A Novel Approach for Immediate, Interactive CT Data Visualization and Evaluation using GPU-based Segmentation and Visual Analysis

author preprint, final version will be published at: `https://www.ndt.net/index.php`

H. Steinlechner[1], G. Haaser[1], B. Oberdorfer[2], D. Habe[2], S. Maierhofer[1], M. Schwärzler[3,1], E. Gröller[1,4]

[1]VRVis Research Center, Donau-City-Str. 11, 1220 Vienna, Austria, e-mail: hs@vrvis.at

[2]Austrian Foundry Research Institute, Parkstrasse 21, Leoben, Austria, e-mail: office@ogi.at

[3]Delft University of Technology   [4]Technische Universität Wien

Abstract

CT data of industrially produced cast metal parts are often afflicted with artefacts due to complex geometries ill-suited for the scanning process. Simple global threshold-based porosity detection algorithms usually fail to deliver meaningful results. Other adaptive methods can handle image artefacts, but require long preprocessing times. This makes an efficient analysis workflow infeasible. We propose an alternative approach for analyzing and visualizing volume defects in a fully interactive manner, where analyzing volumes becomes more of an interactive exploration instead of time-consuming parameter guessing interrupted by long processing times. Our system is based on a highly efficient GPU implementation of a segmentation algorithm for porosity detection. The runtime is on the order of seconds for a full volume and parametrization is kept simple due to a single threshold parameter. A fully interactive user interface comprised of multiple linked views allows to quickly identify defects of interest, while filtering out artefacts even in noisy areas.
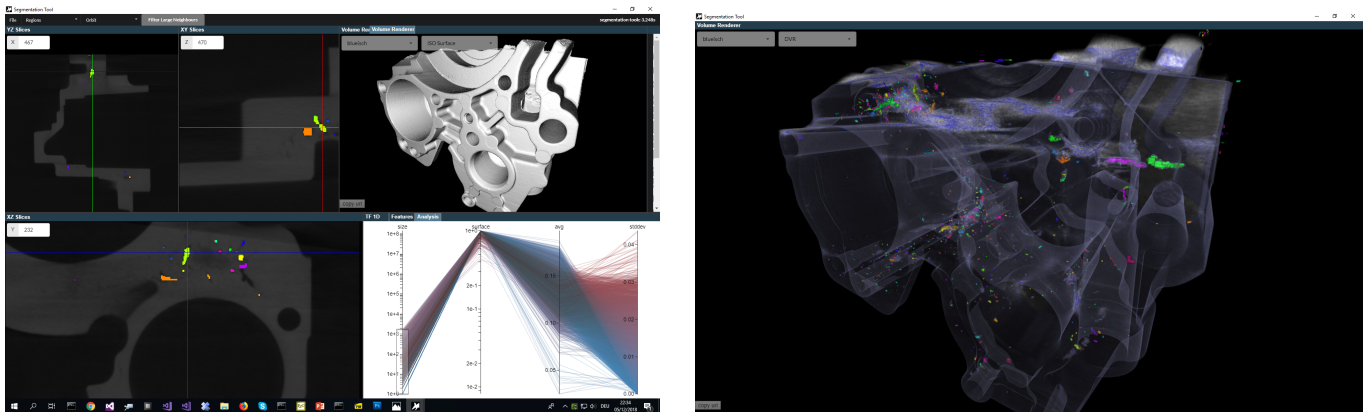
Figure 1: Left: Our novel segmentation tool performing a porosity analysis on an aluminum cast part with multiple linked views: 2D slices, Direct Volume Rendering, and interactive parallel coordinates. Right: CT data set rendered at real-time frame rates using Direct Volume Rendering.

Keywords: CT, GPU, Inclusion Detection, Interactive Visualisation, Visual Analysis, Parallel Coordinates, Volume Rendering

1 Introduction and Motivation

For the quality assessment of cast metal parts the accurate detection of casting defects such as pores or shrinkage cavities is a crucial factor. Computed tomography (CT) is a common method for this task. CT datasets of cast parts with non-ideal geometries are often afflicted with image artefacts, which make an effective defect segmentation rather difficult.

For instance, with beam hardening artefacts showing up as dark regions in the CT dataset, simple global threshold-based detection algorithms usually fail to deliver meaningful results. As an example, Figure 2 (left) shows a CT image of a common cast part, with a global threshold-based segmentation of the pores. False-positive pores are detected on the dark regions and false-negative pores are missed in the bright regions.

2 State of the Art

Volume rendering [12] is a standard technique in the field of non-destructive testing to detect and visualize volume defects [1–5]. In addition to standard 1D transfer functions, multi-dimensional transfer functions [13] can be used to visualize interesting parts and to emphasize different materials. Hadwiger et al.[6] use a region growing approach to extract regions of similar density (pores, defects etc). They use multiple parameters for their segmentation method, which can later be explored interactively by using multi-dimensional transfer functions. Weissenböck et al. [7] combine parameter space exploration for segmentation
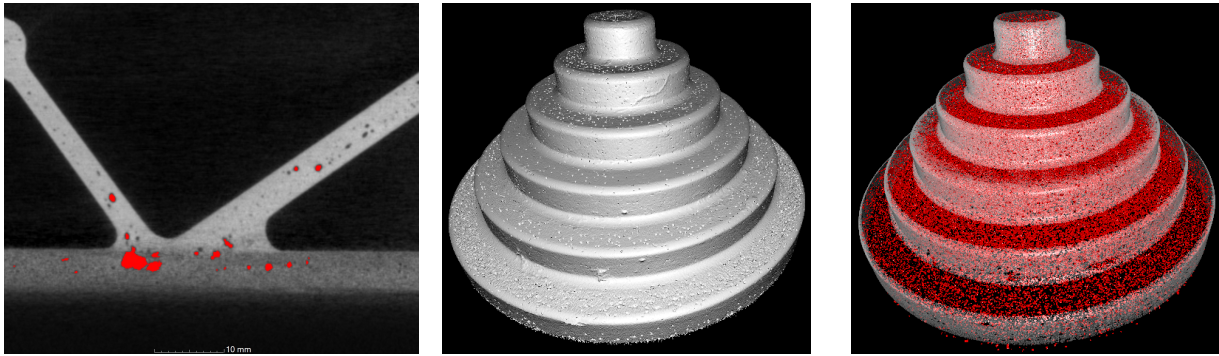
Figure 2: Global threshold-based segmentation applied to engine mount CT scan resulting in false-positive pores in the dark region and false-negative pores in the brighter regions (left). ISO surface rendering of the step cylinder used in our case study comprising artefacts (middle) and detected pores colored red in the Direct Volume Rendering (right).

parameters with highly customizable segmentation pipelines. Our system, in contrast, works with a simple single-parameter segmentation approach combined with real-time interactive analysis for a selected set of feature measures.

3 Our Approach: An Overview

In contrast to time-consuming parameter space exploration approaches, where a fixed result set can be filtered after a preprocessing step, we propose a novel interactive analysis framework. The main idea is to perform a very fast over-segmentation based on a single parameter with local threshold adjustment using a statistical region merging approach [8]. Afterwards we apply live filtering and merging building on the explored findings. The segmentation algorithm exploits the massive parallelism available on GPUs and can also handle out-of-core data larger than GPU- and system memory. For live exploration, we use a *Parallel Coordinates* view to visualize descriptive properties (e.g., size, average density, standard deviation over density, volume/surface ratio, etc.) of all detected artefacts. Through interactive filtering by brushing and selecting regions on the coordinates, a fast and intuitive exploration of the desired defects is made possible *without prior knowledge* of multiple analysis parameters.

After the initial segmentation, a user typically employs the *size* descriptor to filter out large features (such as the surrounding air). Furthermore, noise in the data is taken into account. Instead of prefiltering the data, we apply our algorithm to the unfiltered, raw input. Therefore, detected regions at this stage either consist of noise, or actual (possibly separated) defects. By interactively filtering out all tiny features with homogenous densities, i.e., *standard deviation* close to zero in the parallel coordinates view, the noise can be removed. We then perform a segmentation-refinement step, which re-applies the merging step, now based on the selected filters. This results in a high-quality segmentation that is robust to noise and non-uniform density values. More information about the system can be found online [11].

Our application combines multiple linked views (2D slices, Direct Volume Rendering, Parallel Coordinates, and Settings) into a single user interface (see Figure 1, left). For in-depth analysis, each view can be made full-screen (see Figure 1, right). The user interface is based on standard web technologies (e.g., Parallel Coordinates are implemented using D3 [9]) and can be used with any modern web browser. Our system runs locally on the user's machine. The architecture has been designed with cloud-based deployment in mind to make users independent of costly hardware (any device with a web browser works) and to enable collaborative and distributed work environments in the future. Cloud-based rendering will also address data transfer (next-generation CT detectors will approach 4000^3 voxels) and security issues.
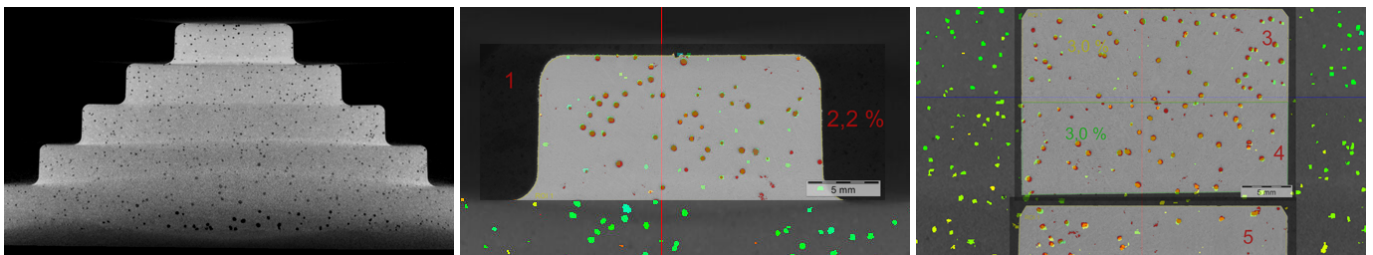

Figure 3: CT image of the step cylinder comprising artefacts (left) and comparison with a metallographic analysis overlaid to our segmentation result (middle, right).

4 Implementation

Our segmentation method is based on the well-known statistical region merging method [8]. Initially, each voxel constitutes a seed region. Subsequent merging steps, iteratively merge together adjacent regions based on some merging criterion. In contrast to the original algorithm we perform independent merge operations for one iteration in parallel, directly on the graphics card using vulkan compute shaders [10]. The main idea of our approach is to allow the user to interact after a first, initial, over-segmentation of the image in order to further refine the segmentation in subsequent steps. In the following sections we call regions, computed by the segmentation method, *features*. Our workflow is not tied to a particular implementation, but works with algorithms that can be modified to allow user interaction between iterations.

Given the input CT scan (stored in memory as a 16 bit 3D texture) the analysis phase associates each voxel with a *feature ID*. In an additional table we store following meta data for each feature, which we use in the Parallel Coordinates view for interactive exploration:

- **Size** is the number of voxels in the region.

- **Surface** describes the region form factor by the ratio of boundary and interior voxels.

- **Average** is the mean density value in the feature.

- **Standard deviation** of the density values in a region, used as measure for homogeneity.

4.1 Interaction and Visualization

Our tool provides a rather simple user interface consisting of multiple, dockable views for (1) 2D slices, (2) 3D Direct Volume Rendering, as well as (3) the Parallel Coordinates view for filtering segmentation results using the aforementioned feature properties (see Figure 1). In addition to Direct Volume Techniques our tool also provides a fixed set of predefined transfer functions, mapping feature properties such as standard deviation, feature surface, and feature IDs to colors. Often we are interested in features of a specific size and average density. For this purpose we provide a 2D transfer function widget similar to Hadwiger et al. [6]. Additionally we offer a gradient-magnitude/density widget for specifying a two dimensional transfer function. Figure 5 (right) shows the engine mount using a gradient magnitude transfer function and features colored using a two dimensional transfer function.

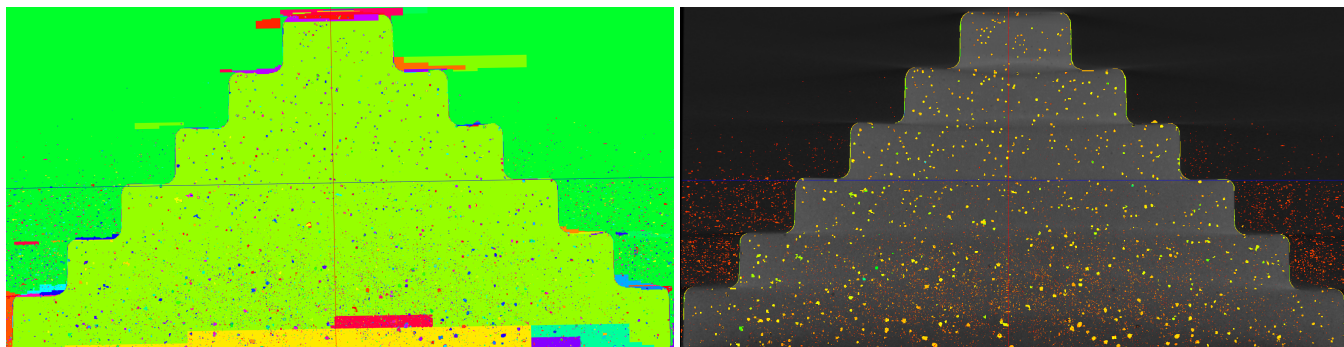## 5 Case Study: Analysis of a Step Cylinder

In this section we show how our system can be used to analyze CT scans in the presence of severe artifacts and noise. We created a worst-case test object, which is an artefact affected aluminium step cylinder with diameters ranging from 20 to 120 mm. It was constructed and cast with intentionally gas-induced melt in order to introduce volume defects such as pores (see Figure 3, left). Additionally, we use various existing component cast parts as test objects, in order to also cover a range of less problematic data sets (see Figure 1). All CT scans were performed on a GE/phoenix v|tome|x L equipped with a 240 kV microfocus cone-beam X-ray tube and a DXR 250 RT flat panel detector. The CT parameters of the cylinder scan are as follows: 180 kV, 200 μA and 0.5 mm Sn prefilter, no beam hardening correction, voxelsize 70 μm, dimensions: 1850 x 1850 x 1000 voxels, filesize 6.37 GB.

In the course of the workflow the volume data is segmented in such a way, that voxels constituting localized inclusions or pores can be detected, analyzed, and visualized, as well as optionally exported for further analysis in other scientific applications. By forming the ratio between pore voxel and scan object voxel numbers a simple global porosity value is calculated.
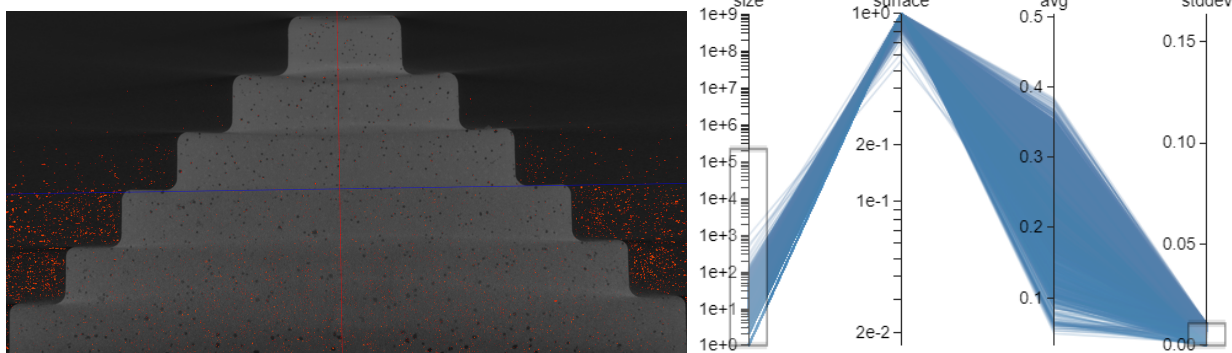
Our workflow is interactive, i.e. the user can immediately see the result of each action and can also quickly undo it. This makes it very easy to respond to individual peculiarities of scans and exploit them to accelerate the analysis and also to improve the quality.

In the following, we describe a representative workflow (see Figure 4) for analyzing the worst-case step cylinder scan described above.
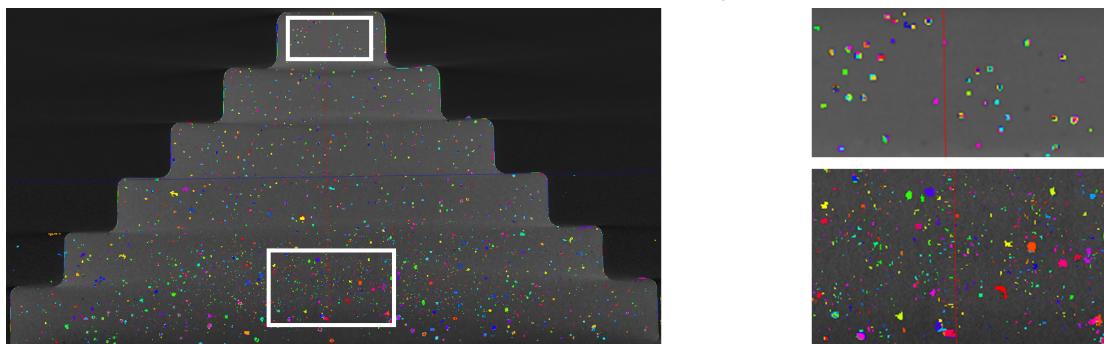
- First, we run the segmentation by using a local threshold value. In order to achieve good results, at this stage we aim to produce an over-segmented result (see Figure 4a). Typically the result consists of huge regions like the background, surrounding air, or large cavities, as well as lots of rather small features representing (potentially fragmented) pores. Optionally, since huge regions can easily be distinguished by their feature size, this initial result can already be used to automatically count voxels classified as surrounding air. After that, the huge regions can be filtered out, since we are not interested in analyzing the background.

- Second, depending on the noise level it might be useful to already remove small features, which can be definitely classified as noise. In this example we are able to remove noise by filtering out tiny features with low standard deviation (see Figure 4b). By inverting the filter we eliminate some but not all noise (see Figure 4c, left). Inspecting the results more closely, we find that especially in the upper areas with small diameters, pores are detected, but they are mostly fragmented. In contrast, the lower areas suffer from noise mixed with regions properly detected to be pores (see Figure 4c).
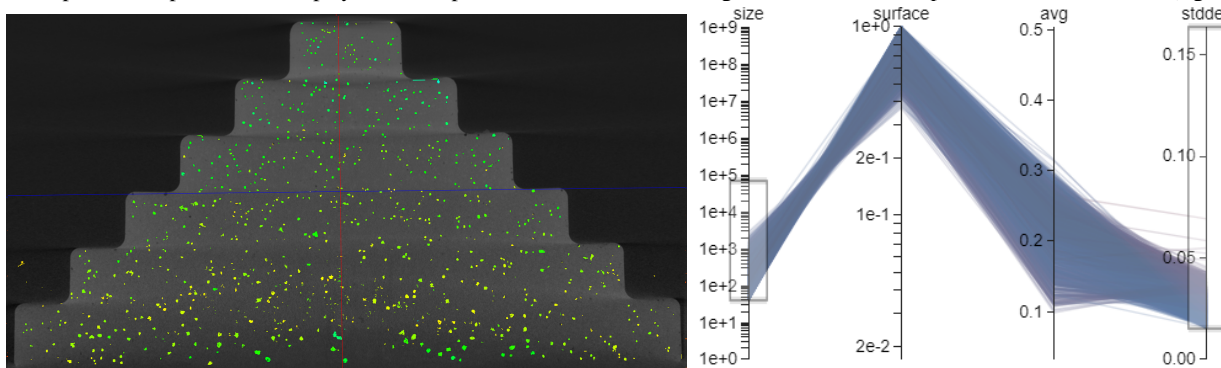
(a) Segmentation applied to step cylinder data set and individual features colored with random colors (left). The same slice after filtering out large regions (right).

(b) Optionally, noise can be filtered out if necessary. Here we used a combination of the size and standard deviation filters to identify noise (left), and the associated filter criterion in the Parallel Coordinates view (right).

(c) Noise removed by inverting the *stddev* filter shown in (b). The remaining features are rendered with random colors (left). Enlarged details of Step 1 and Step 5-6 of the step cylinder. Step 1 still suffers from over-segmentation, while Step 5-6 still contains noise (right).

(d) The final segmentation result (left) after re-applying the merging procedure and the currently active filters (right).

Figure 4: The proposed workflow applied to an artefact affected six-step aluminum cylinder with diameters ranging from 20 to 120 mm. After removing noise, the resulting features are merged again in order to eliminate remaining noise and produce the final segmentation result.

- Third, as filtering out small features would also eliminate desired features as well, we stop for the moment and re-run the segmentation process, which respects the active filters and merges connected components. With the background already removed, this effectively merges fragmented features encountered in the brighter areas with small diameters. Since feature properties might have changed due to the merging procedure, we again employ filters based on size and stddev in order to cancel out noise again, if any. In this stage we can finally filter the results for features, e.g. of specific density, size or form. It is crucial that all interactions can be double-checked with the interactively updated visualizations. In our example, the final results are given in Figure 4d. This is also the result we use for our ground-truth comparison in Section 6.

6 Evaluation

In order to evaluate real-world practicability, we applied our method to the previously shown engine mount cast part. In contrast to the global-threshold based segmentation method (see Figure 2 (left)) the volume defects could be detected in both, dark and bright areas. The final analysis results can be seen in Figure 5.
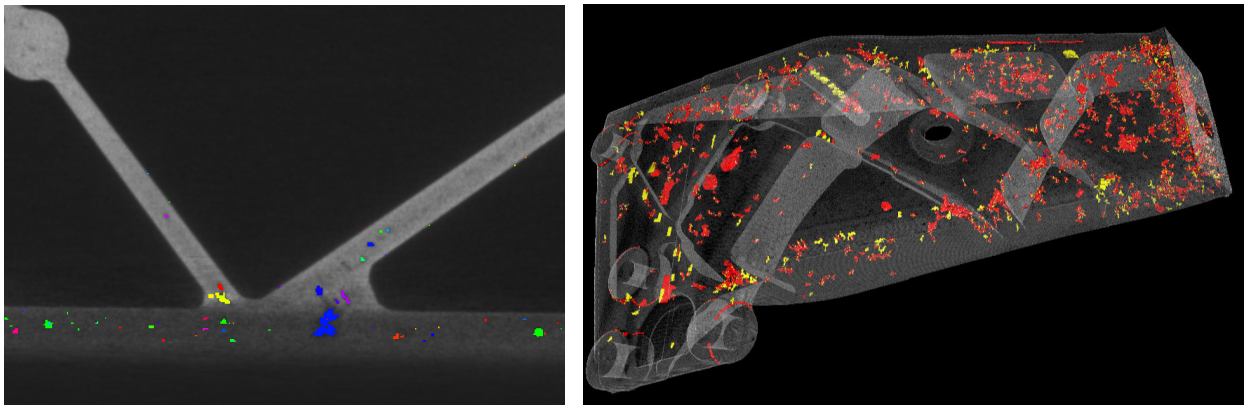

Figure 5: Pores detected with our local region merging method. Pores are detected in both, the dark as well as the brighter regions (left). Direct Volume Rendering using a two dimensional gradient-magnitude/density transfer function, big pores marked in red, smaller ones in yellow (right).

In order to check the accuracy of the porosity analysis, the cast step cylinder was cut, subdivided into six regions corresponding to the six steps, embedded, ground, polished and analysed metallographically. The determined porosity values were used as ground-truth data for analysis by means of CT. Since the initial segmentation uses a single parameter we quickly found the appropriate value. We followed the workflow presented in Section 5 and achieved an acceptable segmentation after a few user interactions. For evaluation, we successfully aligned digital image data and the physical cut used for metallography in order to double-check the segmentation result (see Figure 3, right). In the brighter, upper areas of the dataset our analysis result and the metallography agreed mostly. However, an exact congruence with metallography due to the limited voxel resolution in CT is hard to achieve. In the lower, darker areas with big diameters we optically detected some false positives owed by some noise, reflected in a low contrast to noise-ratio of 3.2 in the thickest step.

7 Future Work and Conclusions

In this paper we present a workflow for segmentation and analysis of industrial CT data based on user interaction instead of fully automated image processing. In our experiments we showed that our approach works for pores that are typically hard to detect due to image artifacts. We achieve high-quality results even in presence of artefacts, as shown by our comparison with metallography. Since our segmentation method does not perform any pre-filtering, we detect even small pores while noise can be filtered out interactively. The software prototype is currently used and tested at the Austrian Foundry Research Institute on a daily basis. Due to the explorative, interactive approach and the simplicity, users adapted to the workflow very fast und successfully. This is also driven by the fact that the high performance segmentation (3-16 seconds for the test data sets) outperforms existing tools, increasing productivity significantly. In future work we plan to explore additional feature parameters such as distance to surface as well as interaction methods for specifying regions of interest.

References

[1] G. Geier, M. Hadwiger, T. Höllt, L. Fritz, T. Pabel, Interaktive Exploration und Quantifizierung von Ungänzen in komplexen Bauteilen, Conf. Proc. Industrielle Computertomographie, Wels, Austria, pp. 103-108, 2008

[2] VG Studio Max http://www.volumegraphics.com/en/products/vgstudio-max.html, 2018 (accessed 20 June 2018).

[3] Amira/Avizo http://www.zib.de/software/amira, (accessed 20 June 2018).

[4] Open_iA http://www.computer-tomographie.at/cms2/index.php/en/software-en/, (accessed 20 June 2018).

[5] MAVI https://www.itwm.fraunhofer.de/de/abteilungen/bv/produkte-und-leistungen/mavi.html, (accessed 20 June 2018).

[6] M. Hadwiger, L. Fritz, C. Rezk-Salama, T. Höllt, G. Geier, T. Pabel, Interactive Volume Exploration for Feature Detection and Quantification in Industrial CT Data, IEEE TVCG, 14(6) pp. 1507-1514, 2008

[7] J. Weissenböck, A. Amirkhanov, E. Gröller, J. Kastner, C. Heinzl, PorosityAnalyzer: Visual Analysis and Evaluation of Segmentation Pipelines to Determine the Porosity in Fiber-Reinforced Polymers, IEEE VAST 2016, pp 101-110, 2016.

[8] R. Nock, F. Nielsen, Statistical Region Merging, IEEE TPAMI, 26(11): 1–7., 2004

[9] M. Bostock, V. Ogievetsky, J. Heer D3, Data-Driven Documents, IEEE TVCG, 17(12), pp 2301-2309, 2011

[10] Vulkan spec 1.0 specification https://www.khronos.org/registry/vulkan/specs/1.0/pdf/vkspec.pdf, (accessed 07 December 2018).

[11] https://aardvark-community.github.io/interactive-segmentation/, (accessed 07 December 2018).

[12] K. Engel, M. Hadwiger, J. M. Kniss, C. Rezk-Salama, D. Weisskopf, Real-time Volume Graphics, A. K. Peters Ltd/CRC Press, 2006

[13] J. Kniss., G. Kindlmann, C. Hansen: Multidimensional Transfer Functions for Interactive Volume Rendering. IEEE TVCG 8, 3 (July 2002), 270–285