

Feature Exploration using Local Frequency Distributions in Computed Tomography Data

M. Falk¹ , P. Ljung¹ , C. Lundström² , A. Ynnerman¹ , and I. Hotz¹ 

¹ Scientific Visualization Group, Linköping University, Sweden

² Center for Medical Image Science and Visualization (CMIV), Linköping University, Sweden, and Sectra AB

Abstract

Frequency distributions (FD) are an important instrument when analyzing and investigating scientific data. In volumetric visualization, for example, frequency distributions visualized as histograms, often assist the user in the process of designing transfer function (TF) primitives. Yet a single point in the distribution can correspond to multiple features in the data, particularly in low-dimensional TFs that dominate time-critical domains such as health care. In this paper, we propose contributions to the area of medical volume data exploration, in particular Computed Tomography (CT) data, based on the decomposition of local frequency distributions (LFD). By considering the local neighborhood utilizing LFDs we can incorporate a measure for neighborhood similarity to differentiate features thereby enhancing the classification abilities of existing methods. This also allows us to link the attribute space of the histogram with the spatial properties of the data to improve the user experience and simplify the exploration step. We propose three approaches for data exploration which we illustrate with several visualization cases highlighting distinct features that are not identifiable when considering only the global frequency distribution. We demonstrate the power of the method on selected datasets.

CCS Concepts

- *Human-centered computing* → *Scientific visualization; Visualization techniques;*
- *Applied computing* → *Life and medical sciences;*

1. Introduction

Volume data from medical imaging modalities are at the core of medical research and diagnostics. It contains a wealth of information and is used for a large variety of applications entailing measurements, diagnosis, teaching, and communication. Many of the related tasks are based on the classification of the volume into features and regions of interest that shall be explored in detail. For this purpose, Transfer Functions (TFs) presets are often utilized in clinical practice. Even though in some cases features can directly be described by specific data values or local statistics this is not always possible. Thus, other intuitive methods enabling effective differentiation of features become essential without introducing too much interaction complexity.

Due to its conceptual simplicity, the 1D TF is an attractive starting point. The TF design process itself is often supported by showing a histogram, i.e., a visual representation of the frequency distribution (FD) of the data. However, finding a good transfer function can still be a challenging task even for experienced users. Dynamic previews can provide some support traversing the TF space [JFY16] though their primary aim is to support the exploration of the dataset. In addition, the possibilities using a 1D TF are limited for a variety of reasons including:

Variability of data, meaning it is not always possible to uniquely assign data values to a tissue due to variations in the imaging modalities. Its values might even vary locally within the same dataset.

Separability of different tissues, meaning different tissues might have similar densities and thus the same data values occluding each other in the histogram.

Locality of features, meaning that features at different locations in the data should be individually accessible.

Some of these challenges can be addressed by increasing the dimensionality of the transfer function design thus providing more descriptive attributes. However, this also comes with an increase in complexity of the transfer function editor while not solving all of these limitations.

In this work, we are addressing these challenges by facilitating the characteristics of neighborhood statistics as local frequency distributions (LFDs) to support the domain scientist in interactive feature selection and TF design. In contrast to regular frequency distributions, LFDs are only defined on small localized regions instead of the entire dataset. Local statistic distributions have distinguishing power between features not found at the global level. They can also be effective in describing relevant domain knowl-

edge, as demonstrated by Lundström et al. [LLY06]. We are going one step further and utilize LFDs in an interactive framework. Our approach proposes automatically identified features to the user by utilizing a clustering of LFDs. This enables high-level feature-based exploration through interaction with a hierarchy of clusters in a linked view environment containing both attribute space and volume-rendered images. It also supports manual identification of local target features using a dense LFD plot and queries for similar features in the volume.

The contributions of our work presented in this paper can be summarized as follows:

- Volume exploration based on neighborhood similarity using LFDs for both classification and TF design.
- Introducing an effective LFD similarity measure for clustering.
- Automatic and interactive feature definitions based on LFDs appropriate for data queries.

The results are manifested in three data exploration approaches: direct selections of representatives and outliers, top-down LFD cluster traversal, and bottom-up similarity selection. We demonstrate the visualization of differentiation capabilities of all three approaches on several medical examples.

2. Related Work

Transfer function design for volumetric data has for quite some time been, and still is, an active research area ranging from 1D scalar TFs to multidimensional TFs in various spaces [KKH02, FHL^{*}17]. For example, Correa and Ma [CM08] define transfer functions based on the physical size of the features. For more information on general TF design in the context of volume rendering, we refer the interested reader to the overview by Ljung et al. [LKG^{*}16]. The idea of using histogram features has also been a research topic for decades. For example, color histograms have been used for object identification by considering parts of an FD as features and matching those with a database of models [SB92].

A core concept in this paper is to use properties of local neighborhoods to assist visual exploration. This idea has been exploited in several previous research efforts. One way is to employ neighborhood analysis to guide the design of a traditional TF. The objective to construct TFs by detecting salient intensity ranges was shared both by the Partial Range Histogram method [LLY06], which subdivides the global histogram according to neighborhood content, and the α -histogram method [LYL^{*}06] where the global histogram is modulated to emphasize peaks of spatially coherent features. Spatial conditioning on LFDs can be used for emphasizing user-defined regions of interest like the iodine uptake in the vicinity of the liver [LLL^{*}10]. LFDs obtained from small spatio-temporal neighborhoods facilitate the visual analysis of contrast-enhanced ultrasound data as well as the extraction of perfusion parameters [ANGH11]. Röttger et al. [RBS05] proposed a TF design approach where an intensity-gradient histogram is divided into distinct features by connecting data points with a close spatial connection. Lan et al. [LWS^{*}17] approach the problem that different structures with similar attributes occupy the same region in TF space by integrating spatial connectivity computation of the boundaries into

the TF design. Another idea for 2D TFs was presented by Tappenbeck et al. [TPD06] which uses the physical distance to pre-defined target structures as second dimension. Duta et al. [DS16] have proposed a distribution-driven method for feature extraction and tracking in time-dependent data without providing an explicit feature definition. Local neighborhood information has also been used in combination with machine learning in the context of volume classification [TLM05]. By considering the volumetric shapes of local neighborhoods, graph-based approaches provide a TF design which allows to selectively render objects and depict adjacent materials in a volume [SAK19].

Another use of neighborhood properties, more closely related to the focus of this paper, is to explicitly derive additional attributes and create a multi-dimensional TF. The typical example, although with very small neighborhoods, is to add the gradient magnitude [KKH02]. Lundström et al. [LLY06] introduced another class of local histogram attributes based on intensity range footprints. By growing voxel neighborhoods, it is possible to compute statistical moments and use the resulting moment curves directly for TF design [PHBG09]. Haidacher et al. [HPB^{*}10] proposed TFs based on mean and standard deviation of adaptively defined voxel neighborhoods. The size of coherent neighborhoods has also been used as a TF attribute dimension [CM08].

Our current work goes beyond these approaches. The main novelties are that full LFDs are used in the TF classification as well as that the link between attribute space and spatial location in the dataset is always retained. Local histograms have also been applied in image processing, for example by Liu and Wang [LW06] who utilize local spectral histograms for segmenting 2D images.

Our proposed method is based on clustering of neighborhoods, creating a hierarchy based on LFD similarity. Previous work has also employed clustering for visual exploration, but instead grouped individual data points based on their attributes.

An early example of clustering-based volume exploration is the work by Tzeng and Ma [TM04] providing a visual user interface for interacting with the cluster space and assigning materials. The method of Ip et al. [IVJ12] creates clusters by direct segmentation in the intensity-gradient histogram image. Hierarchical clustering has also been used for visualization of multivariate multidimensional data [LvLRR08, VLL09], including a combination with volume rendering by means of an opacity mask [DVLL11]. Cai et al. [CNC017] propose a semi-automatic TF design with a two-level clustering approach which uses self-organizing maps (SOM) in a high dimensional feature space. Based on the results of the two-level clustering, they propose a top-down volume exploration guided by an informative volume overview of interesting structures that also can be modified by the user.

LFD processing can be demanding due to the size and complexity of the data. To tackle this challenge, a simplifying decomposition of the LFDs can be employed. An established approach to such histogram analysis is to adopt a Gaussian mixture model for which the best fit is derived [PB91, CFC02]. An alternative way to achieve high-performance processing of LFDs was proposed by Lee and Shen [LS13]. In their method LFDs are encoded in a compressed wavelet representation and queried through integral histograms, an extension of summed area tables.

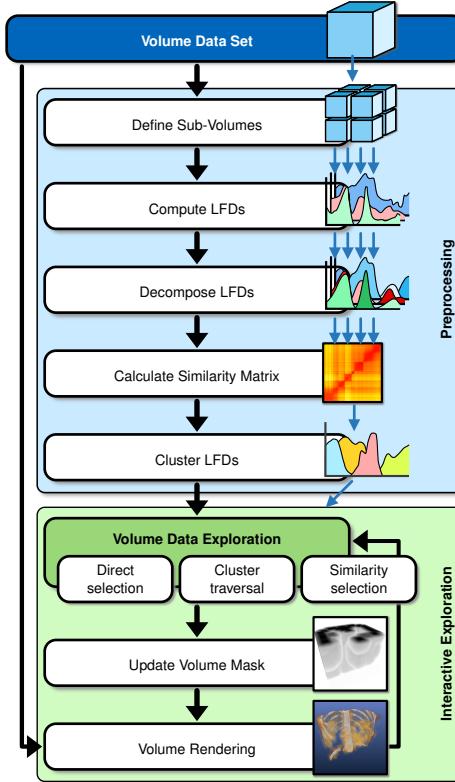


Figure 1: Workflow of our approach for LFD decomposition and volume exploration.

To compare multiple histograms, a wide selection of different metrics is available. Commonly used metrics include the Bhattacharyya distance [Bha43], the χ^2 distance [Coc52], and the Kullback-Leibler distance [Kul59]. The similarity measure presented by Rezk-Salama et al. [RSHSG00] relies on non-linear time warping for aligning two normalized histograms. Support-vector machines have also been used for measuring differences between histograms with the purpose of image classification [CHV99]. Cao and Petzold [CP06] proposed a density distance in the context of stochastic simulations, which is also used in our work. They compare their approach with the Kolmogorov distance [Kol33, Kol41] and show that both can equally be used to measure accuracy with respect to histograms.

3. Our Approach

The algorithm of our proposed LFD decomposition and volume exploration is shown in Figure 1. The method consists of a preprocessing step and interactive volume exploration.

In the preprocessing stage, the volume dataset is divided into multiple sub-volumes of similar size. A local FD is computed for each of these sub-volumes and subsequently decomposed into a set of feature LFDs. Each feature distribution is stored with information on its origin, i.e., the sub-volume. Next, we create a similarity matrix by comparing all features from all sub-volumes. The simi-

larity matrix is then used as input to a hierarchical agglomerative clustering whose results are used in the second, interactive stage.

We visualize the clusters in attribute space either as individual histograms color-coded according to cluster (Figure 5) or in a more abstract hull representation (e.g., Figure 3, top). Three interaction modes are available for exploring the attribute space by selecting clusters or individual feature distributions. A selection is immediately applied to the volume rendering in a second view. Here, we use the explicitly stored connections between feature distributions and sub-volumes to create a masking volume based on the current selection. The volume mask provides an effective way to highlight regions with a locally-restricted TF or, alternatively, to desaturate non-selected regions and reduce their opacity. The volume rendering is performed with a state-of-the-art ray casting algorithm, modified to consider the volume mask and the optional overlay TF. To reduce brickling artifacts, we apply a distance transform on the masking volume to yield a smooth transition between selected and non-selected sub-volumes.

3.1. LFD Computation for Sub-Volumes

Frequency distributions (FD), often visualized as histograms, are obtained by binning the underlying data and counting the number of occurrences, i.e., voxels matching the respective bins. Likewise, local frequency distributions (LFD) are being computed only for subsets of the data. In this work, before we compute the LFDs, the dataset is divided into sub-volumes of a block size given in physical lengths, which are mapped to volume space depending on the volume resolution. This yields distinct local regions or blocks. The block size defines the scale of interest and, hence, determines the range of local statistics. Larger block sizes will emphasize more global features whereas small sub-volumes are most likely to enhance noise occurring in the data. However, choosing an exact block size only plays a minor role since we focus on individual features in the LFD, which are similar for slight variations in block size.

A 3D scalar field, or scalar volume, is a map s which maps a position x onto a scalar value

$$s : \Omega \longrightarrow \mathbb{R}, \quad x \longmapsto s(x), \quad (1)$$

where $\Omega \subset \mathbb{R}^3$ denotes the 3D Euclidean space. For CT datasets, we can assume that the discrete scalar values $s(x)$, i.e., voxel values, are in the range $[1, m]$. The FD H for s is then given by

$$H = \{H(i) : i \in [1, m]\}, \quad (2)$$

where

$$H(i) = |\{s(x) = i : x \in \Omega\}|. \quad (3)$$

That is, the original data is summarized by the bins $H(i)$, where each bin represents the number of voxels containing the scalar value i , i.e., the statistical frequency. Plotting H as bar plot yields the respective histogram.

The LFDs are computed for each sub-volume and are normalized with respect to the number of enclosed voxels. This allows us to interpret the distribution values directly as volume percentages which, in turn, can directly be associated with feature size.

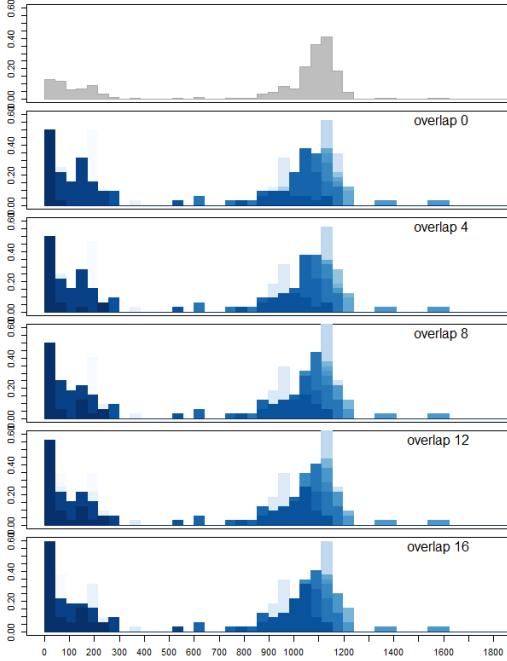


Figure 2: Frequency distributions for varying overlaps of 1D blocks taken from the lower chest of the Upper Torso dataset. The shapes of the LFDs are only marginally affected by the different overlaps. The global histogram is shown on top and the block size is 32. The blue-white coloring refers to the block index of the LFD.

For the shape of sub-volumes we use a regular cuboid although other shapes for local sampling are possible (see [LYL*06] for a brief discussion). Although overlapping blocks can be computed with a small overhead using for example a sliding window approach, features are well represented even if there is no overlap between blocks. Figure 2 shows the similarity between frequency distributions for different overlaps. We therefore chose non-overlapping blocks since there were no significant differences visible with respect to features in the LFDs. Artifacts introduced by large block sizes could potentially be reduced, but not entirely avoided, using an overlap at the cost of additional LFDs.

3.1.1. LFD Decomposition

The goal of this step is to decompose LFDs into multiple parts where each part represents a single feature. Thus, it is possible to cluster similar features, i.e., close-by peaks. This decomposition also ensures that the feature detection in attribute space is more or less independent of the block size. First, it ensures that the major n peaks of a single LFD are considered individually in the clustering process. Secondly, assuming certain tissue characteristics, choosing a smaller block size should more or less result in similar peaks if the tissue is still dominant.

A smooth FD is identical to a mixture of Gaussian distributions [ZHPZ96]. We can use this property to decompose each local FD into independent feature distributions. By applying a low-pass filter like a Gaussian window, we obtain a smoothed FD. For the

decomposition of the LFD we apply the algorithm proposed by Chang et al. [CFC02]. First, local minima and maxima are identified as initial guesses for the Gaussian distributions. An optimal interval window is then determined for each maxima for which the proportion between the windowed part of the distribution and the entire FD is computed. This proportion is used to update the boundaries of the intervals, i.e., the most likely support of the respective Gaussian. In a last step, the parameters for the individual Gaussians are determined and then used to create Gaussian-shaped LFDs.

We have conducted tests using parts of the original FD data for each feature while zeroing out everything outside the feature's support. However, the estimated Gaussian distributions for features yield better results in the subsequent clustering since zeroing parts of the LFDs might lead to sudden jumps in the histogram.

3.1.2. FD Similarity Measures

Frequency distributions sharing a similar shape are characterized by a high congruency and are, thus, likely to share the same features. This is in particular the case since we decompose the LFDs into individual features. To be able to cluster the feature LFDs, we need to define a similarity measure for comparing frequency distributions. Since we decompose the LFDs into Gaussian distributions, the Gaussian parameters lend themselves to be the ideal candidates. However, depending on the chosen clustering method, LFDs are recombined or averaged resulting in a Gaussian mixture model. We therefore decided to use the LFDs directly, which also allows for computing similarities without the Gaussian decomposition.

By accumulating the pair-wise distance between two FDs we obtain a density distance metric as described by Cao and Petzold [CP06]. The distance between two distributions H_1 and H_2 is then given by

$$D(H_1, H_2) = \sum_1^m |H_1(i) - H_2(i)|. \quad (4)$$

We also include the Bhattacharyya distance [Bha43] for comparison, but in our application scenarios the differences were hardly noticeable.

In addition, other properties of the LFD can be incorporated into the similarity measure for further adjustments. The locality of features regarding certain density values is emphasized by utilizing the L_1 distance between features in attribute space. Likewise, the Euclidean distance between the centers of sub-volumes is used to describe spatial coherence.

3.2. Hierarchical Clustering

We apply an agglomerative, hierarchical clustering to the LFDs based on their similarity. The hierarchical clustering has several advantages over commonly applied methods like K-means clustering. The hierarchy can be traversed without recomputation or clustering the subtree and it does not require an initial guess for the number of clusters. One of the downsides is the increased memory footprint which becomes the limiting factor with respect to the number of LFDs. However, this did not pose any problems in our tests based on the chosen datasets and block sizes but certainly has to be considered.

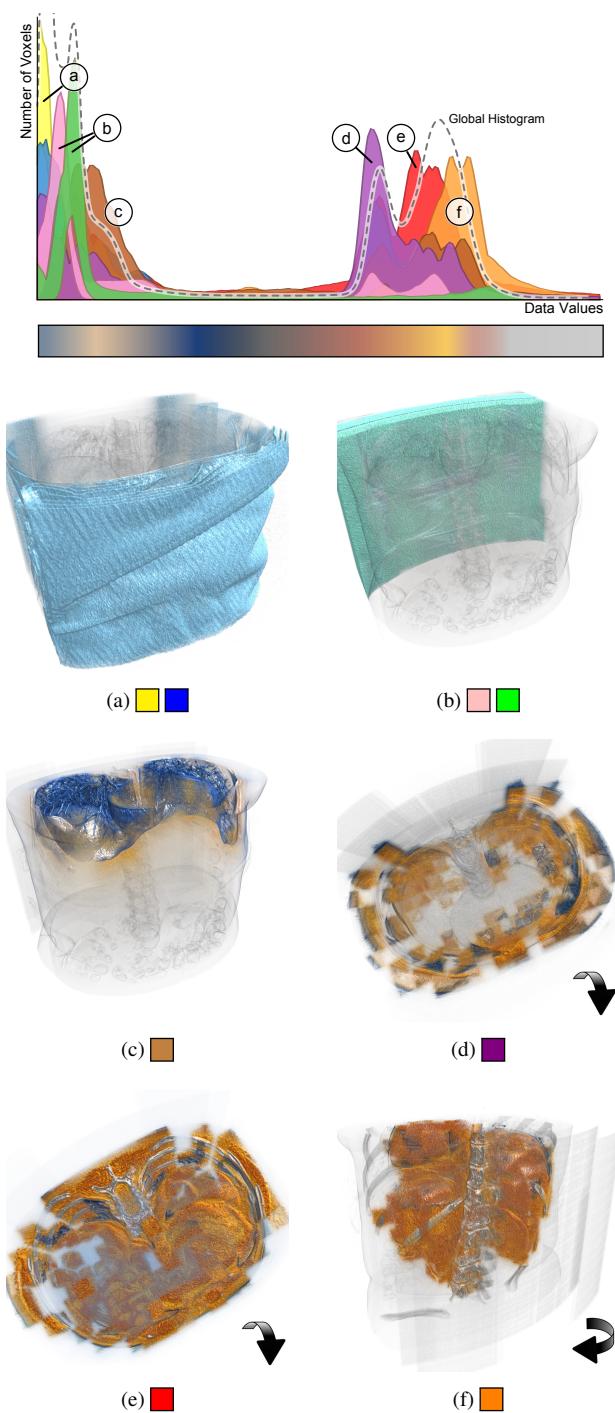


Figure 3: Clustering example for a CT dataset of the upper torso and utilized TF. The initial cluster results are color-coded and relate to different features. (a) blanket. (b) examination table. (c) lung. (d) sub-volumes filled with epidermal tissue and air. (e) tissue, fat, bone, and liver tumor. (f) internal organs and spinal cord. Volume parts corresponding to the individual clusters are highlighted.

A similarity matrix is computed using all LFDs and the chosen distance metric, which is then used as input for the clustering. We experienced the best clustering results by applying Ward's method [WJ63]. The result is a hierarchical tree T , where the height of each node corresponds to the similarity of the two subtrees. This is exploited in Section 4.3 for masking similar regions within the volume. For each cluster node within the hierarchy we compute one representative FD. When interacting with the LFDs, the visual complexity can be reduced by drawing only the representative FD of the cluster instead of all contributing distributions.

Figure 3 shows the initial clustering results for a CT scan of a torso. On top, the global histogram is indicated with a dashed line besides the LFDs which are colored according to the clustering results. Images (a) to (f) depict the respective cluster-based masking of the blocks in the volume. In this example, the first eight clusters already reveal more features contained in the data than the global histogram. The block artifacts visible in this figure can be suppressed by smoothing out the block boundaries as described in the next section. Here, they illustrate the spatial locality of the individual clusters.

3.3. Volume Rendering

Dividing the volume into sub-volumes and using these during volume rendering to highlight features will lead to block artifacts. Smoothing the masking volume by applying a distance transform with a user-defined cut-off reduces these artifacts (Figure 4(b)-(e), bottom row). It is important to keep in mind that that our approach is primarily an exploration tool and that it is utilizing state-of-the-art rendering techniques to support exploratory interaction.

Masked-out volume parts receive a higher transparency and are slightly desaturated to emphasize the selection. Alternatively, a custom, localized TF can be applied to the selection (Figures 5 and 10). To account for the volume masking and local TFs only a minor adjustment is required in a regular volume ray casting algorithm. In particular, masking and local TF are applied between classification and compositing.

4. Volume Data Exploration

In classic volume exploration, the user is editing the TF while looking at the volume visualization at the same time. The global histogram serves as essential context for the TF design. However, when dealing with a multitude of LFDs instead of a single histogram new means of interaction and depiction are necessary. We therefore propose three approaches to interact with feature distributions.

4.1. Direct Selection of Representative FDs and Outliers

The naive approach for interacting with LFDs is to render them as histograms in attribute space and allow the user to perform a rubber-band selection. This approach is well-suited for investigating outliers and determining their origin within the volume dataset. Here, an outlier is a part of an LFD which sticks out visibly from all the others. As soon as the user hovers or selects one or multiple distributions, the masking volume is updated and immediately applied

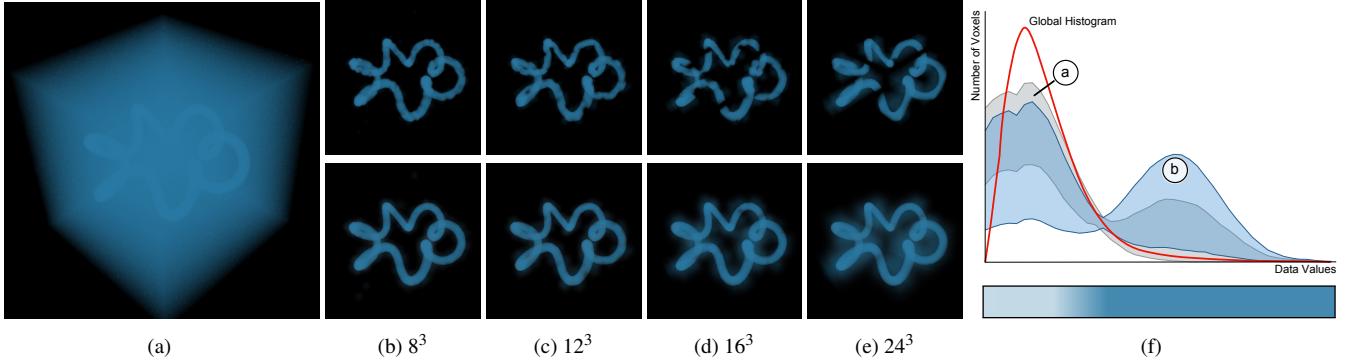


Figure 4: Synthetic Vessel dataset, 192^3 , 8bit. A small vessel is simulated with a spiraling tube having a cross-section of 12×12 voxels. (a) vessel dataset. (b)-(e) effect of varying block size on the vessel cluster with smoothing disabled (top) and enabled (bottom). (f) global histogram (red), two clusters (surrounding (a) and vessel (b)), and utilized TF.

to the volume rendering. This provides a direct link between feature LFDs and the volume data, i.e., the feature location within the volume. In Figure 5, a peak is investigated by selecting all feature LFDs whose extrema lie within the selected region. The respective sub-volumes corresponding to the selected features are highlighted in the volume rendering and reveal that they belong to metastases in the liver. The structure on the right becomes visible because it exhibits a similar FD as the tumor.

The feature LFD selection is straightforward and intuitive, but only for a small number of distributions. In complex data the amount of LFDs leads to visual clutter by the massive overdraw caused by several thousands of distributions. There is also no explicit indication of features except given by visible accumulation.

4.2. Top-down Cluster Traversal

In this approach, a number of clusters representations is shown instead of individual distributions. Its primary use case is to deal with data of largely unknown content, for example a backpack filled with different items. Each cluster is depicted by its mean curve and a hull given by minima and maxima values as shown in Figure 3, top. This reduces overdraw and visual complexity dramatically. When hov-

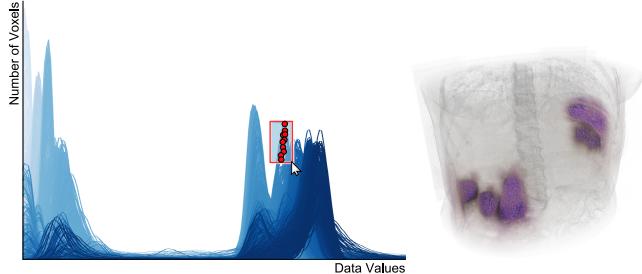


Figure 5: Selection of feature LFDs and outliers in attribute space by dragging a rubber band (left) and the resulting volume rendering (right). Red dots indicate the maxima of selected LFDs. The blue-white coloring refers to the block index of the LFD.

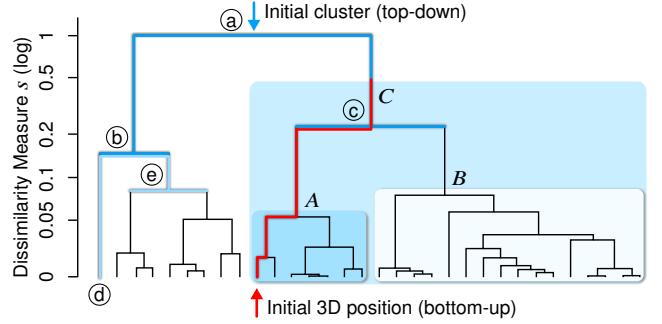


Figure 6: The cluster hierarchy, represented as a tree, can be explored with the cluster traversal (top-down, blue line) or the similarity selection (bottom-up, red line). The cluster traversal allows to drill down to interesting features while only n_{vis} subclusters are shown at a time. Similarity selection begins at a single LFD associated with the respective 3D position.

ering a cluster hull, the volume mask is updated to contain only sub-volumes which are linked to the LFDs of the subtrees.

The number of visible clusters n_{vis} is a user-defined parameter initially set to eight, which provides a good overview on the clustering. Due to the hierarchical clustering scheme, adjusting n_{vis} is equivalent to cutting the hierarchical tree T in such a way that the representatives of the first n_{vis} subtrees are shown. Figure 6 shows an example of such a tree. A value $n_{vis} = 3$ will depict (d), (e), and (c) since cluster (b) will be split due to its high dissimilarity measure.

The user can traverse the hierarchy vertically by selecting the individual sub-clusters or their parent and adjusting n_{vis} . Thus, he or she is able to find and isolate specific features, i.e., peaks in attribute space. Figure 6 illustrates this traversal for $n_{vis} = 2$ starting with (b) and (c) being visible. Exploring (b) reveals cluster (d), consisting of only a single FD, and cluster (e).

The cluster traversal is appropriate for large numbers of LFDs since the visual complexity scales with the number of shown clus-

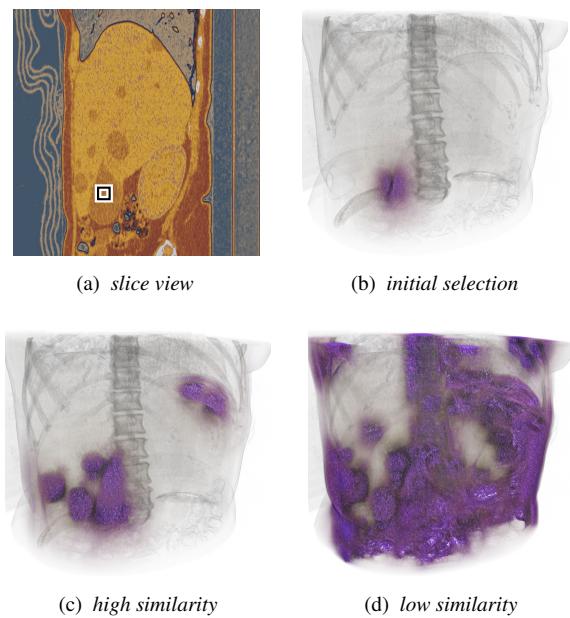


Figure 7: The dissimilarity value s affects the hierarchy traversal and, thus, the selection. (a), (b) the initial marker is positioned within a tumor inside the liver. (c) by increasing s slightly, i.e., indicating high similarity, similar regions are shown. (d) when s is too large other features are selected as well.

ters and not the total number of distributions. The number of feature distributions n affects the height of the cluster tree T , which is $n - 1$ in the worst case. Depending on the chosen similarity metric and clustering metric, the visualization might be cluttered due to overlap between similar clusters. This issue can, however, be alleviated by reducing the number of shown clusters, while requiring one additional interaction to drill down.

4.3. Bottom-up Similarity Selection

Here, exploration begins at a single point in data space and its associated sub-volume, i.e., targeting a specific material like metastases. To select the initial position within a 3D volume we use a single 2D slice. Based on the position of the slice and the location of the selection point we determine the closest corresponding sub-volume block and its LFD to be used as starting point.

The hierarchy is traversed upward from the selected distribution until the similarity of the current cluster exceeds the user-defined dissimilarity parameter s . All LFDs which are part of the current tree are highlighted. In Figure 6, an exemplary traversal of the cluster hierarchy is shown in red. Increasing the dissimilarity value s adds more and more subtrees to the initially selected block. For low values, all nodes in the subtree below A will be added and only if s is larger than that of node C subtree B will be included in the selection as well. The effects of adjusting s are illustrated in Figure 7.

Similarity-based masking visualizes similar regions simply by setting the initial selection and adjusting one parameter. Domain

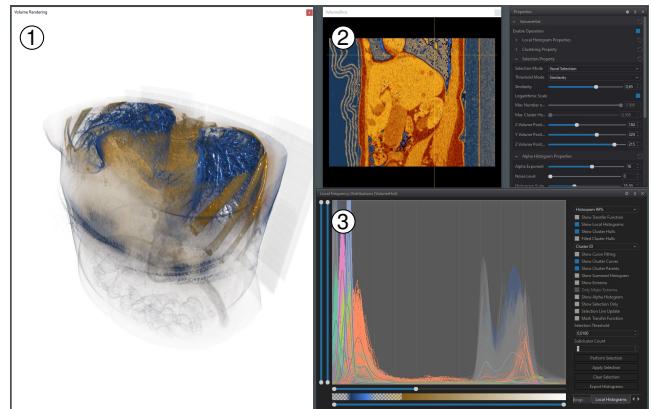


Figure 8: The user interface of our system consists of three main parts: the volume rendering incorporating the volume mask reflecting the current selection ①, a slice view for similarity-based selection ②, and the LFD viewer which combines the interactive cluster traversal with a TF editor ③.

knowledge is required to some extent to detect interesting features on the slice. However, this technique can also be used in a more explorative manner by moving the selection point over the slice.

5. Results

5.1. Performance

The workflow of our approach as proposed in Section 3 was implemented in C++ and OpenGL. We use OpenMP to parallelize the computation of the LFDs for each sub-volume as well as the similarity matrix. The implementation was benchmarked on an Intel Xeon CPU, 2 GHz, with 6 cores and hyperthreading enabled, 16 GiB RAM. Regular volume ray casting is performed on an NVIDIA GeForce GTX TITAN graphics card with 6 GiB of VRAM and the same CPU. Figure 8 depicts the basic UI components of our framework. The 3D view shows the volume rendering with the current selection applied by means of the volume mask (Section 3.3). Similarity-based selection (Section 4.3) is enabled via an adjustable slice view of the dataset and point-based selection. The LFD viewer is basically an extended 1D TF editor which supports the interactive top-down cluster traversal (Section 4.2). It can show both representative FDs and their hulls as well as the respective LFDs.

The time required by the preprocessing step depends mostly on the number of LFDs, i.e., the proportion of block size to volume dimension as well as the number of features per distribution. The most expensive operation in both, time and memory footprint, is the calculation of the similarity matrix followed by the clustering as both have a complexity of $O(n^3)$ in the number of feature LFDs. Figure 9 depicts the necessary time for preprocessing depending on the LFDs. For up to 7,500 LFDs the entire preprocessing step is finished after ten seconds. Computing the similarity matrix for 25,000 LFDs takes up to forty seconds whereas the clustering is slightly faster and requires only 30 seconds. It is noted that the results of the clustering can be stored with the volume dataset for

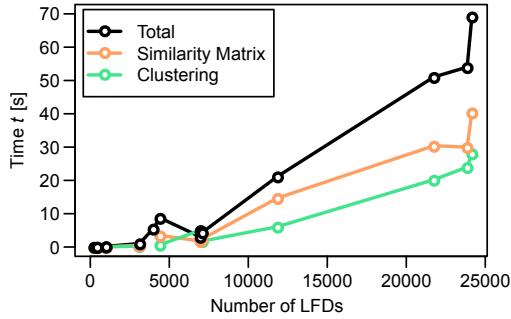


Figure 9: Time required for the preprocessing step correlated to the number of LFDs.

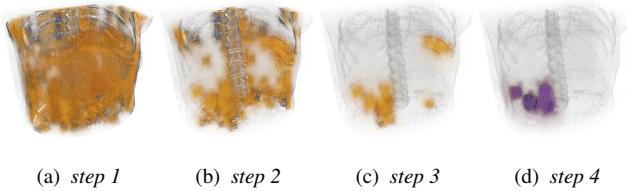


Figure 10: Exploration of cluster ⑩ shown in Figure 3 employing the cluster traversal on the most prominent feature.

different sub-volume sizes unless a recomputation is necessary due to a parameter change. The exploration phase is highly interactive and the update and use of the masking volume does not affect the rendering.

5.2. Exploratory Power

We applied our LFD decomposition approach to a variety of datasets to extract features hidden in the data. These include one synthetic dataset from literature and three additional 12 bit CT scans: Upper Torso dataset containing liver tumors ($512 \times 512 \times 247$), Heart dataset ($512 \times 512 \times 338$), and a scan of a backpack ($512 \times 512 \times 373$).

Upper Torso: The first medical dataset is a scan of a torso where the patient suffered from tumors in the liver. The initial results of applying our approach are shown in Figure 3 revealing features like lungs, or different tissues, organs, the blanket and observation table (Figure 3(a) to (f)). In the top-level clustering, the tumors are responsible for the left-most peak in cluster ⑩ which exhibits multiple features. Due to the very high similarity of the surrounding tissue, this cluster needs to be explored in more detail before revealing the individual features. Figure 10 depicts the four necessary steps using the cluster traversal while focusing on the aforementioned peak. The result (Figure 10(d)) is a clear separation of the liver tumors from their surrounding. The TF utilized for this dataset was designed with respect to the LFDs, i.e., highlighting individual features such as the tumor (cf. Figure 5).

Similarity-based masking can be used, if the user detects an anomaly in a certain volume region or a 2D slice of the data. For example, the tumors located in the liver show up on the slice in

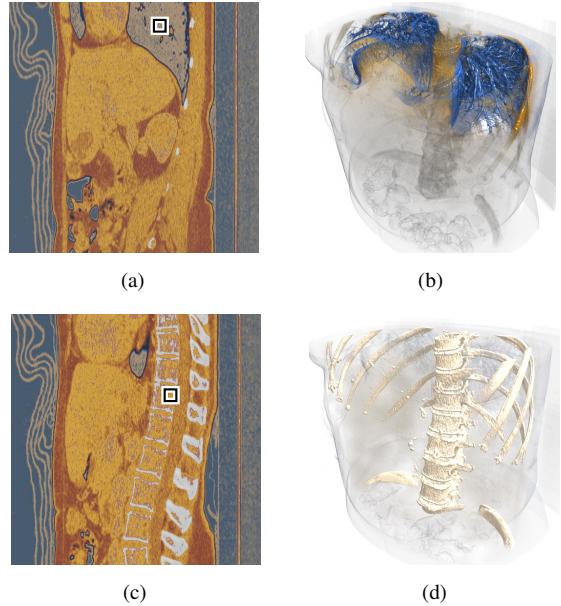


Figure 11: Similarity-based masking applied to the Upper Torso. The user positions a marker on a 2D slice (left) and similar areas are shown in the corresponding volume visualization. (a), (b) probing the lungs. (c), (d) probing the spine.

Figure 7(a). Selecting similar regions then reveals more tumors by adjusting a single parameter. The similarity masking is used for probing the lungs and the spine in Figure 11.

Heart: The second real-world dataset contains a heart with an injected contrast agent. In Figure 12, the clustering results as well as the volume renderings of the individual clusters are shown. Again, the lung is represented as a single, prominent cluster. Different types of tissue can be distinguished as well as the left ventricle filled with the contrast agent. However, the contrast agent is already visible in the global histogram as a feature. But only after having seen the LFDs and the respective clustering, we can ensure that super-positioning plays only a minor role for this feature since cluster ⑩ matches the feature.

Backpack Dataset: This dataset contains a backpack filled with different things including cables, a lighter, and several fluid containers among other things. Haidacher et al. [HPB*10] reported that they were able to distinguish three different fluid containers in the data. Our approach also reveals these three in the initial clustering, two spray cans and a filled tube. The comparison between the global histogram and the clustering in Figure 13 shows that the tube is already visible as a feature in the global histogram (cluster ⑩) whereas the spray cans are part of cluster ⑩ beside other objects. The cluster traversal for cluster ⑩ is shown in Figure 14. By investigating the sub-clusters, we are able to separate and identify the objects contained in this cluster: a box, a round tin, and the two spray cans.

5.3. Block Size

Choosing the correct block size for creating the local histograms is largely depending on the size of the feature one is looking for. If the size is not known beforehand, as in the most common case when investigating a new dataset, we found that a subdivision into 16^3 blocks provides a good overview. This yields about four to eight thousand local histograms depending on the number of features without being affected too much by noise. For the Heart and the Backpack dataset we chose a block size of $32 \times 32 \times 24$ and $32 \times 32 \times 16$ for the Upper Torso.

Synthetic Vessel: Lundström et al. [LYL^{*}06] presented this dataset containing a tubular spiral around a torus to imitate a blood vessel (Figure 4). The vessel-like structure is not visible in the global histogram. The LFDs immediately uncover two clusters not being visible in the global FD (Figure 4(f)). For this dataset the clustering is largely independent on the chosen block size. Figures 4(b)-(e) depict different sizes. As the blood vessel has an approximate cross-section of 12×12 voxels, the vessel cannot entirely be captured by block sizes larger than 12^3 resulting in small gaps which can be suppressed by smoothing. Block sizes of 8^3 and 12^3 can be used to extract the vessel structure clearly.

6. Conclusion and Future Work

In this paper we establish a link between the attribute space and the actual physical domain of data by using the notion of local frequency distributions (LFDs) for feature exploration in volumetric CT datasets. A novel algorithm is presented where LFDs are decomposed into multiple features and are then subject to hierarchical clustering. The cluster hierarchy can be explored using three different approaches. We have demonstrated the applicability of our approach to a variety of datasets. For all tested datasets, LFDs reveal more features than originally visible in the global histogram—a finding coinciding with the idea of an α -histogram, which enhances local features. The different proposed techniques for interactive exploration proved to be useful and are complementing each other, supporting an effective work flow implementation. A potential application scenario for this work would be the creation of feature templates based on representative FDs resulting from the clustering step. Such pre-generated templates could then be stored in a database and applied as presets to other datasets thereby supporting and, potentially, speeding up the exploration process.

One limitation of our approach is the choice of the size of the sub-volumes since this will affect the outcome of the algorithm. However, subdividing the volume into $16 \times 16 \times 16$ blocks provides a good starting point and reflects the scale of interest of the dataset used. To overcome this limitation, we are in future work considering an adaptive scheme to subdivide the volume until each sub-volume contains only one feature. Further research will, then, also be required with respect to the subdivision scheme. The currently most time-consuming part of the method is the computation of the similarity matrix during preprocessing. We are investigating different options to avoid this computation and the associated memory requirements by computing the similarity between distributions on-the-fly. Another limitation of the technique is the focus on CT data. Both MRI and ultrasound data feature an inherent inhomogeneity

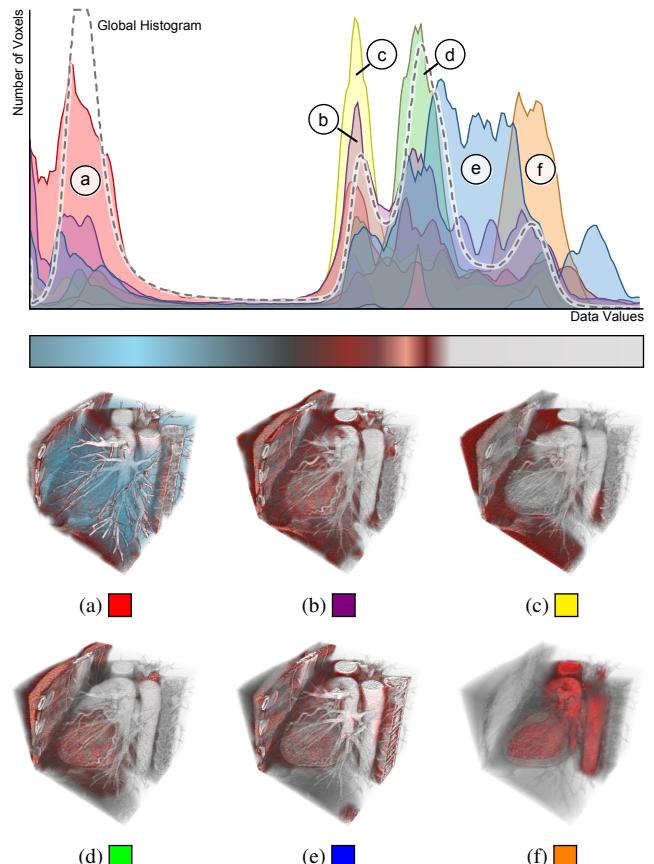


Figure 12: LFD clustering of the Heart CT scan and utilized TF. The initial cluster results are color coded and relate to different features. (a) lung. (b) fat and tissue. (c) fat. (d) soft tissue and muscles. (e) heart muscle. (f) contrast agent in left ventricle (red highlight).

which in turn affects the detection of reasonable features. The next step to further increase the utility of the approach will, thus, be to consider these types of data and to incorporate multi-dimensional data, e.g., multi-spectral data or gradients, into the similarity computations in attribute space.

Acknowledgments

This work was supported through grants from the Excellence Center at Linköping and Lund in Information Technology (ELLIIT) and the Swedish e-Science Research Centre (SeRC). The Backpack dataset is courtesy of Kevin Kreeger, Viatronix Inc., USA. The presented concepts have been developed and evaluated in the Inviwo framework [JSS^{*}19] (www.inviwo.org).

References

- [ANGH11] ANGELELLI P., NYLUND K., GILJA O. H., HAUSER H.: Interactive visual analysis of contrast-enhanced ultrasound data based on small neighborhood statistics. *Computers & Graphics* 35, 2 (2011), 218–226. Virtual Reality in Brazil Visual Computing in Biology and Medicine Semantic 3D media and content Cultural Heritage. [doi:10.1016/j.cag.2010.12.005](https://doi.org/10.1016/j.cag.2010.12.005).

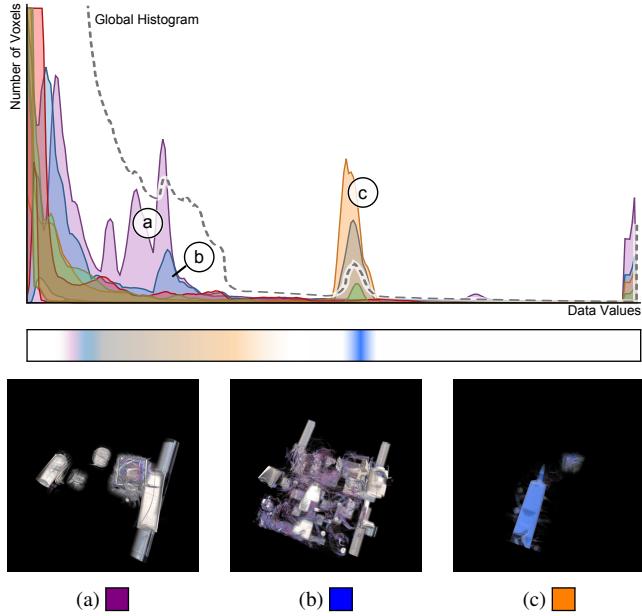


Figure 13: Clustering results for the Backpack dataset and utilized TF. (a) two spray cans, one tin, and a box. (b) cables. (c) filled tube.

- [Bha43] BHATTACHARYYA A.: On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society* 35 (1943), 99–109. 3, 4
- [CFC02] CHANG J.-H., FAN K.-C., CHANG Y.-L.: Multi-modal gray-level histogram modeling and decomposition. *Image and Vision Computing* 20, 3 (2002), 203–216. doi:10.1016/S0262-8856(01)00095-6. 2, 4
- [CHV99] CHAPELLE O., HAFFNER P., VAPNIK V. N.: Support vector machines for histogram-based image classification. *IEEE Transactions on Neural Networks* 10, 5 (1999), 1055–1064. 3
- [CM08] CORREA C., MA K.-L.: Size-based transfer functions: A new volume exploration technique. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (2008), 1380–1387. doi:10.1109/TVCG.2008.162. 2
- [CNCO17] CAI L., NGUYEN B. P., CHUI C.-K., ONG S.-H.: A two-level clustering approach for multidimensional transfer function specification in volume visualization. *The Visual Computer* 33, 2 (2017), 163–177. 2
- [Coc52] COCHRAN W. G.: The χ^2 test of goodness of fit. *The Annals of Mathematical Statistics* 23, 3 (1952), 315–345. 3
- [CP06] CAO Y., PETZOLD L.: Accuracy limitations and the measurement of errors in the stochastic simulation of chemically reacting systems. *Journal of Computational Physics* 212, 1 (2006), 6–24. doi:10.1016/j.jcp.2005.06.012. 3, 4
- [DS16] DUTTA S., SHEN H.-W.: Distribution driven extraction and tracking of features for time-varying data analysis. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (2016), 837—846. 2
- [DVLL11] DOBREV P., VAN LONG T., LINSEN L.: A cluster hierarchy-based volume rendering approach for interactive visual exploration of multi-variate volume data. In *Vision, Modelling and Visualization (VMV 2011)* (2011), pp. 137–144. doi:10.2312/PE/VMV/VMV11/137-144. 2
- [FHL*17] FALK M., HOTZ I., LJUNG P., TREANOR D., YNNERMAN A., LUNDSTRÖM C.: Transfer function design toolbox for full-color volume datasets. In *IEEE Pacific Visualization Symposium (PacificVis 2017)* (2017), pp. 171–179. 2
- [HPB*10] HAIDACHER M., PATEL D., BRUCKNER S., KANITSAR A., GRÖLLER M. E.: Volume visualization based on statistical transfer-function spaces. In *IEEE Pacific Visualization Symposium (PacificVis 2010)* (2010), pp. 17–24. doi:10.1109/PACIFICVIS.2010.5429615. 2, 8
- [IVJ12] IP C. Y., VARSHNEY A., JAJA J.: Hierarchical exploration of volumes using multilevel segmentation of the intensity-gradient histograms. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (2012), 2355–2363. doi:10.1109/TVCG.2012.231. 2
- [JFY16] JÖNSSON D., FALK M., YNNERMAN A.: Intuitive exploration of volumetric data using dynamic galleries. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (2016), 896–905. 1
- [JSS*19] JÖNSSON D., STENETEG P., SUNDÉN E., ENGLUND R., KOTTRAVEL S., FALK M., YNNERMAN A., HOTZ I., ROPINSKI T.: Invivo—a visualization system with usage abstraction levels. *IEEE Transactions on Visualization and Computer Graphics* (2019). 9
- [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. doi:10.1109/TVCG.2002.1021579. 2
- [Kol33] KOLMOGOROV A. N.: Sulla determinazione empirica di una legge di distribuzione (on the empirical determination of a distribution law). *Giornale dell'Istituto Italiano degli Attuari* 4 (1933), 83–91. 3
- [Kol41] KOLMOGOROV A. N.: Confidence limits for an unknown distribution function. *The Annals of Mathematical Statistics* 12 (1941), 461–463. 3
- [Kul59] KULLBACK S.: *Information Theory and Statistics*. Wiley, New York, 1959. 3
- [LKG*16] LJUNG P., KRÜGER 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 (2016), 669–691. 2
- [LLL*10] LINDHOLM S., LJUNG P., LUNDSTRÖM C., PERSSON A., YNNERMAN A.: Spatial conditioning of transfer functions using local material distributions. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 1301–1310. 2
- [LLY06] LUNDSTRÖM C., LJUNG P., YNNERMAN A.: Local histograms for design of transfer functions in direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics* 12, 6 (2006), 1570–1579. doi:10.1109/TVCG.2006.100. 2
- [LS13] LEE T.-Y., SHEN H.-W.: Efficient local statistical analysis via integral histograms with discrete wavelet transform. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2693–2702. doi:10.1109/TVCG.2013.152. 2
- [LvLRR08] LINSEN L., VAN LONG T., ROSENTHAL P., ROSSWOG S.: Surface extraction from multi-field particle volume data using multidimensional cluster visualization. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (2008), 1483–1490. doi:10.1109/TVCG.2008.167. 2
- [LW06] LIU X., WANG D.: Image and texture segmentation using local spectral histograms. *IEEE Transactions on Image Processing* 15, 10 (2006), 3066–3077. doi:10.1109/TIP.2006.877511. 2
- [LWS*17] LAN S., WANG L., SONG Y., PING WANG Y., YAO L., SUN K., XIA B., XU Z.: Improving separability of structures with similar attributes in 2d transfer function design. *IEEE Transactions on Visualization and Computer Graphics* 23, 5 (2017), 1546–1560. 2
- [LYL*06] LUNDSTRÖM C., YNNERMAN A., LJUNG P., PERSSON A., KNUTSSON H.: The α -histogram: Using spatial coherence to enhance histograms and transfer function design. In *IEEE VGTC Symposium on Visualization* (2006), pp. 227–234. doi:10.2312/VisSym/EuroVis06/227-234. 2, 4, 9

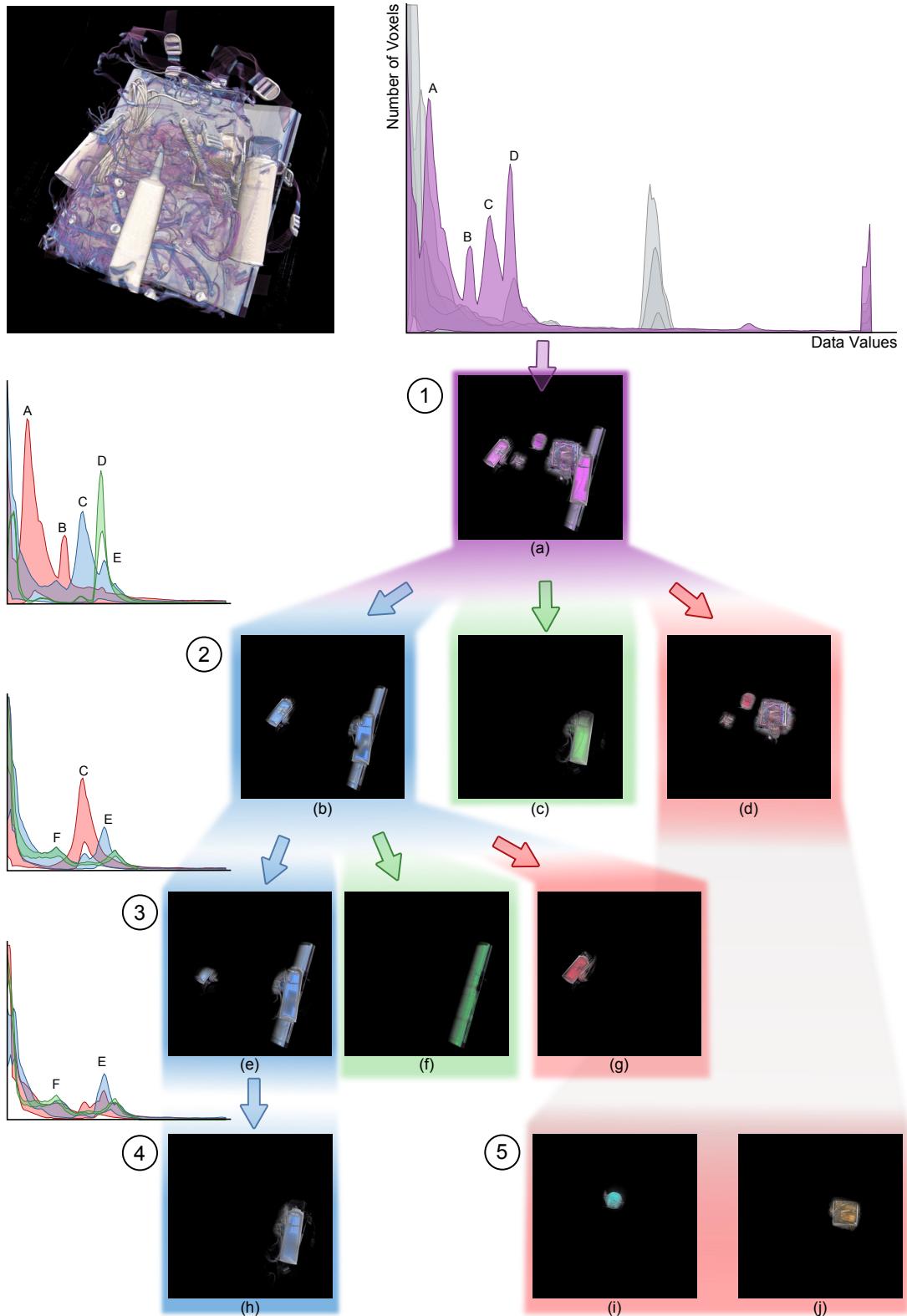


Figure 14: Exploration of the cluster @ in the Backpack dataset (Figure 13). Step 1 depicts the contents of cluster @. In step 2, the three sub-clusters of (a) are shown. Following the blue-colored cluster leads to step 3 and subsequently to step 4. The results (i) and (j) in step 5 were obtained by following the sub-clusters of (d). Capital letters A-F at feature peaks are for reference only.

- [PB91] PAL S., BHATTACHARYYA P.: Multipeak histogram analysis in region splitting: a regularisation problem. *IEE Proceedings E (Computers and Digital Techniques)* 138, 4 (1991), 285–288. [2](#)
- [PHBG09] PATEL D., HAIDACHER M., BALABANIAN J.-P., GRÖLLER M. E.: Moment curves. In *IEEE Pacific Visualization Symposium 2009 (PacificVis 2009)* (2009), pp. 201–208. [2](#)
- [RBS05] ROETTGER S., BAUER M., STAMMINGER M.: Spatialized transfer functions. In *Eurographics/IEEE VGTC Symposium on Visualization* (2005), pp. 271–278. [2](#)
- [RSHSG00] REZK-SALAMA C., HASTREITER P., SCHERER J., GREINER G.: Automatic adjustment of transfer functions for 3D volume visualization. In *Vision, Modelling and Visualization (VMV 2000)* (2000), pp. 357–364. [3](#)
- [SAK19] SHARMA O., ARORA T., KHATTAR A.: Graph-based transfer function for volume rendering. *Computer Graphics Forum* 39, 1 (2019), 76–88. [2](#)
- [SB92] SWAIN M. J., BALLARD D. H.: Indexing via color histograms. In *Active Perception and Robot Vision*. Springer, 1992, pp. 261–273. [2](#)
- [TLM05] TZENG F.-Y., LUM E. B., MA K.-L.: An intelligent system approach to higher-dimensional classification of volume data. *IEEE Transactions on Visualization and Computer Graphics* 11, 3 (2005), 273–284. [2](#)
- [TM04] TZENG F.-Y., MA K.-L.: A cluster-space visual interface for arbitrary dimensional classification of volume data. In *Eurographics/IEEE VGTC Symposium on Visualization* (2004). [doi:10.2312/VisSym/VisSym04/017-024](#). [2](#)
- [TPD06] TAPPENBECK A., PREIM B., DICKEN V.: Distance-based transfer function design: Specification methods and applications. In *SimVis* (2006), pp. 259–274. [2](#)
- [VLL09] VAN LONG T., LINSEN L.: MultiClusterTree: Interactive visual exploration of hierarchical clusters in multidimensional multivariate data. *Computer Graphics Forum* 28, 3 (2009), 823–830. [doi:10.1111/j.1467-8659.2009.01468.x](#). [2](#)
- [WJ63] WARD JR J. H.: Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association* 58, 301 (1963), 236–244. [5](#)
- [ZHPZ96] ZHUANG S., HUANG Y., PALANIAPPAN K., ZHAO Y.: Gaussian mixture density modeling, decomposition, and applications. *IEEE Transactions on Image Processing* 5, 9 (1996), 1293–1302. [doi:10.1109/83.535841](#). [4](#)