Dynamic Volume Lines: Visual Comparison of 3D Volumes through Space-filling Curves

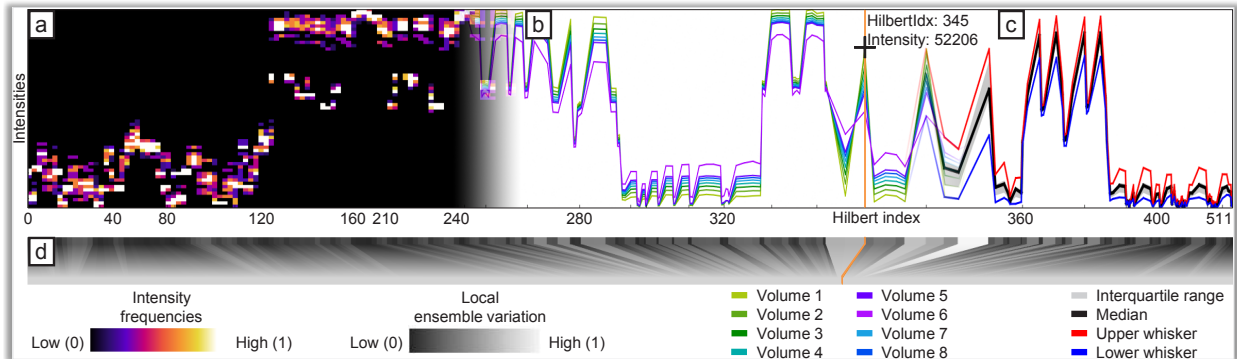Johannes Weissenböck, Bernhard Fröhler, Eduard Gröller, Johann Kastner and Christoph Heinzl

Fig. 1. *Dynamic Volume Lines* depicts sets of volumes (a) in an overview visualization, i.e., an interactive nonlinearly scaled histogram heatmap, which encodes intensity frequencies, or (b) in a detailed view, as interactive nonlinearly scaled 1D Hilbert line plots. Based on the individual 1D Hilbert line plots, functional boxplots (c) are generated on demand. The scaling widget (d) depicts the ensemble variation on each level of detail.

Abstract— The comparison of many members of an ensemble is difficult, tedious, and error-prone, which is aggravated by often just subtle differences. In this paper, we introduce *Dynamic Volume Lines* for the interactive visual analysis and comparison of sets of 3D volumes. Each volume is linearized along a Hilbert space-filling curve into a 1D Hilbert line plot, which depicts the intensities over the Hilbert indices. We present a nonlinear scaling of these 1D Hilbert line plots based on the intensity variations in the ensemble of 3D volumes, which enables a more effective use of the available screen space. The nonlinear scaling builds the basis for our interactive visualization techniques. An interactive histogram heatmap of the intensity frequencies serves as overview visualization. When zooming in, the frequencies are replaced by detailed 1D Hilbert line plots and optional functional boxplots. To focus on important regions of the volume ensemble, nonlinear scaling is incorporated into the plots. An interactive scaling widget depicts the local ensemble variations. Our brushing and linking interface reveals, for example, regions with a high ensemble variation by showing the affected voxels in a 3D spatial view. We show the applicability of our concepts using two case studies on ensembles of 3D volumes resulting from tomographic reconstruction. In the first case study, we evaluate an artificial specimen from simulated industrial 3D X-ray computed tomography (XCT). In the second case study, a real-world XCT foam specimen is investigated. Our results show that *Dynamic Volume Lines* can identify regions with high local intensity variations, allowing the user to draw conclusions, for example, about the choice of reconstruction parameters. Furthermore, it is possible to detect ring artifacts in reconstructions volumes.

Index Terms—Ensemble data, comparative visualization, visual analysis, Hilbert curve, nonlinear scaling, X-ray computed tomography

✦

1 INTRODUCTION AND MOTIVATION

Synthetic foams are widely used, for example, as packaging, as thermal insulating materials, or even as lightweight components [19]. The mechanical behavior of foamed polymers is mainly influenced by the foam density, cell size and diameter, foam hardness, and the deformation rate. For closed-cell and open-cell foams, the determination of the foam

- *Johannes Weissenböck is with the University of Applied Sciences Upper Austria, Wels, Austria. E-mail: johannes.weissenboeck@fh-wels.at*
- *Bernhard Fröhler is with the University of Applied Sciences Upper Austria, Wels, Austria. E-mail: bernhard.froehler@fh-wels.at*
- *Eduard Gröller is with TU Wien, Vienna, Austria and VRVis Research Center, Vienna, Austria. E-mail: groeller@cg.tuwien.ac.at*
- *Johann Kastner is with the University of Applied Sciences Upper Austria, Wels, Austria. E-mail: johann.kastner@fh-wels.at*
- *Christoph Heinzl is with the University of Applied Sciences Upper Austria, Wels, Austria. E-mail: christoph.heinzl@fh-wels.at*

module is important. For this purpose, the spatial distribution of the matrix material in the cell walls must be accurately described [10].

A major challenge in three-dimensional material characterization with conventional industrial 3D X-ray computed tomography (XCT) systems are low densities and thin cell walls, especially at low physical resolutions. One method that overcomes these challenges is Talbot-Lau grating interferometer XCT (TLGI-XCT) [37]. It is a non-destructive testing method, which fully delivers 3D volume information of the scanned specimen at a high resolution to precisely capture external and internal structures (e.g., cracks) in a single scan. TLGI-XCT is one of the most important X-ray technology innovations in the past ten years [28]. This method provides three complementary modalities in one scan of the specimen: (1) the attenuation contrast (AC), (2) the differential phase contrast (DPC), and (3) the dark-field contrast (DFC).

Currently, it is common practice to reconstruct the data of the three modalities separately, without simultaneously using the present and instructive complementary information. To reconstruct the data from the three modalities, the conventional filtered back-projection algorithm by Feldkamp, Davis, and Kress (FDK) is used [6]. This reconstruction algorithm is well suited for XCT data from the AC modality, as it is a fast and accurate method. However, for the DPC and DFC modalities, the FDK reconstruction is not optimal because the prior knowledge and the inherent physical effects of the different modalities

are not considered [17, 33]. Experts in the field of computed tomography reconstruction are therefore developing new algorithms based on appropriate mathematical models. These correspond to the physical characteristics of the DFC and DPC modalities in order to achieve satisfying reconstruction results with regard to conventional methods.

The domain specialists compare the results of the different reconstruction algorithms and their parameterizations with each other and with a reference reconstruction. Regions in the volume with a high ensemble variation of the intensities (e.g., feature edges on interfaces) are of great interest to the experts, as the behavior of the reconstruction algorithm can be deduced if changing specific parameters. The comparison is typically done visually. It is based on 2D gray value slices through the volumes arranged side by side. Experts perform this comparison on a case-by-case basis. They try to determine promising algorithms and suitable parameters, for instance, by checking for noise and artifact suppression. The intensity differences between the various volumetric reconstructions are typically rather small. Therefore it is difficult for an expert to judge whether a particular algorithm or parameter set provides better results (e.g., sharper edges) than another one. This problem gets even worse if several or even many different reconstruction results of an algorithm are compared (e.g., due to parameter variations).

Due to a close collaboration with reconstruction specialists, we were able to analyze their workflow and identify the following tasks when comparing many volumes in specific regions of minor intensity differences:

Task 1: Compare several reconstruction volumes with each other

Task 2: Identify interesting spatial regions based on high local intensity variations

Task 3: Reveal repeating patterns in the spatial domain, which are of high variance among all ensemble members

Task 4: Find the most suitable volume in the ensemble

To address domain-specific requirements, we introduce *Dynamic Volume Lines* for the interactive visual analysis and comparison of 3D volumes using nonlinearly scaled 1D Hilbert line plots (see Figure 1). We build upon the work of Demir et al. [3], who also use the Hilbert linearization to analyze 3D data. As space-filling curve the Hilbert curve traverses the entire 3D volume. For the line plot the space-filling curve is straightened along the horizontal axis. On the vertical axis the intensities at the specific volume positions are shown. We extend the approach of Demir et al. by an adaptive, automatic, and dynamic nonlinear scaling of the horizontal axis, which allows the user to focus on interesting regions in the volumes. The nonlinear scaling highlights regions of high variation in the ensemble and optionally hides uninteresting background regions. Abstracting and reformatting 3D volumes as line plots is chosen because the domain experts are familiar with line graphs, which they use on a daily basis. The workflow of *Dynamic Volume Lines* starts with the extraction of a region of interest (ROI), which is for each volume to be analyzed at the same position and of the same size (see Figure 2a). A Hilbert space-filling curve for the extracted ensemble volumes is generated by mapping voxel coordinates and the corresponding intensities in 3D to 1D Hilbert indices (see Figure 2b). The nonlinear scaling of a Hilbert line plot is built by summing up the local ensemble variations to formulate a cumulative importance-function (see Figure 2c), which serves as basis for the interactive visualization techniques (see Figure 2d–f). Our main contributions are:

- Design and development of *Dynamic Volume Lines*, to compare ensembles of 3D volumes, including the following key features:

  - Nonlinear scaling of the 1D Hilbert line plot, which is built from a cumulative importance-function (see Section 3.2)

  - Interactive nonlinearly scaled histogram heatmap, which encodes the intensity frequencies (see Section 3.3.1)
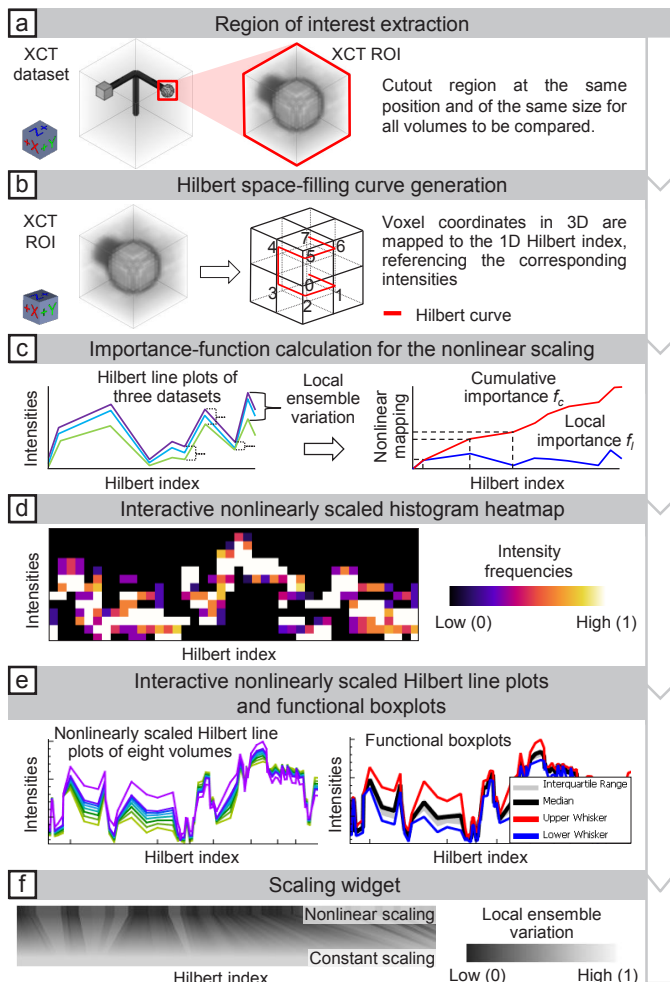

Fig. 2. The *Dynamic Volume Lines* workflow: (a) extraction of interesting regions, (b) generation of a space-filling Hilbert curve, (c) nonlinear scaling of the Hilbert line plots, (d) interactive nonlinearly scaled histogram heatmap, (e) interactive nonlinearly scaled 1D Hilbert line plots, (f) and interactive scaling widget.

  - Interactive nonlinearly scaled 1D Hilbert line plots of the individual volumes in the ensemble (see Section 3.3.2)

  - Interactive scaling widget, which illustrates the locally varying scaling factor (see Section 3.3.3)

- Evaluation of the tool based on two case studies from the XCT domain

In the subsequent Section 2 we review the related work on comparative and ensemble visualization. In Section 3 we explain *Dynamic Volume Lines*, which includes the Hilbert curve generation, the nonlinear scaling based on a cumulative importance-function, and the interactive visualization techniques. Section 4 describes the data acquisition and the used datasets. Section 5 presents the evaluation of the developed tool based on two case studies. We conclude and point out potential future work in Section 6.

2 RELATED WORK

The related work is mainly in the areas of comparative and ensemble visualization, but also in the fields of visual parameter space analysis and interactive visual analysis.

Gleicher et al. [9] provide a taxonomy to group visual designs into one of three basic categories: juxtaposition, superposition, and explicit encoding. In the work of Malik et al. [21], slices from different

volumes are compared in a hexagonal arrangement. This visual representation works only for a rather small number of datasets to be compared. Schmidt et al. [35] present a tool for the comparative visual analysis of 3D meshes, which enables the simultaneous comparison of several meshes and allows for the interactive exploration of their differences. In an earlier work [34], they present a method for visualizing differences and similarities in large sets of images. The technique preserves contextual information, but also allows the user to perform a detailed analysis of subtle variations. Based on magnetic resonance cartilage-imaging techniques, Mlejnek et al. [25] propose 3D glyphs, called *Profile Flags*, for the probing of sets of underlying curve data. Weissenböck et al. [42] introduce a system to evaluate the porosity in 3D volumes. They provide 2D slice views and 3D renderings to compare the different pore segmentation results due to varying segmentation parameters. Many of the ensembles investigated in these works result from analyzing the parameter space of some algorithm, for which Sedlmair et al. [36] present a conceptual framework to guide and systematize research endeavors.

Ensemble visualization often uses statistical summaries for the comparison of many similar datasets. *Ensemble-Vis* by Potter et al. [30] is a framework consisting of an interactively linked overview and statistical displays for the discovery and evaluation of simulated meteorology outcomes. Jarema et al. [16] provide a visual-analysis user-interface with multiple linked views to support the comparative exploration of 2D vector-valued ensemble fields. Fröhler et al. [7] present an interactive tool for exploring and analyzing the parameter space of multi-channel segmentation algorithms and the corresponding ensemble of segmentation results. Several works have been published, which embed descriptive statistics measures such as minimum, median, and maximum in functional boxplots [39], contour boxplots [43], and curved boxplots [24]. Genton et al. [8] developed *surface boxplots* for the visualization and exploratory analysis of samples of images to detect potential outliers. They use the notion of volume depth to order the images interpreted as hightfields. Raj et al. [31] examine the effectiveness of contour boxplots in the medical domain of brain atlas analysis. They extend contour boxplots to 3D to visualize and interact with ensembles of 3D isosurfaces. Demir et al. [4] determine the most central shape from a given set, to quantify a region-wise centrality, and to compute the locally most representative shape. Konyha et al. [18] and Matkovic et al. [22] focus on the interactive visual analysis of ensembles of curves called *families of curves* or *families of surfaces* using data aggregation and attribute derivation. Piringer et al. [29] extend the work on feature-preserving downsampling of 2D functions. They discuss a design study of an interactive approach for the comparative visual analysis of 2D function ensembles.

In the context of visual analysis, the state of the art report by Heinzl and Stappen [15] closes a gap between visual computing and material science. Torsney-Weir et al. [40] propose *Sliceplorer* to visually examine multi-dimensional continuous scalar functions with 1D slices. Their technique combines the benefits of topological views, i.e., screen space efficiency, with those of slices, that are a close resemblance of the underlying function. Another work of Demir et al. [3], presents *Multi-Charts*, an interface to visually analyze 3D scalar ensemble fields by linearizing the 3D data points along a space-filling curve. Our approach is similar to *Multi-Charts*, as we also use the Hilbert space-filling curve to linearize 3D volumes and represent the volumes as 1D line plots. Demir et al. represent the individual ensemble members as multiple stacked and combined bar and line charts at different levels of detail. For analyzing such regions the user has to zoom in and out. This leads to a loss of context. A significant difference in our work is the computation of a nonlinear scaling of the x-axis based on local ensemble variations. The nonlinear scaling allows us to depict all 3D volumes as 1D line plots, which can be presented in one visualization. As a result, uninteresting regions (with low ensemble variance) are compressed in the line plots, and interesting regions (with high ensemble variance) are expanded. Thereby, we can optimally use the available screen space and no zooming is necessary in the first place, as we provide insight into the interesting regions from the initial overview state. The individual line plots of the corresponding 3D volumes can be aggregated

using functional boxplots. Thus, we provide a statistical overview of the ensemble. In addition, the scaling widget indicates the nonlinear scaling of the data. Finally, we support an importance-driven selection by defining ranges based on a cumulative importance-function.

3   DYNAMIC VOLUME LINES

In this section we explain the generation of the Hilbert curve based on the 3D voxel intensities and compare the line plots of the Hilbert curve with the line plots of the scan line curve (see Section 3.1). Furthermore, we describe the nonlinear scaling of the Hilbert line plots based on local ensemble variations (see Section 3.2).

The basic motivation of *Dynamic Volume Lines* is to linearize 3D volumes along a space-filling curve. The resulting line plots are a familiar representation to engineers. Without occlusion many volumes can be compared through their line plots. A comparison of many volumes in their original 3D space that differ only slightly from each other turns out to be difficult with traditional methods. For example, a direct volume visualization of many datasets is plagued by severe clutter and occlusion problems. This effect is further reinforced by the increasing number of volumes to compare. The same is true for 2D slice views. Arranging two or four slice views of different volumes side by side would be feasible, but with an increasing number of volumes to compare, e.g., six, it is nearly impossible to find regions where the volumes differ. For example: if individual voxels differ by 5000 intensities, this difference is difficult to perceive as brightness difference, even for an expert. For a line plot in a range of 65000 intensities on the y-axis, this amplitude drop would be 7% and thus easier to recognize as positional difference. A positional encoding is much more effective than color coding to indicate subtle differences. Comparing intensities is much easier through line plots, in contrast to having first to match them in two or more 2D (or 3D) views and then comparing their color encoding. One could apply statistical aggregation to determine a volume where the voxels contain the local ensemble variances. But even in such a reduced data, rendering is affected by clutter and occlusion. For example, to make differences visible in the interior of the volume, the opacity must be set to a low value. However, larger differences are then only vaguely recognizable. Small differences will be lost. In addition, statistical aggregation volumes provide only a summary or overview and would require additional detailed visualizations to compare specific members in a region of interest. In our approach the differences between the individual members can easily be inspected by comparing line plots.

A drawback of linearizing volumes is the loss of spatial coherency. Among the many possibilities of space-filling curves we decided for one which is preserving the spatial coherency as much as possible. The wigglyness of the Hilbert curve ensures that very often neighboring voxels in the 3D volumes are mapped to nearby locations in the straightened Hilbert line plot. We compare the spatial coherence of the Hilbert curve with respect to another simple volume linearization, i.e., the scan line curve where the volume is traversed slice by slice and scan line by scan line within a slice. Switching scan lines or slices introduce large spatial incoherences. Such incoherencies appear with Hilbert curves as well, but much less.

3.1   Hilbert Space-Filling Curve Generation

In general, with space-filling curves, n-dimensional regular grids can be completely traversed and the grid points can be brought into a one-dimensional linear order. In this sense volumes are three-dimensional regular grids. There are many different space-filling curves such as the Hilbert curve, the Peano curve, or the Z-curve. An obvious approach to linearize the intensities of a 3D volume is to traverse the voxels in z-, y-, and x-axis order along a scan line curve, line by line and slice by slice (see Figure 3a). The disadvantage of this approach is that there are large jumps in the scan line curve between the last voxel of the previous row and the first voxel of the next row and the last voxel of the previous slice and the first voxel of the current slice, respectively. In contrast, the Hilbert space-filling curve traverses every point of a square, a cube, or more generally, an n-dimensional hypercube, by preserving locality much better. Points close to each other in the n-dimensional space are very often close in their order along the Hilbert curve and vice
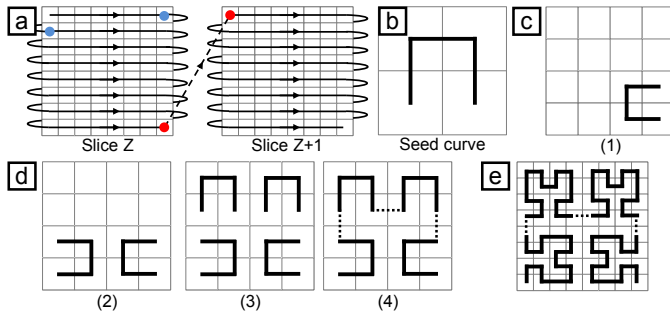
Fig. 3. (a) shows the scan line curve of a 3D volume dataset by traversing the voxels along their appearance. The blue dots mark a large jump due to a line change and the red dots mark a large jumps due to a slice change. (b) depicts the Hilbert curve of order one (initial seed curve). (c–d) illustrate the individual steps to generate a Hilbert curve of order two in 2D space. (e) shows the Hilbert curve of order three.
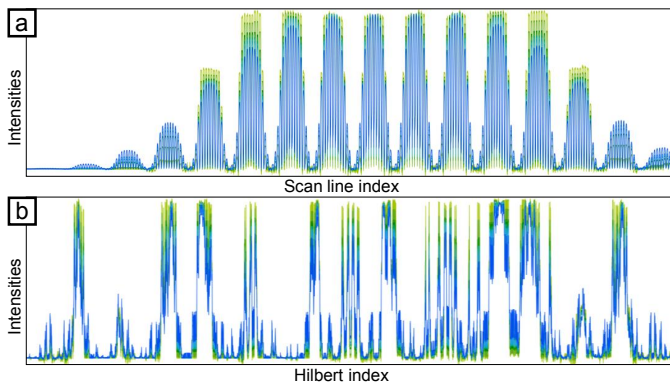

Fig. 4. Difference between the scan line curve and the Hilbert curve due to large jumps and different locality preservation. The six datasets of the artificial specimen (see Section 4.2) are represented by color-coded line plots (from light green to blue). The line plots are based on (a) the scan line curve and (b) on the Hilbert curve.

versa [11]. This results in fewer large jumps. Therefore in this work, we focus on the Hilbert curve, as it preserves the locality best [26].

In the following we describe the generation of the space-filling Hilbert curve exemplarily in 2D space [13]. Consider the initial seed curve defined on a 2×2 grid as shown in Figure 3b. It is called order 1 Hilbert curve. Based on an order k Hilbert curve defined on a $2^k \times 2^k$ grid, we define the order $k+1$ Hilbert curve on a $2^{k+1} \times 2^{k+1}$ grid according to four steps (see Figures 3c–d):

1. Place a copy of the curve in the lower right cell and rotate it 90° counter-clockwise.
2. Place a copy of the curve in the lower left cell and rotate it 90° clockwise.
3. Place a copy of the curve in each of the upper cells.
4. Connect the curves with each other.

The resulting space-filling curve visits every voxel exactly once and assigns it a scalar index resulting from the traversal order (Hilbert index). For example, Figures 3b, 3c(4), and 3e show the Hilbert curves of order one, two, and three. Figure 2b (on the right) shows the Hilbert curve of order one in 3D space.

In the case of *Dynamic Volume Lines*, all 3D volume datasets are linearized using the implementation of Hamilton and Rau-Chaplin [14]. It generates a Hilbert space-filling curve where the volumes do not need to have the same number of voxels along the x-, y-, and z-axis. The resolution along the individual axes also need not be a power of two. Figure 4 illustrates the difference between linearizing along the Hilbert curve and the scan line curve. It depicts the six volumes of the artificial

specimen (see Section 4), each represented by a color-coded line plot (from light green to blue). The scan line plots in Figure 4a fluctuate more, due to less spatial coherence, as compared to the Hilbert line plots (in Figure 4b).

3.2 Nonlinear Scaling of the Hilbert Line Plots

When generating the Hilbert curve for a dataset of $16 \times 16 \times 16$ voxels in size, 4096 Hilbert indices are created. Currently the horizontal screen resolution of a standard PC monitor is typically between 2000 and 3000 pixels. If one wants to display all the 4096 Hilbert indices as points in a 1D Hilbert line plot on the monitor, it turns out, that the horizontal screen resolution is not sufficient to assign each Hilbert index to its own pixel column. This problem becomes even more severe with increasing volume size. Linearizing a volume reduces it to a simple line plot, but with a tremendous horizontal resolution, i.e., number of voxels. This requires automatic scaling along the horizontal axis so that important regions get the screen space they need. Unimportant regions like background can be drastically reduced in their screen space or even removed. To counteract the problem of the limited screen space, we apply a nonlinear scaling to the Hilbert line plot. Since the domain experts are interested in those regions of the dataset where the variation of the intensities is high, we compute the maximum local ensemble variation V_h for every Hilbert index as follows:

$$V_{\mathrm{h}} = \max_{\forall m \in M_h} Intensity_h(m) - \min_{\forall m \in M_h} Intensity_h(m) \tag{1}$$

m defines an ensemble member of a local ensemble M_h at a discrete Hilbert index h, $Intensity_h$ defines the intensity at Hilbert index h for member m.

Inspired by the work of Mindek et al. [23] and Lindow et al. [20], we formulate the discrete local importance-function f_l based on the maximum local ensemble variation V_h:

$$f_l(h) = \left(\frac{V_h}{\max V_h} \right)^p \tag{2}$$

To be able to filter for a specific importance-value, we do a normalization by the maximally occurring local ensemble variation. To influence the nonlinear scaling, we introduce an exponent p, which can be adapted by the user. Setting this parameter to zero means equal importance for all Hilbert indices, setting it higher than zero increases the importance-value for Hilbert indices with a high variance in the ensemble. Exponent p can be adapted by the user to fine-tune the importance-value according to the individual application scenario. By summing the local importance-function values, we define the cumulative importance-function f_c as follows:

$$f_c(h) = \sum_{i=0}^{h} f_l(i) \tag{3}$$

The cumulative importance-function values serve as a nonlinear mapping to compress the distances between the Hilbert indices (on the x-axis). Figure 2c illustrates the calculation of the nonlinear mapping. Figure 5 illustrates the effect of nonlinearly scaling the Hilbert line plots. The regions indicated with red in Figure 5b are compressed (respectively uncompressed in Figure 5a) by the nonlinear scaling because of the low ensemble variation in these regions. In addition, we allow the user to set a threshold for intensities that are not of interest (e.g., areas of air in the dataset). For these background areas (see Figure 6) f_l is fixed to a value of 0.025. This value ensures that the background areas are still sufficiently visible, but at the same time occupy little screen space. By increasing the parameter p, the width of the background regions can be adjusted.

3.3 Visualization Techniques

Dynamic Volume Lines provides multiple linked views and follows the visual information-seeking mantra by Ben Shneiderman, "overview first, zoom and filter, then details on demand" [38]. Two charts, one with nonlinear scaling and one with constant scaling, are arranged on
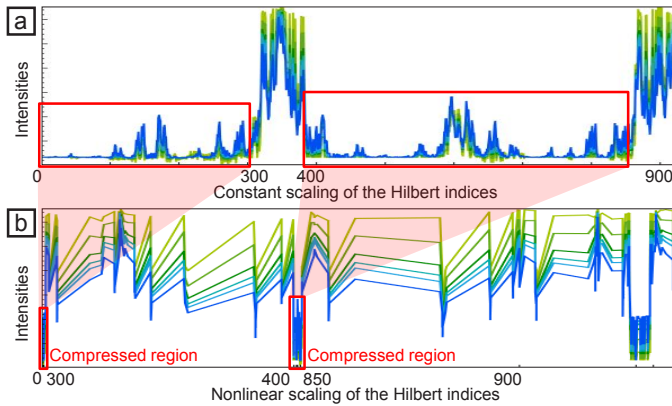
Fig. 5. Effect of the nonlinear scaling. (a) Constantly scaled 1D Hilbert line plots of six 3D volumes. (b) nonlinearly scaled 1D Hilbert line plots of the same six 3D volumes.

top of each other with a scaling widget in-between. An overview is provided through the nonlinearly scaled histogram heatmap visualization. Background regions can be filtered out. If the user zooms into the chart, details for each volume are provided through the 1D Hilbert line plots. An orientation widget displays the size and position of the currently visible chart area in light blue compared to the overall chart size, which is shown in gray (see bottom of Figures 6 and 7). In addition, we enable brushing and linking to immediately highlight the affected voxels or Hilbert indices when performing a selection in the 1D Hilbert line plots or in the 3D spatial view, respectively.

3.3.1 Interactive Histogram Heatmap Visualization

The histogram heatmap visualization provides an overview of the intensity distribution in the volume ensemble. To be able to fit the full volume into a chart with the width of the screen, we split the x-axis of the nonlinearly scaled chart into intervals of equal width. Due to the nonlinear scaling, the intervals may include a varying number of Hilbert indices. For each interval we then compute a single histogram of the intensities at the included Hilbert indices, over all ensemble members. The single histogram is visualized as a heatmap through a vertical bar, all histograms together form a histogram heatmap. The default width of each histogram bar is 10 pixels, and by default each histogram has 64 bins. These parameters can be adapted by the user to best fit the size and intensity distribution of the currently analyzed volumes. To improve the performance in computing the histogram heatmap, we adapt a segment tree [2]. A segment tree is a binary tree used for storing segments in the range $0...n-1$. Each node represents a segment and is assigned the corresponding histogram. The segment of a leave node covers only one Hilbert index (voxel), the segment of the root node covers all Hilbert indices. The segment tree partitions an arbitrary interval into a minimal set of segments (of varying sizes). With this, we avoid to compute the histogram of an interval based on single voxels. Instead we combine the histograms of the highest-level nodes, whose segments are included in the requested interval. The root node of the segment tree contains the entire segment $[0, n-1]$. A leaf node represents an elementary segment, which corresponds to one Hilbert index, i.e., one voxel. It is assigned the histogram of all the intensities at that voxel throughout the entire volume ensemble. The internal nodes merge the segments of their child nodes. The segment tree can be serialized using an array of size $n-1$. The internal nodes are stored in the first half of the array, the leaf nodes are stored in the second half of the array. The left child of each node at index i can be found at index $2*i+1$, the right child at index $2*i+2$. The segment tree is built bottom-up by taking the pairs of nodes with indices $(2*i+1, 2*i+2)$ and aggregating their histograms into the histogram of the parent at index i. The construction of the segment tree is performed for every ensemble and takes $O(n)$ time. In our case, each node of the segment tree contains the histogram for a segment of a certain contiguous range of Hilbert indices. A vertical bar of the

histogram heatmap is created by summing up each histogram of the individual volumes. Due to the nonlinear scaling, many Hilbert indices may be covered by a vertical bar of the histogram heatmap. Without the segment tree, a histogram would have to be calculated from all these Hilbert indices. With the segment tree, a histogram for a vertical bar can be generated in $O(\log n)$ time, since only a few nodes need to be traversed. The segment tree therefore supports efficient rescaling of the charts.

In the figures in this paper, the extended black body scheme proposed by Moreland [27] is used as color map for the histogram heatmap, but other predefined color maps can be selected by the user. Figure 6 shows an example histogram heatmap. White regions denote a high concentration of intensities in a single bin. This is the case if all ensemble members agree on a small range of intensities, i.e., if the variation is low in that region. In contrast, a high variation is indicated by a broader distribution in violet, red, and yellow colors from the middle of the color map. The focus in XCT images is typically on regions containing an object, therefore areas containing only background, i.e., air, are not of interest for the analysis. The background areas, where the intensities of all ensemble members are below a user-defined threshold, are assigned a very low importance-value and therefore are highly compressed in the nonlinear scaling. Setting the background threshold is optional, a value of zero means that the background is left out altogether. Instead of showing a histogram, these background regions are depicted by light orange boxes. In this fashion, we can identify interesting regions, i.e., those with high local variation, which addresses Task 2 from Section 1.

3.3.2 Interactive 1D Hilbert Line Plot Visualization

In the 1D Hilbert line plots, for each volume, the intensities are plotted on the y-axis over the Hilbert indices on the x-axis. Each plot is assigned a distinctive color taken from the metro color scheme of MaterialUI [1], as can be seen in Figure 7a. When zooming into the histogram heatmap, the 1D Hilbert line plots are activated automatically as soon as the range of currently visible Hilbert indices fits on the screen without aggregation. The 1D Hilbert line plots can also be shown as an overlay on the histogram heatmap. Specific ensemble members can always be activated and deactivated by clicking on them in the legend. The current mouse position is highlighted by a position marker line (the orange line in Figure 7a and c), augmented with a tool-tip displaying the Hilbert index and intensity at that position. For better visibility and to better visually link the nonlinearly scaled and the linearly scaled chart, this line is updated simultaneously in both charts. Visualizing two or more 1D Hilbert line plots side by side enables a detailed analysis and comparison of all ensemble members, addressing Task 1 and Task 4 from Section 1.

Dynamic Volume Lines provides several ways to select regions of interest. The user can perform rectangular multi-selections directly in the charts. Additionally, all Hilbert indices within a specified importance-range can be selected by defining upper and lower bounds. Furthermore, the user can perform a selection in the 3D spatial view, by marking a rectangular region with a dragging interaction. This selects all voxels inside of the cuboid region that is spanned by projecting the rectangle from the near plane of the viewing frustum along the viewing direction to the far plane. In either case, all intervals in the Hilbert line plot falling into the selected region are highlighted. They are emphasized in the scaling widget as well, and the respective regions in the selected ensemble members are displayed in a separate 3D visualization. As a result, different or similar areas in the ensemble of reconstruction volumes can be identified and thus repeating patterns in the data (e.g., ring artifacts) can be exposed. This addresses Task 3 from Section 1. Optionally, we provide an aggregated view of the individual 1D Hilbert line plots using functional boxplots [39]. The functional boxplots show statistical properties such as the lower and upper whisker, the median, and the interquartile range (see Figure 1c).

3.3.3 Scaling Widget Visualization

Dynamic Volume Lines depicts a nonlinearly scaled chart at the top and a constantly scaled chart at the bottom, both showing the histogram
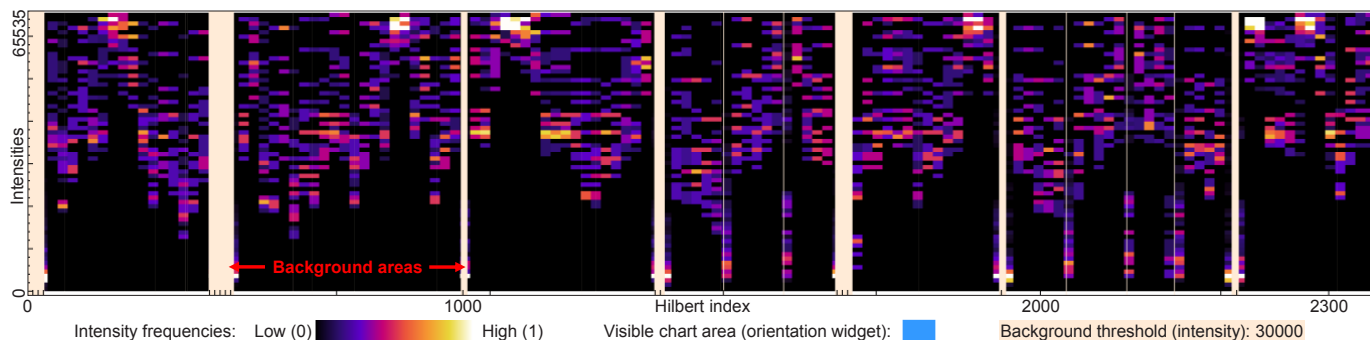
Fig. 6. The histogram heatmap overview visualization with the extended black body heatmap. White areas indicate a high concentration of intensities. This is the case if all ensemble members agree on a small range of intensities, i.e., the variation in this region is low. A high variation is indicated by a broader distribution in violet, red, and yellow. Light orange background areas hide the uninteresting intensities below 30000.
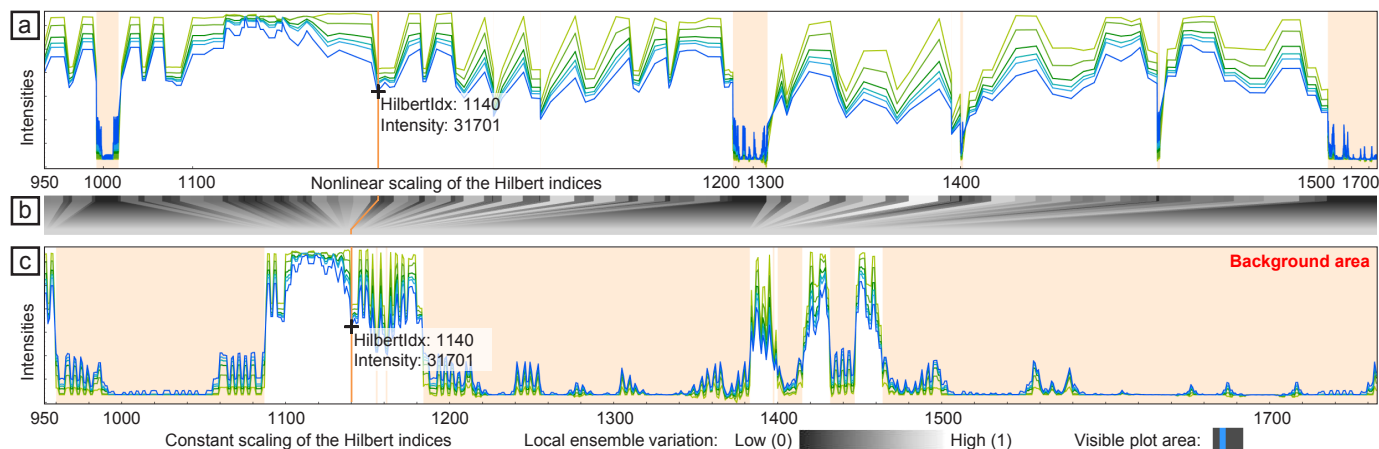

Fig. 7. (a) The nonlinearly scaled chart shows the differently colored 1D Hilbert line plots for each volume. The intensities are plotted on the y-axis over the Hilbert indeces on the x-axis. The current mouse position is highlighted by an orange position marker line, which shows the Hilbert index and intensity at that position. (b) depicts the scaling widget, which emphasizes the nonlinear scaling. (c) shows the constantly scaled chart with the individual 1D Hilbert line plots.

heatmap visualization as well as the Hilbert line plot visualization. Early prototypes just had those two charts on top of each other. When working with those early prototypes together with the domain experts, we realized that an explicit visualization of the nonlinear scaling is necessary. For this purpose, the current design includes the scaling widget, as shown in Figure 7b. Each single histogram of the histogram heatmap in the nonlinearly scaled chart is mapped to the corresponding histogram in the constantly scaled chart. If the line plots are visible, each Hilbert index in the nonlinearly scaled chart maps to the corresponding index in the constantly scaled chart. Small rectangles at the top of the scaling widget each represent a histogram or Hilbert index. Their gray values encode the local ensemble variation. Black represents a low variation, white a high variation. From these rectangles at the top, trapezoids extend to the bottom. Their color is gradually shifting to an average gray value, which represents the constant scaling applied in the lower plot. The position marker line is also shown in the scaling widget, linking the nonlinearly scaled chart at the top with the constantly scaled chart at the bottom, as shown in Figure 7.

4 DATASETS

This section briefly explains how the industrial 3D X-ray computed tomography (XCT) data is acquired (see Section 4.1) and which datasets are used (see Section 4.2).

4.1 Data Acquisition

XCT provides a volumetric representation of a scanned specimen. The specimen is placed on a rotary table between the X-ray source and the detector. While the specimen is rotating, the source emits cone-beam X-rays. The detector collects the X-rays attenuated by the specimen. The attenuation depends on the density and atomic number of the material and on the penetration thickness of the specimen. The detector converts the radiation intensity into a series of digital projection images. In the reconstruction stage, an algorithm is applied on the projection images in order to reconstruct the 3D volume of the specimen. Talbot-Lau grating interferometer XCT (TLGI-XCT) delivers, in contrast to conventional XCT, three complementary modalities, i.e., attenuation contrast (AC), differential phase contrast (DPC), and dark-field contrast (DFC) in a single scan. The three modalities are perfectly registered to each other. AC provides information on the attenuation of the X-ray beam intensity and thus is equivalent to conventional X-ray imaging. DPC is related to the index of refraction and image contrast, which is achieved by the local deflection of the X-ray beam. DFC reflects the total amount of radiation scattered at small angles, e.g., caused by microscopic structures in the sample like particles, pores, fibers, struts, or cracks. In addition, the DFC modality produces a strong signal and a high contrast at interfaces and reveals information that is undetected by AC and DPC imaging.

4.2 Dataset Description

The first ensemble consists of 3D reconstruction datasets of an artificial specimen from simulated XCT [32] of the AC modality with intensities between 0 and 65535. The size of each dataset is $128 \times 128 \times 128$ voxels and the data type is unsigned short (see Figure 8a). The artificial projection images are generated by calculating penetration lengths of primary monochromatic X-rays through the specimen. The specimen is represented by surface models of three cylinders, one sphere, and
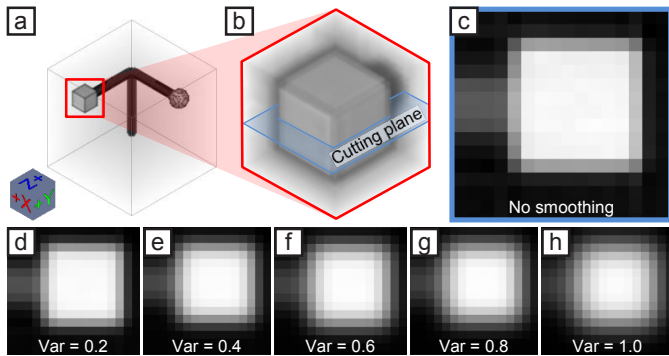
Fig. 8. (a) 3D reconstruction dataset of the artificial specimen from simulated XCT of the AC modality, (b) with an ROI cutout of a cube. (c) shows an xy-slice view of the dataset without Gaussian smoothing. (d–h) depict the xy-slice views of the five datasets with an increasing variance of the Gaussian smoothing between 0.2 and 1.0.
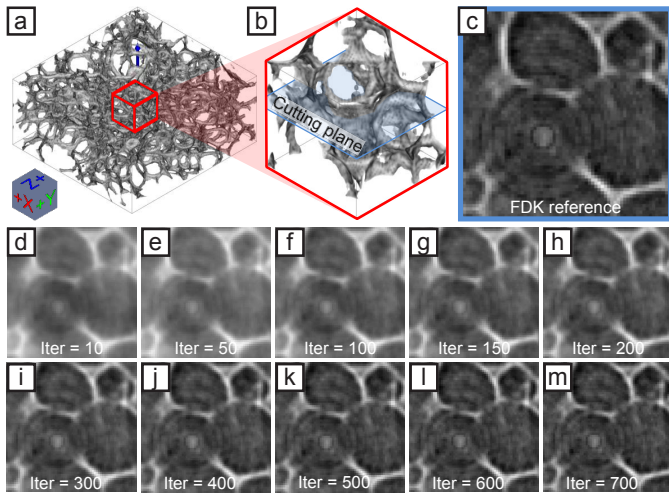

Fig. 9. (a) 3D reconstruction dataset of a real-world TLGI-XCT scanned foam specimen of the DFC modality (b) with an ROI cutout. (c) shows an xy-slice view of the FDK reconstructed reference dataset with ring artifacts in the center. (d–m) depict the xy-slice views of the SIRT datasets with iteration parameters increasing from 10 to 700 exemplarily.

one cuboid. Attenuations are calculated by applying Lambert-Beer's law. The X-ray scatter and the blurring effects were disabled for the simulation. The virtual projection images are processed using the FDK reconstruction algorithm. We applied a Gaussian filter with increasing smoothing effect to generate five additional volumes. No smoothing was applied to the first dataset. The individual variances of the Gaussian smoothing in the range [0.2, 1.0] are increased by a step of 0.2. In a preprocessing step, a user-defined ROI is selected and applied to all ensemble volumes (see Figure 8b). The resolution of the ROI cutouts are $16 \times 16 \times 16$ voxels. Figure 8c depicts an xy-slice of the dataset without smoothing. Figures 8d–h show xy-slices of the different results of the Gaussian smoothing filter.

The second ensemble consists of 16 datasets from a real-world open-cell polyurethane foam specimen, which was scanned with a Bruker Skyscan 1294 TLGI-XCT device at a resolution of 11.4 microns. The size of each dataset is $550 \times 550 \times 250$ voxels and the data type is unsigned short. To compare the individual datasets, we normalized the intensities (itk::NormalizeImageFilter) by setting the mean to zero and the variance to one. We then rescale the intensities between 0 and 65535. Figure 9a shows the thin cell walls of the foam specimen, which are revealed by the DFC modality. In the middle of the foam specimen we cut out an ROI of $64 \times 64 \times 64$ voxels (see Figure 9b). Figure 9c depicts an xy-slice of the cutout reference dataset, which

was reconstructed from 900 projections using the FDK algorithm. The center of Figure 9c shows ring artifacts. The other 15 datasets were reconstructed using the simultaneous iterative reconstruction technique (SIRT) [12] with 900 projections and the following increasing iteration parameters: 10, 50, 100, 150,...,700. Figures 9d–m present the SIRT volumes with increasing iteration parameters.

5 RESULTS

In this section we present two case studies that reflect the domain-specific requirements and present the capabilities of *Dynamic Volume Lines*. First, we analyze reconstruction data from the simulated XCT specimen, and in our second case study, we analyze the real-world XCT specimen.

5.1 Case study 1: Simulated XCT Dataset

The specimen analyzed here is an artificial dataset from simulated XCT with three cylindrical bars orthogonal to each other (see Section 4.2). Two of the bars have attachments at their ends, one of which is a sphere, the other one is a cube. Different levels of smoothing produce a volume ensemble, as shown in Figure 8. The ROI cutout for this analysis covers the end of the bar with the attached cube, as shown in Figure 10a.

The analysis goal for this dataset mainly has been to determine interesting regions, which correspond to areas where the most changes happen in the ensemble. We use a synthetic dataset in order to show the basic behavior of *Dynamic Volume Lines* under well-defined conditions. We set a filter for regions with high local variations, in this case we select an importance-range between 0.5 and 1. The 3D spatial view displays the respective regions in each member, as can be seen in Figure 10b. The voxels displayed there clearly indicate that the regions with most changes are located at the edges of the cube. The selection also gets highlighted in the nonlinearly scaled Hilbert line plot and the scaling widget shown in Figure 10c and 10d. The scaling widget visualizes the importance through the color coding (white to light-gray for the selection) and by the trapezoidal shapes, which are much broader at the top row as compared to the bottom. The background threshold is set to 30000. Intensities below this threshold correspond to air and are not of interest for this analysis. Figure 10c shows these background regions, which are marked with light orange boxes. The parameter p to influence the nonlinear scaling (see Section 3.2, Equation 2) is set to 1.4. For this dataset, this setting ensures a good balance between emphasizing regions with high variances, but still keeping background regions and regions with low variances visible.

5.2 Case study 2: Real-world XCT Dataset

We analyze the ensemble of 16 volumes from the foam specimen (see Section 4.2) as shown in Figure 9. Each volume of the ensemble is represented by a Hilbert curve of length 262144. The histogram heatmap displayed in Figure 11a shows that there is a broad variation in the lower intensities, indicated by the white region. There are less intensities in the upper range, and the histogram colors indicate lower frequencies there. This implies that most of the space is occupied by voids, the remaining space is left to the cell walls. It is also a hint that some volumes do not represent the cell walls very well.

When zooming in and viewing the 1D Hilbert line plots as shown in Figure 11b, one can see at a glance that the volumes differ vastly in their local intensity variation. The ensemble member corresponding to the bottom, dark green Hilbert line plot is nearly flat, indicating a low contrast of the intensities. The topmost, light-green plot shows highly varying intensities, revealing a high contrast. Figure 11c gives the functional boxplots of the 1D Hilbert line plots (shown in Figure 11b). The gray colored interquartile range covers the volumes with iterations from 250 to 650. The volume with 500 iterations is the median. In areas of low local intensity variation, the minimum and maximum of the functional boxplots are very close to each other. The background threshold is set to zero, because in this analysis scenario we are interested in low intensities as well. Since there is no compression due to the background, there are more voxels competing for available screen space. The parameter p to influence the nonlinear scaling (see Section 3.2, Equation 2) is increased to 2 in order to compress regions
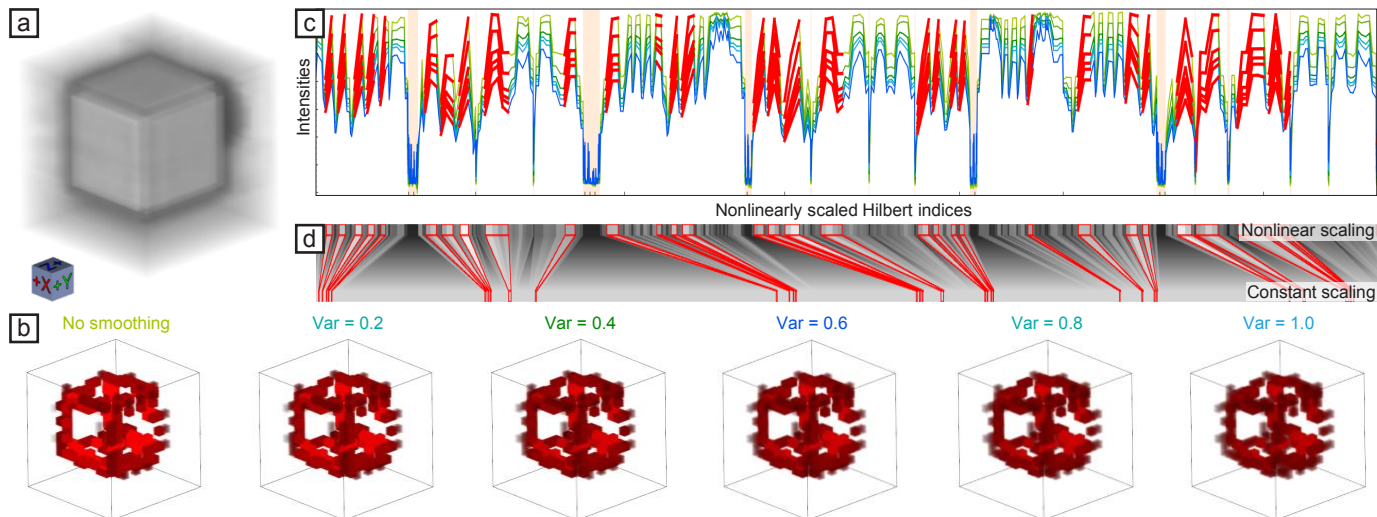
Fig. 10. (a) 3D rendering of the volume without smoothing. (b) 3D volume renderings of the selected voxels for all six ensemble members. (c) Corresponding Hilbert line plots with highly important regions selected. (d) Scaling widget with selection highlighted in red.

with low variance more and to expand interesting regions with high variance.

We select regions with an importance-value between 0.1 and 1 (high variation of the intensities). As can be seen in Figure 11d, only the cell walls show such high variations. Based on the importance-function, the cell walls can be separated easily. The selected indices are highlighted in the scaling widget shown in Figure 11e. Inspecting the Hilbert line plots (see Figure 11b) and the 3D views of the different parameter variations (see Figure 11d), we can see that the volumes produced by the SIRT algorithm with less than 250 iterations result in low contrast between cell walls and voids. Starting from approximately 350 iterations, the contrast converges to the FDK reference volume. Since doubling the number of iterations also means doubling the required reconstruction time, we can conclude that we can stop the reconstruction at 400 iterations without a major loss of accuracy.

When selecting regions with a low importance-value between 0 and 0.001 (low variation of intensities), the ring artifacts are selected, as shown in Figure 11f. This indicates that the SIRT iterations do not suppress the ring artifacts. Both FDK and SIRT are affected by them in exactly the same way. The importance-function could therefore be used to detect such artifacts. They are the only features in the volumes, which show only little variation with different reconstruction parameters. Figure 11g shows in the Hilbert line plots that in the lower intensity range there is in general a low variation in all the volumes. There is even less variation in the highlighted regions of the ring artifacts. We therefore hypothesize that there is potential to use the local ensemble variation to develop a ring-artifact reduction algorithm for the reconstructed volumes. This is an insight that would not have been possible without the knowledge gained from the analysis with *Dynamic Volume Lines*.

5.3 Performance Measurements

The experiments were performed on a desktop machine with an Intel(R) Core(TM) i7-3770 CPU, 32 GB RAM and an NVIDIA GeForce GTX 1080 with 8 GB RAM. The *Dynamic Volume Lines* were implemented in C++. ITK 4.9 has been used to perform basic image processing, to load the datasets, and to generate the Hilbert curves [41]. The 3D views were rendered with VTK 7.0. The histogram heatmap and the 1D Hilbert line plots were built with the QCustomPlot 2.0 library [5]. The scaling widget was implemented in Qt 5.8.

Computing the Hilbert curves for example for 16 datasets with a size of $64 \times 64 \times 64$ voxels (see Figure 8) takes approximately 12 seconds. This includes creating the nonlinear scaling, building the segment tree, and the initial rendering of the charts. Dragging and zooming within the charts works in real-time. The performance of selecting 1D Hilbert

line plots and rendering the corresponding 3D views depends on the number of chosen line segments and on the number of chosen volumes, but typically takes less than 5 seconds.

6 CONCLUSION AND FUTURE WORK

In this paper we introduce *Dynamic Volume Lines* for the interactive visual analysis and comparison of ensembles of 3D volumes using 1D Hilbert line plots. Volumes are linearized along a space-filling Hilbert curve. We introduce an aggregate overview visualization for volume ensembles as a histogram heatmap, which encodes the intensity frequencies. We provide nonlinear scaling to emphasize regions with high local variation, and to optimally utilize the available screen space. We illustrate the scaling in an interactive scaling widget. We investigate the usefulness of *Dynamic Volume Lines* with two case studies. Using a simulated XCT dataset, we investigate the general usefulness of the tool in detecting local variations in the ensemble. On a real-world foam dataset we showed that our importance-function, based on local variations, can be used to detect structures such as cell walls, and to discover unwanted ring artifacts.

Our domain expert collaborators were very positive about the possibility to compare multiple volumes at once, as they previously had no comprehensive tool available to support this analysis scenario. They also positively mentioned the guidance towards interesting regions, i.e., areas with high local variations. It was important for them to be able to investigate selected regions in more detail, and to retain the relation to the spatial domain. This allowed them to draw conclusions and gain insights for adaptations in their algorithm development.

Due to a technical implementation detail, the current prototype limits the maximum region of interest that can be analyzed to $256 \times 256 \times 256$ voxels. We are confident that with minor adaptations this limitation can be removed to support the analysis of much larger volumes. In general, the concept of representing an ensemble of volumes as nonlinear 1D Hilbert line plots is not limited to 3D space, but can also be applied to (abstract) n-dimensional spaces.
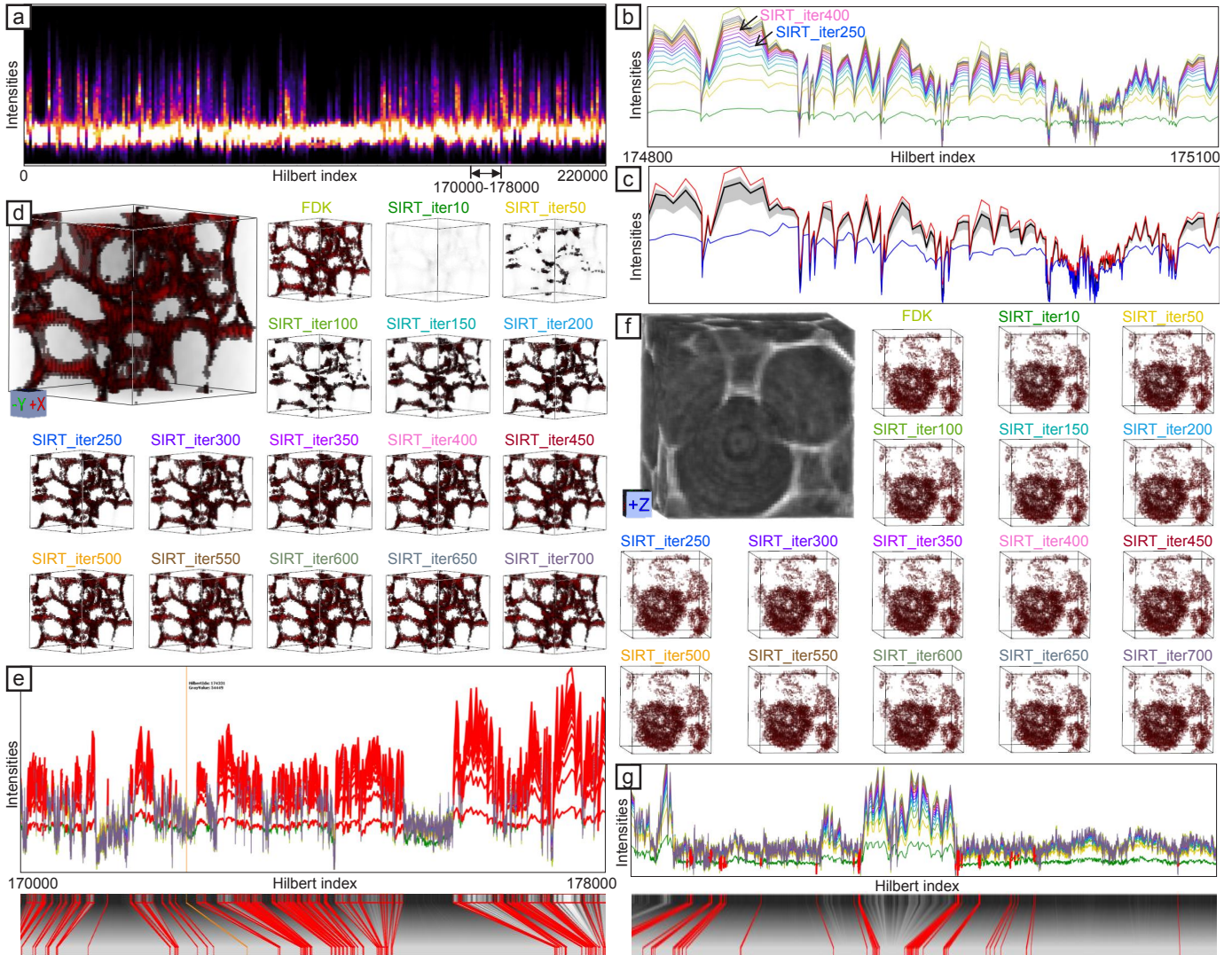
Fig. 11. (a) Histogram heatmap of the foam volumes ensemble showing most intensities in the lower range. (b) 1D Hilbert line plots zoom-in, enabling a detailed comparison of the reconstruction volumes. (c) Functional boxplots of the 1D Hilbert line plots in (b). (d) 3D views of regions with selected high importance, coinciding with the foam cell walls. (e) Regions of high importance as in (d), selected in the 1D Hilbert line plots. (f) 3D views of regions with selected low importance, coinciding with the ring artifacts. (g) Regions of low importance as in (f), selected in the 1D Hilbert line plots.

REFERENCES

[1] 7span. MaterialUI. https://www.materialui.co (accessed on 26-Jun-2018).

[2] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars. *Computational Geometry: Algorithms and Applications*, chap. More Geometric Data Structures, pp. 219–241. 2008. doi: 10.1007/978-3-540-77974-2_10

[3] I. Demir, C. Dick, and R. Westermann. Multi-Charts for Comparative 3D Ensemble Visualization. *IEEE Transaction on Visualization and Computer Graphics*, 20(12):2694–2703, 2014. doi: 10.1109/TVCG.2014.2346448

[4] I. Demir, M. Jarema, and R. Westermann. Visualizing the Central Tendency of Ensembles of Shapes. In *SIGGRAPH ASIA 2016 Symposium on Visualization*, SA '16, pp. 3:1–3:8, 2016. doi: 10.1145/3002151.3002165

[5] E. Eichhammer. QCustomPlot. http://www.qcustomplot.com/ (accessed on 26-Jun-2018).

[6] L. A. Feldkamp, L. C. Davis, and J. W. Kress. Practical cone-beam algorithm. *Journal of the Optical Society of America A*, 1(6):612–619, 1984. doi: 10.1364/JOSAA.1.000612

[7] B. Fröhler, T. Möller, and C. Heinzl. GEMSe: Visualization-Guided Exploration of Multi-channel Segmentation Algorithms. *Computer Graphics Forum*, 35(3):191–200, 2016. doi: 10.1111/cgf.12895

[8] M. G. Genton, C. Johnson, K. Potter, G. Stenchikov, and Y. Sun. Surface boxplots. *Stat*, 3(1):1–11, 2014. doi: 10.1002/sta4.39

[9] M. Gleicher, D. Albers, R. Walker, I. Jusufi, C. D. Hansen, and J. C. Roberts. Visual Comparison for Information Visualization. *Information Visualization*, 10(4):289–309, 2011. doi: 10.1177/1473871611416549

[10] R. Gosselin and D. Rodrigue. Cell morphology analysis of high density polymer foams. *Polymer Testing*, 24(8):1027 – 1035, 2005. doi: 10.1016/j.polymertesting.2005.07.005

[11] C. Gotsman and M. Lindenbaum. On the metric properties of discrete space-filling curves. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, vol. 3, pp. 98–102, 1994. doi: 10.1109/ICPR.1994.577130

[12] J. Gregor and T. Benson. Computational Analysis and Improvement of SIRT. *IEEE Transactions on Medical Imaging*, 27(7):918–924, 2008. doi: 10.1109/TMI.2008.923696

[13] C. H. Hamilton and A. Rau-Chaplin. Compact Hilbert Indices for Multi-Dimensional Data. In *1st International Conference on Complex, Intelligent and Software Intensive Systems*, pp. 139–146, 2007. doi: 10.1109/CISIS.2007.16

[14] C. H. Hamilton and A. Rau-Chaplin. Compact Hilbert Indices: Space-filling Curves for Domains with Unequal Side Lengths. *Information Processing Letters*, 105(5):155–163, 2008. doi: 10.1016/j.ipl.2007.08.034

[15] C. Heinzl and S. Stappen. STAR: Visual Computing in Materials Science. *Computer Graphics Forum*, 36(3):647–666, 2017. doi: 10.1111/cgf.13214

[16] M. Jarema, I. Demir, J. Kehrer, and R. Westermann. Comparative visual analysis of vector field ensembles. In *IEEE Visual Analytics Science and Technology*, pp. 81–88, 2015. doi: 10.1109/VAST.2015.7347634

[17] I. Jerjen, V. Revol, P. Schuetz, C. Kottler, R. Kaufmann, T. Luethi, K. Jefimovs, C. Urban, and U. Sennhauser. Reduction of phase artifacts in differential phase contrast computed tomography. *Optics Express*, 19(14):13604–13611, 2011. doi: 10.1364/OE.19.013604

[18] Z. Konyha, A. Lež, K. Matković, M. Jelović, and H. Hauser. Interactive Visual Analysis of Families of Curves Using Data Aggregation and Derivation. In *Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies*, i-KNOW '12, pp. 24:1–24:8, 2012. doi: 10.1145/2362456.2362487

[19] S.-T. Lee, C. B. Park, and N. S. Ramesh. *Polymeric Foams: Science and Technology*. CRC Press, 2006.

[20] N. Lindow, D. Baum, and H.-C. Hege. Perceptually Linear Parameter Variations. *Computer Graphics Forum*, 31(2pt4):535–544, 2012. doi: 10.1111/j.1467-8659.2012.03054.x

[21] M. M. Malik, C. Heinzl, and E. Gröller. Comparative Visualization for Parameter Studies of Dataset Series. *IEEE Transaction on Visualization and Computer Graphics*, 16(5):829–840, 2010.

[22] K. Matković, D. Gracanin, B. Klarin, and H. Hauser. Interactive Visual Analysis of Complex Scientific Data As Families of Data Surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1351–1358, 2009. doi: 10.1109/TVCG.2009.155

[23] P. Mindek, G. Mistelbauer, E. Gröller, and S. Bruckner. Data-sensitive Visual Navigation. *Computers and Graphics*, 67(C):77–85, 2017. doi: 10.1016/j.cag.2017.05.012

[24] M. Mirzargar, R. Whitaker, and R. Kirby. Curve boxplot: Generalization of boxplot for ensembles of curves. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2654–2663, 2014. doi: 10.1109/TVCG.2014.2346455

[25] M. Mlejnek, P. Ermes, A. Vilanova, R. van der Rijt, H. van den Bosch, F. Gerritsen, and E. Gröller. Profile Flags: a Novel Metaphor for Probing of T2 Maps. In *Proceedings of IEEE Visualization 2005*, pp. 599–606, 2005.

[26] B. Moon, H. v. Jagadish, C. Faloutsos, and J. H. Saltz. Analysis of the Clustering Properties of the Hilbert Space-Filling Curve. *IEEE Transactions on Knowledge and Data Engineering*, 13(1):124–141, 2001. doi: 10.1109/69.908985

[27] K. Moreland. Why We Use Bad Color Maps and What You Can Do About It. *Electronic Imaging*, 2016(16):1–6, 2016. doi: 10.2352/ISSN.2470-1173.2016.16.HVEI-133

[28] F. Pfeiffer, C. Kottler, O. Bunk, and C. David. Hard X-Ray Phase Tomography with Low-Brilliance Sources. *Physical Review Letters*, 98:108105, 2007. doi: 10.1103/PhysRevLett.98.108105

[29] H. Piringer, S. Pajer, W. Berger, and H. Teichmann. Comparative Visual Analysis of 2D Function Ensembles. *Computer Graphics Forum*, 31(3pt3):1195–1204, 2012. doi: 10.1111/j.1467-8659.2012.03112.x

[30] K. Potter, A. Wilson, P.-T. Bremer, D. Williams, C. Doutriaux, V. Pascucci, and C. R. Johnson. Ensemble-Vis: A Framework for the Statistical Visualization of Ensemble Data. In *IEEE Workshop on Knowledge Discovery from Climate Data: Prediction, Extremes*, pp. 233–240, 2009.

[31] M. Raj, M. Mirzargar, J. S. Preston, R. M. Kirby, and R. T. Whitaker. Evaluating Shape Alignment via Ensemble Visualization. *IEEE Computer Graphics and Applications*, 36(3):60–71, 2016. doi: 10.1109/MCG.2015.70

[32] M. Reiter, M. Erler, C. Kuhn, C. Gusenbauer, and J. Kastner. SimCT: a simulation tool for X-ray imaging. In *Proceedings of the 6th Conference on Industrial Computed Tomography*, 2016.

[33] J. Sanctorum, E. Janssens, A. den Dekker, S. Senck, C. Heinzl, J. D. Beenhouwer, and J. Sijbers. A workflow to reconstruct grating-based X-ray phase contrast CT images: application to CFRP samples. In *Proceedings of the 4th conference on X-ray and neutron phase imaging with gratings*, pp. 139–140, 2017.

[34] J. Schmidt, M. E. Gröller, and S. Bruckner. VAICo: Visual Analysis for Image Comparison. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2090–2099, 2013. Demo: https://users.cg.tuwien.ac.at/jschmidt/vaico/ (accessed on 26-Jun-2018).

[35] J. Schmidt, R. Preiner, T. Auzinger, M. Wimmer, E. Gröller, and S. Bruckner. YMCA - Your Mesh Comparison Application. In *IEEE Visual Analytics Science and Technology*, 2014. doi: 10.1109/VAST.2014.7042491

[36] M. Sedlmair, C. Heinzl, S. Bruckner, H. Piringer, and T. Möller. Visual Parameter Space Analysis: A Conceptual Framework. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2161–2170, 2014. doi: 10.1109/TVCG.2014.2346321

[37] S. Senck, B. Plank, J. Kastner, V. Revol, K. Dobes, and M. Scheerer. Non-destructive evaluation of defects in polymer matrix composites for aerospace applications using X-ray Talbot-Lau interferometry and micro CT. In *58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA SciTech Forum*, pp. 1–9, 2017.

[38] B. Shneiderman. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *Proceedings of the IEEE Symposium on Visual Languages*, VL '96, pp. 336–343, 1996.

[39] Y. Sun and M. G. Genton. Functional Boxplots. *Journal of Computational and Graphical Statistics*, 20(2):316–334, 2011. doi: 10.1198/jcgs.2011.09224

[40] T. Torsney-Weir, M. Sedlmair, and T. Möller. Sliceplorer: 1D Slices for Multi-dimensional Continuous Functions. *Computer Graphics Forum*, 36(3):167–177, 2017. doi: 10.1111/cgf.13177

[41] N. Tustison. Why do walks with Hilbert seem to take so long? Insight Journal, 2011. http://hdl.handle.net/10380/3324 (accessed on 26-Jun-2018).

[42] J. Weissenböck, A. Amirkhanov, E. Gröller, J. Kastner, and C. Heinzl. PorosityAnalyzer: Visual Analysis and Evaluation of Segmentation Pipelines to Determine the Porosity in Fiber-Reinforced Polymers. In *Proceedings of IEEE Visual Analytics Science and Technology*, pp. 101–110, 2016. doi: 10.1109/VAST.2016.7883516

[43] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. Contour Boxplots: A Method for Characterizing Uncertainty in Feature Sets from Simulation Ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2713–2722, 2013. doi: 10.1109/TVCG.2013.143