world space.

Generally, we recommend using 1024² 2D textures and the chordal ray marching method with Δt = 2, Δs = 2. There is no major performance difference between peak finding with small and large 2D transfer functions. Aliasing in the transfer function domain is a major source of rendering artifacts; even at 256² it is easy to specify features in transfer function space that yield artifacts when undersampled in world space. We believe it is better to use peak finding to improve classification quality and performance, rather than to identify peaks at given pixel in transfer space. However, multidimensional peak finding with scanline sampling makes this approach feasible if it is desired. For 256² and 512² 2D transfer functions such as the ones we used in our gradient magnitude classification examples (Section 5.2) we find the DDA method is slightly faster and better at finding peaks than ray marching. Not needing to control Δs can be seen as an advantage. For analytically constructed transfer functions, it is usually more efficient to use ray marching than sampling into a texture and applying DDA.

5.1. Quality Comparison

In Figure 6, we compare results of postclassification, separable peak finding, ray marching and DDA with chordal parameterization, and ray marching with interpolating splines. To compare results we use a simple Gaussian analyti-


cal function with sharp peak ridges at (1.1, 12.4, 10.3, 7.5, 12.0 and 8.0 fps, respectively. We use a 512 transfer function (joint histogram overlaid for illustration only), and render at 1024x768. Frames (a-g) run at 8.0, 1.1, 12.4, 10.3, 7.5, 12.0 and 8.0 fps, respectively.

Figure 7: Classification of matter density ($f^1$) and dark matter density ($f^2$) in an Enzo computational astrophysics dataset [NBH^{07}]. We approximate regression lines in the transfer function to illustrate maximal presence of both matter quantities, where red-yellow features illustrate contain primarily dark matter and pink-blue features show normal matter. We use a 512 transfer function (joint histogram overlaid for illustration only), and render at 1024x768. Frames (a-g) run at 8.0, 1.1, 12.4, 10.3, 7.5, 12.0 and 8.0 fps, respectively.

Peak finding exhibits similar behavior with non-separable functions, such as the relatively thin regression line in Figure 1. While lines in transfer space should correspond to isosurfaces in the volume, they are always undersampled by world-space sampling techniques. One disadvantage of scale-invariant volume rendering is that isosurfaces occlude features behind them. Using peak finding we can control how we reproduce surfaces features from a transfer function, better understanding surfaces by adjusting both $\Delta s$ and $\Delta t$, and maintaining interactivity.

In Figure 7, we examine several approaches from Section 4.2 in classifying a subset of an Enzo computational astrophysics dataset. We note that peak finding (c) delivers similar results to postclassification (b) with a 16x higher sampling rate, and at 11x the performance. In panel (d), we show the results of integrating along the chord using a Riemann sum similar to preintegration. This illustrates why scale-invariant volume rendering (peak finding) is a better approach than scale-variant integration when sampling the volume and transfer function separably. Although the differences between e-g are slight, we see that interpolating splines (e) provide slightly less aliased results, DDA (f) detects some features that chordal ray marching omits, and DDA slabs (g) are better still. However, these approaches are slower and not significantly better than chordal ray marching (c); judging by the postclassified reference (b) they can detect false peaks.

5.2. 2D Gradient Magnitude Classification

One drawback of 1D peak finding is that 1D transfer functions provide limited classification of noisy data from CT and biological sources. 2D functions mapping value and inverse gradient magnitude of univariate data offer better classification of material boundaries. Picking surface features in gradient space is an alternative to isosurfacings; but we find that peak finding is still useful in its ability to render specified features more accurately at lower sampling cost.

Figure 8 shows several examples. The backpack and christmas tree are moderate-size CT scans with considerable noise. With the backpack, peak finding clearly reproduces sharp features in the transfer function that are omitted by standard volume rendering at the same base sampling rate. Even when features are not particularly sharp in 2D TF space, as with the christmas tree, peak finding frequently allows us to reproduce equivalent quality at a lower base sam-

submitted to Eurographics/IEEE-VGTC Symposium on Visualization (2011)
Figure 8: 2D classifications of value and inverse gradient magnitude, without (left) and with (right) peak finding. From top left to bottom right, these render at 8.0, 7.1, 5.3, 4.0, 12.3 and 9.8 fps, respectively, using scanline DDA sampling on 512^2 transfer functions and varying ∆t.

Ray marching reduces our search in transfer space to 1D regardless of the dimension of our classification. In Figure 9, we have classified a 4-dimensional CFD combustion simulation [KWCI00], plotting entropy against volume mixture in one 2D transfer function ν(f0, f1), and vorticity and a mixture fraction on another 2D function µ(f2, f3). We map our transfer functions on a subset of joint histograms from each data channel (the grayscale histogram plots shown in the figure are for illustration only, and are not embedded in their respective transfer functions. To create a 4D function, we convolve ν and µ using ρ(f) = ρα(f0, f1, f2, f3) = νµ, then employ an asymptotic interpolation between µ and ν to blend the colors associated with each function. We use the chordal ray marching technique with ∆s = 0.125 to peak-find directly in this 4D function space.

Due to convolution of multiple variables, high frequencies are more common with multidimensional classification. As seen in Figure 9 (top), standard DVR neglects contributions from sharp isolines, and exhibits noise even at a high sampling rate (8 times the voxel Nyquist limit). As usual, peak finding succeeds in detecting more of these features at the same sampling rate. Though some features may appear to be noise, they do not disappear with higher a peak-finding sampling rate, which indicates that they are actual features specified in the transfer function.

An even stronger argument can be made for peak finding with relatively low-frequency transfer functions in higher dimensions. With convolution of 4 variables, even the large block functions shown in the bottom examples of Figure 9 can begin to exhibit high frequencies. Again, these high frequencies go unnoticed without an explicit algorithm for detecting them, and our multidimensional peak finding method excels at reproducing these features.

6. Conclusions and Future Work

Multidimensional peak finding is as good as 1D peak finding qualitatively, and useful for the same goals: rendering noisy volume data and specifying more precise transfer functions. As it samples at peaks on every segment, it is scale-invariant, producing results closer to ground truth than either 1D peak finding or preintegration. We have applied this method to 2D gradient-magnitude classification of scalar volume data, and to direct volume rendering of regression-line features in joint histograms of multifield data. Ray marching reduces peak finding in any dimension to a 1D search, making this method applicable to 3D and higher-dimensional classifications. We have demonstrated this for 4D multifield classification, and higher dimensions are possible. We believe this technique can be a powerful tool for comparative volume visualization.

The main disadvantage of multidimensional peak finding is that it is not needed if classification is sufficiently smooth. However, as we have seen, high frequencies occur even more easily in multidimensional space than in 1D scalar fields. Since isosurfaces cause occlusion, scale-invariant volume rendering with peak finding may not be the best modality for all visualizations. In instances, postclassified rendering could be more useful than the peak-finding. Fortunately, our method provides some control over this behavior via the transfer space sampling rate ∆s. Lastly, unlike in 1D, multidimensional peak finding must sample transfer space between every world-space segment. This is expensive and unnecessary wherever f is monotonic. However, it does ensure scale-invariance, and in practice the cost of peak finding everywhere is small compared to its benefits.
In future work, a comprehensive toolset for modeling precise 2D and higher-dimensional transfer functions is all but necessary. This paper has mostly considered how to better render volume data with multidimensional classification, but determining how to perform multidimensional classification is at least equally important. We are extremely interested in automatic or semi-automatic means of determining regression lines from joint histograms, which could greatly simplify multifield classification. Lastly, to avoid unnecessary peak finding on monotonic ray segments, topological methods might prove useful.

References


submitted to Eurographics/IEEE-VGTC Symposium on Visualization (2011)