CS6170: Topology Guided Volume Exploration

Will Usher, Qi Wu

May 1, 2017

1 Introduction

Volume rendering [Lev88] is a standard technique for directly visualizing univariate or multivariate volumetric data. By assigning colors and opacities to samples in the domain based on univariate or multivariate [KIL*03,KKH02] field values users can classify the data to pick out different features of interest. With these approaches volume rendering can be used to provide an effective overview of the data, or subregions of the data, however; there remain a few challenges to creating effective visualizations of complex datasets.

An effective volume rendering can require a complex transfer function, and while automated techniques have been proposed to generate transfer functions they can have mixed results, requiring further adjustment. Specifying the function can be laborious, requiring a time consuming trial-and-error process, especially in the case of complex multidimensional transfer functions. Additionally, the transfer function classification assumes that field values in the dataset can be directly related to physical properties or features. However, in many datasets different features can share the same or overlapping ranges of field values, but have distinct spatial distributions. When restricted to specifying a global transfer function per-scalar field these features can be challenging, if not impossible, to classify separately. For example, when an interesting feature is surrounded by noise with similar field values, applying a single transfer function to the field cannot hide the noise without also effecting the visibility of the feature. For a recent summary of the state-of-the-art in transfer function specification and generation we refer to a recent survey by Ljung et al. [LKG*16].

Different features in the volume can also be classified based on their topological structure, providing a method to disambiguate cases where features share similar field values. In this paper, we report on our work implementing a topology guided volume exploration tool, similar to that proposed by Weber et al. [WDC*07]. Our tool provides users the ability to classify volumetric data both from a topological and traditional transfer function perspective. By tying transfer function classification with topological segmentation via the Contour, Join or Split tree users can easily remove background noise and classify features.

2 Related Work

For univariate data previous work has successfully employed Contour trees, Reeb graphs, and the Morse-Smale complex in tracing the evolution of features in volumetric datasets for classification and segmentation [CSvdP10, TTNF04,TTF04,WDC*07,XTY*11,GNPH07,GKK*12,CS03]. In the multivariate setting structures such as Fiber Surfaces [CGT*15], Jacobi Sets [EH02] and Joint Contour Nets [CD14] have been applied for data representation, filtering and segmentation.

The Contour tree, introduced by Boyell and Ruston [BR63], effectively counts the connected components of a level set on the field and has proven useful for exploring volumetric datasets. Carr et al. [CSA03] introduced the Join and Split tree which they use to quickly compute the Contour tree, in our system we allow for viewing any of these three trees as one may be better at selecting some desired feature. In the context of volumes this can be used to pick out disjoint features in the data, and for example, guide in selecting or filtering isosurfaces of the data [CS03].

Our project is similar to the work of Weber et al. [WDC*07], who develop a topology-controlled volume rendering system. In their system they use the Contour tree to segment the data set, after which the user is able to assign different transfer functions to the segments of the data corresponding to branches in the tree. Compared to their system we did not implement the (likely needed) algorithm for finding the branch which a sample in a voxel corresponds too. Besides this missing feature our system is quite similar, after computing the tree and simplifying based on persistence the user can select and view individual segments and apply different transfer functions to the selections. For computing the topological features required in our tool, the Contour, Join or Split tree and persistence curves and diagrams, we use the Topology ToolKit (TTK) by Tierny et al. [TFL*17].

3 Contribution

We implemented a user-friendly tool for analyzing volumetric data which combines topological data analysis techniques with traditional transfer function driven classification (Figure 1). By employing the Contour, Split or Join tree, optionally simplified to some persistence threshold, we can compute a meaningful segmentation of 3D scalar fields. The persistence simplification is important for analysis of larger volumes or those with many repeated field values (e.g. background), as it can be used to clean up extraneous critical points introduced by the simulation of simplicity [EM90], which would otherwise produce an extreme over-segmentation. The segmentation of the data computed through the Contour, Split or Join tree correspond to distinct features in the data, see Figures 2 to 4 for examples. By classifying

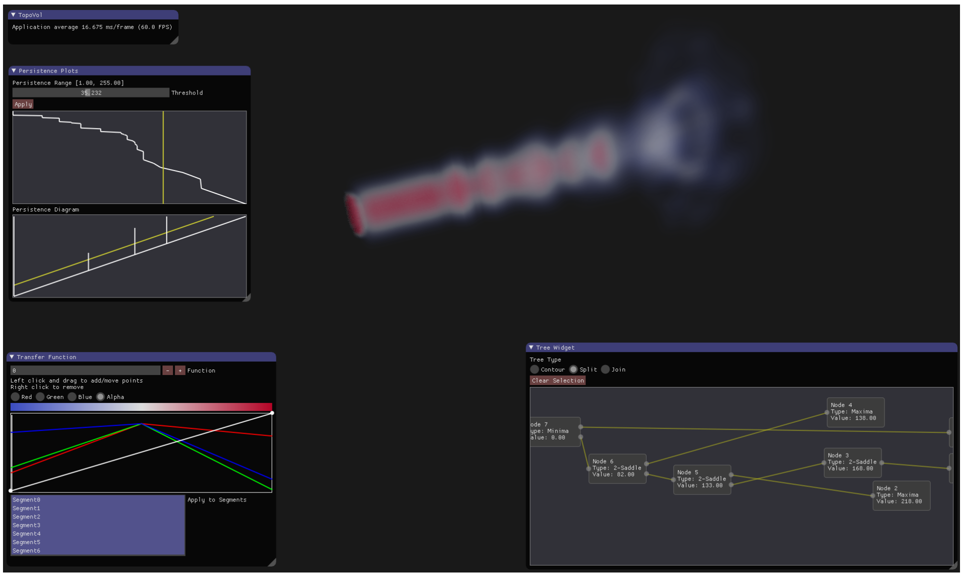


Figure 1: Examining the *Fuel* dataset. The Persistence Plots panel displays the persistence curve and diagram, allowing the user to threshold critical points in the tree by adjusting the slider. This moves the yellow line in each plot to reflect the location of the threshold. The Transfer Function panel allows for changing the transfer function and adding new ones to be applied only to certain segments, selected in the "Apply to Segments" list in the panel. The Tree Widget displays the Contour, Split or Join tree where the user can select branches of the tree to filter the volume data being shown to just voxels on those branches. The histogram in the transfer function panel is updated to show just the values in the selected segments.

these features individually or in meaningful groups with different transfer functions, our tool can be used to hide background noise and view features which would otherwise be occluded.

4 Technical Details

Our project, TopoVol, is an interactive application for topology guided volume exploration which leverages TTK for computing topological structures and a custom rendering system to allow for interactive exploration. We also provide a user interface built using ImGui. TopoVol is based on the idea that the Contour, Split and Join trees capture interesting regions of the data which the user may want to classify independently, in ways not possible with a 1D transfer function. Further, to avoid overwhelming the user with low-persistence features, e.g. from noise or over segmentation, we provide an interface for simplifying the computed tree based on the persistence of the critical points. An overview of our tool being used to study the *Fuel* dataset is shown in Figure 1.

While we use TTK and VTK [SML06] to perform the computations and threshold operations selected by the user, we employ a custom rendering system to allow for interactively selecting branches (segments) of the tree and specifying segment masks for transfer functions. This system is built on a standard volume raycaster in GLSL, e.g. as described by Hadwiger et al. [HLSR08], with modifications to skip samples if they are in a segment which is not selected. This is done by uploading an additional volume containing the segment id for each voxel to the GPU, as we raymarch the volume we first check if this segment is active by getting its segment id from this volume and checking against a shader storage buffer (SSBO) storing a flag for the active segments. To allow for per-segment or segment group transfer function specification we lookup which function is assigned to the voxel's segment in the same SSBO when determining how to classify a sample.

5 Results

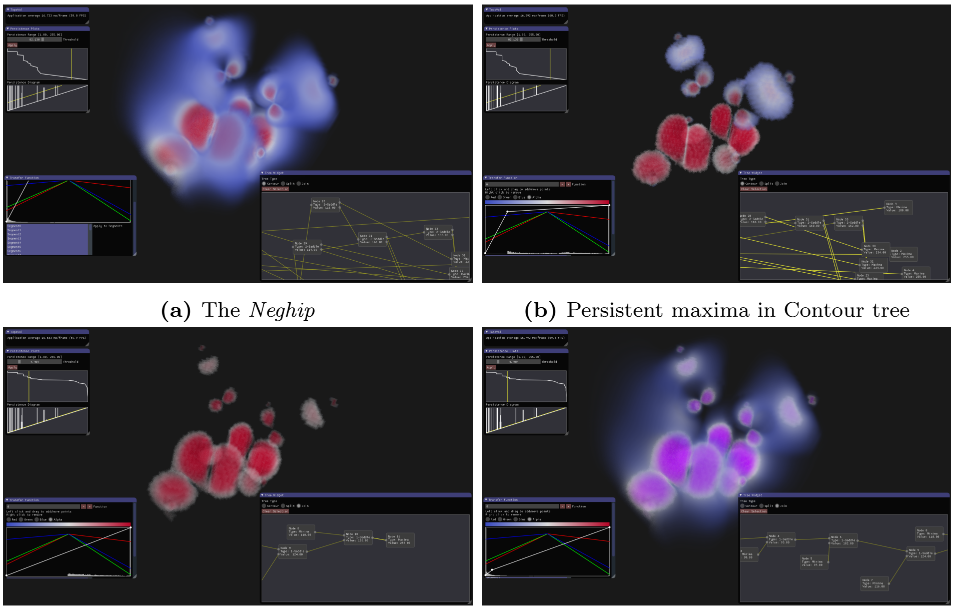
To validate this topology driven approach to volume exploration and segmentation we use our tool to study three different datasets. Two, *Neghip* and *Nucleon*, are from simulations and are smooth and noise free, while the *Tooth* is an MRI scan of a human tooth containing background noise. Both the *Nucleon* and *Tooth* contain either other features or noise at similar or the same range of values as interesting features, making them challenging to classify well with a single 1D transfer function.

5.1 Neghip Analysis

The *Neghip* is a 64^3 volume from a simulation of the spatial probability distribution of electrons in a high potential protein molecule. The analysis session is shown in Figure 2, where the user makes use of both the Contour and Join tree to pick out features of the data. Here the Contour tree is relatively complex even at high thresholds making it time consuming to pick out each branch leading to a maxima (Figure 2b). However, by switching to the Join tree we find a single segment contains all the peaks of the data, as the dataset merges into this single segment as the function value increases (Figure 2c). By applying a different transfer function to this segment we can separately classify the peaks of the field from the lower values (Figure 2d).

5.2 Nucleon Analysis

The *Nucleon* dataset is a 41^3 volume from a simulation of the probability distribution of a nucleon in an ^{16}O nucleus. The analysis session can be seen in Figure 3. The *Nucleon* data contains two minima within shell-like structures of higher values, and eventually a minima valued outer shell and empty space section. This makes extracting the two minima inside the data challenging with a single 1D transfer function, when trying to hide the background or tweak the outer shell we also effect the classification of



(a) The *Neghip* **(b)** Persistent maxima in Contour tree

(c) Segment 10 of the Join tree **(d)** Classifying segment 10 separately

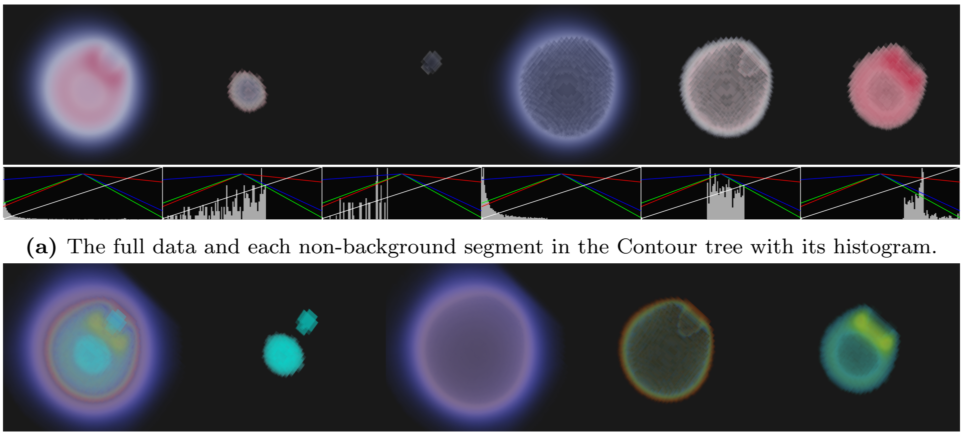
Figure 2: Analysis session of the *Neghip* dataset using our tool. The user begins by viewing the Contour tree at a high threshold in (a) and then selects just the maxima in this tree (b). When inspecting the Join tree they find segment 10 in the tree corresponds to the peaks in the data (c) and apply a different transfer function to the segment (d).

the minima. Similarly, the larger minima feature has some overlap with the middle shell (segment 4, Figure 3a), making the features difficult to separate.

In the Contour tree for the *Nucleon* we find five branches, as shown in Figure 3a. Branches one and two segment out the inner minima, while three is the outermost shell, four the middle shell and five the innermost shell. This precisely segments the volume into its interesting component features, and by separately classifying these components we can build a more useful visualization of the data (Figure 3b). In this case we chose to classify the two minima with a single transfer function to pair them together visually. The ability to classify groups of segments with a single function is valuable to pair disjoint features which belong to the same conceptual feature of interest, here the inner minima of the field.

5.3 Tooth Analysis

The *Tooth* is a $103 \times 94 \times 161$ MRI scan of a human tooth. It is challenging to classify not only due to the noise in the data, but because the noise and outside air region around the tooth has the same value as an inner air pocket inside the tooth. Both of these regions are open space, and thus on the scan appear at the same value range. The tooth analysis session is shown in Figure 4. Due to the high noise in the data we begin by computing the



(a) The full data and each non-background segment in the Contour tree with its histogram.

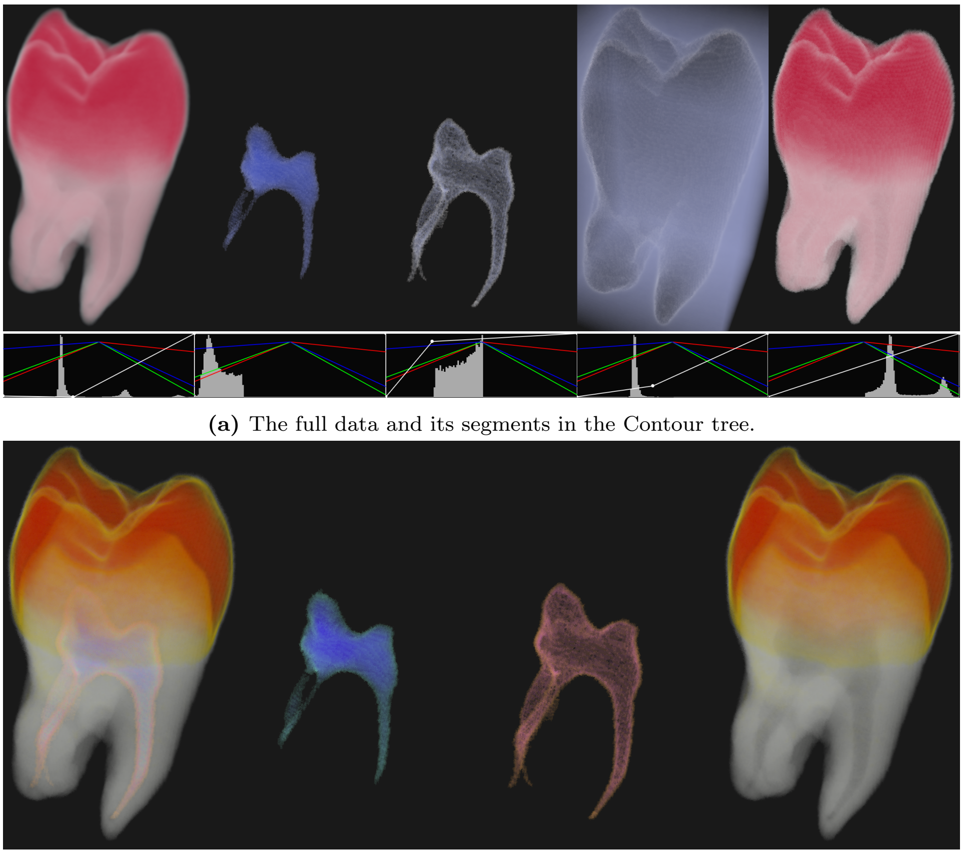
(b) After individual classification has been applied. Segments 1 and 2 are classified together.

Figure 3: Analysis session on the *Nucleon*. The histograms for each segment in (a) show that the two minima, segments 1 and 2, overlap the value range of segments 3 and 4. With per-segment or segment group transfer functions we can classify the individual features of the data we are interested in, (b).

Contour tree and thresholding at a relatively high value to simplify it to just a few persistent features.

At this persistence threshold there is some over-segmentation of the data, specifically the inner air pocket (panel 2, Figure 4a) in fact consists of a selection of four different branches of the tree. However, moving to a high enough threshold to simplify these to a single feature results in merging them with the inner pocket boundary (panel 3, Figure 4a) which may not be desirable. The background air region outside the tooth (panel 4, Figure 4a) appears at the same value as the inner air pockets, making it challenging if not impossible to remove the background without losing the inner features with a global transfer function.

In the Contour tree the background appears as a separate segment from both the bone and inner air pockets, making it trivial to remove by applying a completely transparent transfer function, or simply hiding the segment. The final rendering produced in our tool after separately classifying the interior pockets, bone, and removing the background segment is shown in Figure 4b. This segmentation and classification effectively conveys the value ranges of the individual segments while ensuring each is visible and understandable in the combined rendering. Moreover, the background region is removed entirely without losing any features of interest in the interior. For a demonstration of the entire workflow in our tool analyzing the *Tooth* please see the accompanying video on YouTube.



(a) The full data and its segments in the Contour tree.

(b) After individual classification and filtering to remove background and highlight features.

Figure 4: Analysis session on the *Tooth*. By working on segments of the Contour tree it is possible to entirely hide the background (panel 4, a) without effecting features at similar value ranges (panels 2-3, a). Panel 2 (a) consists of four different segments which are classified together to select the inner air pocket of the tooth. The air pocket, its boundary and the bone are classified separately to create the final visualization (b).

6 Conclusions

Our tool provides users the ability to create effective visualizations allowing them to better explore and understand univariate volumetric datasets. We achieve this by combining topological segmentation and simplification with familiar transfer function classification in an intuitive interface. Users can easily view individual or groups of segments corresponding to topological features in the volume and classify them independently or together, allowing to quickly remove noise and prevent occlusion of different features with similar value ranges. As demonstrated in Section 5 our tool provides a compelling approach to visualizing volumetric datasets. Our tool is available open-source on Github.

A key limitation of our tool is that we did not implement the algorithm for computing the branch a sample lies on described by Weber et al. [WDC*07], instead we pick the branch assignment based on the voxel's branch ID stored in the corresponding ID volume. As a result we do not represent the segment boundaries exactly, instead of a continuous border between segments, potentially bisecting voxels, we display a discrete boundary which is restricted to voxel faces. This limitation is easiest to see in the *Nucleon* analysis, where the individual segments have clear discrete boundaries instead of a smooth segmentation (Figure 3).

Acknowledgements

The datasets used in evaluating our tool were acquired from the Open SciVis Datasets collection.

References

[BR63] Boyell R. L., Ruston H.: Hybrid techniques for real-time radar simulation. In *Proceedings of the November 12-14, 1963, Fall Joint Computer Conference* (New York, NY, USA, 1963), AFIPS '63 (Fall), ACM, pp. 445–458.

[CD14] Carr H., Duke D.: Joint Contour Nets. *IEEE Transactions on Visualization and Computer Graphics 20*, 8 (Aug. 2014), 1100–1113.

[CGT*15] Carr H., Geng Z., Tierny J., Chattopadhyay A., Knoll A.: Fiber Surfaces: Generalizing Isosurfaces to Bivariate Data. *Computer Graphics Forum 34*, 3 (June 2015), 241–250.

[CS03] Carr H., Snoeyink J.: Path Seeds and Flexible Isosurfaces Using Topology for Exploratory Visualization. In *Eurographics / IEEE VGTC Symposium on Visualization* (2003), The Eurographics Association.

[CSA03] Carr H., Snoeyink J., Axen U.: Computing contour trees in all dimensions. *Computational Geometry 24* (2003), 75–94.

[CSvdP10] Carr H., Snoeyink J., van de Panne M.: Flexible isosurfaces: Simplifying and displaying scalar topology using the contour tree. *Computational Geometry 43*, 1 (Jan. 2010), 42–58.

[EH02] Edelsbrunner H., Harer J.: Jacobi sets of multiple Morse functions. *Foundations of Computational Mathematics, Minneapolis* (2002), 37–57.

[EM90] Edelsbrunner H., Mcke E. P.: Simulation of simplicity: a technique to cope with degenerate cases in geometric algorithms. *ACM Transactions on Graphics (TOG) 9*, 1 (1990), 66–104.

[GKK*12] Gyulassy A., Kotava N., Kim M., Hansen C. D., Hagen H., Pascucci V.: Direct Feature Visualization Using Morse-Smale Complexes. *IEEE Transactions on Visualization and Computer Graphics 18*, 9 (Sept. 2012), 1549–1562.

[GNPH07] Gyulassy A., Natarajan V., Pascucci V., Hamann B.: Efficient computation of Morse-Smale complexes for three-dimensional scalar functions. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (2007), 1440–1447.

[HLSR08] Hadwiger M., Ljung P., Salama C. R., Ropinski T.: Advanced illumination techniques for GPU volume raycasting. In *ACM Siggraph Asia 2008 Courses* (2008), ACM, p. 1.

[KIL*03] Kniss J., Ikits M., Lefohn A., Hansen C., Praun E.: Gaussian transfer functions for multi-field volume visualization. In *Visualization, 2003. VIS 2003. IEEE* (2003), IEEE, pp. 497–504.

[KKH02] Kniss J., Kindlmann G., Hansen C.: Multidimensional transfer functions for interactive volume rendering. *IEEE Transactions on visualization and computer graphics 8*, 3 (2002), 270–285.

[Lev88] Levoy M.: Display of Surfaces from Volume Data. *IEEE Computer Graphics and Applications 8*, 3 (1988), 29–37.

[LKG*16] Ljung P., Krger J., Groller E., Hadwiger M., Hansen C. D., Ynnerman A.: State of the Art in Transfer Functions for Direct Volume Rendering. *Computer Graphics Forum 35*, 3 (June 2016), 669–691.

[SML06] Schroeder W., Martin K., Lorensen B.: *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*, fourth ed. Kitware, Inc., 2006.

[TFL*17] Tierny J., Favelier G., Levine J., Gueunet C., Michaux M.: *The Topology ToolKit*. Tech. rep., 2017.

[TTF04] Takahashi S., Takeshima Y., Fujishiro I.: Topological volume skeletonization and its application to transfer function design. *Graphical Models 66*, 1 (2004), 24 – 49.

[TTNF04] Takahashi S., Takeshima Y., Nielson G. M., Fujishiro I.: Topological volume skeletonization using adaptive tetrahedralization. In *Geometric Modeling and Processing, 2004. Proceedings* (2004), pp. 227–236.

[WDC*07] Weber G. H., Dillard S. E., Carr H., Pascucci V., Hamann B.: Topology-Controlled Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics 13*, 2 (March 2007), 330–341.

[XTY*11] Xiang D., Tian J., Yang F., Yang Q., Zhang X., Li Q., Liu X.: Skeleton Cuts – An Efficient Segmentation Method for Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics 17*, 9 (Sept. 2011), 1295–1306.