Quantitative analysis of microstructures

MAVI

Modular Algorithms
for Volume Images

User manual

Fraunhofer ITWM

July 24, 2020

Fraunhofer-Institut für Techno- und Wirtschaftsmathematik ITWM, Abteilung
Bildverarbeitung, Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany,
www.mavi-3d.de, mavi@itwm.fraunhofer.de

Contents

1 **MAVI** 's purpose and scope

MAVI is a software for processing and analysis of volume images as produced e.g. by micro computed tomography. It is a laboratory tool intended to complement any device for 3D image acquisition.

MAVI is particularly suited for quantitative analysis of materials microstructures like fiber reinforced composites, foams, non-woven, concrete, powders. Due to its modular design and the universal basic geometric concepts used, **MAVI** is capable to deal with three-dimensional images of other structures like bone or snow too.

MAVI focuses on the characterization of the complex geometry of microscopically heterogeneous structures. Volume, surface, integrals of curvature and Euler number are determined for the whole structure or isolated objects. Anisotropies and preferred directions are not only found but their strength is measured, too. Special functions have been tailored for the analysis of open foams or fiber reinforced composites. Metrology in the sense of dimensioning of parts and comparison between actual and nominal dimensions of a part is not in **MAVI** 's scope.

MAVI 's analysis core is complemented by various image processing and segmentation techniques. 2D slices provide fast visual control of processing steps while the volume rendering yields a spatial visualization.

This handbook intends to

- give the non-expert user a starting point to get into **MAVI**
- give the experienced user detailed information on the algorithms used
- provide a reference for finding the appropriate processing chain for a particular problem.

2 **MAVI** versions and licensing models

2.1 Demo version

A demo version of **MAVI** is available via www.mavi-3d.de (center column, first item in download list).

The demo version is fully functional except all functions for saving or exportingimage or analysis data being disabled.

The image size you can work is restricted to 128^3 pixels,however. The **Import**:Image Raw Data dialog offers the possibility to Crop such a sniplet out of your 3D image data.

2.2 Full license

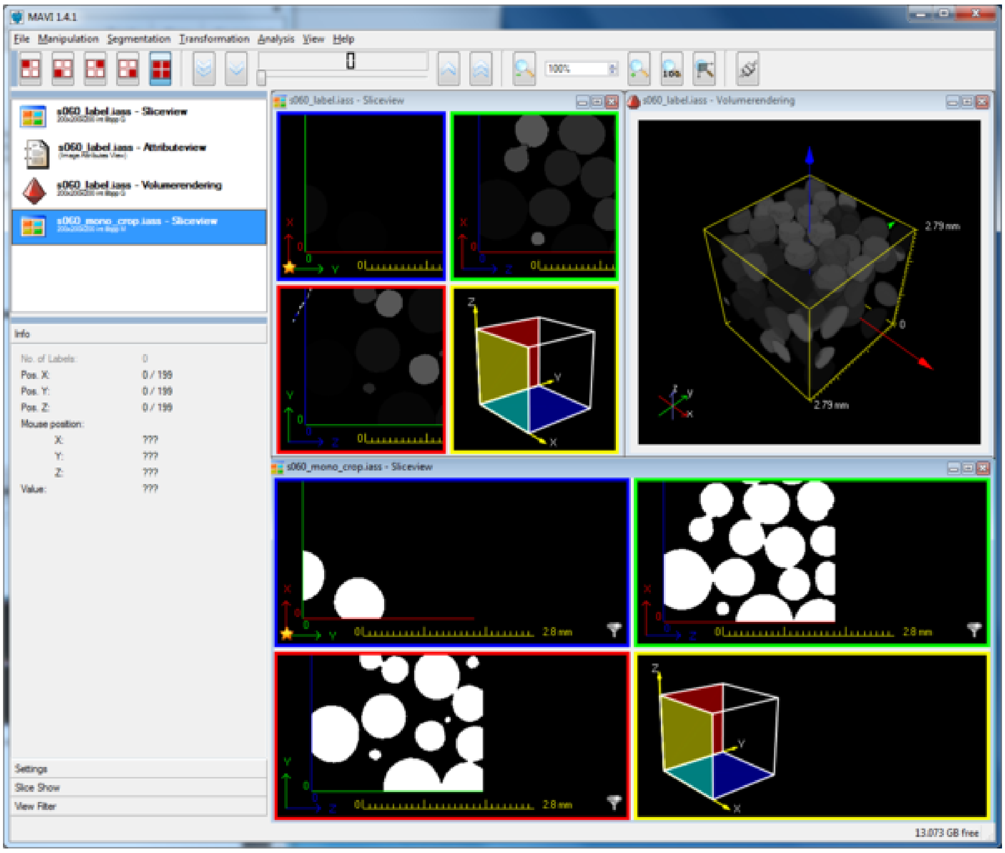
MAVI 's current licensing model is a single user floating license controled by an USB dongle. To run a fully licensed **MAVI** , you need this special dongle, the matching license file, and the **MAVI** installer (Windows) or rpm (Linux). Please contact mavi@itwm.fraunhofer.de to get information on how to obtain them.

If the dongle or a valid license file is missing, **MAVI** issues a warning and proceeds in demo mode.

3 Framework

MAVI has a multi document interface, which allows for processing of multiple documents at the same time. The application main window provides menus, a toolbar, several docking windows and a statusbar surrounding a large central workspace, see Figure 1.

The main menu provides access to the algorithms. See Sections 8.1-8.7 for detailed descriptions of the groups:

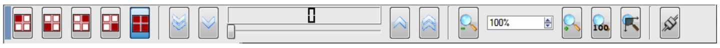

MAVI 1.4.1
File Manipulation Segmentation Transformation Analysis View Help
100%
s060_label.iass - Sliceview
s060_label.iass - Attributeview
s060_label.iass - Volumerendering
s060_mono_crop.iass - Sliceview
Info
Settings
Slice Show
View Filter
2.79 mm
2.8 mm
13.073 GB free

No. of Labels: 0
Pos. X: 0 / 199
Pos. Y: 0 / 199
Pos. Z: 0 / 199
Mouse position:
X: ???
Y: ???
Z: ???
Value: ???

Figure 1 **MAVI** 's main window.

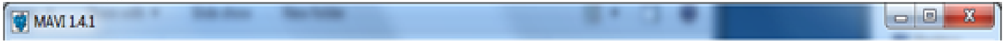
- File
- Manipulation
- Segmentation
- Transformation
- Analysis
- View
- Help.

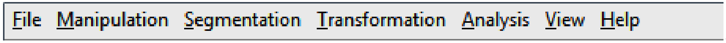
A menu entry is active (i.e. can be chosen by clicking on it) only if the corresponding function can be applied to the selected image (that is, if the image is of a valid gray value typeand size).

The toolbar  offers control functionality for the current view window. For more details, please refer to the Slice View Section 8.6.1.

The central workspace contains all view windowsavailable in the application. Each view is associated with a specific document and displays its content. The foremost view of the workspace is assigned to the active document. All operations will be applied to this document. Activate a window by clicking on it or its symbol in the view navigator window. It is possible to have different kinds of views for one document at the same time. By default, each view is maximized. That means the view fills the complete workspace when it appears for the first time.

Navigation in the view navigator window by cursor is not possible. Under Windows operating systems this can however easily be done using Ctrl and Tab keys.



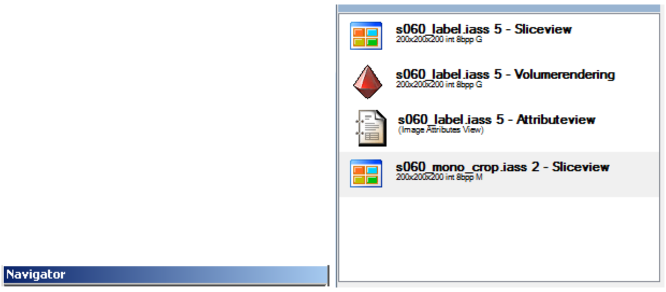


The titlebar of a view window consists of the view icon on the left hand side, which provides a window menu when clicked, the name of the view window, and three control buttons on the right hand side. The window menu allows you to close, resize and move the window. The ▁ and □ buttons minimize and miximize the window, respectively, and the ✕ button closes it. If the last view window of a document is closed, the document is closed too. When a window is maximized, the view icons and the control buttons are displayed on the left and right hand side of the menu bar . In this case, the □ button turns into a 🗗 button which restores the original size of the window when clicked.

At startup the **MAVI** framework has two windows docked on the left border of the workspace, a view navigator (Figure 2(b)) on the top and a view control window below. The windows can be moved around by using the mouse on the docking edge (Figure 2(a)). If the window is not docked to a side of the workspace, then it becomes a floating standalone window. By clicking on a docking window or on the menu bar with the right mouse button you obtain a control list of all available docking windows which allows to activate and deactivate them.

The view navigator window shows a list of all available views. A list entry consists of the view icon, the view label and an additional description provided by the view. The active entry is marked by a different background color. When an entry is selected, the assigned view becomes active and is moved to the foreground of the workspace.

The view control window provides several functions and information for the current view. Its content is different for different kinds of views. See Sections 8.6.1- 8.6.3 for details on the particular view types and Figure 45 for the view control window of the Slice View.



s060_label.iass 5 - Sliceview
200x200x200 int 8bpp G

s060_label.iass 5 - Volumerendering
200x200x200 int 8bpp G

s060_label.iass 5 - Attributeview
(Image Attributes View)

s060_mono_crop.iass 2 - Sliceview
200x200x200 int 8bpp M

(a) Docking Edge

(b) View Control

Figure 2 **MAVI** 's framework: Docking Edge and View Control Window.

4 Fundamental concepts

To take advantage of **MAVI** 's full potential, a basic understanding of the following concepts is needed. However, the material presented here is certainly not exhaustive. Thus references to the relevant literature are provided for the user interested in more background information.

4.1 Images in MAVI

MAVI is built on the idea of pixels (or voxels) being the vertices of a grid and holding information on the image content as gray values. More precisely, let $\mathbb{L} = (s_1, s_2, s_3)\mathbb{Z}^3$ be a three-dimensional cuboidal lattice with lattice spacings $s_1, s_2, s_3 > 0$. For data from micro-computed tomography, it holds usually that $s_1 = s_2 = s_3$ and with a slight abuse of notation one speaks of an “isotropic” lattice. Analogously, lattices with different spacings are refered to as “anisotropic” lattice.

MAVI 's core analysis functions as well as essential transforms like the Euclidean distance tranform, work for image data with pairwise different lattice spacings e. g. from so-called FIB tomography, too. However, filters as well as morphological transforms are pixel based and thus do not correct for “anisotropic” lattice spacings.

4.1.1 Image attributes

An image is not just described by its pixel's gray values. **MAVI** stores the following image attributes in the image header:

Creator A string containing information about the image creator.
Description A string containing information about the content of the image.
History A string summarizing the transforms applied to the image.
Offset A triple of integer values allowing to relate to some global coordinate system.
Spacing A triple of floating point values describing the physical size of pixel spacings or edges in [m].

The image header entries will either be created when loading an image from a file 8.1 containing the respective information or can be edited by the user via the Edit Header function. Before applying analysis functions, it should always be ensured that the image spacings are set correctly. If no spacing is set, the base unit for all measurements is [Pixel].

4.1.2 Image types

So far, **MAVI** deals with scalar images only. The only exception from this rule are COMPLEX images as arising from the Fast Fourier Transformation.

image type	description	range
MONO	just 2 components – foreground and background	0 (background), 1 (foreground)
GRAY8	8 bit integer gray values	$[0, 255]$
GRAY16	16 bit integer gray values	$[0, 65.535]$
GRAY32	32 bit integer gray values	$[0, 4.294.967.295]$
GRAYF	floating point gray values (single IEEE754 precision)	approx. $\left[1.17e^{-38}, 3.40e^{+38}\right]$
COMPLEX	complex floating point gray values (single IEEE754 precision)	approx. $\left[1.17e^{-38}, 3.40e^{+38}\right]^2$ for both real and imaginary components

Table 1 **MAVI** 's image types

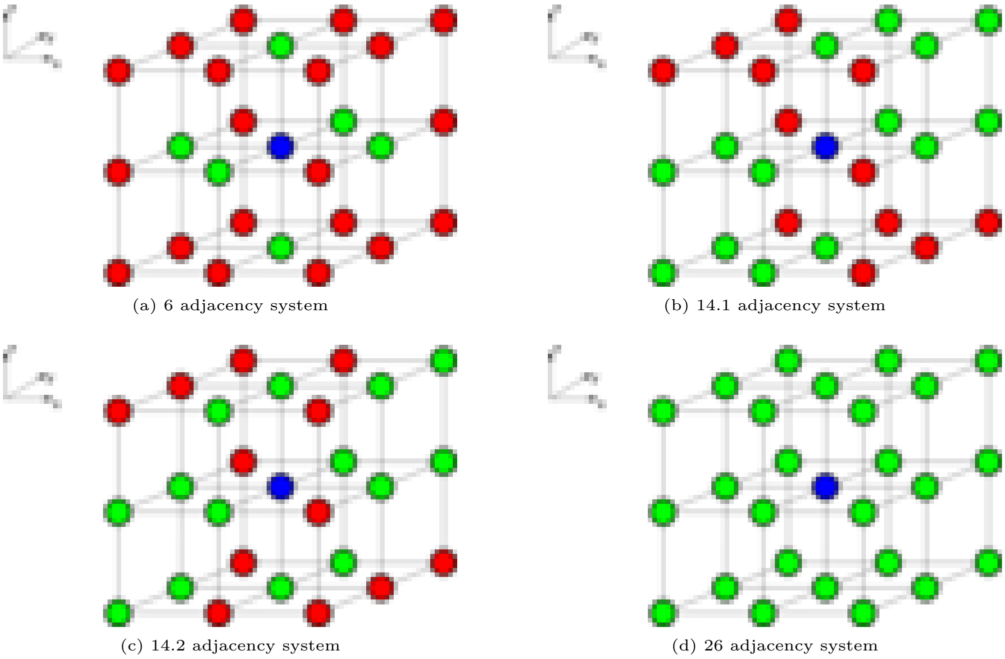
4.2 MAVI 's coordinate system

MAVI has a right-handed coordinate system and the origin $(0, 0, 0)$ is interpreted to be the lower left front corner of the cuboid representing the observation window, see also Slice View.

When importing image data**MAVI** assumes the data to be X local if not explictely stated otherwise by the original image format. That is, the data is read with the X coordinate running the fastest and the Z coordinate the slowest resulting in e. g. pixel $(1, 0, 0)$ being read before pixel $(0, 0, 1)$.

4.3 Discrete connectivity by adjacency systems

Usually, discrete neighborhood graphs are used to define how the pixels of an image are connected. However, in 3D, neighborhood graphs do not yield a unique description anymore. Therefore **MAVI** uses adjaceny systems consisting of vertices, edges, faces, and cells instead. For 3D images, the adjacency systems described in [12] are used – 6, 14.1, 14.2, and 26 adjacency system, see Figure 3 for a sketch. For 2D images, **MAVI** uses the 4, 6.1, 6.2, and 8 adjacency systems.



(a) 6 adjacency system

(b) 14.1 adjacency system

(c) 14.2 adjacency system

(d) 26 adjacency system

Figure 3 3D adjacency systems, direct neighbors (green) of the central pixel (blue).

There is no overall best adjacency system, see [10] for a rigourous proof. It is important to be consistent, in particular to segment using the same adjacency system as used later on e. g. for a connectivity analysis. When aiming at separating image components, the 6 or 4 adjacency systems are a better choice than the 26 or 8 adjacency systems, respectively. It's exactly the other way round when the goal is to preserve thin connections.

4.4 Directions given by the cuboidal lattice

The unit cell of the cuboidal lattice determines a number of discrete directions. **MAVI** uses two sets of 13 (3D) or 8 (2D) directions for morphological transforms 8.4.2 and some global analyses provided by the Field Features and Subfield Features. One set of 13 directions in 3D is given by lines connecting vertices of the lattice unit cell – 3 coordinate directions, 6 face diagonal directions, 4 space diagonal directions as specified in the following Table 2. For an illustration see Figure. The spacing vector(s_1, s_2, s_3) is the one introduced in 4.1 above.

The second set of 13 directions in 3D, used by Field Features and Subfield Features are the normal directions to planes defined by vertices of the unit cell. In the case of an isotropic lattice, the two sets of directions coincide, just the order differs. However, for anisotropic lattices this does not hold and therefore

Direction	vector	description
00	$(1, 0, 0)$	parallel to the x-axis
01	$(0, 1, 0)$	parallel to the y-axis
02	$(0, 0, 1)$	parallel to the z-axis
03	$(s_1, s_2, 0)/\sqrt{s_1^2 + s_2^2}$	face diagonal
04	$(-s_1, s_2, 0)/\sqrt{s_1^2 + s_2^2}$	face diagonal
05	$(s_1, 0, s_3)/\sqrt{s_1^2 + s_3^2}$	face diagonal
06	$(-s_1, 0, s_3)/\sqrt{s_1^2 + s_3^2}$	face diagonal
07	$(0, s_2, s_3)/\sqrt{s_1^2 + s_3^2}$	face diagonal
08	$(0, -s_2, s_3)/\sqrt{s_1^2 + s_3^2}$	face diagonal
09	$(s_1, s_2, s_3)/\sqrt{s_1^2 + s_2^2 + s_3^2}$	space diagonal
10	$(-s_1, s_2, s_3)/\sqrt{s_1^2 + s_2^2 + s_3^2}$	space diagonal
11	$(s_1, -s_2, s_3)/\sqrt{s_1^2 + s_2^2 + s_3^2}$	space diagonal
12	$(-s_1, -s_2, s_3)/\sqrt{s_1^2 + s_2^2 + s_3^2}$	space diagonal

Table 2 Discrete directions given by the unit cell of the cuboidal lattice.

the normal directions are listed in Table 3 below.

For 2D images, the 8 directions – 4 coordinate directions and 4 diagonal directions – are given by their angle to the x-axis in radians.

4.5 Edge treatments

Image transforms like filters or morphological transforms have to apply some kind of edge treatment for determining the new values of the pixels at or close to the image borders.

The following table lists the different edge treatments offered by **MAVI** 's algorithms. The initials are those appearing in the image history, see 4.1.1. The choice of available edge treatments depends on the respective algorithm.

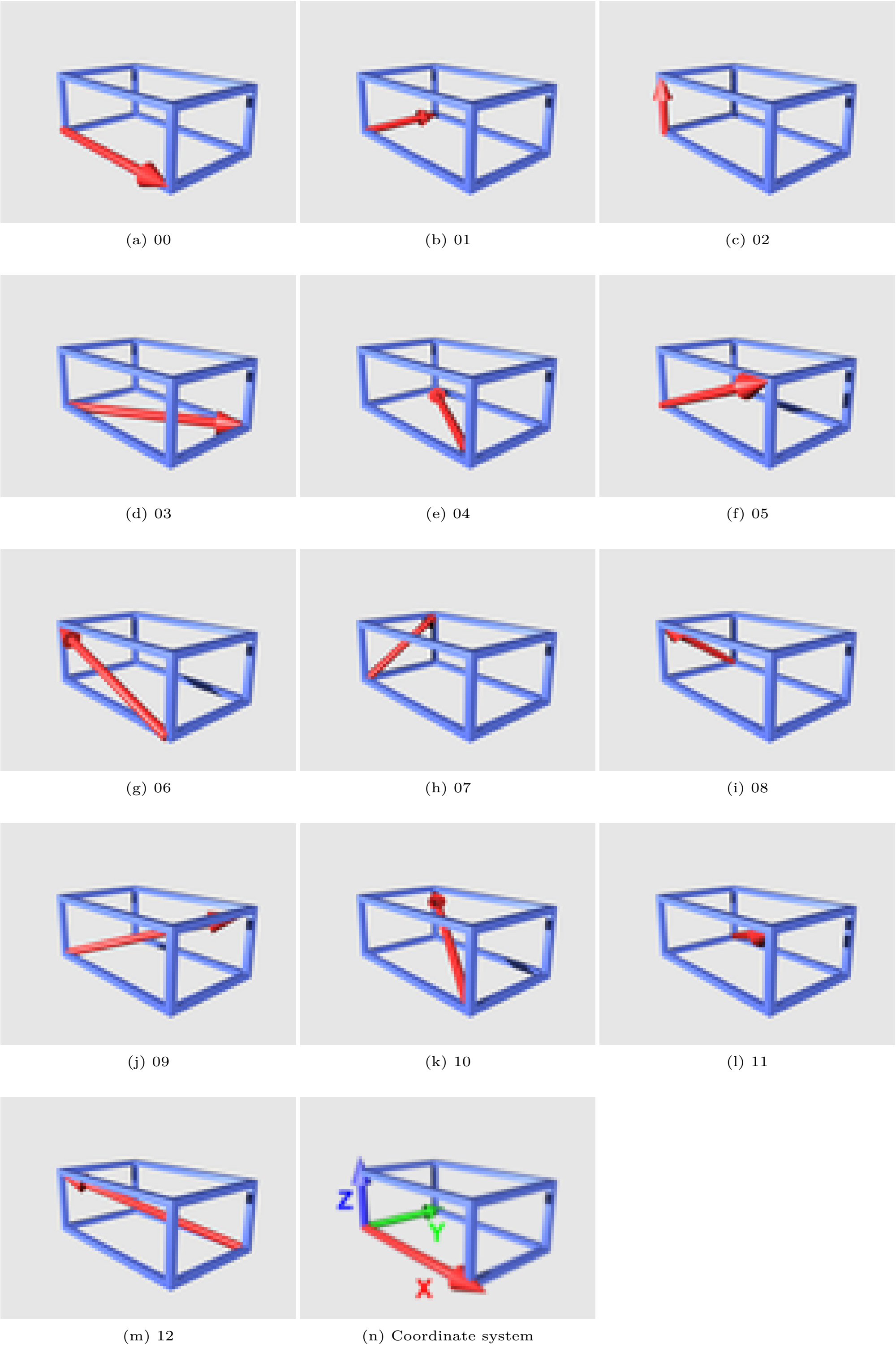
Remark 1 The edge treatments given here refer to image transforms only. See Section 8.5 for the edge correction options of the analysis algorithms.

Direction	vector	description
00	$(0,0,1)$	normal to xy-plane
01	$(0,1,0)$	normal to xz-plane
02	$(1,0,0)$	normal to yz-plane
03	$(0,-s_3,s_2)/\sqrt{s_2^2+s_3^2}$	normal to plane cutting 4 vertices
04	$(0,s_3,s_2)/\sqrt{s_2^2+s_3^2}$	normal to plane cutting 4 vertices
05	$(-s_3,0,s_1)/\sqrt{s_1^2+s_3^2}$	normal to plane cutting 4 vertices
06	$(s_3,0,s_1)/\sqrt{s_1^2+s_3^2}$	normal to plane cutting 4 vertices
07	$(s_2,s_1,0)/\sqrt{s_1^2+s_2^2}$	normal to plane cutting 4 vertices
08	$(-s_2,s_1,0)/\sqrt{s_1^2+s_2^2}$	normal to plane cutting 4 vertices
09	$(-s_2s_3,-s_1s_3,s_1s_2)/\sqrt{(s_1s_2)^2+(s_1s_3)^2+(s_2s_3)^2}$	normal to plane cutting 3 vertices
10	$(s_2s_3,-s_1s_3,s_1s_2)/\sqrt{(s_1s_2)^2+(s_1s_3)^2+(s_2s_3)^2}$	normal to plane cutting 3 vertices
11	$(s_2s_3,s_1s_3,s_1s_2)/\sqrt{(s_1s_2)^2+(s_1s_3)^2+(s_2s_3)^2}$	normal to plane cutting 3 vertices
12	$(-s_2s_3,s_1s_3,s_1s_2)/\sqrt{(s_1s_2)^2+(s_1s_3)^2+(s_2s_3)^2}$	normal to plane cutting 3 vertices

Table 3 Discrete normal directions given by the unit cell of the cuboidal lattice.

Edge Treatment	Description	Abbrev.
No Edge Treatment	No edge treatment employed	NET
Embedded	The image is embedded into background; the background value is set by the algorithm.	EMB
Periodic	The image data is extended periodically.	PER
Reflective	The image data is extended by reflection on the border pixels.	REFL
Reflective Double Edge	The image data is extended by reflection outside the image borders, i.e. border pixels are repeated.	RDE

Table 4 Edge treament options for image transforms.



(a) 00

(b) 01

(c) 02

(d) 03

(e) 04

(f) 05

(g) 06

(h) 07

(i) 08

(j) 09

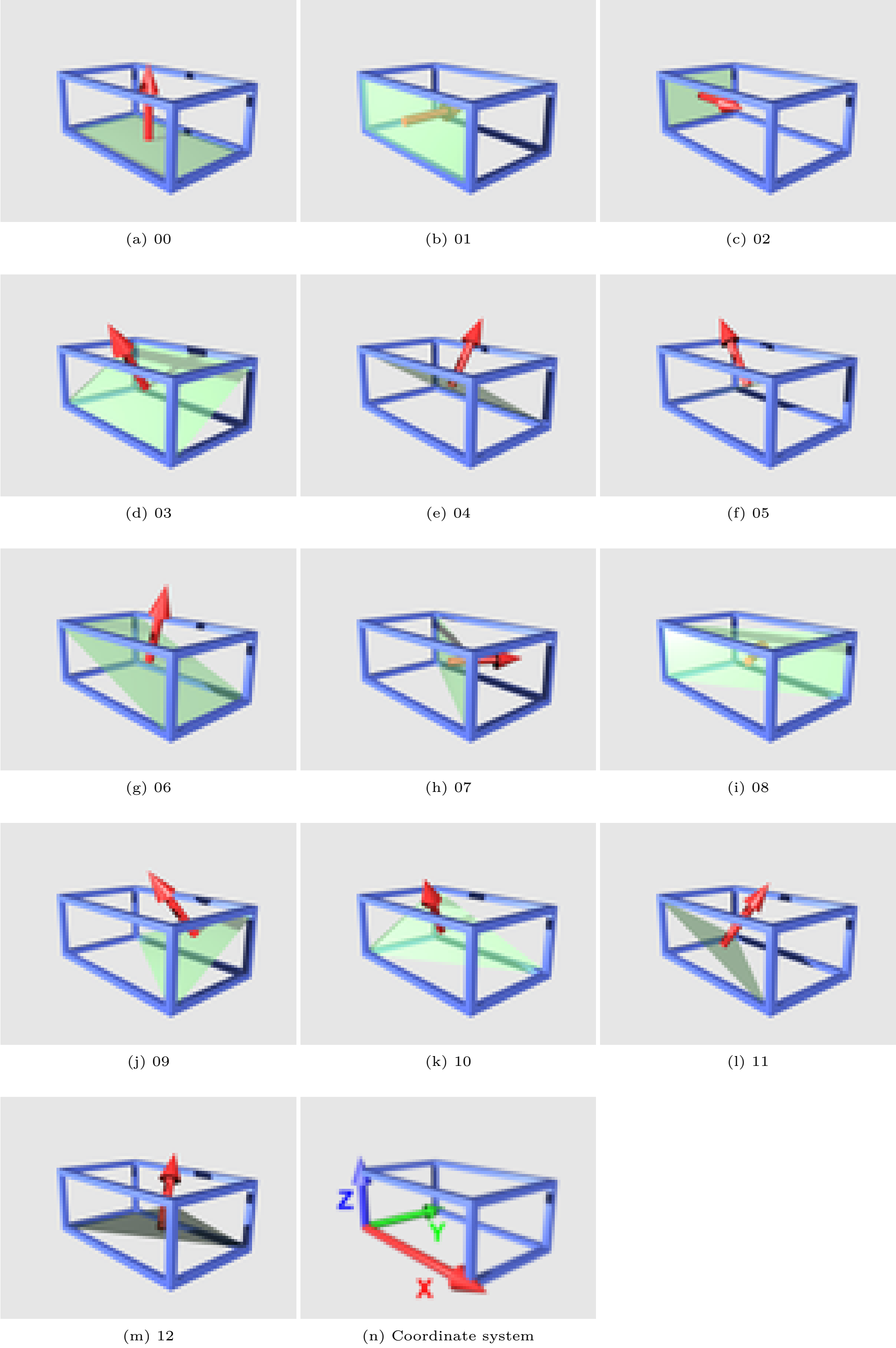
(k) 10

(l) 11

(m) 12

(n) Coordinate system

Figure 4 Discrete directions given by the unit cell of the cuboidal lattice.



(a) 00

(b) 01

(c) 02

(d) 03

(e) 04

(f) 05

(g) 06

(h) 07

(i) 08

(j) 09

(k) 10

(l) 11

(m) 12

(n) Coordinate system

Figure 5 Discrete normal directions given by the unit cell of the cuboidal lattice.

5 First analysis

In order to facilitate the very first steps with **MAVI** , this section describes step by step how to solve a very simple yet typical analysis task: Determining the specific surface areaof a closed cell metal foam.

(1) Start **MAVI** . The window is empty except for the title bar, the main menu, and the status bar on the bottom.

(2) Open the demo image AlFoam128.iass by choosing Open Image from the **File** menu. **MAVI** opens a dialog for specifying path and file name.

(3) The Slice View now shows the first slice in each coordinate direction. Clicking into one of the three views and subsequently moving of the slider in the toolbar above enables navigation through the slices.

(4) Choosing Binarization from the **Segmentation** menu opens a dialog box with preview for choosing a global gray value threshold. Ticking "Whole image histogram" activates the button "Otsu's threshold". Clicking that one in turn yields a lower threshold of 34. OK closes the dialog and creates a new image appearing in the document list as AlFoam128.iass 2.

(5) Now the Field Features from the **Analysis** menu can be applied and the view switches to an attribute view.

(6) Extending the attributes tree by clicking the Field Features and the Surface entries now shows the specific surface area ("surface density") measured in [1/m].

6 The typical **MAVI** workflow

The typical workflow contains a lot more steps and is sketched in the following:

(1) Import image data using **File**:**Import**. Either the file type needed is listed there or the generic importing functionsImage Raw Data or Image Stack have to be used.

(2) Check whether the pixel spacings are set correctly by **View**:Image Attributes View and extending the spacings subtree there. Correct the spacings if needed using **File**:Edit Header.

(3) Navigate through the data sets using the slice views or the Slice Showto get an impression of the structure. Use different view filters if needed.

(4) Reduce the observation window to the region containing structural information by **Manipulation**:Crop

(5) Remove noise using e. g. the Filter from **Transformation**:**Filter**

(6) Segment the component of interest by one of the binarization methods from the **Segmentation** menu.

(7) Obtain global geometric characteristics of the foreground component by computing the Field Features.

(8) Export the analysis results using **File**:**Export**:Export

(9) Import the csv-file into your preferred statistics package like Excel or R. Ensure that ";" is used as separator and "." as decimal point.

Further processing, segmentation, and analysis steps will follow depending on the application. The tutorials presented in the following Section 7 cover the most common cases.

7 Tutorials

This section offers several step-by-step tutorials for frequently occuring analysis tasks.

7.1 Open cell nickel foam

This tutorial consists of two parts. The first one demonstrates how to measure mean features of an open foam like porosity, specific surface area, or mean cell size based on the original gray scale image obtained by computed tomography (CT). In particular, the tutorial shows all processing steps neccessary before the actual analysis.

The second part of the tutorial shows how to measure the empirical size distribution of the foam cells by image analytical cell reconstruction.

7.1.1 Porosity and model based mean cell size

Figure 6 shows slices from the reconstructed CT image of an open cell nickel foam:

(1) Clearly, the structure does not fill the image, but is surrounded by a lot of empty space along the image borders. These empty areas affect the Field Features' result and therefore have to be removed using **Manipulation**:Crop.

(2) **Segmentation**:Binarization with Lower Threshold 70 and Upper Threshold 256 segments the solid component.

(3) Some small foreground components not connected to the solid structure are left. **Segmentation**:Labeling identifies them. **Manipulation**:Extract by label with label 1 yields a binary image without these crumbs.

(4) Now the **Analysis**:Field Features yield porosity, specific surface area, and various other global characteristics.

(5) The mean cell size is deduced from the Euler number density and a (mild) model assumption [18]. Thus the pores inside the struts have to be closed. To this end **Transformation**:**Morphology**:Closure with structuring element Cube and size 3 fills the small holes inside the edges of the foam.

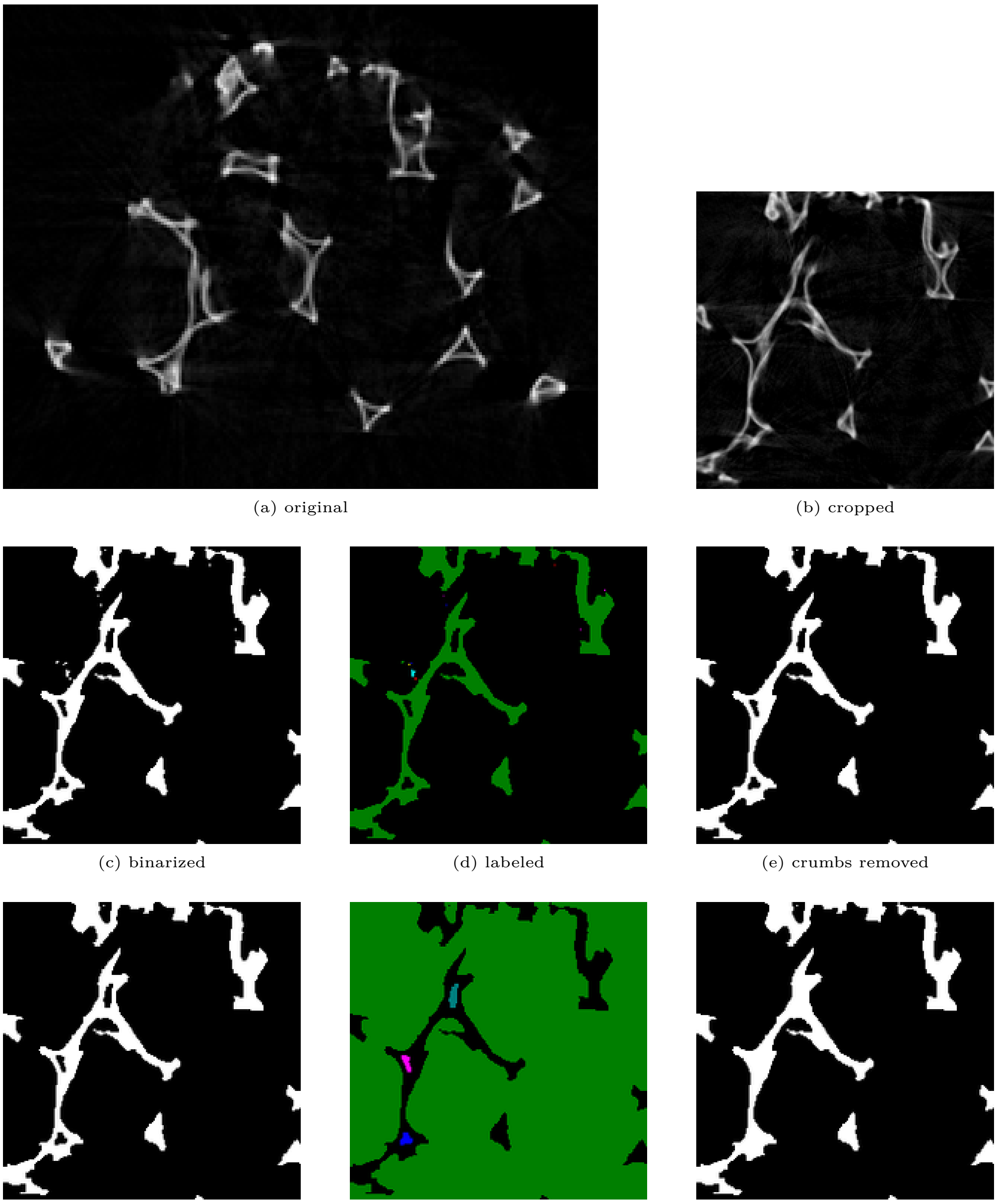
(6) Another Labeling, this time applied to the background ("apply to foreground" NOT ticked) and Extract by label with label 1

Now Field Features and Open Foam Features from the **Analysis** menu yield porosity and specific surface area and mean cell size, respectively.

7.1.2 Cell size distribution via cell reconstruction

In order to deduce empirical distributions of cell characteristics, the pore space has to be divided. To this end, apply **Segmentation**:Cell Reconstruction to the result of Section 7.1.1 shown in Figure 6(h). Use Dynamic 19 to obtain the cell system as shown in Figure 7

Now the Object Features compute a variety of geometric characteristics of the reconstructed cells. To export them, choose **Export**:**Object Features as CSV** form the **File** menu. Import the csv-file into your preferred statistics package like Excel or R. Ensure that “;” is used as separator and “.” as decimal point. Make sure to correct for edge effectswhen estimating means, standard deviations, or other moments of the respective distributions.



(a) original

(b) cropped

(c) binarized

(d) labeled

(e) crumbs removed

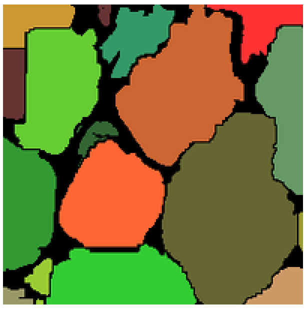
(f) closed

(g) background labeled

(h) inner pores removed

Figure 6 Slices through the open nickel foam image illustrating the tutorial steps.

Cell size distribution via cell reconstruction



(a) cells reconstructed

Figure 7 Slice through the open nickel foam image after cell reconstruction.

7.2 Closed cell zinc foam

This tutorial shows how to reconstruct foam cells if the cell size varies strongly. A closed cell zinc foam sample is used. The 3D image data set was obtained by computed tomography at the BAM line of BESSY. The pixel spacing is 5.5μm. Both sample and image data are courtesy of the Helmholtz-Zentrum Berlin.

Figure 8 uses x-y-slice 126 to illustrate the tutorial steps:

(1) Load kombin172-171.orig.iass.gz (a)

(2) Reduce noise by applying **Transformation**:**Filter**:Filter (b)

(3) Use global thresholding (**Segmentation**:Binarization) with lower threshold 98 to binarize (c)

(4) Apply the **Transformation**:**Distance Transformation**:Euclidean Distance Transformation on the background (no edge treatment, exact values)

(5) Transform the resulting float values into integers by **Manipulation**:Spread with Output type 8bit integers, Grayvalue characteristic linear, and Spread from grayvalue range of interest? not ticked (d)

(6) Smooth the image by **Transformation**:**Geodesic Transformation**:Adaptive H-Extrema choosing H-Maxima, 26 Neighborhood, Dynamic 3 at 0 and 11 at maximum

(7) Add the value 1 to all pixels by **Manipulation**:Unary Operations

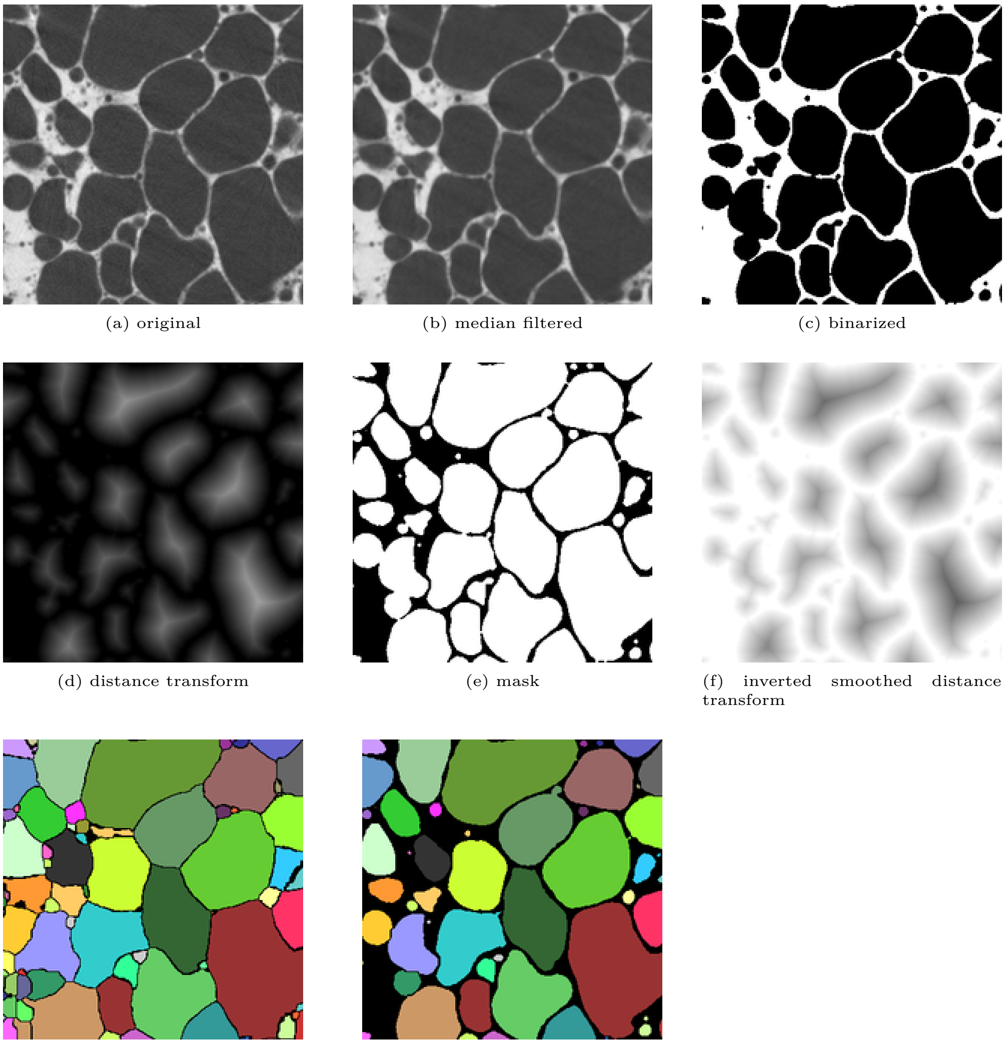
(8) Use **Manipulation**:Complement to invert the binary image obtained in step (3) (e)

(9) Use this binary image to mask the smoothed lifted distance image: Choose **Manipulation**:Mask Image and the result from (7) as image that should be masked. These three step ensure that the foam structure has everywhere in the image strictly lower gray values than the pore space.

(10) Invert by **Manipulation**:Complement to turn the cell centers into local minima (f)

(11) Apply **Segmentation**:Watershed Transformation with 26 neighborhood (g)

(12) Return to the mask image (8), mask the watershed result with this (h)



(a) original

(b) median filtered

(c) binarized

(d) distance transform

(e) mask

(f) inverted smoothed distance transform

(g) watershed transform

(h) masked watershed transform

Figure 8 Slices through the closed zinc foam image illustrating the tutorial steps.

7.3 Glass fiber reinforced composite

This tutorial shows how to analyse the fiber orientations in a fiber reinforced composite. The sample provided by Institut für Verbundwerkstoffe (IVW) Kaiserslautern features long fibers and a fiber volume fraction above 50%. The 3D image data set with a pixel spacing of 3.0μm was obtained by computed tomography at the IVW as well.

Figure 9 illustrates the tutorial steps using x-y-slice 250:

(1) Load GF60-crop512.iass.gz (a)

(2) Explore the gray value distribution by the **Analysis**:Grayvalue Profile. The global decline of the mean gray value in y-direction will cause global thresholding methods to fail. Therefore a so-called shading correction will be performed in the following steps.

(3) Create a strongly smoothed version of the input image by **Transformation**:**Morphology**:Opening, choosing "Cube" as structuring element (3D Struct. Element) with size 60 (b)

(4) Subtract the smoothed image from the original one by returning to the original image and applying **Manipulation**:**Binary Operations**:Binary Operations. Choose GF60-crop512.iass.gz 2 as second image for Subtracting (the subtrahend). The resulting image is much darker as gray value dynamic gets lost, however, the mean gray value in y-direction changes only mildely now. (Check using the Grayvalue Profile again.) (c)

(5) Now the (Sub-)Field Fiber Directions from the **Analysis** menu can be applied.

  (a) The glass fibers are known to have a cross section diameter of approximately 10μm. Thus the fiber radius is 1-2 pixels. Select 6e-06 ($6 \cdot 10^{-6}$m = 6μm) corresponding to 2 pixels.

  (b) Tick "Select subfield size". Enter 135e-06 ($135 \cdot 10^{-6}$m = 135μm) corresponding to 45 pixels for x, accept this value for y and z, too.

  (c) Tick "Advanced options". Change the "Otsu factor" from its default value of 1.25 to 1.0. (Due to the shading correction.) Push "OK". This step takes a couple of minutes.

(6) Use **Transformation**:Map Subfield Fiber Directions To Image to visualize the rheological effects. Select the Orientation tensor/yy. This is the diagonal entry of the orientation tensor in y-direction (a_{22} from (15)). Tick "Crop result". Push OK. (d)

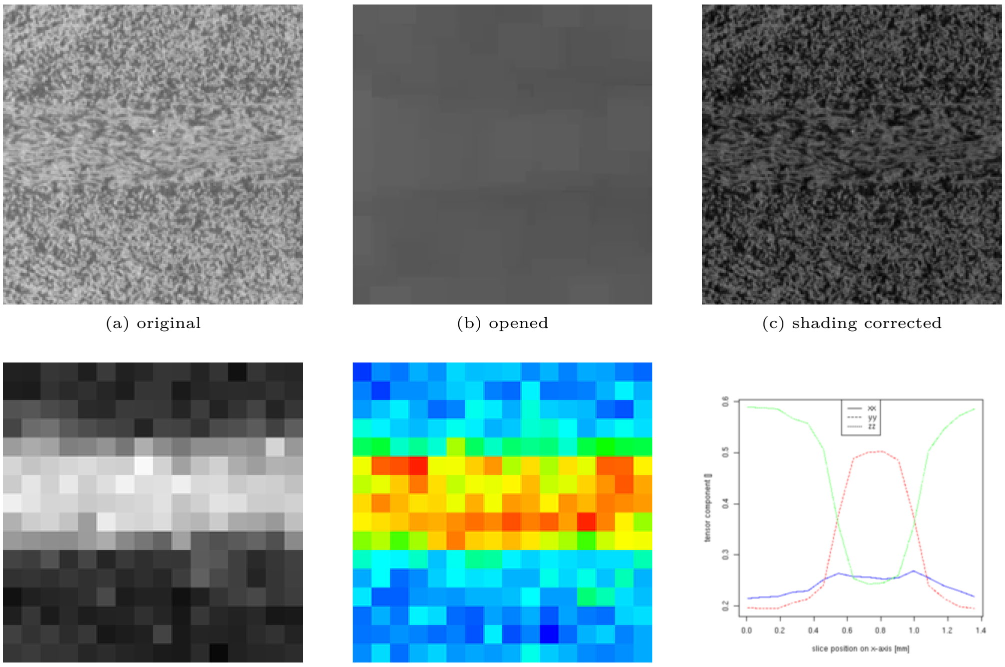
(7) A color view filter would increase the visual effect. However, to apply a View Filter, the resulting GRAYF image has to be Spread (**Manipulation**) to 8bit integer gray values.

(8) Now apply the Power Spectrum (blue -> red) (e) color table.

(9) Return to the attributes view and **File**:**Export**:**(Sub-)Field Fiber Directions as CSV**.

(10) Import the csv-file into your preferred statistics package like Excel or R. Ensure that ";" is used as separator and "." as decimal point. Plot

“Orientation tensor/xx”, “Orientation tensor/yy”, “Orientation tensor/zz” over “slice position”. (f)



(a) original

(b) opened

(c) shading corrected

(d) orientation tensor diagonal entry in y-direction

(e) orientation tensor yy, colored

(f) orientation tensor profile plot

Figure 9 Slices through the GFRP image illustrating the tutorial steps.

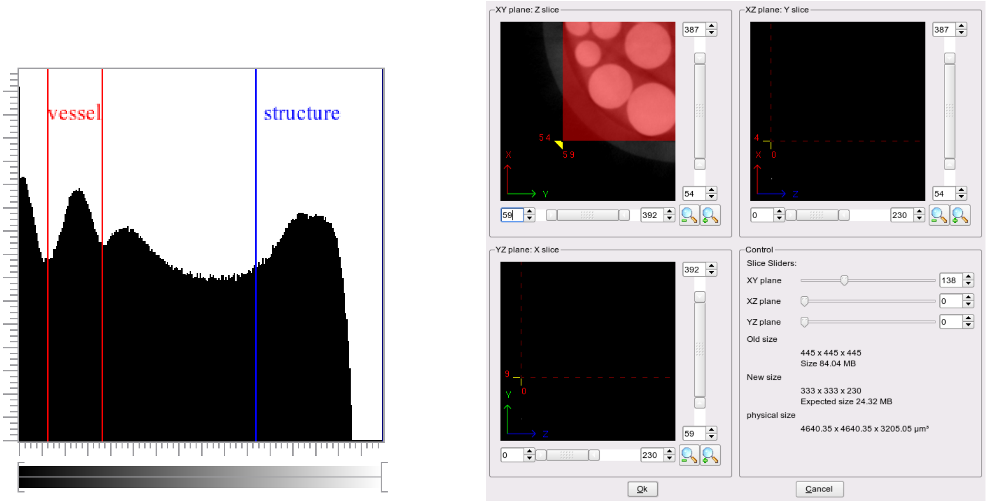
7.4 Sintered copper

This tutorial shows how to separate interconnected particles in order to analyze them indivdually. The sample provided by Institute of Materials Science at University of Dresden was obtained by computed tomography at the BAM with a pixel spacing of 14μm, see [9] for more background information.

Figure 10 illustrates the tutorial steps using x-y-slice 125:

(1) Load s060.iass.gz (a)

(2) Before starting to work on the sintered copper, the balls have to be separated from the cylindrical crucible holding them. This can be easily achieved choosing the lower gray value threshold for **Segmentation**:Binarization appropriatly as the aluminium crucible is darker (has lower gray values) than the copper balls. Figure (2)(a) shows how this is appears in the **Segmentation**:Binarization preview window.



XY plane: Z slice: 59 – 392, 387 – 54

XZ plane: Y slice: 0 – 230, 387 – 54

YZ plane: X slice: 0 – 230, 392 – 59

Slice Sliders:
XY plane 138
XZ plane 0
YZ plane 0

Old size
445 x 445 x 445
Size 84.04 MB

New size
333 x 333 x 230
Expected size 24.32 MB

physical size
4640.35 x 4640.35 x 3205.05 µm³

(a) Binarization dialog

(b) Crop dialog

(3) Since there is much space without any information in the resulting image, crop by **Manipulation**:Crop with parameters choosen as in the dialog displayed in Figure (2)(b). The resulting image should now have a size of 333x333x230 pixels.

(4) An attempt to binarize by **Segmentation**:Binarization with lower threshold value 150 shows a quite rough structure, see Figure (2)(b). Therefore, choose the previous image (after cropping) and apply **Transformation**:**Filter**:Filter on it. Choose "Double Reflective Edge" as Edge Treatment and a Filter Size of 3.

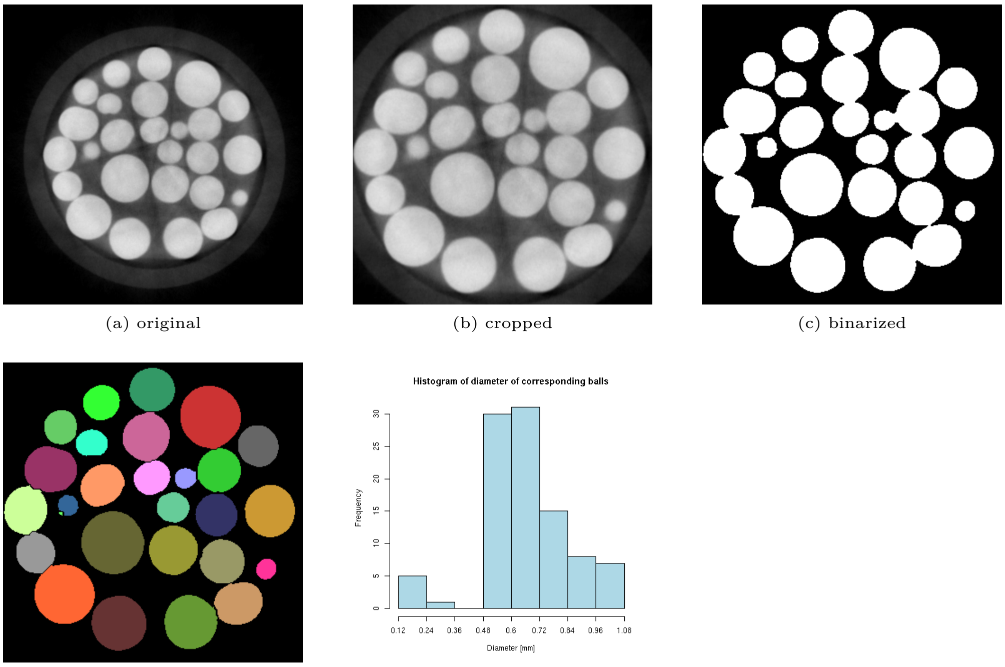
(5) Now binarize the result by **Segmentation**:Binarization with lower threshold value 150 Figure 10(c). The resulting surface is much smoother and the image is suitable for particle separation.

(6) MAVI provides automatic particle separation in the **Segmentation** menu. Apply Particle Separation with an area threshold of 20 (Figure 10(d))

(7) Compute the **Analysis**:Object Features for the result. Do not exclude the background. If the "Object Statistics" are computed does not matter if you postprocess the analysis data externally.

(8) Export the Object Features using Export from the **Export** submenu of the **File** menu.

(9) Import the csv-file into your preferred statistics package like Excel or R. Ensure that “;” is used as separator and “.” as decimal point. Plot the frequency of some statistics, e.g. “Diameter” (Figure 10(e)). Note that in this particular case, no edge correctionis needed as all balls are completely contained in the image.



(a) original

(b) cropped

(c) binarized

(d) particle separation

(e) diameters frequency plot

Figure 10 Slices through the sintered copper image illustrating the tutorial steps.

8 Reference manual

8.1 File

The **File** menu provides load and save function for **MAVI** 's native 3D image data format (.iass or .iass.gz) and various 2D image formats as well as generic import and export nodes for 3D image data either given as a block of raw data or as a stack of 2D images. The **Export** submenu subsumes various functions for saving **Analysis** results.

When opening or importing a file, **MAVI** chooses the last path used for opening or importing as default path. The last path used in a specific **MAVI** session will be stored and used in the next session.

If you work with the demo-versionof **MAVI** , **MAVI** restricts the size of the images you are able to work with to 128^3 pixels. It is nevertheless possible to openparts of larger images by croppingthem to at most 128^3 pixels. All save and export functions are however disabled in the demo-version.

Remark 2 Image meta data as available via Edit Header is completely loaded and saved when using ***MAVI*** 's .iass or .iass.gz formats. When importing or exporting other image data formats, make sure to provide and store essential meta data, respectively. Check the description of the functions for details.

Open Image This function opens IASS and IASS.GZ files and 3D TIFF files as 3D images in **MAVI** . The supported 2D file types are BMP, TIFF, PNG, PBM, PGM, PXM, JPG, and JP2 for 2d data.

3D TIFF does not contain information about the pixel spacing in z-direction. Therefore please use **File**:Edit Header to insert the correct pixel spacings. Otherwise, all **Analysis** results for this image will have [pixel] as base unit.

Save Image This function saves the current image as IASS, IASS.GZ, or 3D TIFF in case it is a 3D image, and as PGM, BMP, or TIF in case it is a 2D image. The path suggested for saving is the one used for the last opening of a file. Image meta data as available via Edit Header is saved in IASS(.GZ), only.

To save a slice of a 3D image as a 2D image, there are two options:

(1) Save the selected slice using the “Save current slice view” option from the “Settings” toolbox of the view control window of the Slice View. This saves the view to the slice, not the slice data itself.

(2) Crop the slice of thickness one pixel from the 3D image. Use Save Image and choose explicitly “2d Image” in the file type entry of the dialog. This option works for slices in the XY plane only.

8.1.1 Import

AVS Data The AVS field data is assumed to be given by two files: a data file of type DAT or REK and an ASCII description file of type FLD which specifies the name of the file containing the actual image data and defines the structure of the AVS field.

Input for **Import**:AVS Data is just the FLD file.

This is one of various possible representations of AVS field data. For more information see www.avs.com.

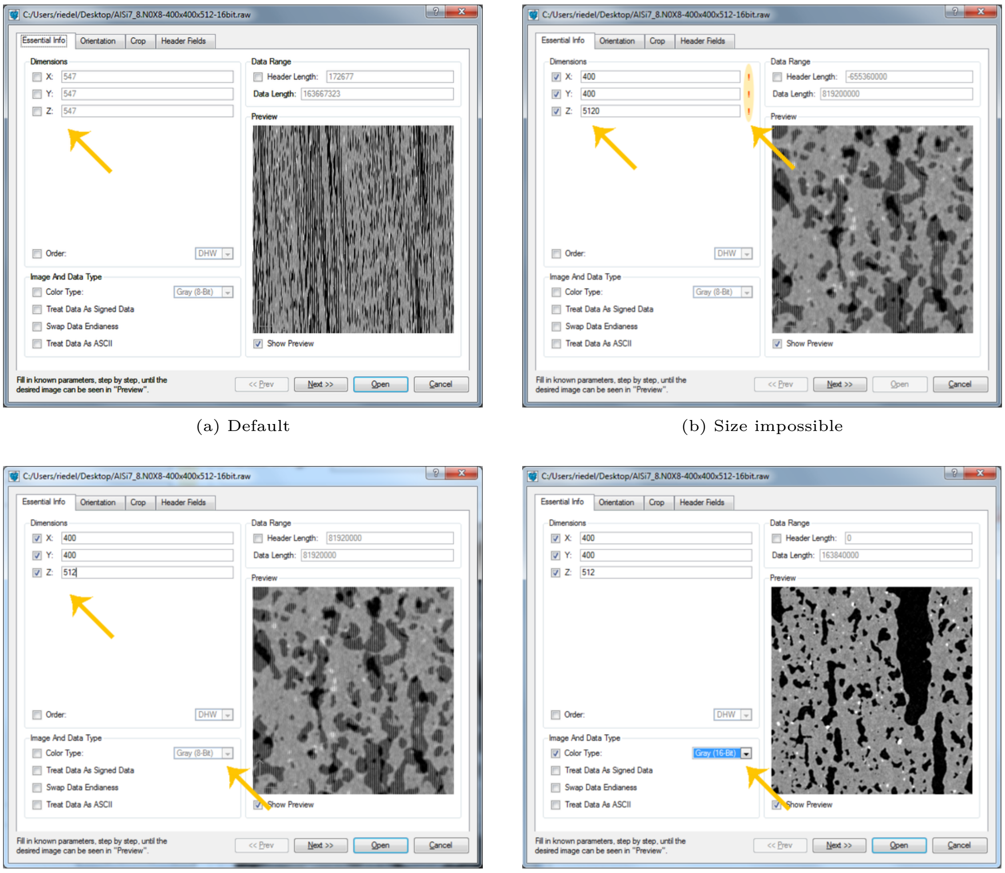
Image Raw Data **MAVI** is capable of importing a large number of additional formats, given they are saved to disk uncompressed. This is done generically, meaning that virtually any file saved in such a format can be opened in **MAVI** .

The user has to specify a few parameters to help **MAVI** understand the format structure. Such an initially unsupported format type is referred to as Raw Data files. Common file suffixes for raw data files are RAW, BIN, VOL, REC, and REK.

Select the desired file and accept. A dialog will appear giving options to fine-tune the import process. **MAVI** guesses the parameters. However, as shown in Figure 11(a), this first guess is very likely to be wrong. In particular, **MAVI** 's guess for the image size is based on the assumption of a cubic observation window. Now the user should provide step by step more information, until the preview window shows a reasonable content. Parameters include the image dimensions, image color type, image orientation, data format and the image header size.

To change a parameter, activate the checkbox next to it. Once a parameter is active, **MAVI** no longer tries to guess its value. If values contradicting previous input or the data are specified, **MAVI** marks the respective input parameters with a red “!”. See Figure 11(b) for an example. These values have to be corrected before importing the image. Once the preview shows the desired image Figure 11(d), click Open. The parameters and their meaning are summarized in Table 5. Please make sure to fill in the pixel spacingscorrectly as all **Analysis** results will be based on the spacings. Please note that the base unit in **MAVI** is always [m]. The spinbox allowing to change the unit is just provided for convenient input of very small nominal resolutions.

After importing the raw image data, pixel spacingscan still be corrected using Edit Header in the **File** menu.



(a) Default

□ X: 547
□ Y: 547
□ Z: 547
□ Header Length: 172677
Data Length: 163667323
□ Order: DHW
□ Color Type: Gray (8-Bit)
□ Treat Data As Signed Data
□ Swap Data Endianess
□ Treat Data As ASCII
☑ Show Preview

Fill in known parameters, step by step, until the desired image can be seen in "Preview".

(b) Size impossible

☑ X: 400
☑ Y: 400
☑ Z: 5120
□ Header Length: -655360000
Data Length: 819200000
□ Order: DHW
□ Color Type: Gray (8-Bit)
□ Treat Data As Signed Data
□ Swap Data Endianess
□ Treat Data As ASCII
☑ Show Preview

Fill in known parameters, step by step, until the desired image can be seen in "Preview".

(c) Size correct

☑ X: 400
☑ Y: 400
☑ Z: 512
□ Header Length: 81920000
Data Length: 81920000
□ Order: DHW
□ Color Type: Gray (8-Bit)
□ Treat Data As Signed Data
□ Swap Data Endianess
□ Treat Data As ASCII
☑ Show Preview

Fill in known parameters, step by step, until the desired image can be seen in "Preview".

(d) Color type correct

☑ X: 400
☑ Y: 400
☑ Z: 512
□ Header Length: 0
Data Length: 163840000
□ Order: DHW
☑ Color Type: Gray (16-Bit)
□ Treat Data As Signed Data
□ Swap Data Endianess
□ Treat Data As ASCII
☑ Show Preview

Fill in known parameters, step by step, until the desired image can be seen in "Preview".

Figure 11 **Import**:Image Raw Data: dialog.

Dicom Data In the open file dialog, see Figure 12(a), choose either a DICOM header file of type DIC or DCM or a DICOM dump file of type TXT and click Open. A list of available 2d image files will be shown, see Figure 12(b). The number of files selected should coincide with the number given in the DICOM header file. However, in most cases **MAVI** is able to open and display the DICOM data correctly even if that is not the case.

In the preview dialog, check if all data are correct and click Open to accept, see Figure 12(c). The tags found in the DICOM header are shown in the table at the top. For details on a specific entry, simply click on it. The information will then be shown in the lower right window. The window in the lower left corner shows a preview of a slice of the actual 3D image. To check the number of 2D files belonging to the DICOM dataset is given in the Section Count entry of the tag table.

Tab	Parameter	Description
Essential info	Dimensions X/Y/Z	image size in pixels
	Order	specifies whether dimensions are given depth-first (DHW) or width-first (WHD)
	Color Type	image type, supported are MONO, GRAY8, GRAY16, GRAY32, and FLOAT
	Treat Data As Signed Data	check if image data contains negative gray values
	Swap Data Endianness	check if byte-order of the image data is big-endian, e. g. coming from a SUN or Power-PC system, button remains inactive, if **MAVI** fails to interpret the data as ASCII
	Treat Data As ASCII	interpret input stream as ASCII
	Header Length	length of the header in bytes, usually correctly filled in by **MAVI** once the Dimensions have been entered
	Data Length	length of the data in bytes, usually correctly filled in by **MAVI**
Orientation	Original Locality	swap coordinate axes if needed
Crop		For large data sets, it might be sensible to crop before actually importing.
	Offset	defines the new lower left front corner of the image w. r. t. the original data
	Size	defines the size of the part of the image to be imported The memory requirement is shown in the lower part of the dialog window.
Header Fields	Spacing	set the pixel spacings
	Description	allows to enter image meta informationlike Creator, detailed Description, and processing History

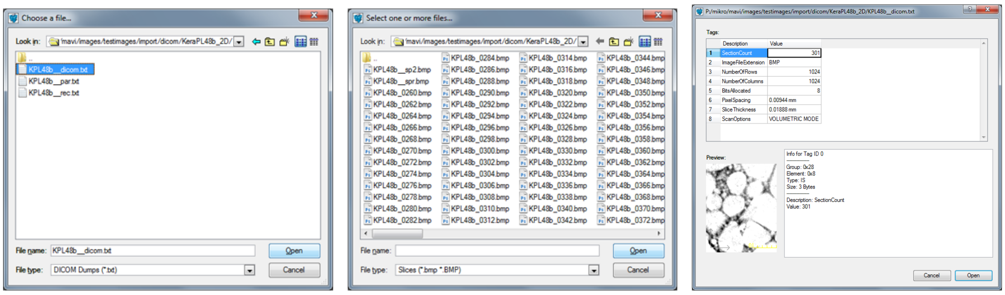
Table 5 Parameters of the **Import**:Image Raw Data: dialog.

After importing, the spacings of the resulting 3D image should be checked. Otherwise, all **Analysis** results may be incorrect.

Image Stack This generic importing function loads 3D image data provided as stack of 2D slices. In the section dialog, use the Shift key to select the desired 2D image files. Alternatively, try Ctrl A to select all and remove the unwanted ones using Ctrl. Press OK to load the files in ascending alphanumerical order.

Supported file types are BMP, (uncompressed) TIFF, PGM, PBM, PXM, JPG, and JP2.

All of the selected files must be two dimensional, and in one of the supported formats BMP, (uncompressed) TIFF, PGM, PBM, PXM, JPG, and JP2. The images must all be of equal dimension. Otherwise, the import will aborted.



Choose a file...

Look in: 'mavi/images/testimages/import/dicom/KeraPL48b_2D/

```
..
KPL48b__dicom.txt
KPL48b__par.txt
KPL48b__rec.txt
```

File name: KPL48b__dicom.txt

File type: DICOM Dumps (*.txt)

(a) Select header

Select one or more files...

Look in: 'mavi/images/testimages/import/dicom/KeraPL48b_2D/

```
..                  KPL48b_0284.bmp   KPL48b_0314.bmp   KPL48b_0344.bmp
KPL48b__sp2.bmp     KPL48b_0286.bmp   KPL48b_0316.bmp   KPL48b_0346.bmp
KPL48b__spr.bmp     KPL48b_0288.bmp   KPL48b_0318.bmp   KPL48b_0348.bmp
KPL48b_0260.bmp     KPL48b_0290.bmp   KPL48b_0320.bmp   KPL48b_0350.bmp
KPL48b_0262.bmp     KPL48b_0292.bmp   KPL48b_0322.bmp   KPL48b_0352.bmp
KPL48b_0264.bmp     KPL48b_0294.bmp   KPL48b_0324.bmp   KPL48b_0354.bmp
KPL48b_0266.bmp     KPL48b_0296.bmp   KPL48b_0326.bmp   KPL48b_0356.bmp
KPL48b_0268.bmp     KPL48b_0298.bmp   KPL48b_0328.bmp   KPL48b_0358.bmp
KPL48b_0270.bmp     KPL48b_0300.bmp   KPL48b_0330.bmp   KPL48b_0360.bmp
KPL48b_0272.bmp     KPL48b_0302.bmp   KPL48b_0332.bmp   KPL48b_0362.bmp
KPL48b_0274.bmp     KPL48b_0304.bmp   KPL48b_0334.bmp   KPL48b_0364.bmp
KPL48b_0276.bmp     KPL48b_0306.bmp   KPL48b_0336.bmp   KPL48b_0366.bmp
KPL48b_0278.bmp     KPL48b_0308.bmp   KPL48b_0338.bmp   KPL48b_0368.bmp
KPL48b_0280.bmp     KPL48b_0310.bmp   KPL48b_0340.bmp   KPL48b_0370.bmp
KPL48b_0282.bmp     KPL48b_0312.bmp   KPL48b_0342.bmp   KPL48b_0372.bmp
```

File name:

File type: Slices (*.bmp *.BMP)

(b) Select slices

P:/mikro/mavi/images/testimages/import/dicom/KeraPL48b_2D/KPL48b__dicom.txt

Tags:

	Description	Value
1	SectionCount	301
2	ImageFileExtension	BMP
3	NumberOfRows	1024
4	NumberOfColumns	1024
5	BitsAllocated	8
6	PixelSpacing	0.00944 mm
7	SliceThickness	0.01888 mm
8	ScanOptions	VOLUMETRIC MODE

Preview:

```
Info for Tag ID 0
---------------
Group: 0x28
Element: 0x8
Type: IS
Size: 3 Bytes
---------------
Description: SectionCount
Value: 301
```

(c) Preview dialog

Figure 12 Importing DICOM data: selecting a DICOM header file, selecting slice images, and checking the import via the preview.

After importing the image stack, please use Edit Header in the **File** menu to insert the correct pixel spacings. Otherwise, all **Analysis** results will be in [pixel].

Fraunhofer GeoDict Volume (GDT) Loads the specified file and sets the pixel spacingscorrectly.

Fraunhofer Raw Volume Data (REK) Loads the specified file and sets the pixel spacingscorrectly. Note that the Fraunhofer Raw Volume Data is based on a coordinate system different from **MAVI** 's. This results in the data being flipped along the Z axis.

3D Electron Density File (MRC) The user has to specify the file and to input the pixel spacings.

Just a subset of the various possible MRC file format versions is supported. The pixel grey value type has to be

- little endian and
- 8 bit unsigned or
- 16 bit unsigned and signed or
- float.

In all other cases **MAVI** will not recognize your MRC file. Then please use **Import**:Image Raw Data.

8.1.2 Export

This submenu offers functions for exporting/saving image data in a choice of image formats or as CSV (comma separated value) and analysis results as CSV files. The CSV files use the semicolon as separator, the decimal separator is the decimal point. Additionally, comment lines marked by a # can appear at the beginning of the CSV file.

The following export functions for image data are available:

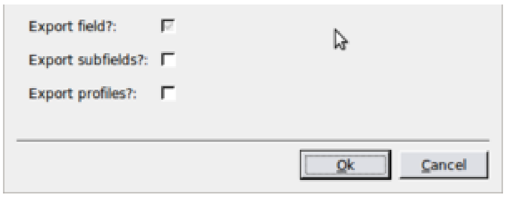
- Image as CSV (Text)
- Image as Fraunhofer GeoDict Volume (GDT)
- Image as Fraunhofer Raw Volume (REK)
- Image as AVS Data
- Image as Raw Data

Note that all four image formats store the pixel spacings but no other image meta data. The CSV files contain nothing but the gray values. For the two generic exporting functions **Export**:**Image as CSV** and **Export**:**Image as Raw Data**, the gray values of the image's pixels are written to the file with the X coordinate running the fastest and the Z coordinate the slowest. That is, pixel $(1, 0, 0)$ is written before pixel $(0, 0, 1)$. For **MAVI** 's coordinate system see Section 4.

The following export functions for analysis results are available:

- Image Atrributes as CSV (Text)
- Field Features as CSV (Text)
- Area Fraction Profile as CSV (Text)
- Grayvalue Profile as CSV (Text)
- Granulometric Data as CSV (Text)
- Object Features as CSV (Text)
- Open Foam Features as CSV (Text)
- Geometric Tortuosity as CSV (Text)
- (Sub-)Field Fiber Directions as CSV (Text)

The export function for the Field Features offers the choice between global and local features, that is between Field Features and Subfield Features, see the dialog in Figure 13 below.



Export field?: ☑
Export subfields?: ☐
Export profiles?: ☐

Figure 13 **Export** dialog for Field Features and Subfield Features as CSV.

Finally, there are export functions for rotation means of analysis results:

- Rotation Mean of Image Data as CSV (Text)
- Rotation Mean of Covariance as CSV (Text)
- Rotation Mean of Bartlett Spectrum as CSV (Text)

8.1.3 Edit Header

This function allows to set or change image meta information as shown by Image attributes, see Section 4.1.1 for a list of fields. When savingthe image as IASS or IASS.GZ, this meta data is stored, too.

The pixel spacingsin the header dialog can be given in micrometers (default), meters, millimeters or nanometers. In the file header, however, it will always appear in meters. Make sure the pixel spacingsare set correctly as all measurements by functions like Field Features or Object Features from the **Analysis** menu use these spacings.

8.1.4 Close View

Closes the current view. If it is the only view on the respective document, then the documentitself is closed.

8.1.5 Close Document

Closes the current document together with all view windows associated with the document.

8.1.6 Close All Documents

Closes all documents listed in the document viewtogether with all view windows associated with these documents.

8.1.7 Quit MAVI

Closes **MAVI** together with all open documents.

8.2 Manipulation

The manipulation menu contains tools to manipulate the observation window of an image and switching the coordinate axes. Moreover, a node for masking an image with a MONO image is provided.

This menu provides basic unary and binary operations to be applied pixelwise e.g. for combining or comparing images. The complement turning local minima into maxima and vice versa is an indispensable tool when feeding distancedata to the Watershed Transformation.

8.2.1 Crop

Cropping an image means to decrease its size by cutting off slices. In the **MAVI** cropping dialog, three slice views are visible, orthogonal to each of the three coordinate axes. The part of the image to be cut out can be specified either using the sliders beside and below these views or by changing the image size in each of the coordinate directions given in the number boxes next to the sliders.

The intervals defining the cropping limits are half open, including the lower but excluding the upper boundary. That is, to select exactly e.g. slice number 128 you need to choose lower boundary 128 and upper boundary 129.

The remaining part of the image is marked in red. The rectangle is filled iff the current slice intersects the new, smaller image, too.

If a particular image size after cropping is desired, then
(1) first type in the upper bounds,
(2) activate the last one of them by clicking into one of the the slice views and
(3) position the marked cuboid by moving the centers of the sliders.

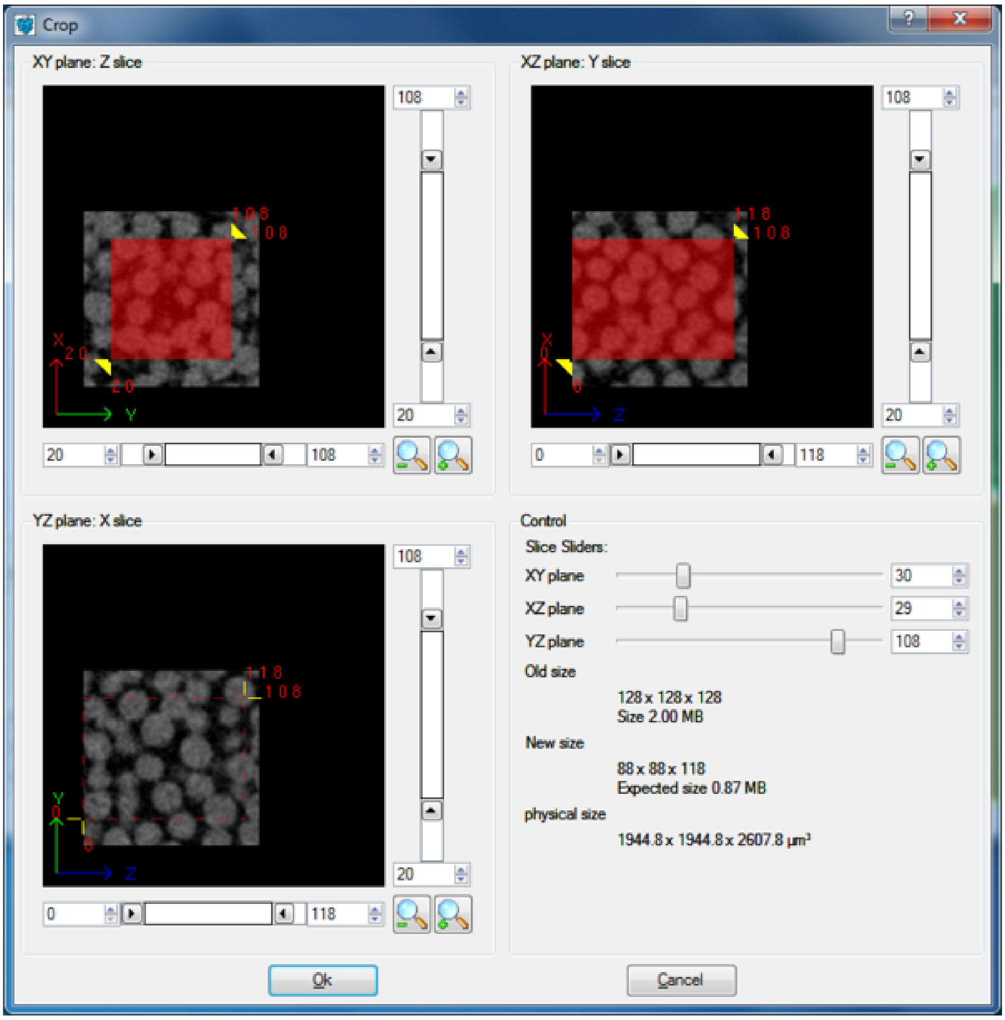
The buttons 🔍 and 🔍 allow to zoom in to get a close-up view of the image slice or to zoom out to see more of the slice at a reduced size, respectively.

The crop control panel in the bottom right part of the crop dialog window provides additional control facilities and some information about the cropping process:

Slice sliders: XY plane/XZ plane/YZ plane Allow to "move" through the image in Z, Y, and X directions respectively. You can change the slice number by moving the sliders or by changing the numbers in the spinboxes (type new number or click on the small arrows).

Old size Shows the original dimensions of the image and the original size of the image in megabytes.

New size Interactively shows the size of the marked part and its expected size



Control

Slice Sliders:

XY plane 30

XZ plane 29

YZ plane 108

Old size

128 x 128 x 128
Size 2.00 MB

New size

88 x 88 x 118
Expected size 0.87 MB

physical size

1944.8 x 1944.8 x 2607.8 µm³

Figure 14 Crop dialog window.

in megabytes.

Physical size Interactively shows the expected physical dimensions of the marked part in micrometers.

Note that cropping a 3D image to a new image size of 1 in one coordinate direction still results in a 3D image. For slices in the XY plane conversion to a 2D image can be achieved by Save Image with “2d Image” as target file type. For differently oriented slices, the “Save current slice view” option from the “Settings” toolbox of the view control window of the Slice View can be used.

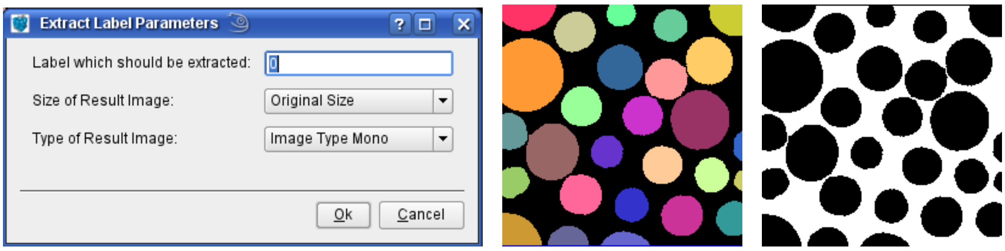
8.2.2 AutoCrop

Images of type can be cropped automatically. Choose a minimum pixel distance to the image border and a minimal grayvalue. **MAVI** cropsthe image such that each pixel with a grayvalue greater than the given minimal one has at least the given distance to the border in the cropped image.

8.2.3 Extract by label

This function transforms a gray value image into an image whose foreground consists of all pixels with the chosen gray value or label in the original image. The user chooses for the resulting image either the type MONO or to preserve the original image type. In the first case the foreground pixels get value 1 while in the latter they keep their original gray value.

In particular, it is possible to extract the background (label 0), too. Extracting the background from a label image and preserving the image type results in label 1 for the new foreground, see Figure 15 for an example.



Extract Label Parameters

Label which should be extracted: 0

Size of Result Image: Original Size

Type of Result Image: Image Type Mono

(a) Dialog (b) Before extracting 0 (c) After extracting 0

Figure 15 Extract by label dialog window and effect of extracting label 0 from a label image.

8.2.4 Pad

Pad embeds the image into a larger image by adding slices at the borders. The number of slices to be added to each side of the image as well as the gray value for the new pixels can be specified in the padding dialog.

Note that the dialog will allow to choose the value to embed the image in within the range of the image typeonly.

Padding a 2D image can result in a larger 2D image, if rows are added in X and Y directions, or in a 3D image, if slices in Z direction are added, too.

8.2.5 Shrink

This function reduces the size of an image by virtually decreasing the lateral resolution. The reduction factor can be set for each coordinate direction separately. The new image size in the respective coordinate direction is the old image size divided by the reduction factor.

In the case of a 3D image, the value of a pixel in the resulting image is the average of all pixel values within a cuboid of size ReductionFactorX x ReductionFactorY x ReductionFactorZ of the input image. For a 2D image, the resulting value is the average of all pixel values within a rectangle of size ReductionFactorX x ReductionFactorY.

8.2.6 Cast

Cast changes the gray value type of the image by simply casting each pixel value to the target gray value type. When casting down to a smaller gray value range, this results in a loss of image information since values exceeding the new gray value range are truncated. For example, when casting to GRAY8, each pixel with a value smaller than 1 is assigned 0 as new gray value while each pixel with an original value greater than 255 is assigned 255.

8.2.7 Convert

Converting is a linear scaling of the gray values from one gray value range to another, where the dynamic range of the gray values remains the same in relation to the value range. Thus Convert preserves the full image information and keeps the relations between gray levels as intact as possible, see Figure 17 for a sketch.

Here dynamic range denotes the difference between maximum and minimum gray value in the image while gray value range refers to the difference between maximum and minimum possible gray values determined by the image type, e.g. 255 for GRAY8 images.

8.2.8 Spread

Spread is a simple transformation of the image type. Depending on the user's choice, the gray values are transformed linearly, logarithmically, or following the square or root functions.

Spreading transforms the dynamic range, that is, the range from minimum to maximum value present in the input image to the full value range of the output image. Spread preserves all image information, i.e. it does not cut off any values. It may however transform different gray levels onto the same one, when spreading down e. g. from GRAY16 to GRAY8 images.

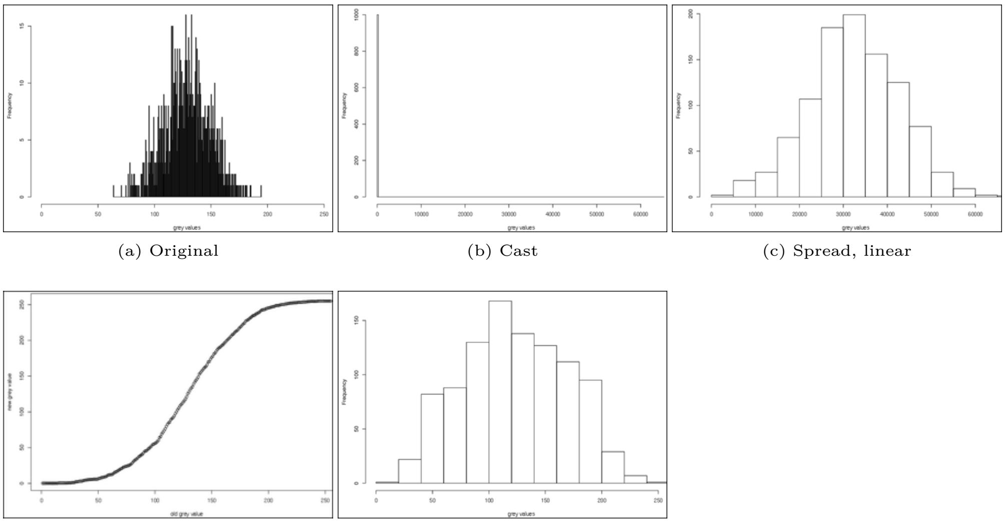
After spreading, the full value range of the resulting image will be used. For example, when a GRAY8 image is spread to GRAY16, the smallest value in the original will be assigned a zero, and the largest value in the original image will be assigned the value $2^{16} - 1$.

8.2.9 Equalize

Equalization is a method that can be used to enhance the contrast of an image. In **MAVI** , equalization to an output image type diffferent from the input image type is possible, too.

Equalize uses a transfer function which is derived from the image's gray value histogram. Hence, the output of equalization depends on the image content. Usually, contrast will be enhanced in the resulting image. That is, structures that were hardly visible before may appear much brighter after equalizing. However, equalization tends to emphasize noise, too.

Figure 16 shows how Cast, Spread, and Equalize affect a GRAY8 image. Figure 17 visualizes the difference of Spread and Convert.



(a) Original

(b) Cast

(c) Spread, linear

(d) Transfer function

(e) Equalize

Figure 16 Changing of image types by an example. Shown are the images' gray value histograms and the transfer function for Equalize.

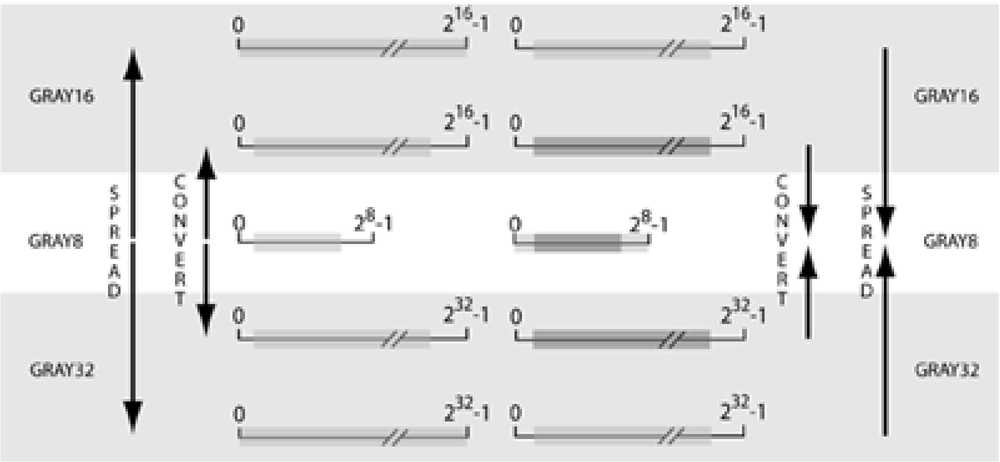


Figure 17 Visualization of the difference of Spread and Convert. Spread always uses the full target gray value range whereas the converted image occupies the same proportion of the gray values of its range as the original did.

8.2.10 Prepare label image for visualization

Reduces the number of labels in images obtained by Labeling or Watershed Transformation to the number N given by the user, in order to facilitate the **View**:Volume Rendering View. Values $N > 20$ are not recommended. See Section8.6.3 for details.

The labels are assigned in turns and spread to GRAY8. That is, if an object had label M in the input image, it is assigned the new label

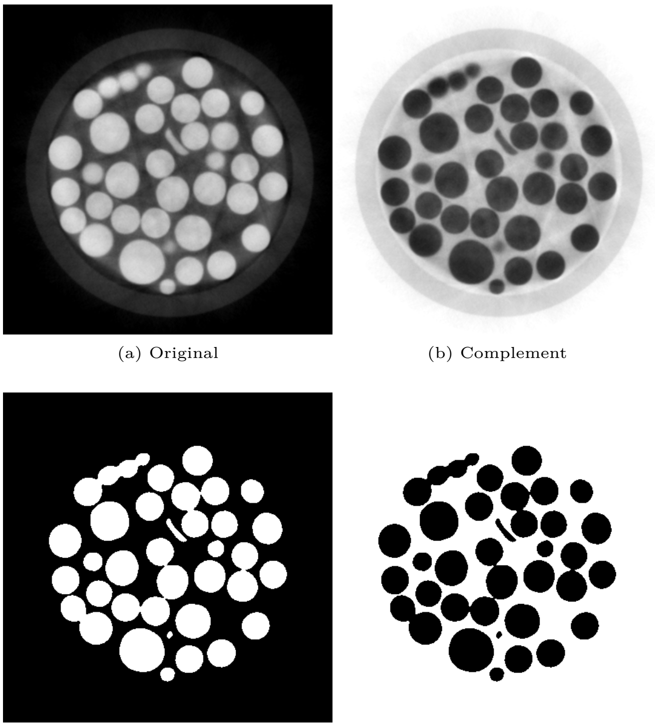
$$\big((M \mod N) + 1\big) * 256/(N + 1).$$

Note that this is a helper function for visualization purposes only. Do not base any analysis on the result of this function as different objects are assigned equal labels.

8.2.11 Complement

This function assigns to each pixel its complementary pixel value, which is obtained by subtracting the current value from the maximum of the gray value range given by the image type.

The complement of a MONO image simply swaps fore- and background, see Figure 18(c) and (d). For an 8 bit gray value image, the largest possible pixel value is $2^8 - 1 = 255$. A pixel value of 12 will be converted into $255 - 12 = 243$ by Complement. See Figure 18(a) and (b) for an example.



(a) Original

(b) Complement

(c) Binarized original

(d) Complement

Figure 18 Example for Complement acting on gray value or binary (MONO) images.

8.2.12 Complex Conjugate

This function computes the complex conjugate of a COMPLEX image, that is, it assigns each pixel with complex value $z = x + i * y$ the new value $z = x - i * y$.

8.2.13 Mask Image

To mask an image means to select certain areas of the image and discard the rest as background. The areas to be selected are specified by the foreground of a second image, called the "mask".

The binary operation Mask Image is active only if a MONO mask image is currently selected. The image on which the operation will be applied can then be selected via the dialog. Only images having the same size as the mask image can be chosen.

A very simple example of Mask Image is shown in Figure 19: The original image is masked by the image having a white rectangle in the foreground. Both images are of type MONO. The foreground of the result image contains only those pixels of the original image that were "covered" by the white rectangle of the mask image. For MONO images Mask Image is equivalent to a pixelwise logical Binary Operations.

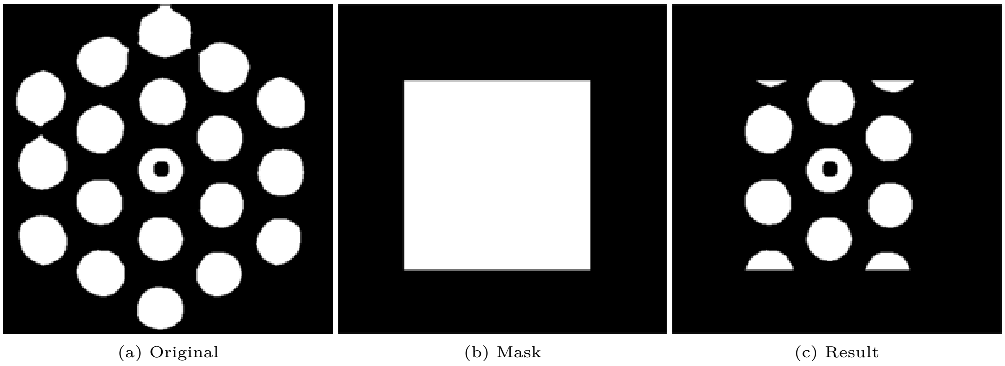

(a) Original (b) Mask (c) Result

Figure 19 Mask Image, simple example.

Figure 20 shows a more realistic example application of Mask Image: Tomographic reconstruction artefacts are removed from the image background by applying Binarization followed by Mask Image.

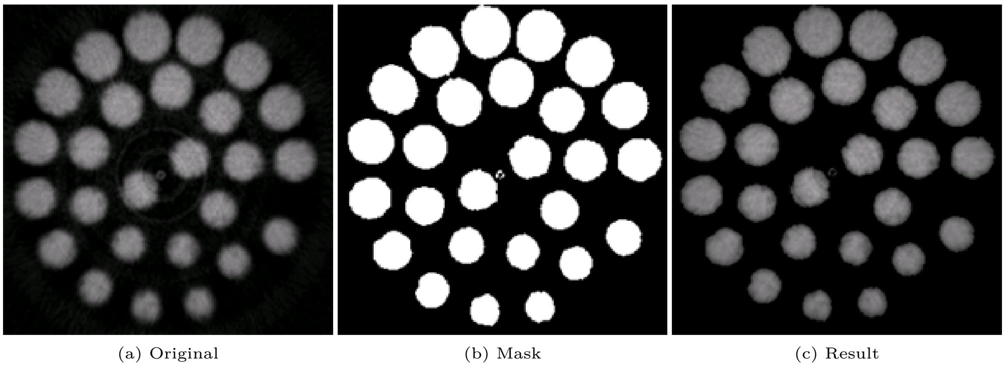

(a) Original (b) Mask (c) Result

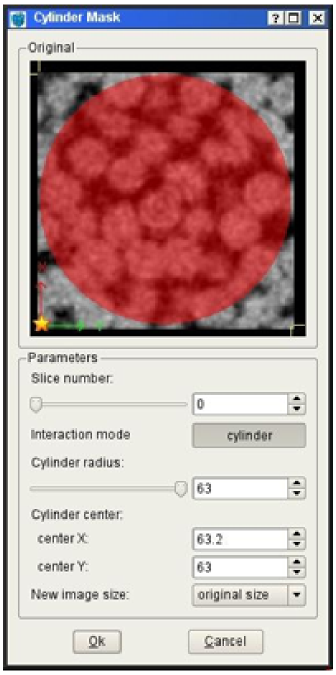
Figure 20 Mask Image, realistic example.

8.2.14 Mask with Cylinder

In order to facilitate analysisof the full volume for cylindrical samples, **MAVI** offers this special function for defining and applying a cylindrical mask. The cylinder's base is parallel to the coordinate plane given by the currently active Slice View. The user specifies the

- Cylinder radius [pixel] and
- Cylinder center [pixel]

in the dialog box. Moreover, the user decides whether the reulting image keeps the size of the original one or is reduced to the bounding box (of the cylinder) by the New image size parameter.



Slice number: 0
Interaction mode: cylinder
Cylinder radius: 63
Cylinder center:
center X: 63.2
center Y: 63
New image size: original size

Figure 21 Mask with Cylinder dialog box.

8.2.15 Unary Operations

Unary operations derive the new image from the original one by pixelwise calculating the new gray value from the old one. The following operations are available:

- Add Value
- Subtract Value
- Multiply by Value
- Divide by Value

The second value is given by the user via the dialog.

Unary operations cannot be applied to MONO images. For gray value images, the pixel values are truncated if the result of the chosen operation exceeds the gray value range defined by the image type.

8.2.16 Binary Operations

A binary operation is a calculation which takes pixel values from two different images at the same spatial location and writes the resulting value to the output image before proceeding the next location within both input images. For example, you can add two images and the value at a location in the resulting dataset will contain the sum of both values at the corresponding locations in the two input images.

Images must be of equal type and dimensions. If the resulting pixel value is greater than the maximum or lower than the minimum of the given value type, the resulting value is set to the maximum respectively the minimum.

In order to link two images of different gray value type, Cast the one with the smaller gray value range to the type of the other image.

The following binary operations are available:

Add: pixelwise sum of the gray values, for MONO images equivalent to the logical operation Or

Subtract: pixelwise difference of the gray values, for MONO images A and B equivalent to the logical A And Complement(B)

Minimum: pixelwise minimum of gray values, for MONO images equivalent to the logical operation And

Maximum: pixelwise maximum of the gray values, for MONO images equivalent to the logical operation Or

And: pixelwise logical and, on GRAY value images pixelwise minimum

Or: pixelwise logical or, on GRAY value images pixelwise maximum

Multiply: pixelwise product of the gray values

Absolute Difference: absolute value of the pixelwise difference, useful for comparison of images.

8.3 Segmentation

Segmentation refers to two tasks – finding the component or image segment of interest in a gray scale image and identifying connected objects or regions. We call the first task "binarization" as the output image is binary (MONO), i. e. it contains only zeros and ones as pixel values. Binarization is a prerequisite for measuring geometric characteristics of structural components like the Field Features.

Currently available binarization methods are global thresholding, hysteresis thresholding[1], and local thresholding according to Niblack[8] and Sauvola[16].

Segmentation methods for identifying foreground regions are labeling and the watershed transformation. Input data for labeling is a MONO image while the watershed transformation expects gray valued input. The output images contain discrete label values identifying connected regions. These are needed to compute geometric features of objects like the Object Features.

Integrated plugins combine several single functions to offer complex functionality at one click. For separation of touching particles like the grains of a powder or for the reconstruction of cells of an open cell foam, Labeling will fail as the objects are connected in the binary image. A combination of the Euclidean Distance Transformation with the Watershed Transformation will however do the job, provided n be prevented without destroying the structural information [12, Section 4.2.6]. An example is given in tutorial 7.1.2.

In most analyses, segmentation is performed rather early within the processing chain. For very noisy images, some sort of smoothing, usually **Transformation**:**Filter**ing, should be applied before segmenting.

8.3.1 Binarization

Global thresholding is the simplest binarization method and based on the idea, that the component of interest can be identified solely based on its brightness. This is for instance surely fulfilled in the case of the CT image of a porous structure.

The binary image with values g is created from the original image with gray values f by comparing the value of each pixel with the lower and the upper threshold values:

$$g(x) = \begin{cases} 1 & \text{if lowerthreshold } \leq f(x) \leq \text{upperthreshold} \\ 0 & \text{otherwise.} \end{cases}$$

The thresholds have to be determined either interactively by the user, usually with the help of the gray value histogram of the image and visual feedback, or by a threshold selection scheme. Slicewise both histogram and preview are available within the Binarization dialog window, see Figure 8.3.1. Checking the Whole Image Histogram box induces **MAVI** to calculate the gray value histogram of the whole image. This also activates the Otsu's threshold button to be used to let **MAVI** suggest a lower global threshold according to Otsu's method. Otsu's threshold maximizes the variance of gray values between the two classes created while keeping the variance within the classes as small as possible:

$$T_{\text{Otsu}} = \text{argmax}_T \frac{\left(\text{Mean}[\{f(x)|f(x) < T\}] - \text{Mean}[\{f(x)|f(x) \geq T\}]\right)^2}{\text{Variance}[\{f(x)|f(x) < T\}] + \text{Variance}[\{f(x)|f(x) \geq T\}]}. \quad (1)$$

Step-by-step, the binarization works as follows:

(1) Use the slider Slice number until a representative slice through the image is shown.
(2) The histogram of the current slice is displayed in the Information box. Switch to the histogram of the whole dataset by checking Whole Image Histogram.
(3) Switch to Logarithmic scale for the gray value-axis in the Histogram plot by pushing the respective button, in case the gray value histogram is hard to read due to the huge number of dark pixels.
(4) Use the sliders to set Lower threshold and Upper threshold, respectively. The result is immediately visible in the Preview box.
(5) Adjust the sliders until the slice preview is satisfying.
(6) Verify that the selected thresholds are suitable for the other slices using the Slice number slider again.
(7) Click OK to apply the selected thresholds to the whole dataset.

Remark 3 The interval defined by upper and lower threshold is half open, including the lower but excluding the upper one. That is, to select exactly the pixels with value e.g. 128 you need to choose lower threshold 128 and upper 129.

8.3.2 Hysteresis thresholding

Hysteresis or double thresholding [1] transforms a gray value image into a MONO image by a two-step procedure. A lower and an upper threshold value $T_l < T_u$ are input. First a simple global thresholding (1) with T_u as lower and the maximal gray value as upper threshold is applied. Subsequently, the foreground is enlarged according to the following rule:

For each pixel neighboring a foreground pixel, it is checked whether its value exceeds the second (lower) threshold T_l. The new value of the current pixel is one if this condition is fulfilled and zero otherwise. This is repeated as long as the foreground still grows, see Figure 23 for a sketch of the algorithm. Hysteresis thresholding yields smooth connected foreground components without emphasizing noise.

8.3.3 Niblack Segmentation

Global thresholding fails whenever noise is strong compared to the image content or when the image features global gray value fluctuations as due for instance to a strongly absorbing material in CT or to the tilted electron beam in FIB-tomography. Local thresholding can overcome this disadvantage of global thresholding by spatially varying the gray value threshold $t(x)$:

$$g(x) = \begin{cases} 1 & \text{if } f(x) \geq t(x) \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Niblack's algorithm [8] deduces the local threshold $t(x)$ for the current pixel x from gray value mean and variance within a cubic window $W(x)$ centered at x:

$$g(x) = \begin{cases} 1 & \text{if } f(x) \geq \text{Mean}_{W(x)}(f(x)) + c\sqrt{\text{Variance}_{W(x)}(f(x))} \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

The window size (edge length) is a parameter of the method and should be chosen according to the size of objects or the structure. The choice of the parameter c depends on the noise. In [21] $c = -0.2$ was recommended. However, that study was restricted to character recognition and thus it can hardly be generalized. In most cases the parameters have to be determined by trial-and-error, which is time consuming and should therefore be tried on a small but representative sub-volume created by **Manipulation**:Crop first.

In the dialog shown in Figure 24, the user has to specify the Window Size corresponding to the number of pixels to inspect in each direction from the current pixel. Thus, the resulting cubic window has the side lengths 2 * Window Size + 1. This parameter should be adjusted to the approximate size of the image features to be segmented. The second parameter Coefficient is the c from (3) and governs over- and under-segmentation, see Figure 25.

In the current version, no edge treatment is implemented. Thus, the resulting MONO image will have a black border exactly Window Size pixels wide.

Local binarization is available for all image gray value types including FLOAT. However, for FLOAT images, the option Use fast implementation must be ticked. Note that the result of Cast to FLOAT followed by Niblack Segmentation can differ from the local binarization of the original GRAY8 or GRAY16 image in some pixel values due to rounding effects.

8.3.4 Sauvola Segmentation

In images with large low contrast regions, Niblack's local thresholding tends to emphasize noise or low frequency variations due to the low local gray value variance. An empirical modification by J. Sauvola and M. Pietikäinen [16] solves this problem:

$$g(x) = \begin{cases} 1 & \text{if } f(x) \geq \mathrm{Mean}_{W(x)}(f(x)) + c\,\mathrm{Mean}_{W(x)}(f(x)) \left(\frac{1}{R}\sqrt{\mathrm{Variance}_{W(x)}(f(x))} - 1\right) \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

The additional parameter R normalizes the variance. In **MAVI** it is always set to half of the image's gray value range, that is $R = 128$ for GRAY8 and $R = 2^{15} = 32\,768$ for GRAY16 images. For FLOAT images f, R is set to $(\max f - \min f)/2$. For negative Coefficients c, the local threshold is now close to the local mean in high contrast areas and significantly above the local mean in low contrast areas.

8.3.5 Labeling

Labeling assigns to every connected object in a MONO image a certain gray value which is unique for each object. By default, the image foreground is labeled. However, the user can switch to labeling the background by un-ticking the corresponding box. **MAVI** 's labeling is based on Labeling by Reconstruction from [25]. An example is shown in Figure 26.

The result of the labeling might depend on the chosen adjacency system, see Section 4.3 for the list of options. If the structure is sufficiently well resolved and smooth, it does not matter which adjacency system is assumed. However, if one desires to keep thin, not well resolved structures connected, the 26 adjacency system should be used in 3D (8 in 2D), while the 6 and 4 adjacency systems are preferrable when touching object are to be separated.

The number of objects found is visible in the Info tab of the View Control Windowand stored in the image header's history entry. The gray value type of the labeled image is automatically adapted to the number of objects found.

8.3.6 Watershed Transformation

The watershed transformation segments a gray value image by simulation of flooding the image starting at the smallest gray values following [24]. Where waters from two sources would collate, a watershed is set. Watersheds get label 0, while each basin is assigned a specific label. The watersheds separate the basins. That is, the basins are not connected with respect to the chosen adjacency system. An example is shown in Figure 27.

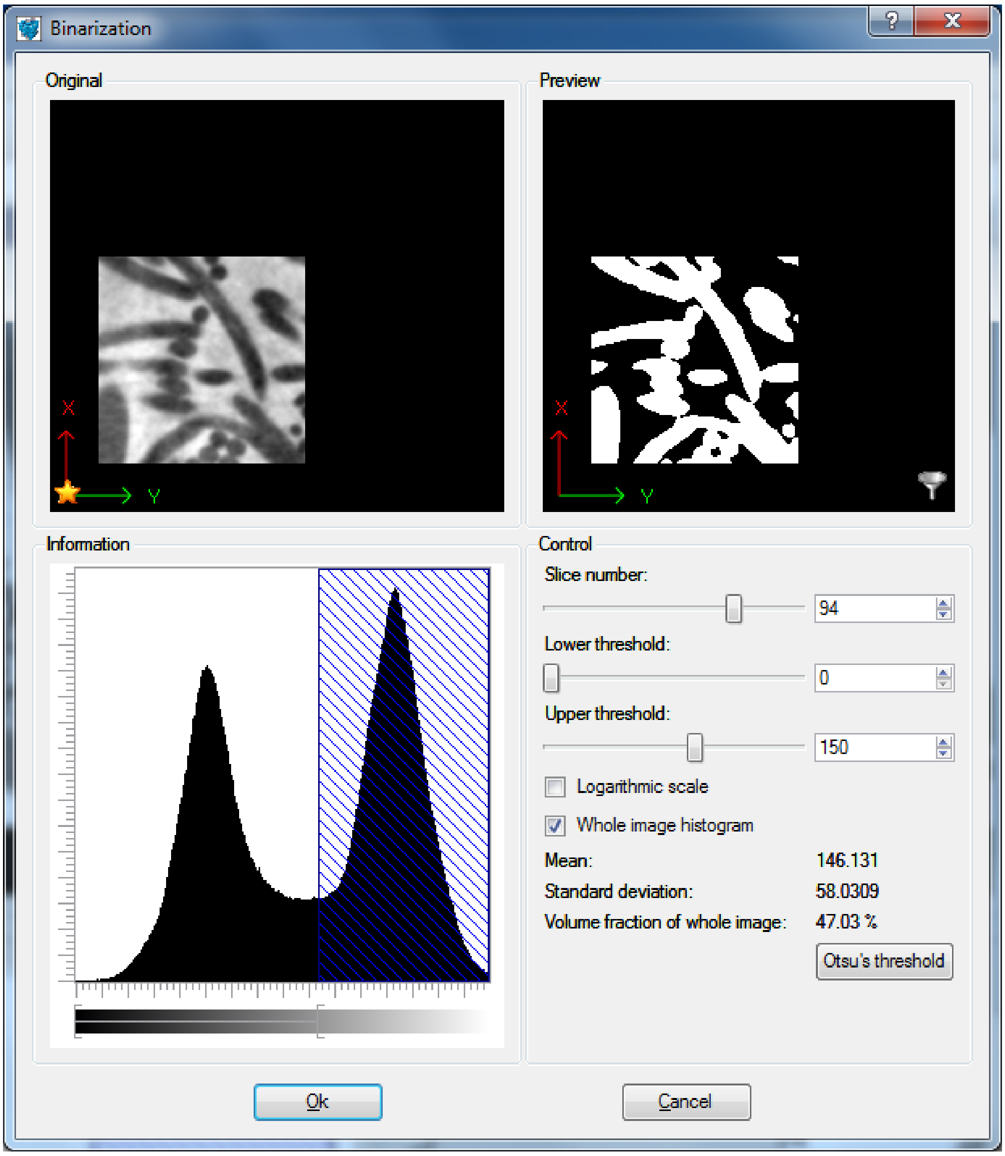
The watershed transformation is an essential ingredient for cell reconstructionas performed by the integrated plugins for this task as well as in the step-by-step tutorials for the open nickel foamand the closed zinc foam. In the same way, the watershed transformation is used by the integrated plugin for particle separation.

Usually, when applied to gray value images, the waterhed transformation results in strongly over-segmented images. A way to avoid over-segmentation is to use the preflooded versioninstead. For other strategies see the integrated plugins for cell reconstruction.

8.3.7 Watershed Transformation, Preflooded

Preflooding is a method to avoid over-segmentation in the Watershed Transformation. When flooding the image, at each step the number of pixels in the created basins is checked. If it is below the minimal pixel number specified as Area Threshold in the dialog, the basin is blocked from creating watersheds. It will be integrated into a larger basin instead. An example is shown in Figure 28.

This procedure is equivalent to
(1) flooding the image until the basins attain the minimal pixel number,
(2) assigning all pixels associated to a specific basin the highest gray value present in that basin, and
(3) using the resulting image as input for the Watershed Transformation.



Slice number: 94

Lower threshold: 0

Upper threshold: 150

☐ Logarithmic scale

☑ Whole image histogram

Mean: 146.131

Standard deviation: 58.0309

Volume fraction of whole image: 47.03 %

Figure 22 Binarization dialog window.

Watershed Transformation, Preflooded

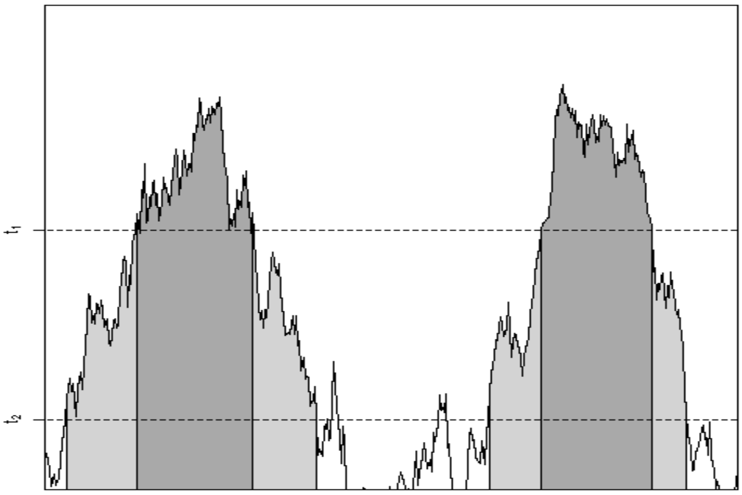
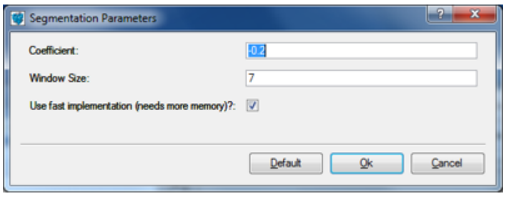


Figure 23 The principle of hysteresis thresholding illustrated by a 1d example: dark gray - foreground due to pixel value higher than higher threshold T_u, light gray - foreground due to being connected to dark gray area and pixel value higher than lower threshold T_l, white - background.

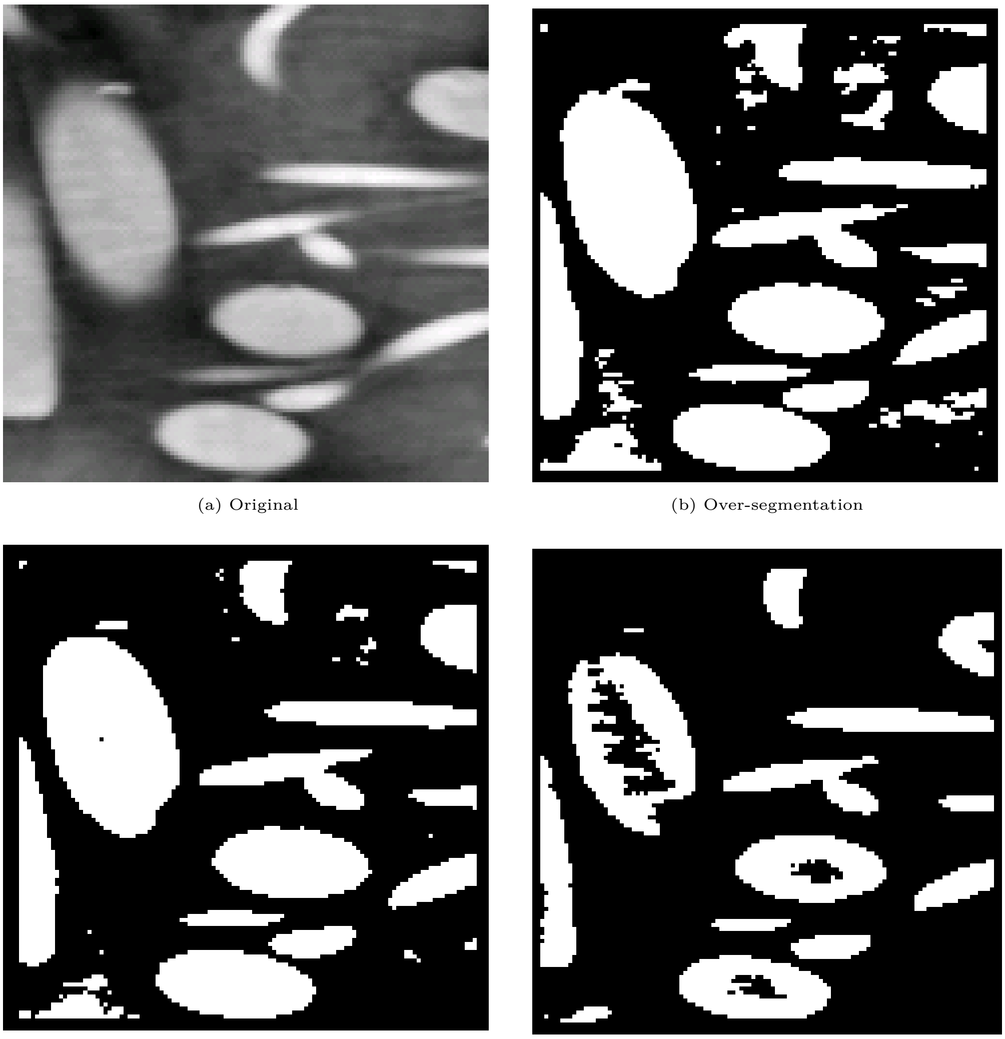


Coefficient: -0.2

Window Size: 7

Use fast implementation (needs more memory)?: ☑

Figure 24 Niblack Segmentation dialog window.



(a) Original

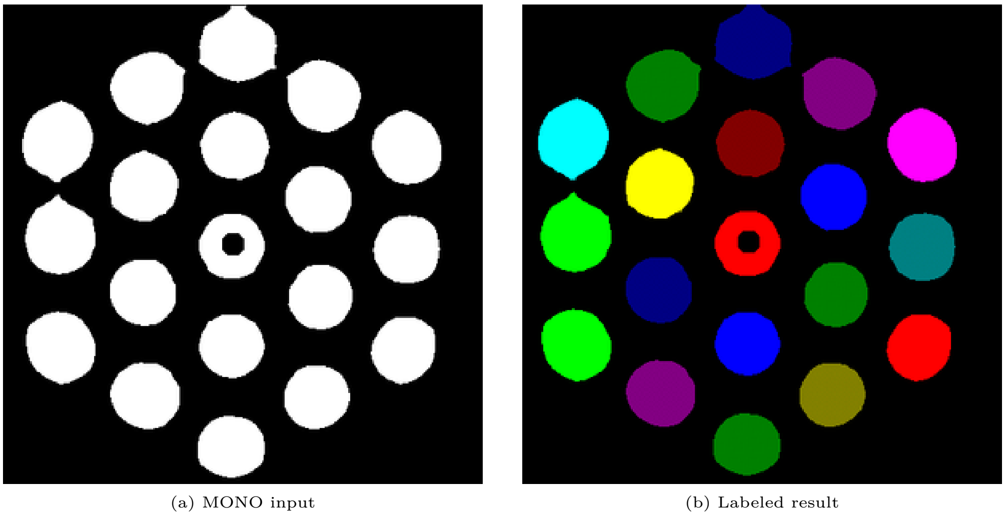
(b) Over-segmentation

(c) Correct

(d) Under-segmentation

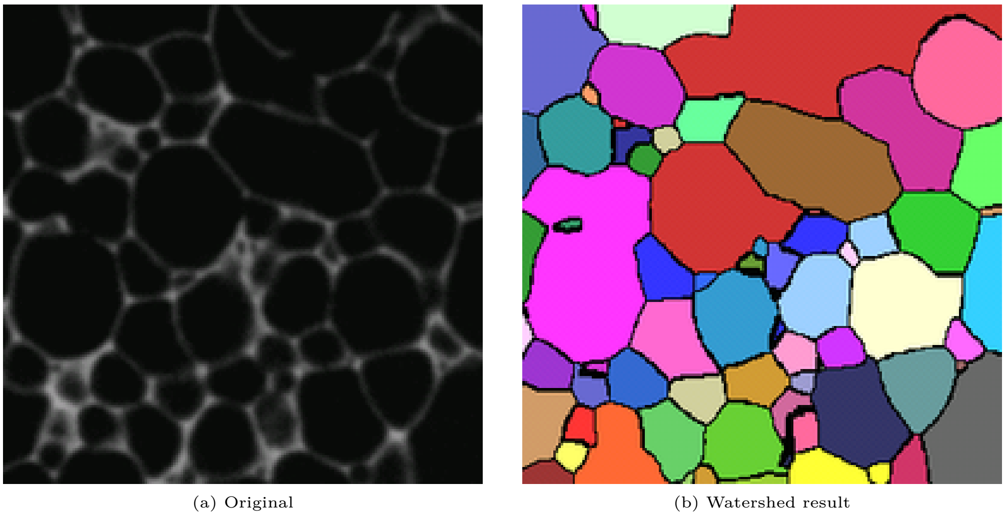
Figure 25 Niblack over- and under-segmentation. For Coefficient=-0.5, over-segmentation occurs: Areas which clearly belong to the background are assigned to the foreground. For Coefficient=-0.1, a good segmentation result is achieved. For Coefficient=0.1, under-segmentation occurs: Areas clearly belonging to the foreground are assigned to the background.

Watershed Transformation,
Preflooded


(a) MONO input

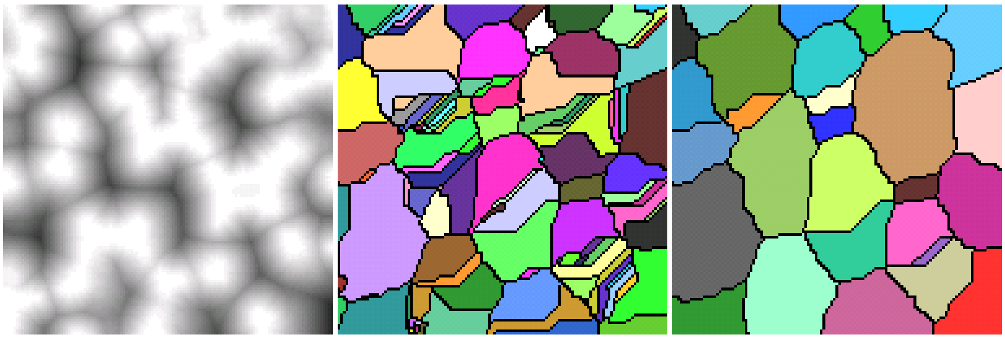
(b) Labeled result

Figure 26 Labeling example.


(a) Original

(b) Watershed result

Figure 27 Watershed Transformation example.



(a) Input

(b) Watershed result

(c) Watershed result

Figure 28 Effect of Watershed Transformation, Preflooded: (b) was obtained from (a) by a simple Watershed Transformation. Over-segmentation is clearly visible in (b) but can be avoided by preflooding as can be seen in (c).

8.3.8 Cell Reconstruction

In the reconstruction of cells, different methods can be used to prevent over- and undersegmentation. Cell Reconstruction uses the Filter to smooth the distance image before the Watershed Transformation. The only parameter to be set is the size of the filtermask. Cell Reconstruction uses the H-Maxima option of the H-Extrema transformation to suppress regional maxima. In the dialog, the s to be set. Finally, Cell Reconstruction exchanges the Watershed Transformation for the prefloodedone. It takes as input parameter the minimal pixel number that all regions created by the watershed transform have to contain.

Cell Reconstruction is the fastest choice. However, this simple gray value smoothing works rarely.

Cell Reconstruction works much better. A good way to choose the parameter “Area Threshold” is either to exploit knowledge about the material to find a lower bound for the cell volume or to use the mean cell volumefrom the Open Foam Features. Divide by the volume of one pixel ($s_1 s_2 s_3$, where (s_1, s_2, s_3) is the spacingvector) to get the mean cell volume in [pixel]. Depending on the structure, use $10 - 40\%$ of that value. The only drawback of the Preflooded Watershed Transformationis, that it slows down considerably for large regions and a high “Area Threshold”.

In that case, Cell Reconstruction is the best alternative. Its parameter “DynamicH” is however harder to choose. “DynamicH” is related to the diameter of the cells.

To get a rough estimate for the parameter for open foams use the mean cell diameter from the Open Foam Features d. Divide by the spacingd/s_1 as the dynamic is given in pixel units. Estimate visually or using the mouse slider in the Slice View the radius of the largest cell R (in [pixel]). This value is mapped to the value 255 by the Spread following the Euclidean Distance Transformation. Thus a starting value for the dynamic is $10 - 40\%$ of $255 \cdot d/(R \cdot s_1)$.

For closed cell structures, the mean chord lengthMIL obtained by the Field Features on the Complement can be exploited. Again, it has to be divided by the spacing and the effect of the Spread has to be taken into account. A starting value for the dynamic is now $1 - 2\%$ of $255 \cdot \mathrm{MIL}/(R \cdot s_1)$.

8.3.9 Particle Separation

The separation of touching particles uses the same algorithmic ingredients as the Cell Reconstruction. The major difference is, that the Euclidean Distance

Transformation is applied to the foreground of the input MONO image instead of the background.

Particle Separation uses the Filter to smooth the distance image before the Watershed Transformation. The only parameter to be set is the size of the filtermask.

Particle Separation uses the H-Maxima option of H-Extrema to suppress regional maxima. The dynamic is user defined. See 8.3.8 for details on the choice of the parameter.

Particle Separation uses the Preflooded Watershed Transformationto prevent oversegmentation. It takes as input parameter the minimal number of pixels that a particle has to contain. See 8.3.8 for details on the choice of the parameter.

Tutorial 7.4 describes the separation of sintered copper particles step-by-step.

8.4 Transformation

8.4.1 Filter

Filters replace the value of a pixel by combining the pixel values in a neighboring region given by a filter mask or structuring element. The filter function can be a linear combination of input values (linear filter) or a more general non-linear function.

Filtering is typically used in the initial steps of the processing chain for denoising, smoothing, and contrast enhancement to facilitate segmentation or analysis at later stages.

Smoothing filters help to denoise images, often at the cost of blurring edges, whereas edge detection filters are used to emphasize edges of an object or within complex structures.

The filter masks are all cubic in pixels, no matter how the image spacings are set. Table 6 lists the filters provided by **MAVI** .

Filter	Mask size	Mask values or function
Smoothing filters		
Mean Filter	$3 \leq N \leq 31$, odd	mean of all values
Binomial Filter	$3 \leq N \leq 31$, odd	convolution of 1D filters with entries $\binom{N-1}{N-1-k}$ at distance k from center, normalized by $2^{3(N-1)}(2^{2(N-1)})$
Weighted Mean Filter 1	3	center = 2, 6(4)-neighbors = 1, all others = 0, normalized by 8(6)
Weighted Mean Filter 2	3	center = 4(3), 6(4)-neighbors = 2, all others = 1, normalized by 36(15)
Gauss Filter	3	center = 4, 6(4)-neighbors = 1(2), all others = 0(1), normalized by 10(16)
Edge detection filters		
High Pass Filter	3	center = 26(8), all others = -1, normalized by 26(8)
Laplace Filter 1	3	center = 6(4), 6(4)-neighbors = -1, all others = 0, normalized by 1
Laplace Filter 2	3	center = 7(5), 6(4)-neighbors = -1, all others = 0, normalized by 2
Nonlinear Filters		
Alpha-Trimmed Mean Filter	3	mean after removing top and bottom alpha portion from sorted value list, alpha=0 – mean, alpha=0.5 – median filter
Median Filter	3, 5, 7	median

Table 6 Filters in **MAVI** with their masks, values, and parameters. Mask size is the edge length of the cubic/square filter mask in pixels. The values in braces refer to the 2D version if this differs from 3D.

A Morphological Mean filter with larger and more flexible filter masks is available in the **Morphology** submenu.

Laplace Filter 2 is a modifies edge detection filter setting former background pixels to 0, edge pixels to the largest possible value, and foreground pixels to a gray value halfway in between.

Rank order filters sort the N^2 or N^3 values of the pixels covered by the filter mask by size and assign the value at the desired rank from this list to the pixel considered. The median is the value at the central position $N^2/2 + 1$ or $N^3/2 + 1$ for 2D and 3D, respectively.

8.4.2 Morphology

Morphological transformations modify the image content with respect to a given structuring element. The structuring elements (SE) available in **MAVI** 's **Morphology** submenu are listed in Tables 7 and 9 for 3D and 2D images, respectively. The structuring element is centered at its central pixel or in the lower left front one of a set of central pixels. Note that this can cause shifting, for instance when a cube of even edge length is used repeatedly. For an isotropic lattice, the SE Approx. Ball is a linear combination (Minkowski addition) of line segments of length Size parallel to the 3 coordinate directions, length Size/$\sqrt{2}$ in the 6 directions defined by the face diagonals of the lattice unit cell, and length Size//$\sqrt{3}$ in the 4 directions defined by the space diagonals of the lattice unit cell. The diameter of the resulting approximate ball is therefore considerably larger than Size, see Table 8 for the approximate diameters. For morphological transformations with balls of intermediate diameters, please use alternatively the Euclidean Distance Transformation. Note that Approx. Ball of Size 2 is nothing but the cube of Size 2.

Name	Description	Parameters
Cube	cube parallel to the coordinate axes	side length Size
Cuboid	cuboid parallel to the coordinate axes	side lengths SizeX, SizeY, SizeZ
Approx. Ball	rough approximation of a ball	Size
Line	line parallel to one of the 13 discrete directionsgiven by the cuboidal lattice	length Size
Custom	Minkowski sum of segments in the 13 discrete directionsgiven by the cuboidal lattice	lengths of the segments

Table 7 Structuring elements for 3D morphology. All size parameters are in [pixel] units.

Structuring elements can be chosen via the dialog box shown in Figure 29

Size	2	3	4	5	6	7	8	9	10	11	12	13	14
Diameter	2.31	11.00	16.26	25.00	26.56	35.22	44.00	45.61	54.27	59.47	64.67	73.32	78.52

Table 8 Size parameter and corresponding approximate diameter of the SE Approx. Ball provided for 3D morphological transformations in the case of an isotropic lattice.

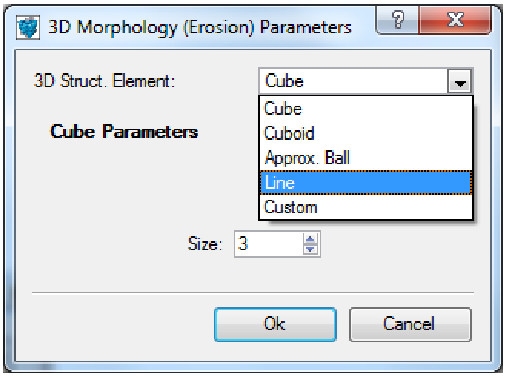
Name	Description	Parameters
Square	square parallel to the coordinate axes	side length Size
Rectangle	rectangle parallel to the coordinate axes	side lengths SizeX, SizeY
Line	line parallel to one of the 4 discrete directions given by the rectangular lattice	length Size
Custom	Minkowski sum of segments in the 4 discrete directions given by the rectangular lattice	lengths of the segments

Table 9 Structuring elements for 2D morphology. All size parameters are in [pixel] units.

Erosion The value of the pixel covered by the central pixel of the structuring elementis set to the minimum gray value of all pixels covered by the structuring element mask. Erosion can be used to remove small objects, to reduce edge deformation, or to break thin connections between objects.

For MONO images Erosion reduces the number of foreground pixels and thus object size or volume of the foreground. For gray value images Erosion is a minimum filter and thus the image becomes darker. See Figure 30 for an example.

Dilation The value of the pixel covered by the central pixel of the structuring elementis set to the maximum gray value of all pixels covered by the structuring element mask. Dilation can be used to remove small disturbances (i.e. holes) within the objects or to connect interrupted objects.


3D Morphology (Erosion) Parameters

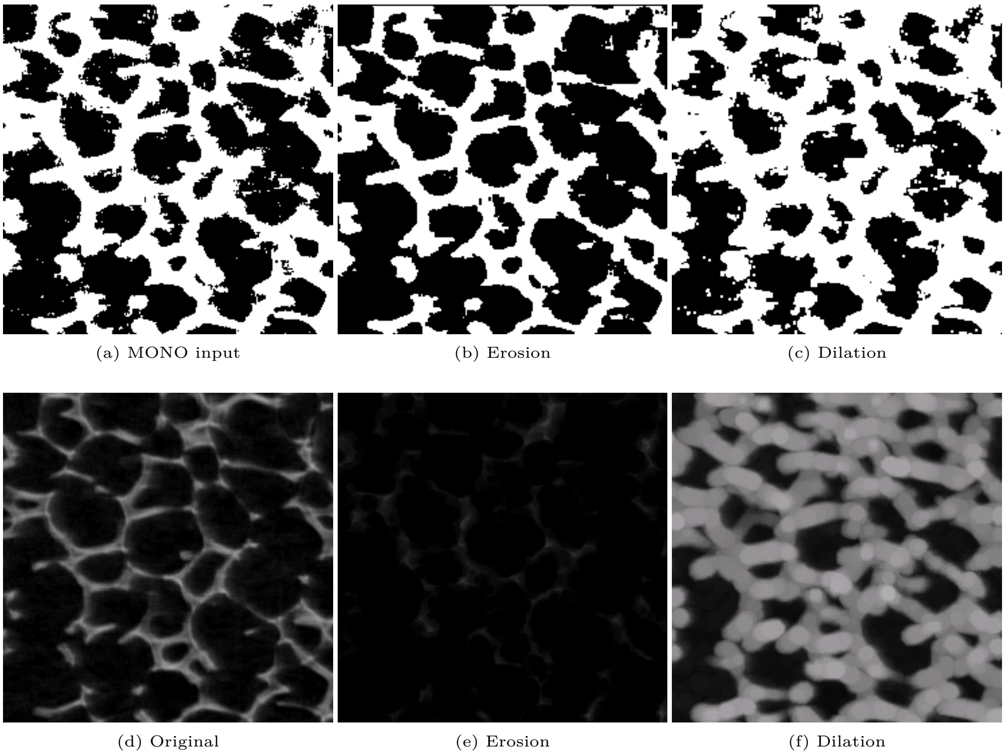
3D Struct. Element: Cube

Cube Parameters

Size: 3

Figure 29 Dialog box for defining a structuring element for morphological transformations.

For MONO images Dilation increases the number of foreground pixels and thus object size or volume of the foreground. For gray value images Dilation is a maximum filter and thus the image becomes brighter. See Figure 30 for an example.


(a) MONO input

(b) Erosion

(c) Dilation

(d) Original

(e) Erosion

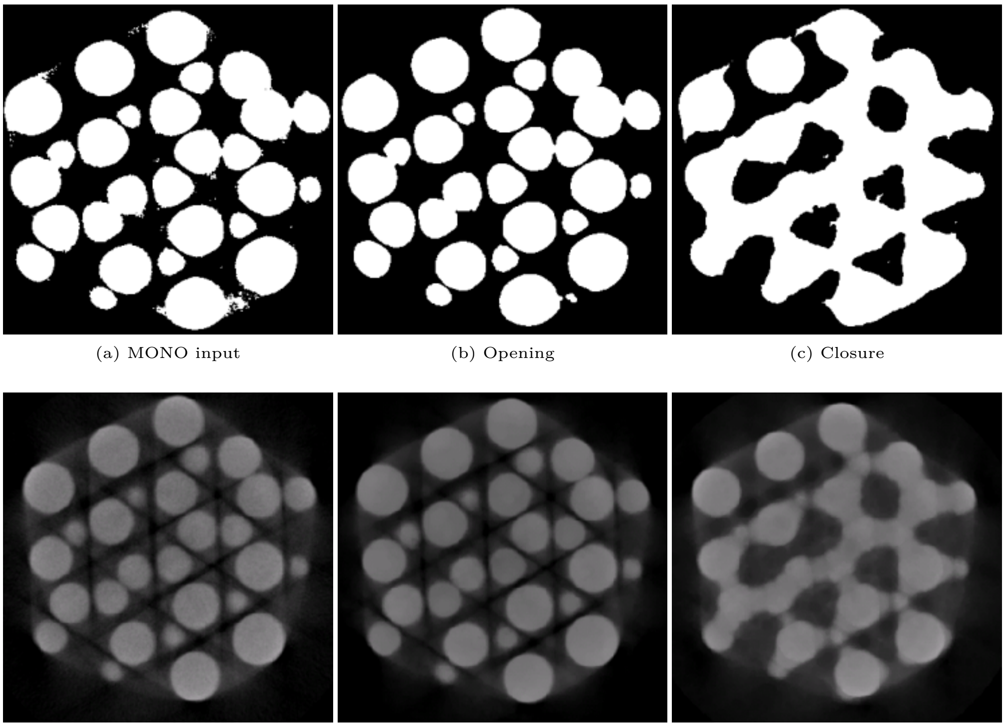
(f) Dilation

Figure 30 Erosion and Dilation examples. The structuring element Approx. Ball of Size 3 was used in all cases.

Opening Opening is an Erosion followed by a Dilation. It can be used to remove small objects, to reduce edge deformation or to interrupt thin connections between objects while keeping the remainder of the image unchanged.

In the case of gray value images, Opening reduces noise and enhances image contrast. See Figure 31 for an example.

Closure Closure is a Dilation followed by an Erosion. It can be used to remove small disturbances (i.e. holes) within the objects or to connect interrupted objects while keeping the remainder of the image unchanged. Gray value images appear “blurred” after a Closure. See Figure 31 for an example.



(a) MONO input (b) Opening (c) Closure

(d) Original (e) Opening (f) Closure

Figure 31 Opening and Closure examples. The structuring element Approx. Ball of Size 3 was used in all cases.

White Top Hat Subtractsthe openedimage from the original one. White Top Hat can be used to remove large objects from the image.

To determine an adequate structuring elementtry opening the image with an element that removes the small objects completely.

Black Top Hat Subtractsthe original image from the closedone. Black Top Hat can be used to emphasize holes within the objects or small background areas in the image.

Algebraic Opening The image is openedwith respect to line segments in all the discrete directionsas structuring element. The union of all opening results is the algebraic opening. For gray value images this translates to the pixelwise maximum of the opening results.

The SE lengths are controled by the Line parameter, input in [pixel]: In the case of an isotropic lattice, the line segments are approximately Line pixels long. In the case of an anisotropic lattice, the segment lengths are approximately Line*(average of spacings).

Algebraic Closure The image is closedwith respect to line segments in all the discrete directionsas structuring element. The intersection of all closing results is the algebraic closure. For gray value images this translates to the pixelwise minimum of the closing results.

The SE lengths are controled by the Line parameter, input in [pixel]: In the case of an isotropic lattice, the line segments are approximately Line pixels long. In the case of an anisotropic lattice, the segment lengths are approximately Line*(average of spacings).

Morphological Gradient The erodedimage is subtractedfrom the dilatedimage. The morphological gradient emphasizes edges of a thickness given by the structuring element.

Morphological Mean The morphological mean acts essentially like the common Filter. It is however a generalization in the sense that it averages the gray values of all pixels covered by the structuring elementinstead of just using a cubic filter mask. Besides being more flexible, the morphological mean filter allows for filter mask sizes up to 128^3 pixels. Moreover, it is much faster for large filter masks.

Toggle Mapping Erosion/Dilation Toggle mapping [19] by erosion/dilation or the morphological shock filter [13] enhances edge contrast by applying pixelwise Erosion or Dilation. The actual transformation applied at the current pixel is chosen such that the result differs the least from the original input image.

Toggle Mapping Opening/Closure Toggle mapping [19] by opening/closure enhances image contrast by applying pixelwise Opening or Closure. The actual transformation applied at the current pixel is chosen such that the result differs the least from the original input image.

8.4.3 Distance Transformation

Distance transformations usually assign each background pixel in a MONO image the distance to the foreground. The distance information can be used to speed up successive morphological Dilations with balls of varying radius as structuring elements: Taking a global thresholdin the distance image is equivalent to a dilation by a ball (w.r.t. the chosen metric).

Moreover, the Euclidean distance transformation is an essential ingredient for Cell Reconstruction and Particle Separation where it serves as input for the Watershed Transformation.

The Euclidean Distance Transformation is not just a valuable processing tool but a means for analysis in its own right as it yields the spherical contact distribution [20] and can be used to investigate correlations in the spatial arrangement of components, see [14].

Euclidean Distance Transformation This function computes the exact or squared Euclidean distance of each background pixel to the closest foreground pixel or of each foreground pixel to the closest background pixel of a MONO image. For an example see Figure 33.

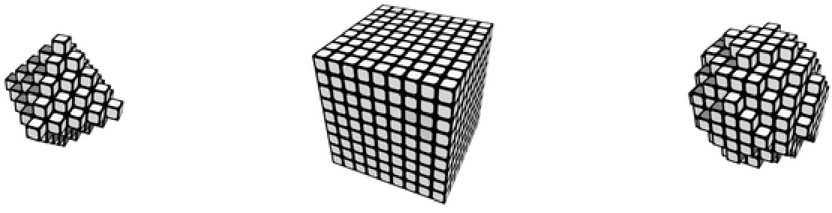
The dialog offers the choice of computation on background or foreground pixels, the choice of an edge treatment, and whether squared rather than exact Euclidean distance values are stored in the result image.

Normally, the algorithm results in a FLOAT image. If the input image contains no spacing information or all spacing entries are set either to 0 or 1, the type of output image can be specified additionally: If GRAY32 is chosen, then the result image contains squared rather than exact Euclidean distance values. If FLOAT is chosen, then the result can be either the squared or exact Euclidean distance image.

This implementation will be replaced by the Euclidean Distance Transformation (Voronoi algorithm).

Euclidean Distance Transformation (Voronoi algorithm) This alternative implementation to Euclidean Distance Transformation based on [7] provides faster computation of the Euclidean distance image. The choice of edge treatmentsis however currently restricted to NET and EMB.

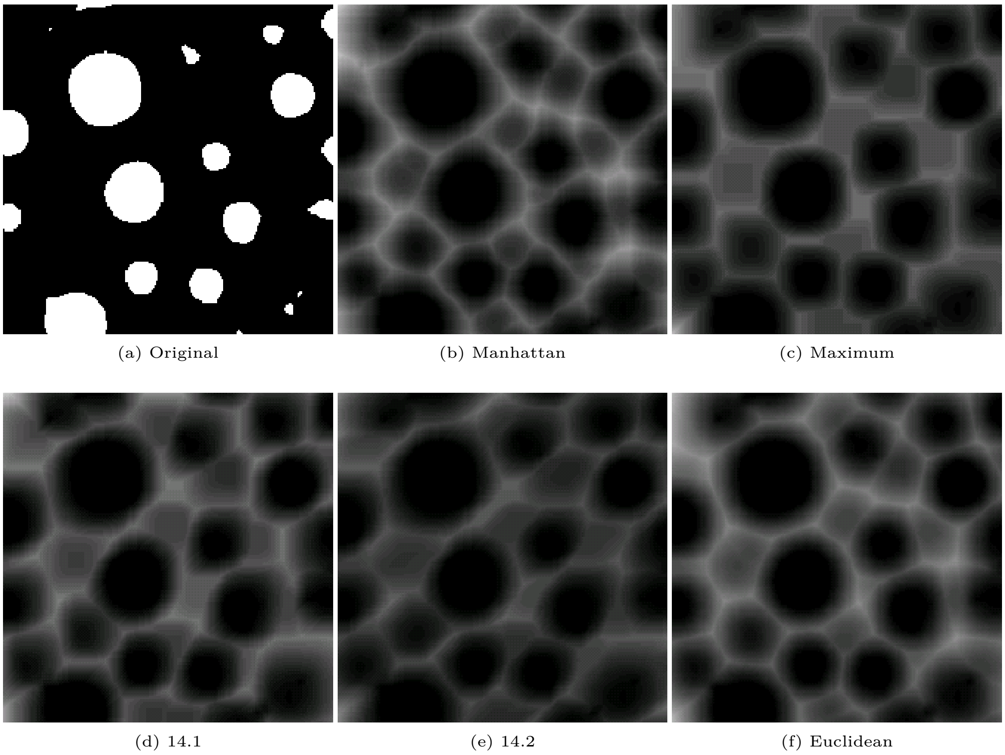
Discrete Distance Transformation This function computes for each background pixel in a MONO image its distance to the closest foreground pixel w.r.t. the chosen discrete metric. See Figure 32 for the choices and Figure 33 for an example.



(a) Manhattan (b) Maximum (c) Euclidean

Figure 32 Discrete metrics visualized by equal radius spheres. Manhattan or city block metric: shortest paths with steps only in the directions of the coordinate axes. Maximum metric: shortest paths with steps only in the coordinate directions. The discrete Euclidean sphere is shown for comparison.

In analogy to the discrete Manhattan and maximum metrics, **MAVI** can also measure discrete distances with respect to the 14.1 and 14.2 adjacency systems. That is, **MAVI** measures the length of the shortest discrete path from a background pixel to the closest foreground pixel that is connected with respect to the given neighborhood.



(a) Original

(b) Manhattan

(c) Maximum

(d) 14.1

(e) 14.2

(f) Euclidean

Figure 33 Distance transformation results for various choices of the metric. The brighter the pixels the higher the distance values.

8.4.4 Geodesic Transformation

In contrast to the morphological operationsoperating on an input image together with a certain structuring element, geodesic transformations use two input images of the same size. The basic procedure is as follows: a morphological operation, that is, erosion or dilation by the elementary isotropic structuring element, is performed on the first image and it is then forced to remain either above or below the second image. New algorithms are created by choice of suitable pairs of input images and iterated application of this basic procedure.

All algorithms in this submenu are based on [19].

Regional Extrema A regional minimum (maximum) of a GRAY8 or GRAY16 image is a connected component of pixels of constant gray value such that any path to a pixel of a lower (higher) gray value includes at least one pixel of a higher (lower) gray value.

The algorithm computes either Regional Minima or Regional Maxima. The adjacency systemdefining the connectivity of the extrema components is chosen by the user, too.

The resulting MONO image has the regional minimum (maximum) pixels as foreground and all the other pixels as background.

H-Extrema The H-Extrema function computes regional extrema of an image under a contrast constraint and modifies the image by use of these regional extrema.

An important concept in the context of regional extrema is that of depth or . The dynamic of an image minimum (maximum) is the minimal height one has to ascend (descend) to get from the given minimum (maximum) to another minimum (maximum) of lower (higher) gray value. The dynamic of the global minimum (maximum) of the image is just the difference between maximal and minimal gray value in the image.

The H-Minima (H-Maxima) transformation reduces the number of regional minima (maxima) of an image by use of a contrast criterion. It suppresses all minima (maxima) whose dynamic is smaller than the threshold Dynamic to be set by the user. This transformation is well suited to prevent oversegmentation in the Watershed Transformation and one option for smoothing in the Cell Reconstruction and Particle Separation algorithms.

The Extended H-Minima (Extended H-Maxima) extracts connected regions surrounding the local minima (maxima). It is equivalent to applying Regional Extrema to the result of H-Minima (H-Maxima). In the MONO output image, all pixels belonging to an H-Minimum (H-Maximum) are set to foreground, while all the other pixels are set to background.

The H-Concave option subtractsthe H-Minima transformed image from the input image. The H-Convex transformation is the differenceof the input image and its H-Maxima.

Adaptive H-Extrema The Adaptive H-Extrema compute regional extrema of an image under a gray value dependent contrast constraint and modify the image by use of the computed regional extrema.

Adaptive H-Extrema is a modified version of H-Extrema with a different contrast criterion. More precisely, the Adaptive H-Extrema adapt the Dynamic parameter to the total height in the gray value relief, whereas in the H-Extrema transformations the Dynamic is a constant independent of the actual gray value of the extremum currently considered.

The dialog offers the same choice of transformations as for the H-Extrema, the Neighborhood, and two parameters for the dynamic: the Dynamic at 0 and the Dynamic at maximum, where maximum is the largest possible value in the range of the image type. The dynamic is interpolated linearly between these two values. This is illustrated in Figure 34. Adaptive H-Minima can save the separation of foreground or background objects if Particle Separation and Cell Reconstruction fail due to strongly diverging sizes of the particles or pores, respectively. For an example of a practical application see the Tutorial 7.2.

Minima Imposition This function helps to remove irrelevant minima as e. g. due to noise. It requires two input images: a MONO marker image marking relevant dark image regions and an image on which the minima are imposed. A suitable marker image can be obtained from the second image by an appropriate transformation such as Regional Extrema.

The Minima Imposition preserves or creates image minima at the locations indicated by the markers and removes those not corresponding to a marker. The MONO marker image has to be selected while the second image is chosen via the dialog.

Minima Imposition is useful for preprocessing to prevent oversegmentation by the Watershed Transformation, provided the marker image contains a nucleus of every segment. See Figure 35 for an example.

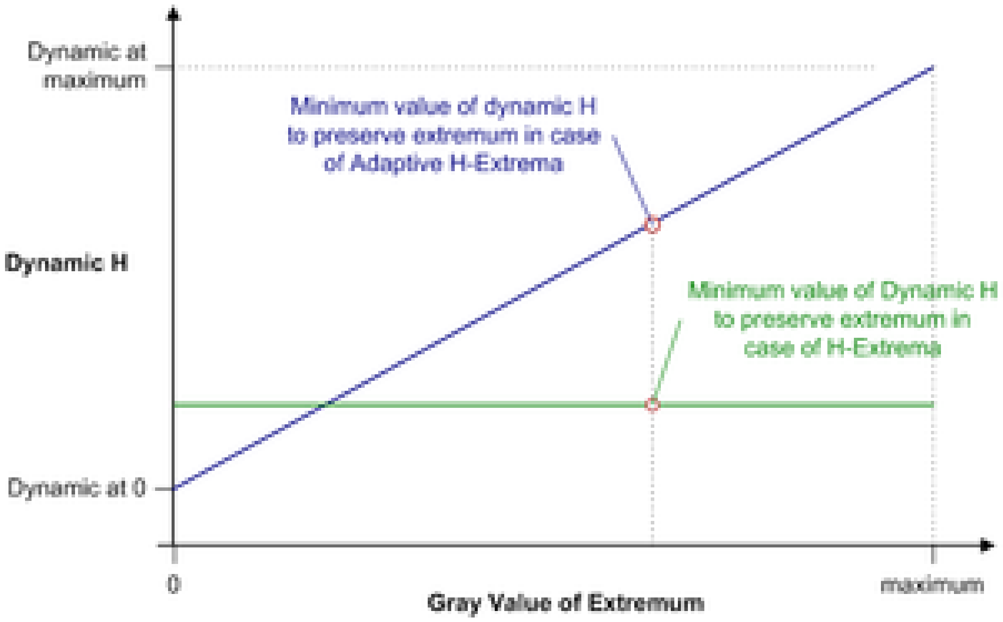


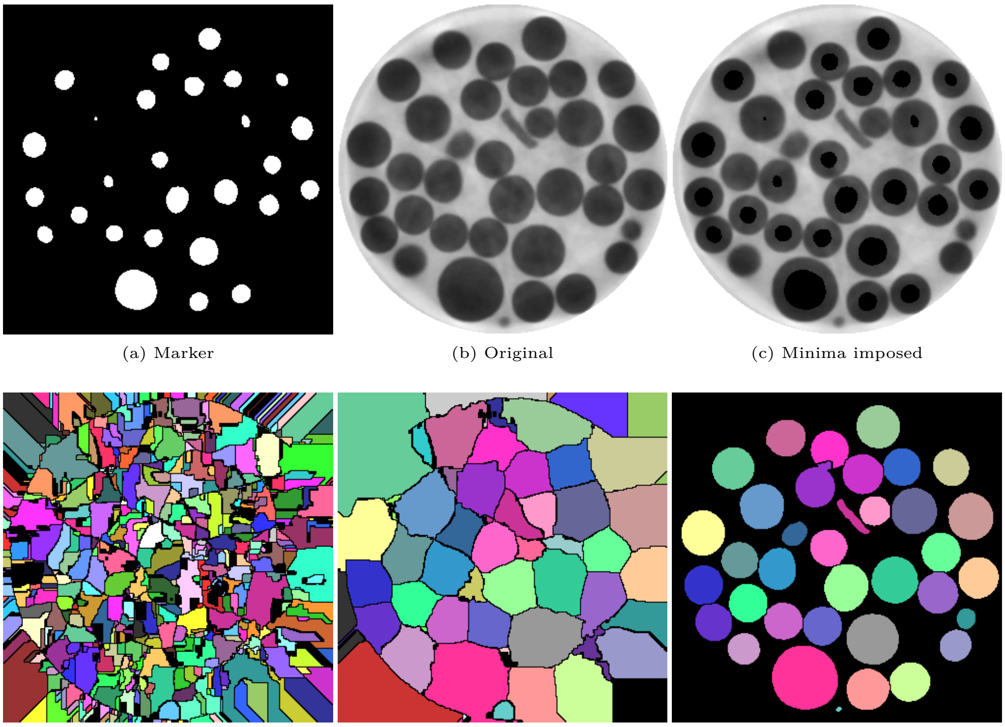
Figure 34 Calculation of dynamic h in Adaptive H-Extrema.

Maxima Imposition This function helps to remove irrelevant maxima as e. g. due to noise. It requires two input images: a MONO marker image marking relevant bright image regions and an image on which the maxima are imposed. A suitable marker image can be obtained from the second image by an appropriate transformation such as Regional Extrema.

The Maxima Imposition preserves or creates image maxima at the locations indicated by the markers and removes those not corresponding to a marker. The MONO marker image has to be selected while the second image is chosen via the dialog.

Ultimate Eroded Set This transform reduces the connected components of the foreground of the MONO input image to their central regions.

The Ultimate Eroded Set is generated by successive erosions of the input image (type MONO) using the structuring element given by the chosen neighborhood. Each time an erosion step would remove a connected component, the remainder of this connected component is kept. That is, the Ultimate Eroded Set is the union of all last pixels of connected components that disappear during successive erosions of the input image. The ultimate eroded set coincides with the regional maximaof the Discrete Distance Transformation on the foreground.


(a) Marker (b) Original (c) Minima imposed

(d) Watershed on (b) (e) Watershed on (c) (f) Mask Image (e) with Binarization

Figure 35 Minima Imposition used as preprocessing for Watershed Transformation.

Ultimate Dilated Set This transform finds the central regions of the connected components of the background of the MONO input image.

The Ultimate Dilated Set is generated by successive dilations of the input image (type MONO) using the structuring element given by the chosen neighborhood. Each time a dilation step would remove a connected component of the background, the remainder of this connected component is kept. That is, the Ultimate Dilated Set is the union of all first pixels of background connected components that appear during successive dilations of the input image. The ultimate dilated set coincides with the regional maximaof the Discrete Distance Transformation on the background.

Reconstruction by Erosion A reconstruction by erosion is a special form of a geodesic erosion and involves two input images: A mask image and a marker image. Both images must be of equal size and the mask image must be pointwise less than or equal to the marker image.

The geodesic erosion of the marker image with respect to the mask image is obtained as follows: First, the marker image is eroded and then the pointwise maximum of the resulting eroded image and the mask image is taken. Thus, the resulting image is forced to remain above the mask image. The iteration of this basic procedure always reaches stability, in the sense that the resulting image can no longer be modified, after a finite number of steps.

The reconstruction by erosion of a mask image from a marker image is defined as an iteration of geodesic erosion of the marker image with respect to the mask image, stopped as soon as stability is reached. To compute the Reconstruction by Erosion in **MAVI** , the marker image has to be selected. The dialog helps in choosing the mask image, which has to be of the same type and pixelwise less than or equal to the marker image. The reconstructed image is pointwise greater than or equal to the mask image and less than or equal to the marker image.

Reconstruction by Dilation A reconstruction by dilation is a special form of a geodesic dilation and involves two input images: A mask image and a marker image. Both images must be of equal size and the mask image must be pointwise greater than or equal to the marker image.

The geodesic dilation of the marker image with respect to the mask image is obtained as follows: First, the marker image is dilated and then the pointwise minimum of the resulting dilated image and the mask image is taken. Thus, the resulting image is forced to remain below the mask image. The iteration of this basic procedure always reaches stability, in the sense that the resulting image can no longer be modified, after a finite number of steps. The reconstruction by dilation of a mask image from a marker image is defined as an iteration of geodesic dilations of the marker image with respect to the mask image, stopped as soon as stability is reached, see Figure 36. To compute the Reconstruction by Dilation in **MAVI** , the marker image has to be selected. The dialog helps in choosing the mask image, which has to be of the same type and pixelwise greater than or equal to the marker image. The reconstructed image is pointwise less than or equal to the mask image and greater than or equal to the marker image.

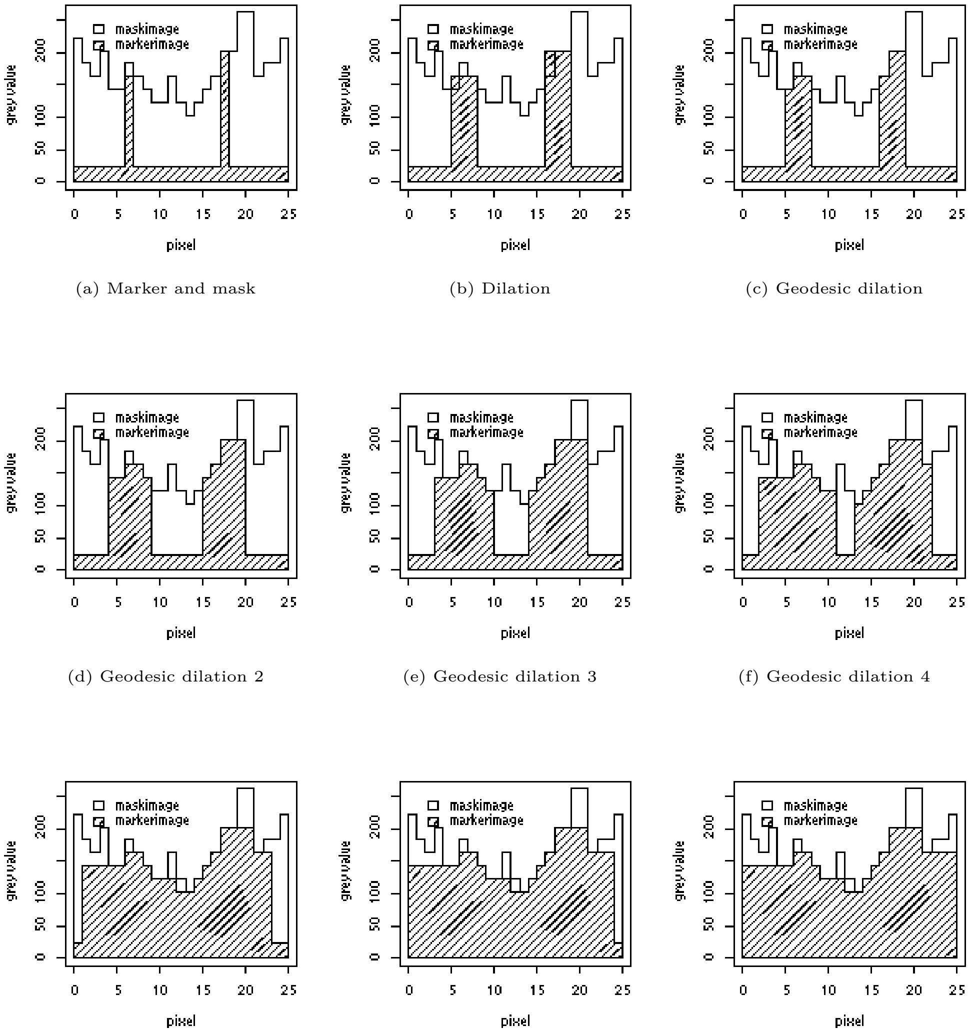
Self Dual Reconstruction This function reconstructs the marker image f by self dual geodesic transformation with respect to the mask image g. Depending on the local relation of the pixel values in marker and mask image, the pixel is assigned the value of the Reconstruction by Erosion or the Reconstruction by Dilation.

The self dual reconstruction is defined by:

$$[R_g(f)](x) = \begin{cases} \left[R_g^d(f)\right](x), & \text{if} f(x) \leq g(x), \\ \left[R_g^e(f)\right](x), & \text{otherwise} \end{cases}$$

Here $R_g^d(f)$ denotes the reconstruction by dilation of f w.r.t. g and $R_g^e(f)$ the reconstruction by erosion of f w.r.t. g.

Self-dual reconstruction can ease segmentation, in particular in cases where contrast and noise are of the same magnitude. To achieve this, use a strongly smoothed version of the original image f as mask image g as provided by the Morphological Mean filter or an Opening.



(a) Marker and mask

(b) Dilation

(c) Geodesic dilation

(d) Geodesic dilation 2

(e) Geodesic dilation 3

(f) Geodesic dilation 4

(g) Geodesic dilation 5

(h) Geodesic dilation 6

(i) Reconstructed image

Figure 36 Reconstruction by Dilation illustrated by a 1D example. The shaded image in (i) is the resulting reconstructed image.

8.4.5 Spectral Transformation

This submenu provides a fast Fourier transform for 3D images as well as it's inverse. The fast Fourier transform allows to determine second order characteristics describing fluctuations of microstructures like the Autocovariance fast and efficiently.

Fast Fourier Transformation Fast Fourier transformation of the input image. Uses FFTW 3.0.1 [2]. Results in a COMPLEX image.

Inverse Fast Fourier Transformation Inverse of the Fast Fourier Transformation. Operates on input images of type COMPLEX. Uses FFTW 3.0.1 [2]. Results in a GRAYF image.

Power Spectrum Power spectrum of input image. Uses FFTW 3.0.1 [2]. Results in a FLOAT image. Use Power Spectrum (Logarithmic) to obtain a result in logarithmic scale.

Magnitude Magnitude of Fast Fourier Transformation of the input image. Uses FFTW 3.0.1 [2]. Results in a GRAYF image. Use Magnitude (Logarithmic) to obtain a result in logarithmic scale.

Phase Phase of Fast Fourier Transformation of the input image. Uses FFTW 3.0.1 [2]. Results in a GRAYF image. Use Phase (Logarithmic) to obtain a result in logarithmic scale.

8.4.6 Cut Hills

The hills of an image are its regional maxima not connected to the image border. The Cut Hills transform removes them by setting them to the lowest gray value they are connected to. Connectivity is defined by the discrete adjacency systemspecified by the user.

The actual hills of the image can be obtained using Extract Hills or by subtractingthe transformed image from the original one.

8.4.7 Fill Holes

The holes of an image are its regional minima not connected to the image border. The Fill Holes transform removes them by setting them to the highest gray value they are connected with. Connectivity is defined by the discrete adjacency systemspecified by the user.

The actual holes of the image can be obtained using Extract Holes or by subtractingthe original image from the transformed one.

8.4.8 Extract Hills

The hills of an image are its regional maxima not connected to the image border. Extract Hills deduces the hills of the image by subtractingthe Cut Hills transformed image from the original one.

8.4.9 Extract Holes

The holes of an image are its regional minima not connected to the image border. Extract Holes deduces the holes of the image by subtractingthe original image from the Fill Holes transformed one.

8.4.10 Skeleton

The skeleton of an object in a MONO image is obtained by thinning the object while preserving its topology. It is a connected set of faces and branches and contains important information about the geometric structure of the object.

The current algorithm is a sequential version of [22] and has been adapted to the other adjacency systems. In the Skeleton dialog, the following parameters can be specified:

Neighborhood adjacency systems w. r. t. which the topology is preserved,

Skeleton Type can be either medial surface, in which case the skeleton will be a connected set of surfaces, or medial axis,

Branching factor is used to control the preservation of end points.

8.4.11 Skeleton Analyzer

The Skeleton Analyzer assigns every Skeleton pixel a gray value representing the point class it belongs to. The classes, their labels, and the corresponding color used in the Slice View are listed in Table 10.

Label	Point class	Color
1	isolated point	🟩
2	inner point of branch	🟦
3	branch end point	🟨
4	junction point of branches	🟪
5	junction point of face and branch	🟦
6	inner point of face	🟥
7	edge point of face	🟩
8	junction point of faces	🟦

Table 10 Classes of Skeleton points specified by Skeleton Analyzer and corresponding colors used for their representation.

8.4.12 Object Filter

The Object Filter allows to remove objects from a label image as generated by Labeling or Cell Reconstruction from the **Segmentation** menu. The removal decision is based on one, or a combination of, the following three characteristics of an object: its volume, its shape factorf_1 (sphericity), and whether or not the object touches the image border. Thus calculation of the Object Features is a prerequisite for applying the Object Filter to obtain the geometric information for the objects.

In the Object Filter dialog window, use the slider Show slice to view a representative cross section. Choose the removal criterion to be applied by clicking on the appropriate tab, then adjust the parameters as described in the following. The result is immediately visible in the preview box labeled Filtered.

For Volume: Use the sliders labeled Minimum Volume and Maximum Volume to set lower and upper volume bounds, respectively. For Border: Select only border objects and only non-border objects to keep the specified objects. For Sphericity: Use the Minimal Shape Factor Sphericity and Maximal Shape Factor Sphericity to set lower and upper boundaries for the shape factor.

Adjust the parameter settings until the preview result is satisfying. Click OK to apply the selected values to the whole dataset. The volume values in the spinboxes are in pixels, the numbers marked Physical are in cubic micrometers. In the lower right corner of the dialog window you can find information about the number of removed/remaining objects.

Remark 4 Note that removing all boundary objects using the Object Filter distorts subsequent statistics on geometric characteristics of the remaining objects as the remaining non-border objects form a size-biased sample. This is due to the fact that large and long objects are more likely to be cut by the image border and thus to be discarded. Statistics on geometric characteristics should rather be based on the Object Features measured on the unfiltered label image together with a treatment for edge effects, see Section 8.5.8.

8.4.13 Map Subfield Feature To Image

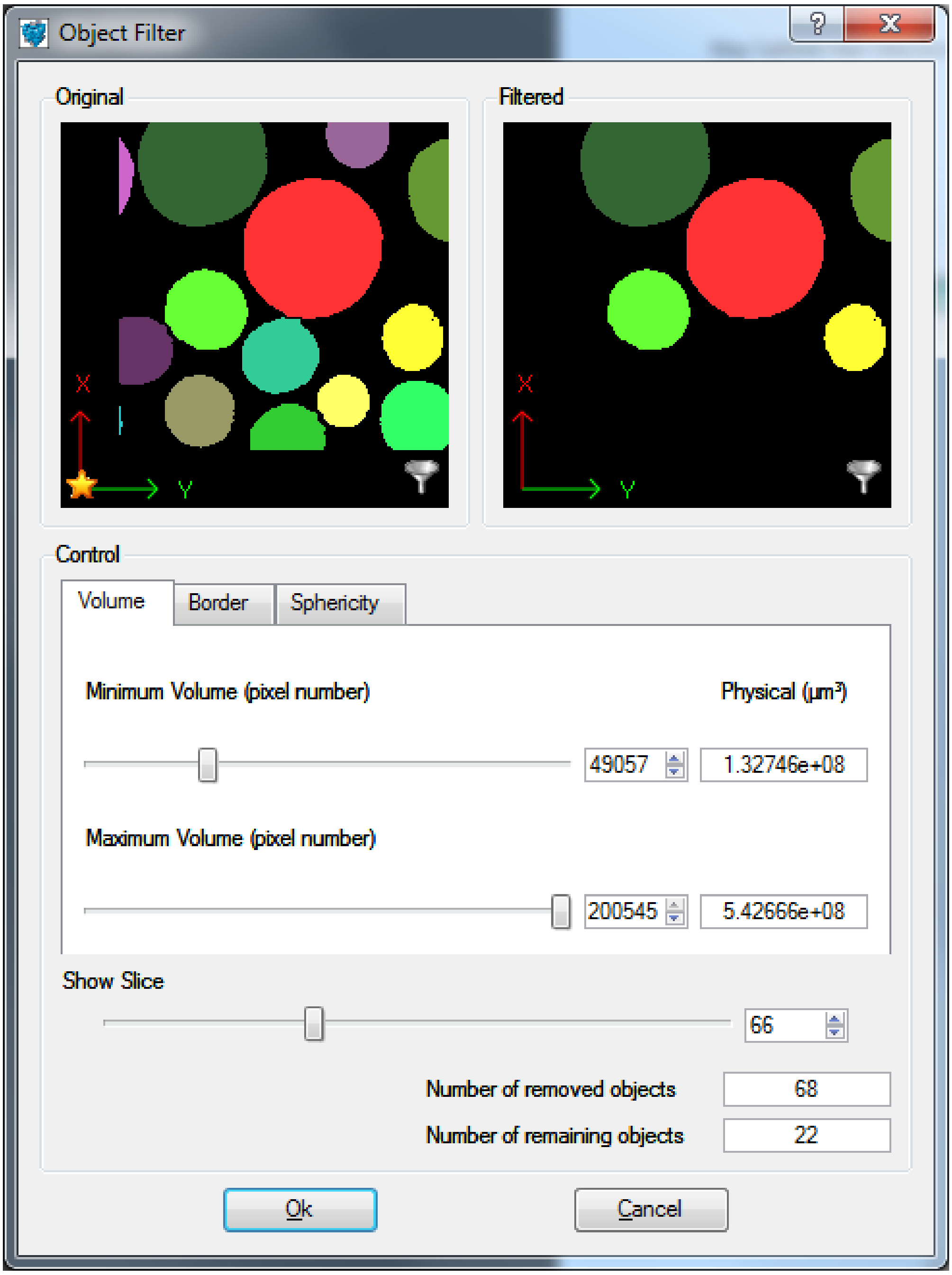
This function is selectable for images for which the Subfield Features have previously been computed. Choose the feature of interest from the dropdown list and click OK. The result is a new GRAYF image containing the numerical value of the selected feature as pixel gray value in each subfield (square for 2D data, cube for 3D data).

The option Crop result in the dialog (see Figure 38) allows to adjust the size of the resulting image such that it does not contain pixels for which no feature data is available. This will happen whenever the image's side lengths are not an integral multiple of the subfields' side lengths. Usually, you should leave this option enabled.

8.4.14 Map Subfield Fiber Directions To Image

This function will be selectable for images for which you have previously computed the (Sub-)Field Fiber Directions. The measurement to be visualized can be chosen via a dialog in a similar manner as described for Map Subfield Feature To Image.

Map Subfield Fiber Directions To Image



Object Filter

Original

Filtered

Control

Volume | Border | Sphericity

Minimum Volume (pixel number): 49057 — Physical (µm³): 1.32746e+08

Maximum Volume (pixel number): 200545 — 5.42666e+08

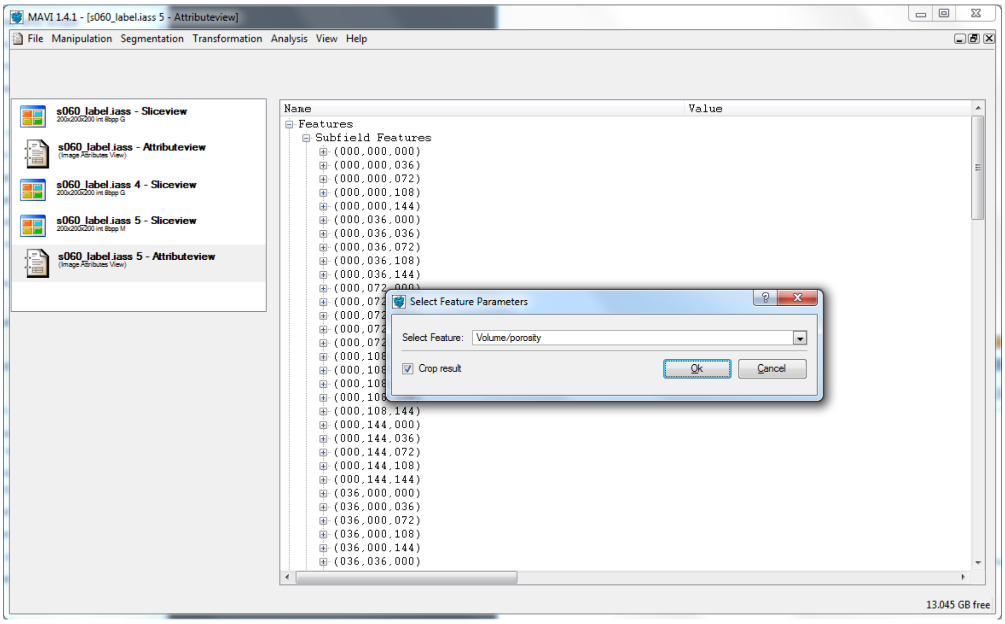
Show Slice: 66

Number of removed objects: 68

Number of remaining objects: 22

Figure 37 Object Filter dialog window.

Map Subfield Fiber Directions To Image



Select Feature: Volume/porosity

☑ Crop result

Figure 38 Map Subfield Feature To Image dialog to select a feature whose value will be written to a new image.

8.5 Analysis

Quantitative geometric analysis of components or objects in images forms the core of **MAVI** . This submenu offers a wide variety of analysis functions, starting with basic gray value Image Statistics and ending with **Integrated Plugins** offering user friendly short cuts for complex image processing chains for Particle Separation and Cell Reconstruction.

Based on efficient algorithms for determining the intrinsic volumes or their densities, the Field Features, Subfield Features, and Object Features yield geometric characteristics for components of microstructures or objects like particles or pores or fibers.

For special classes of structures the Open Foam Features, (Sub-)Field Fiber Directions, and Geometric Tortuosity yield mean characteristics of open foams, local fiber orientations, and a means to quantify how "warped" the pore space is.

The Subfield Features as well as Area Fraction Profile and Grayvalue Profile allow to detect and quantify inhomogeneities.

Moreover, spectral analysis functions like Covariance and Bartlett Spectrum are available.

8.5.1 Image Statistics

Image Statistics shows the gray value histogram of the image as well as minimum, maximum, and mean gray value and the standard deviation. The histogram can be exported in CSV format by clicking on the Export button.

8.5.2 Rotation Mean

The rotation mean of an image is a one-dimensional function giving as ith value the average value of all pixels at the following distance from the image center:

$$i s_{\min} d, \quad \text{where} \quad s_{\min} = \min\{s_1, s_2, s_3\} \tag{5}$$

denotes the minimum of the lattice spacingsin the three coordinate directions. The user defined parameter d controls how fine the rotation mean is discretized. The default value $d = 1$ results in a bin width of one pixel (for an isotropic lattice) while $d = 0.25$ samples the distances four times finer. Note however that for distances smaller than $2s_{\min}$ this fine sampling is impossible due to lack of values.

8.5.3 Field Features

Field features measure geometric characteristics of the foreground of MONO images. It is not neccessary that the foreground consists of isolated and topologically simple objects. It is however assumed that the foreground pixels form a discrete representation of a subsample of a macroscopically homogeneous microstructure. That is, the structure extends beyond the image without changing its average properties.

The field features are essentially the intrinsic volumes and their densities, also known as Minkowski functionals or quermass integrals. For a mathematical definition of the field features and their interpretation for constituents of macroscopically homogeneous microstructures see [12, Chapter 5]. The algorithm for measuring the field features goes back to [6] and is comprehensively described in [10]. Note that all measurements rely on local 2x2x2 pixel configurations only and do not require a surface meshing. Edge effects therefore occur at the last slices of the image only. These border configurations are treated differently such that the final results are free of edge effects.

Generalized projections onto one- and two-dimensional subspaces are computed as intermediate steps of the measurement of the integral of mean curvature and the surface area, respectively. Here 'generalized' means measured with multiplicity as illustrated in Figure 39 below. These generalized projections carry information about anisotropies and preferred directions. Discrete distributions of the fiber direction in the typical fiber point and of the surface normal direction in the typical surface point can be deduced. They are concentrated on the respective set of 13 discrete directions given by the lattice unit cell (see 4.4, Tables 2 and 3) and represented in the output of the field features by the weights for these directions.

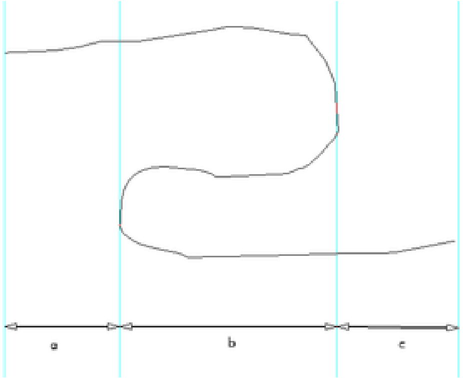


Figure 39 Generalized projection of a bent curve: The length of the projection of this curve onto the x-axis is a+b+c. The length of the generalized projection is however a+3b+c since all three 'layers' contribute.

For special cases of microstructures the field features have to be interpreted as appropriate, the Open Foam Features are provided as special function.

The base unit for all features is one meter, the only exception to this rule is an image without pixel spacings (see 4.1.1). In this case, the base unit is one pixel.

The following Table 11 lists the Field Features measured for 3D images together with their description and the unit they are measured in.

Name as in attributes view	unit	description
total volume V	[m^3]	of the image (foreground and background)
volume of foreground	[m^3]	
volume density	[ ]	volume fraction V_V, foreground volume $/V$
surface	[m^2]	area of the inner surface S
surface density	[m^{-1}]	specific surface area $S_V = S/V$
surface fractal dimension	[ ]	box dimension related to S
area of projections	[m^2]	perpendicular to the 13 directions (Table 2)
orientation distribution of surface normals	[ ]	relative frequencies
integral of mean curvature	[m]	integral of Germain's curvature M
integral of mean curvature density	[m^{-2}]	$M_V = M/V$
integral of mean curvature fractal dimension	[ ]	box dimension related to M
lengths of projections	[m]	on lines in the 13 normal directions (Table 3)
orientation distribution of fibers (edges)	[ ]	relative frequencies
specific fiber length	[m^{-2}]	(6)
fiber aspect ratio	[ ]	(7)
integral of total curvature	[ ]	integral of Gaussian curvature
total curvature density	[m^{-3}]	integral of total curvature $/V$
Euler number for all neighborhoods	[ ]	Euler numbers w.r.t. 26, 14.1, 14.2, and 6 adjancency systems
Euler density	[m^{-3}]	Euler number $/V$
mean chord lengths	[m]	mean chord lengths in 13 directions (Table 2), (8)
mean chord lengths mean	[m]	mean of the 13 mean chord lengths
structure model index	[ ]	shape factor, (9)

Table 11 Field Features measured for 3D images.

The density of the integral of mean curvature M_V yields the specific fiber length L_V:

$$L_V = 4M_V/\pi \tag{6}$$

Comparison of the generalized projection lengths in the coordinate directions can reveal dilation or compression of a fibrous structure: The ratio

$$2\text{projection length}_\mathrm{z}/(\text{projection length}_\mathrm{x} + \text{projection length}_\mathrm{y}) \tag{7}$$

equals 1 for an isotropic structure, variations indicate dilation or compression in z-direction. Of course, these characteristics are meaningful for fiber structures only.

The mean chord lengths (MIL) in the 13 discrete directions are deduced from the areas of projection via

$$\mathrm{MIL} = \mathrm{V_V}/\text{projection area}. \tag{8}$$

The structure model index SMI – a shape factor for components – can be derived from volume fraction, specific surface area and density of the integral of mean curvature [11]

$$\mathrm{SMI} = 12\mathrm{V_V M_V/S_V^2}. \tag{9}$$

It assumes values 0 for ideal planar structures, 3 for spherical cylinders and 4 for non-overlapping balls.

The following Table 8.5.3 lists the Field Features measured for 2D images together with their description and the unit they are measured in.

Name as in attributes view	unit	description
total area A	[m^2]	of the image (foreground and background)
area of foreground	[m^2]	
area density	[ ]	area fraction A_A, foreground area $/A$
boundary length	[m]	
boundary length density	[m^{-1}]	specific boundary length, total length per area
lengths of projections	[m]	in the 8 discrete directions
integral of total curvature	[ ]	integral of Gaussian curvature
total curvature density	[m^{-2}]	integral of total curvature $/V$
Euler number	[ ]	Euler number w.r.t. 6.1 adjancency system
Euler density	[m^{-2}]	Euler number $/A$
mean chord lengths	[m]	mean chord lengths in 8 discrete directions
mean chord lengths mean	[m]	mean of the 8 mean chord lengths

Table 12 Field Features measured for 2D images.

Remark 5 Both field features and object features are based on the intrinsic volumes. Nevertheless, the total values of e. g. volume or surface area in the field features might differ from those obtained by the object features. This is due to the fact that the latter are measured just for the present object, while

MAVI implies the whole space to be filled with an on average similar structure when measuring the Field Features of the foreground.

Field features and object features coincide if you ensure that the foreground does not intersect the image border, e. g. by ***Manipulation***:Padding.

Precision of the Field Features' measurement There is a variety of sources for measurement errors when determining the Field Features, starting with the imaging conditions, continuing with discretization effects, and finally binarization. It is therefore impossible to quantify the overall measurement error. However, there are some rules of thumb:

Specific surface area as well as the density of the integral of mean curvature are accompanied by the respective so called fractal dimensions. These values indicate resolution dependence of the measurements. That means, if the surface fractal dimension is larger than 2.2, then the resolution of the image was not sufficient to measure the specific surface area properly. The same holds true for values of the integral of mean curvature fractal dimension deviating significantly from 1.

Large differences of the Euler numbers with respect to the different discrete neighborhoods are another indicator for the resolution being not sufficient for the Euler number measurement on the given structure.

If you are nevertheless sure that the resolution was good enough then apply a smoothing operationbefore or after segmentation.

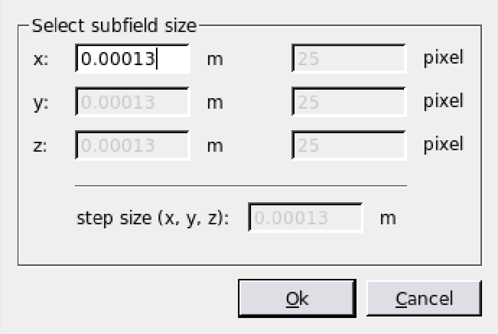
Robustness of the field features decreases with their dimension. The values for volume fraction or specific surface area can still be trusted even if the integral of mean curvature fractal dimension is 2 or the Euler numbers differ strongly. On the other hand, if the surface fractal dimension is 2.5, then not only the measured specific surface area but also the integral of mean curvature and the Euler number should be regarded with suspicion.

8.5.4 Subfield Features

Please note that in the current version of MAVI, Subfield Features are available for 3D images of type MONO, only.

The Subfield Features calculate the Field Features locally, thus allowing to quantify structural differences within one sample without cutting up the respective image. As the Field Features, the Subfield Features are measured on the foreground of MONO images. The cubic (3D) subfields of user-specified size form a tiling of the image. That is, subfields do not overlap and pixels at the image edges not being covered are not taken into account.

The side length of the subfields needs to be specified by the user in units of meters, see Figure 40.



Select subfield size

x:	0.00013	m	25	pixel
y:	0.00013	m	25	pixel
z:	0.00013	m	25	pixel

step size (x, y, z): 0.00013 m

Figure 40 Subfield Features: dialog for specifying the subfield size.

For a list of the resulting measurements in each subfield see the Field Features. In the resulting view, each subfield can be found by the coordinate of its first point, in [pixel]. Subfield Feature results can be saved using **Field Features as CSV** from the **File**:**Export** menu and visualized using Map Subfield Feature To Image from the **Transformation** menu.

8.5.5 Area Fraction Profile

For macroscopically homogeneous microstructures, the volume fractioncoincides with the area fractionin a 2D slice. Conversely, changes of the area fraction indicate inhomogeneities, reveal layered structures or structures with a gradient.

The Area Fraction Profile calculates for MONO images the area fraction of each slice in the coordinate directions. The results can be exported as a CSV file via **File**:**Export**: **Area Fraction Profile as CSV**.

Optionally, a mask image can be provided. In that case, the area fraction is obtained as area of foreground within mask divided by area of mask in the current slice. This allows e. g. to quantify changes of material density within a complex shaped part.

For 2D images, the Area Fraction Profile yields the profile of line fractions. That is, the length fraction of section lines covered by foreground.

8.5.6 Grayvalue Profile

The Grayvalue Profile calculates for gray value images the mean gray value in each slice in the coordinate directions. Strong or systematic changes indicate global gray value fluctuations either due to imaging artefacts like beam hardening or due to structural inhomogeneities like layers or a gradient.

The Grayvalue Profile can be exported as a CSV file via **File**:**Export**: **Grayvalue Profile as CSV**.

Optionally, a mask image can be provided. In that case, only the gray values of those pixels are averaged that are foreground in the mask image.

For 2D images, the Grayvalue Profile yields the mean gray value in each line in the coordinate directions.

8.5.7 Granulometry

This function performs a fast exact spherical granulometry transform on MONO images. The user defines whether foreground (default) or background are analyzed.

The spherical granulometry assigns each pixel the diameter of the largest ball completely contained in the foreground or background, respectively, and covering the pixel. It thus yields the local structure thickness or size or the local pore size.

More precisely, let X be the structure to be analyzed, B_r the ball of radius r, and $X \circ B_r$ the morphological Opening of X with B_r. The granulometry distribution function

$$G(d) = 1 - \frac{1 - V_V(X \circ B_{d/2})}{1 - V_V(X)}, \qquad d \geq 0$$

yields a volume weighted generalized size distribution. It can be measured from the MONO image representing X as foreground by successive morphological openings with growing balls until all foreground pixels have disappeared. Finally each pixel x is assigned the diameter of the ball used as structuring element in the step where x disappeared for the first time. The resulting image is called ultimately opened or granulometry image. For the actual calculation a more efficient algorithm based on the medial axis coinciding with the crest lines of the Euclidean distance image is used.

Results are the granulometric curve (also called pattern spectrum) or a GRAYF image holding the local sizes (diameters). In the latter case, the granulometric curve is computed and attached to the image attributes, too, visible via the Image Attributes View.

The granulometric curve is the histogram of the volume weighted size distribution. Peaks in this curve indicate the predominant structure or pore size. Results can be saved using **File**:**Export**: **Granulometric Data as CSV**. The granulometric curve contains the raw analysis results, that is, each measured discrete diameter forms its own bin. Thus, one either has to smooth it, e. g. by a coarser binning, or the cumulative distribution function should be used.

For a color coded visualization of the granulometry image, Spread to GRAY8 and choose the Rainbow color table (see 18) in the control windowof the Slice View. If several images shall be compared, then define the same "Gravalue range of interest" for all images when Spreading.

Remark 6 This algorithm's run time depends highly on the local structure thickness or pore size and can increase to an impracticable length. Consider alternatives like the Euclidean Distance Transformation if diameters larger than 200 pixels are expected.

8.5.8 Object Features

An object in a 3D image is a connected component representing for instance a particle, a pore, a cell, or a fiber. Object Features can be applied to MONO images or label images as obtained by a Labeling or the Watershed Transformation. In the first case, the foreground is treated as the one and only object. In the case of a label image, the geometric characteristics of each of the objects are computed.

The volume is measured by simply counting the pixels of the object. For details on the measurement method for the intrinsic volumes refer to the Field Features section. The following Table 8.5.8 lists the features measured for objects in 3D images together with their description and the unit they are measured in. The bounding box is the smallest cuboid with axes parallel to the coordinate axes and containing the object.

For an object with volume V the diameter d of the equal volume ball is just

$$d = \sqrt[3]{6V/\pi}. \tag{10}$$

Note that the diameters of an object in the 13 normal directions (Table 3) coincide with those for its convex hull formed by planes perpendicular to those directions.

Name as in attributes view	unit	description
center	[Pixel]	central pixel of bounding box
diameter of corresponding ball	[m]	diameter of equal volume ball, (10)
diameters	[m]	lengths of projections on lines in the 13 normal directions (Table 3)
Euler number	[ ]	w.r.t. 14.1 adjancency system
integral of mean curvature	[m]	integral of Germain's curvature M
integral of total curvature	[ ]	integral of Gaussian curvature
no. of voxels	[ ]	
object	[ ]	number of the object, 0 refers to background
offset	[Pixel]	lower left front corner of bounding box
on border	true/false	indicates whether object touches image border
shape factor 1	[ ]	sphericity, (11)
shape factor 2	[ ]	(12)
shape factor 3	[ ]	(13)
size	[Pixel]	edge lengths of bounding box
status		if not OK, a numerical problem occured, e. g. due to the object being too small
surface area	[m^2]	area of inner and outer surface
surface of convex hull	[m^2]	surface area of convex hull formed by planes perpendicular to the 13 normal directions (Table 3)
volume	[m^3]	
volume of convex hull	[m^3]	volume of convex hull formed by planes perpendicular to the 13 normal directions (Table 3)
weight Miles-Lantuejoul	[ ]	edge correction factor, (14)

Table 13 Object features measured for 3D images.

The Euler number is a topological characteristic. The following examples illustrate its meaning: The Euler number of an object having no holes ('caves') and 'tunnels' is 1, if the object has a hole then its Euler number is 2, if the object is topologically equivalent to the torus, we get 0, if the object has two 'caves', we get Euler number 2, and if it has two 'tunnels' one obtains -1. Euler number and the integral of total curvature carry the same information about the object. They differ by the constant factor 4π.

Additional characteristics for 3D objects like length, width, and thickness or elongation are contained in the module Particle Features.

For 2D objects, **MAVI** determines the following characteristics:

Remark 7 When measuring the Object Features, ***MAVI*** minimizes the mean discretization error for an object in "general" position. Therefore, the quality of the measurements can not be judged by comparing to theoretical

Name as in attributes view	unit	description
area	[m^2]	
boundary length	[m]	
center	[Pixel]	central pixel of bounding box
diameter of corresponding circle	[m]	diameter of equal area circle, (10)
diameters	[m]	lengths of projections on lines in 8 directions
Euler number	[ ]	w.r.t. 6.1 adjancency system
integral of curvature	[ ]	integral of Gaussian curvature
mean chord length	[m]	
no. of pixels	[ ]	
object	[ ]	number of the object, 0 refers to background
offset	[Pixel]	lower left front corner of bounding box
on border	true/false	indicates whether object touches image border
size	[Pixel]	edge lengths of bounding box
status		if not OK, a numerical problem occured, e. g. due to the object being too small

Table 14 Object features measured for 2D images.

values for objects in a particular position. In particular, the surface area and the integral of mean curvature of a cube parallel to the coordinate axes are underestimated, while they are overestimated for the same cube rotated e. g. by 20° about the z-axis, flipped, and rotated by 20° about the z-axis again.

Isoperimetric shape factors The isoperimetric shape factors

$$\begin{aligned} f_1 &= 6\sqrt{\pi}V/\sqrt{S^3} && (11)\\ f_2 &= 48\pi^2 V/M^3 && (12)\\ f_3 &= 4\pi S/M^2 && (13) \end{aligned}$$

are derived from the intrinsic volumes, see Field Features. These shape factors are normalized such that $f_1 = f_2 = f_3 = 1$ for a ball. Deviations from 1 thus describe various aspects of deviations from spherical shape. The shape factor f_1 is often called sphericity. Examples for some simple objects and the theoretical values of the shape factors 1-3 for them are given in Table 15. More shape information can be derived e. g. from the ratio of the volumes of an object and its convex hull or the ratio of minimal and maximal diameter.

Edge correction When determining distributions of characteristics like volume or diameter of objects in an image, objects intersecting the boundary should not be included. However, bigger objects have a higher probability of intersecting the boundary than smaller ones. The Miles-Lantuejoul weight

Object	description	f_1	f_2	f_3
ball		1.00	1.00	1.00
prolate spheroid	long:short axis as 2:1	0.90	0.76	0.89
oblate spheroid	long:short axis as 2:1	0.87	0.80	0.94
cylinder	height:diameter as 1:2	0.75	0.68	0.93
cylinder	height:diameter as 1:1	0.82	0.71	0.91
cube		0.72	0.57	0.85
cuboid	length:depth:height as 1:2:3	0.62	0.42	0.78

Table 15 Isoperimetric shape factors. Examples.

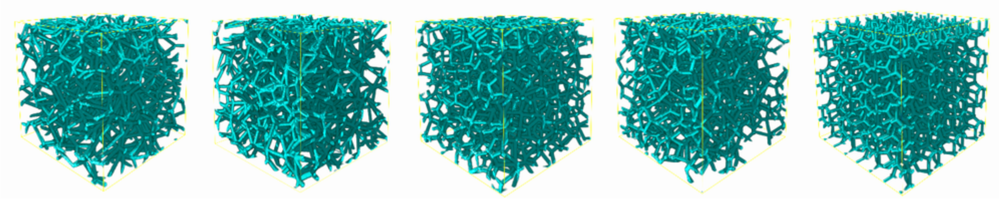
corrects boundary effects caused by this size dependent sampling. Each object is assigned a weight proportional to the reciprocal of its probability of being sampled. Roughly speaking, large objects not hitting the boundary are hard to observe. Once this unlikely event happens, a high weight is assigned to this object:

$$\text{total volume/volume of window eroded by bounding box of object.} \tag{14}$$

This factor appears as “weight Miles-Lantuejoul” in the Object Features. For border intersecting objects, it is 0. When estimating moments etc. from the object features, sum all values multiplied by the weight. Subsequently, normalize by dividing by the sum of the weights, see [12, Section 5.2.6].

8.5.9 Open Foam Features

The Open Foam Features are mean characteristics of open foams, calculated from the Field Features using a model assumption: The typical cell of the open foam is assumed to be up to a scaling factor the same as the typical cell of one of the 5 choosable models. The Laguerre tessellation models are generated by force biased packings of balls. The ball volumes follow a log normal distribution, with low (0.2) and high (2.0) coefficient of variation cv, respectively. See [18] and [12, Section 7.6] for theoretical background.



(a) Poisson Voronoi (b) Hardcore Voronoi (c) Laguerre, cv 0.2 (d) Laguerre, cv 2.0 (e) Weaire-Phelan foam

Figure 41 Models used by the Open Foam Features. (a)-(d) – random tessellation models.

Open Foam Features can be measured from 3D MONO images only. The foreground should represent the strut system of an open foam. That is, the structure is supposed to have no closed walls and the struts have to be closed. Hollow struts or isolated closed walls should therefore be closed or removed, respectively, using morphological transformations 8.4.2.

Remark 8 Open Foam Features do NOT support 2d images.

The scaling factor can be deduced from the strut length density (total strut length per unit volume) or the density of the Euler number, see [17] for details. In a simulation study, the Euler number based scaling proved to be more precise than the strut length based one. It is therefore now the default scaling used by **MAVI** . The strut length based scaling is included for backward compatibility only.

System	characteristic	unit
Cells	mean cell volume	[m^3]
	mean number of cells per unit volume	[m^{-3}]
	mean number of cells per unit length	[m^{-1}]
	ppi-value (pores per inch)	[ ]
	mean cell surface area	[m^2]
	mean cell diameter	[m]
	mean cell chord legth	[m]
Faces	mean number of faces per unit volume	[m^{-3}]
	mean face diameter	[m]
	mean face area	[m^2]
Edges	mean number of edges per unit volume	[m^{-3}]
	mean edge diameter	[m]
	mean edge circumference	[m]
	mean edge cross section area	[m^2]
	length density of the edges	[m^{-2}]
Nodes	mean number of nodes per unit volume	[m^{-3}]

Table 16 Characteristics calculated by the Open Foam Features.

The Open Foam Features compute mean values of the cells, faces, and edges of the foam structure in the image. In order to analyze properties of individual cells or to estimate empirical distributions, the cells have to be reconstructed by an image analytical Cell Reconstruction. Subsequently, the Object Features can be calculated on the cell system. The cell reconstruction process is applied to the open nickel foamand zinc foamexamples.

8.5.10 Geometric Tortuosity

The geometric tortuosity of a porous medium is the ratio of the length of the shortest path between two points within the foreground component to their

Euclidean distance.

All algorithms measuring the geometric tortuosity are based on the concept of geodesic distance of two foreground pixels [19]. It's the length of the shortest digital path connecting the two pixels and not leaving the foreground component. **MAVI** measures on MONO images, for each coordinate direction, the ratio of geodesic and Euclidean distance of border foreground pixels to the opposite image border. The geometric tortuosity is then calculated by a weighted averaging of all such ratios. More precisely, the ratios are squared and weighted by the frequency of their occurence.

The results can be exported as a CSV file via **File**:**Export**: **Geometric Tortuosity as CSV**.

8.5.11 (Sub-)Field Fiber Directions

The (Sub-)Field Fiber Directions estimate the distribution of the local direction in the typical point of a fiber system. Contrary to the majority of the **Analysis** functions they operate on GREY8 images. Images to be analyzed should contain a randomly distributed fiber system and the fibers should have a larger gray value than the matrix or pore space pixels. Furthermore, the fibers should have an approximately constant and known radius. These assumption are in particular fulfilled by μCT images of glass fiber reinforced composites. A typical example is discussed in 7.3.

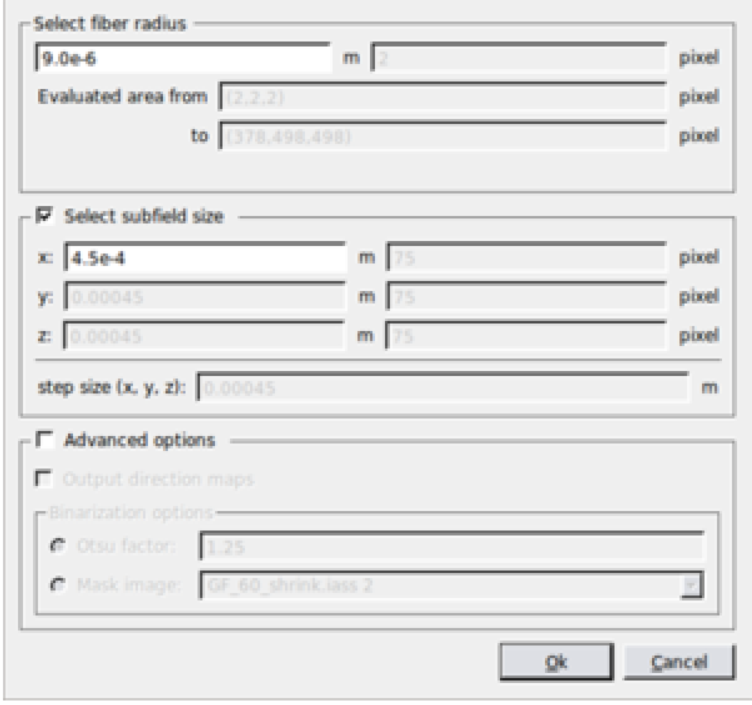
Remark 9 (Sub-)Field Fiber Directions do NOT support 2D images.

The (Sub-)Field Fiber Directions can operate on the whole image as the Field Features or on subvolumes of user defined size as the Subfield Features.

When calling the (Sub-)Field Fiber Directions, the user has to specify the radius of the fibers in meters, see the dialog in Figure 42. The fiber diameter should be resolved with 5-10 pixels. The function proved however to work at the absolute minimum of 3 pixels per fiber diameter too, see the tutorial 7.3. The (Sub-)Field Fiber Directions calculate the local fiber direction in each pixel without segmentation of individual fibres. **MAVI** uses the method based on the Hessian matrix of second order partial derivatives of the gray values as described in [15].

The gray value image is smoothed by a Filters in **MAVI** with their masks, values, and parameters. Mask size is the edge length of the cubic/square filter mask in pixels. The values in braces refer to the 2D version if this differs from 3D with parameter adjusted to the fiber radius. Subsequently, the Hessian matrix $H(x)$ of second order gray value derivatives of the smoothed image is calculated in each pixel x. The eigenvectors of H carry information about directions of the fiber system at x. The least gray value variation is expected along the fibre. Thus the eigenvector corresponding to the smallest eigenvalue of the Hessian matrix $H(x)$ at x is interpreted as the local direction ν_1, ν_2, ν_3 in pixel x.

Finally, the result is restricted to the pixels belonging to the fiber system using either a mask image provided by the user or an automatically computed mask image. The former is specified by the user via the dialog (Figure 42) while the latter is obtained by global thresholding (see 8.3.1). The threshold according to Otsuis multiplied by 1.25 (default) or a user defined value.



Select fiber radius

9.0e-6 m 2 pixel

Evaluated area from (2,2,2) pixel

to (378,498,498) pixel

☑ Select subfield size

x: 4.5e-4 m 75 pixel

y: 0.00045 m 75 pixel

z: 0.00045 m 75 pixel

step size (x, y, z): 0.00045 m

□ Advanced options

□ Output direction maps

Binarization options

○ Otsu factor: 1.25

○ Mask image: GF_60_shrink.iass 2

Figure 42 (Sub-)Field Fiber Directions: dialog for entering the fiber radius and an optional mask image.

Remark 10 The rational behind the increased threshold is the observation that the resulting directional distribution is not heavily distorted if the cores of the fibers are regarded only.

The averaged outer product of the components ν_1, ν_2, ν_3 is an estimator for the so-called second order orientation tensor widely used in simulation of materials properties in order to incorporate anisotropic local behaviour:

$$a_{ij} = \sum_{x \in \text{image and fiber system}} \nu_i(x)\nu_j(x). \tag{15}$$

The eigenvector corresponding to the largest eigenvalue l_3 of the orientation tensor now yields the principal fiber direction. This principal direction is however only meaningful if the fiber system features a preferred direction at least locally (in the subvolumes). If the fibers are isotropically distributed, then the three eigenvalues coincide. The more anisotropic the fiber system gets, the stronger differ the maximal and the minimal eigenvalue, l_3 and l_1, respectively. Thus, a measure for the degree of anisotropy is

$$1 - l_1/l_3. \tag{16}$$

This expression assumes values between 0 and 1, where 0 corresponds to an ideal isotropic fiber distribution, 1 to either all fibers being parallel to the main fiber direction or all fibers being parallel to a plane.

The (Sub-)Field Fiber Directions yield the following information for the whole field and – if applicable – for each subfield: If subfields have been specified, the

Name	description
Anisotropy value	(16)
Eigenvalues of the orientation tensor	$l_3 \geq l_2 \geq l_1$
Orientation tensor	3x3 symmetric matrix (15)
Mean fiber direction	eigenvector to l_3, princial fiber direction
Status	Warns about numerical problems, e. g. due to the subfield not containing any pixels belonging to the fiber system. Should state "OK".

Table 17 (Sub-)Field Fiber Directions results.

Averaged Profile holds additionally the characteristics from Table 8.5.11 averaged over all subfields in the same layer. That is, for each coordinate direction, the characteristics are averaged over all subfields with the same respective coordinate.

If subfields have been specified, **MAVI** offers the possibility to visualize selected fiber measurements using Map Subfield Fiber Directions To Image from the **Transformation** menu. Results can be saved using **File**:**Export**: **Subfield Fiber Directions as CSV**.

8.5.12 Spectral Methods

Second order characteristics describe fluctuation of microstructures. They as well as their counterparts in frequency space can be measured from 3D images of the microstructures. The fast Fourier transform allows to determine these quantities like the covariance function fast and efficiently.

Convolution The convolution is a local moving average operation involving two images, where the second image serves as a filter mask. For each pixel in the first image, the second image is shifted so that its origin is located at the given pixel and the average is computed over all pixels covered by the second image. For calculation of the average, pixel values in the first image are weighted by the corresponding values in the second image.

The convolution of two images is equal to the Inverse Fast Fourier Transformation of the product of their Fast Fourier Transformations. Accordingly, the implementation of the convolution in **MAVI** is based on Fourier methods using the FFTW 3.0.1 library [3].

Both images have to have the same dimensions.

Covariance The covariance of two images (a and b) is basically the Inverse Fast Fourier Transformation of the product of the Fast Fourier Transformation of image a and the Complex Conjugate of the Fast Fourier Transformation of image b. High gray values in the resulting image indicate strong similarity in the structures of the input images.

Covariance (Rotation Mean) Rotation mean of the Covariance – a one-dimensional function giving as ith value the average covariance at distance $is_{\min}d$ where $s_{\min}$ is the minmum of the lattice spacings in the coordinate directions. Shortcut for Rotation Mean applied to the Covariance saving computation time and memory. The parameter d is determined by the user with values 1 (default) or 0.25. For details see Section 8.5.2. The result can be exported using **File**:**Export**: **Rotation Mean of Covariance as CSV**.

Autocovariance The autocovariance of an image is the Covariance of this image with itself, that is the Inverse Fast Fourier Transformation of its Power Spectrum.

The autocovariance of a MONO image yields an unbiased estimator for the covariance of the foreground seen as realization of a spatially homogeneous random closed set (see [5]). It is called two-point probability function or two-point correlation function, too.

Autocovariance (Rotation Mean) Rotation mean of the Autocovariance – a one-dimensional function giving as ith value the average autocovariance at distance $is_{\min}d$ where $s_{\min}$ is the minmum of the lattice spacings in the coordinate directions. Shortcut for Rotation Mean applied to the Autocovariance saving computation time and memory. The parameter d is determined by the user with values 1 (default) or 0.25. For details see Section 8.5.2. The result can be exported using **File**:**Export**: **Rotation Mean of Autocovariance as CSV**.

Bartlett Spectrum The Bartlett spectrum of a MONO image is basically the Fast Fourier Transformation of the Autocovariance of that image. It is the counterpart to the covariance in frequency space and – contrary to the Power Spectrum – not distorted by the image size. It solely captures fluctuations of the analyzed microstructure.

Bartlett Spectrum (Rotation Mean) Rotation mean of the Bartlett Spectrum – a one-dimensional function giving as ith value the average Bartlett Spectrum at distance $is_{\min}d$ where $s_{\min}$ is the minmum of the lattice spacings in the coordinate directions. Shortcut for Rotation Mean applied to the Bartlett Spectrum saving computation time and memory. The parameter d is determined by the user with values 1 (default) or 0.25. For details see Section 8.5.2. The result can be exported using **File**:**Export**:**Rotation Mean of Bartlett Spectrum as CSV**.

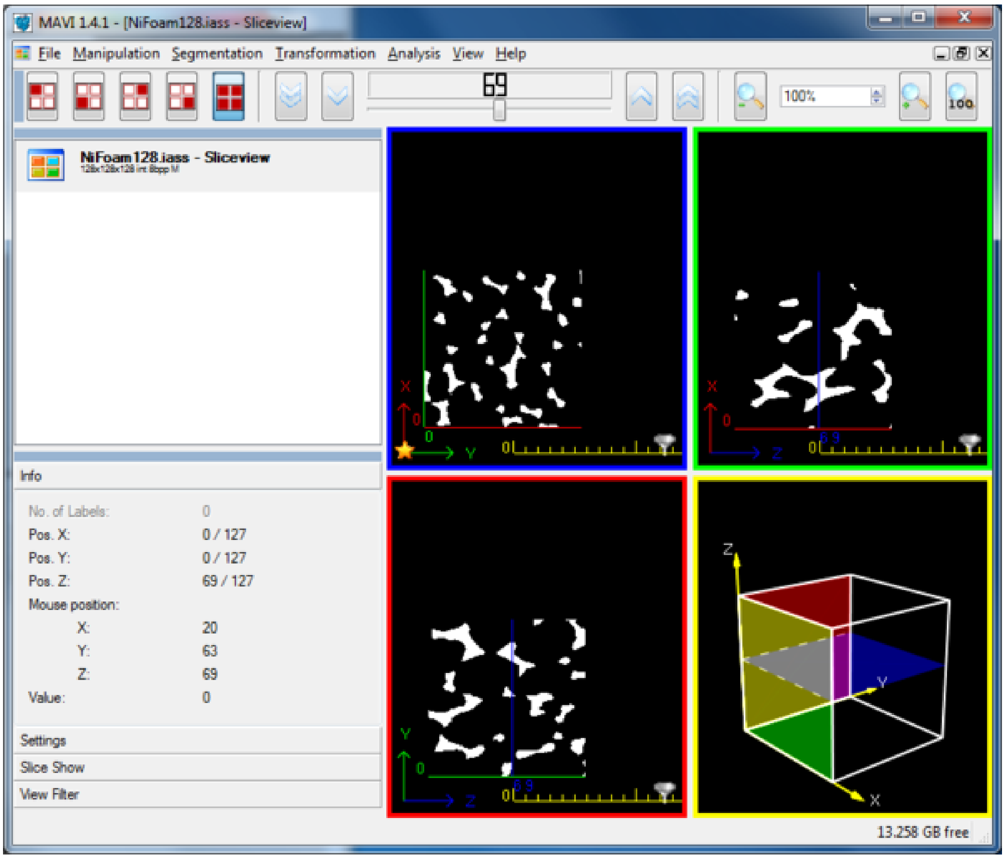
Bartlett Spectrum From Autocovariance The Bartlett Spectrum can be calculated directly from the Autocovariance. If the latter has already been computed Bartlett Spectrum From Autocovariance saves time and memory.

8.6 View

8.6.1 Slice View

The slice view is the standard view in **MAVI** . For 2D images, it just shows the image, the coordinate axes and a scale. For 3D images, it provides three view windows showing 2D slices orthogonal to the coordinate axes and a fourth window holding a cuboid illustrating the location of the slices within the 3D image.

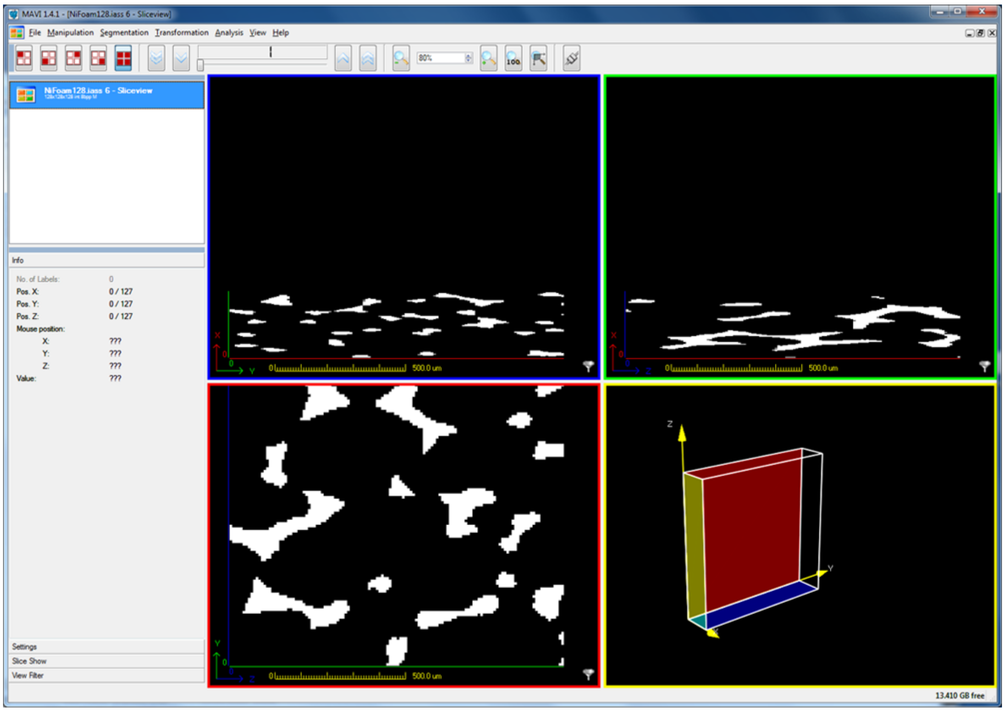
Exactly one of the three view windows is active at a time. Activation is indicated by a yellow star in the lower left corner of the view window, see Figure 43.


MAVI 1.4.1 - [NiFoam128.iass - Sliceview]
File Manipulation Segmentation Transformation Analysis View Help
69
100%
NiFoam128.iass - Sliceview
Info
Settings
Slice Show
View Filter
X
Y
Z
0
69
13.258 GB free

No. of Labels:	0
Pos. X:	0 / 127
Pos. Y:	0 / 127
Pos. Z:	69 / 127
Mouse position:	
X:	20
Y:	63
Z:	69
Value:	0

Figure 43 **MAVI** 's Slice View for a 3D image.

MAVI supports realistic representation of anisotropic grids, that is, if an image has different spacingsalong the coordinate axes, this will be reflected in the representation. Figure 44 illustrates this using the example from Figure 43 with different spacings.


MAVI 1.4.1 - [NiFoam128.iass 6 - Sliceview]
File Manipulation Segmentation Transformation Analysis View Help
80%
NiFoam128.iass 6 - Sliceview
Info
Settings
Slice Show
View Filter
500.0 um
500.0 um
500.0 um
13.410 GB free

No. of Labels: 0
Pos. X: 0 / 127
Pos. Y: 0 / 127
Pos. Z: 0 / 127
Mouse position:
X: ???
Y: ???
Z: ???
Value: ???

Figure 44 **MAVI** 's Slice View representation of a 3D image with anisotropic spacings. (Same image as used in Figure 43, spacing in x direction changed using Edit Header.)

Slice View Control Toolbar The toolbar offers functionality to control or modify the view windows.

You can zoom in to or out of the currently activated slice view by clicking on one of the buttons. The button resets the current view to a zoom level of 100% while the button zooms to the maximum keeping the current slice completely visible.

For 2D images, these are the only buttons present in the toolbar.

For 3D images, the first five buttons control what the slice viewer actually shows. Clicking one of the first four buttons to the left maximizes the corresponding view window. Clicking on the fifth button will restore the standard slice view layout with the four windows.

The buttons facilitate navigation through the image. The index of the slice visible in the currently activated slice view is displayed. You can scroll through the stack of slices either by moving the slider underneath the display or by clicking the up or down buttons. The buttons with one arrow move on by one slice and the buttons with two arrows move by ten slices.

The button offers the chance to disconnect the slice-slider from the update of the slice. It is relevant when working remotely, only. By default, it is set to “connect”. If it is turned to “disconnect” , the slider can be moved to the desired slice without updating the view. The view is not updated until the slider is disengaged.

Slice View Control Window The view control window contains several tabs containing further functions to control and modify the view windows. A tab can be activated simply by clicking on it.

The Info tab contains information on the current view window, such as location of slices and mouse position within the image, see Figure 45(a).

The Settings tab (Figure 45(b)) contains the functions

Reset window: undoes any translations, magnifications or changes of background color on the currently activated view window

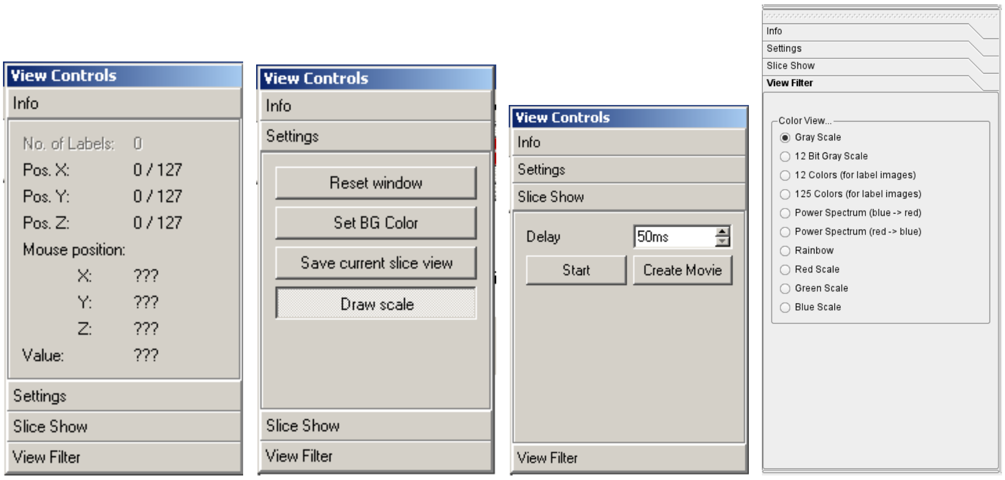
Set BG Color: offers a choice of background colors via a color selection dialog

Save current slice view: saves a 2D image of the current slice view in the active view window including the view filter if applicable but not the zoom. Available file types are BMP, PNG and XPM. An additional dialog offers the possibility to decide whether a scale should be drawn into the 2d image.

The Slice Show tab provides automatic scrolling through the slices of a 3D image in the current view window, see Figure 45(c). Adjust the scrolling speed in the Delay field. The animation will start in the currently activated view window and can be shifted to another one by simply clicking into the desired window. Using the Create Movie button, an animated walk through the image in the direction given by the active view window can be saved. On Windows operating systems, an avi file is created. On Linux operating systems, a dialog pops up allowing to save the slices as bmp or png files to a designated directory. A Windows compatible avi can then be created using the following expression on the command line:

```
mencoder "mf://*.bmp" -mf fps=12 -o output.avi -ovc lavc -lavcopts
vcodec=wmv1
```

The View Filter tab provides filters to modify the color representation of an image in the Slice View, Figure 45(d). Applying a view filter does not change the actual pixel values of an image but simply offers another way of looking at the data. See Figures 46 and 47 for examples of view filters and their effect. As can be seen in Figure 47, gray values may be difficult to distinguish visibly. In such a case, a false color representation adds clarity and information can be extracted more easily from the image on a visual basis. Table 18 lists the color tables available for viewing gray scale images. For COMPLEX images, the view filter tab provides the choice between real and imaginery part. For images of type GRAYF (float) the pixel values are mapped to an integer gray value range that can be displayed.



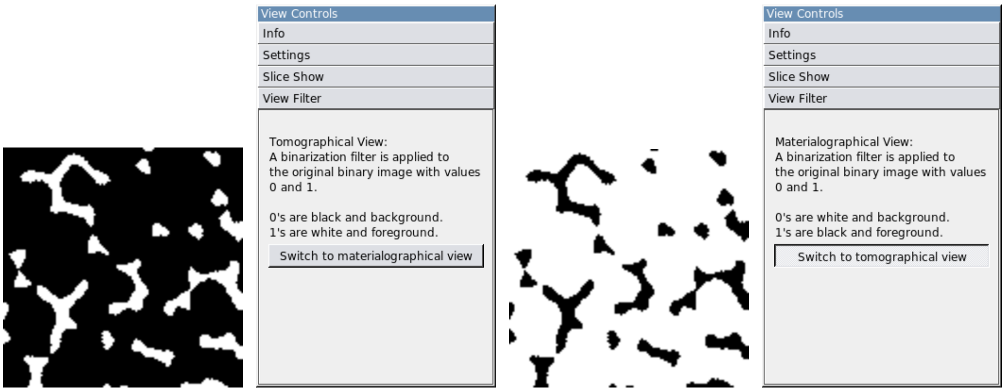
No. of Labels: 0
Pos. X: 0 / 127
Pos. Y: 0 / 127
Pos. Z: 0 / 127
Mouse position:
X: ???
Y: ???
Z: ???
Value: ???

Delay 50ms

Color View...
● Gray Scale
○ 12 Bit Gray Scale
○ 12 Colors (for label images)
○ 125 Colors (for label images)
○ Power Spectrum (blue -> red)
○ Power Spectrum (red -> blue)
○ Rainbow
○ Red Scale
○ Green Scale
○ Blue Scale

(a) Info tab (b) Settings tab (c) Slice Show tab (d) View Filter tab

Figure 45 The tabs in the View Control Window of the Slice View.


View Controls
Info
Settings
Slice Show
View Filter
Switch to materialographical view
Switch to tomographical view

Tomographical View:
A binarization filter is applied to the original binary image with values 0 and 1.

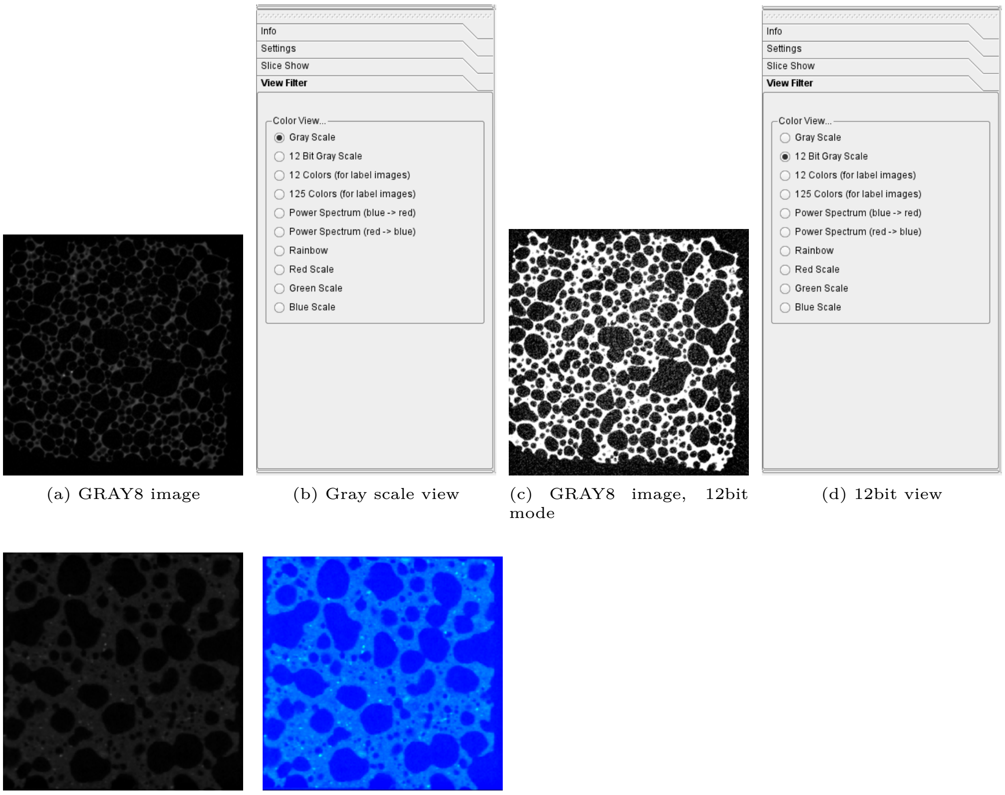
0's are black and background.
1's are white and foreground.

Materialographical View:
A binarization filter is applied to the original binary image with values 0 and 1.

0's are white and background.
1's are black and foreground.

(a) MONO image, default (b) Tomographic view (c) MONO image, materialographic (d) Materialographic view

Figure 46 View filters for MONO images in the Slice View.



Color View...

● Gray Scale
○ 12 Bit Gray Scale
○ 12 Colors (for label images)
○ 125 Colors (for label images)
○ Power Spectrum (blue -> red)
○ Power Spectrum (red -> blue)
○ Rainbow
○ Red Scale
○ Green Scale
○ Blue Scale

Color View...

○ Gray Scale
● 12 Bit Gray Scale
○ 12 Colors (for label images)
○ 125 Colors (for label images)
○ Power Spectrum (blue -> red)
○ Power Spectrum (red -> blue)
○ Rainbow
○ Red Scale
○ Green Scale
○ Blue Scale

(a) GRAY8 image (b) Gray scale view (c) GRAY8 image, 12bit mode (d) 12bit view

(e) GRAY8 image (f) GRAY8 image, blue-to-red mode

Figure 47 View filters for GRAY8 images in the Slice View.

8.6.2 Image Attributes View

In addition to the actual pixel gray values, an image in **MAVI** contains various attributes, which can be viewed by clicking on Image Attributes View. Moreover, **Analysis** results like Field Features or Object Features will be visible here. It is however not possible to copy data from the Image Attributes View window. Image attributes and features can instead be exportedin CSV format via **File**:**Export**.

8.6.3 Volume Rendering View

Volume Rendering View allows visualization of MONO and GRAY8 3D images. The Volume Rendering View windowopens in the central workspacewhen the function is chosen via the **View** menu. The corresponding list entry consisting of the view icon, the view label, and an additional description appears in the view navigator window.

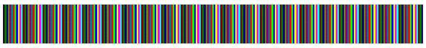
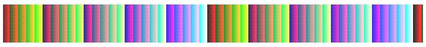
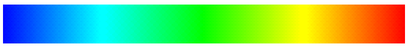
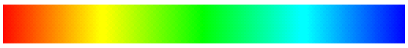
Color table	Description	Color range
Gray Scale	resets the display to gray scale (default)	
12 Bit Gray Scale	shows only the first 12 bits of the pixel gray values, for GRAY16 images where not the full 16bit gray value range is used (like in some images from computed tomography), has no effect on GRAY8 images	
12 Colors (for label images)	for a better display of label images, assigns one of 12 colors to each label, adjoining labels receive distinct colors	
125 Colors (for label images)	for a better display of label images, assigns one of 12 colors to each label, adjoining labels receive distinct colors	
Power Spectrum (blue - red)	power spectrum false color palette from blue to red	
Power Spectrum (red - blue)	power spectrum false color palette from blue to red	
Rainbow	rainbow false color palette	
Red Scale		
Green Scale		
Blue Scale		

Table 18 Color tables in the Slice View. Color tables are applicable to gray scale integer valued image only, that is to images of typeGRAY8, GRAY16, or GRAY32.

Volume Rendering View step-by-step

(1) Start the Volume Rendering View, the corresponding window opens, Figure 48(a)

(2) Check the position of your sample w.r.t. the view

(3) Change the background color to check whether imaging artefacts or noise might disturb the final rendering, Figure 48(b)

(4) Adapt the transfer functionto the gray value distribution in your image: Adjust the red, green, blue, and alpha channels, Figure 48(b)

(5) Save the thus derived color table in between.

(6) Use reset to return to the initial settings if needed.

(7) Adapt the blend scale, Figure 48(c)

(8) Adjust the lights, Figure 48(d), with the help of the light source positions if needed, Figure 48(e).

Volume Rendering View toolbar The Volume Rendering View window has its own menu. Its icons allow to move, rotate, zoom the visualized image. The view control window now contains now tools and functions relevant for the

Volume Rendering View.

On the top-left side of the widget, the 🪄 button starts animation of the image, rotating about the X axis. Another click on that button stops the animation. Moving the scrollwheel on top of the widget zooms in and out. On the right hand side of the zoomwheel, there is a group of three buttons which select the action to be performed when the user moves the mouse over the render area with a mouse button pressed: The ↻ button rotates the image around different axes, depending on the mouse button that is pressed. The different axes with their corresponding buttons are listed in the following Table 19. The ✥ button moves the image inside the render area. The 🔍 button zooms in and out of the image when the mouse is moved in vertical direction with the left mouse button pressed. This is the same operation as possible using the upper scrollwheel. On the bottom of the render widget, there are three scrollwheels which rotate the displayed image about the three different axes. The first one rotates the image view about the X axis, the second one about the Y axis and the third about the Z axis.

Button	Mouse movement	Rotation axis
Left button	vertical	Y axis
	horizontal	X axis
Right button	vertical	Z axis
	horizontal	X axis

Table 19 Axis and button relation in rotation mode.

Volume Rendering View control window On the lower left hand side of **MAVI** 's main window there is a toolbox, containing different advanced tools to modify the visual appearance of the rendered image.

Transfer function The first and most important entry in the toolbox is the transfer function editor controling how the gray values of your image are mapped onto the displayed colors red, green, and blue (RGB), and the alpha (A) channel. The RGB channels are separate functions mapping a gray value to the corresponding color channel. These three fundamental colors are mixed to generate the pixel's color. The alpha channel governs the transparency of the pixel, that is, how much of the pixel behind that pixel is visible.

Initially, the transfer function is the identity function resulting in a gray level visualization with decreasing transparency. Black source pixels are mapped to be fully transparent and white source pixels to appear totally opaque (non-transparent).

In the background of the function widget, the gray value histogram of the image is shown.

To modify the transfer function, first select one of the channels red, green, blue, and alpha below the function widget. Clicking on the function graph creates a new control point, which can be moved to modify the run of the curve. The rendered image is updated each time the mouse button is released. In case the update fails, then change the blend mode shortly and return to your previous blend mode. Existing control points for the selected channel can be moved any time. Right mouse click removes control points. The transfer function can be reset to identity using the “Reset” button. The “Save table” button saves a created transfer function in color table (.ct) format. The “Load table” button opens a file dialog to load previously saved transfer functions (color tables).

Blending On the bottom of the transfer function editor, a spinbox allows to choose a blend mode. Default is “normal” blending, adjusting transparency to alpha value.

Alternative blend modes are MIP (maximum intensity projection) and X-ray. The first one maps the pixel with the largest gray value (the brightest pixel) to the destination pixel, the latter accumulates the values of successive pixels, generating an X-ray like visualization. Both methods create very bright, highly oversaturated images. The saturation can be scaled by reducing the “Blend scale” with the slider on the bottom of the transfer function editor.

Isosurface editor The isosurface editor allows to define a threshold value to set pixels to totally transparent (A=0). You can choose if the values below, above, or exactlythe same as the threshold should be set to transparent by clicking on one of the three radio buttons below the function widget.

This is useful if the object in your image has a fuzzy environment which should not be displayed. It is also possible to set these black environment pixels to transparent by using the transfer function editor and modifying the alpha channel. But this is the faster method.

Clipping editor The clipping editor enables a clipping plane cutting the image and providing a view of the interior of your image, see Figure 50(a). With the three scrollwheels it is possible to rotate the plane around all three axes. If the “View aligned clipping plane” button is checked, the clipping plane always faces towards the view direction.

To change the clipping direction, you can either

(1) rotate the object using the ⟳ button in the toolbar(Figure 50(c)) or
(2) deactivate the “View aligned clipping plane” button and choose the plane’s direction using the X, Y, Z axis wheels.

Note that activating the “View aligned clipping plane” button again changes back, the chosen plane direction is reset.

To change the clipping plane’s position w. r. t. the image, move the cutting plane through the image with the position slider at the bottom (Figure 50(d) and (e)).

Note that this position slider can be used to generate frames for a movie showing the successive cutting of the structure, too.

Light editor Enable lighting to emphasize the 3D effect. Color nuances can be enhanced by modification of the light color. Showing the light position can help fine-tuning. The threshold controls the degree to which the light is absorbed by the structure within the image.

View parameter editor Determine the position of the object to be rendered setting the “Rotate Local” and “Rotate Global” parameters, see Figure 51(a). Useful to recover exactly the same view angle when renderings of two different images shall be comparable.

The “Render Window”parameters control the size of the 2D image holding the rendering result if it is saved, see Figure 51(b).

Preferences editor You can customize the rendering process and enable/disable debugging information in the preferences editor:

Draw bounding box: enables/disables the display of the yellow bounding box and the green direction indicator

Draw scale: enables/disables the display of the along the axes of the bounding box

Draw red-green: enables a red-green pseudo 3d rendering mode which must be viewed with red-green goggles. Note that the rendering performance will be lower in this mode, because the image has to be drawn twice in one render pass.

Draw proxy geometry: disables texture application and draws the underlying geometry used by the render algorithm

Enable progressive refinement: temporarily Shrinks the image when rotating

Set Background Color: opens a color editor window to choose a new background color

Set Boundingbox Color: opens a color editor window to choose a new color for the bounding box

Set Label Color: opens a color editor window to choose a new color for the axes labels 0, x, y, z, and the scale limits

At the bottom of the editor widget, there is another tab, which shows a list of all the OpenGL extensions your graphic card supports. This is useful for debugging purposes but there is nothing to adjust here. The best rendering procedure for your graphic card is chosen automatically.

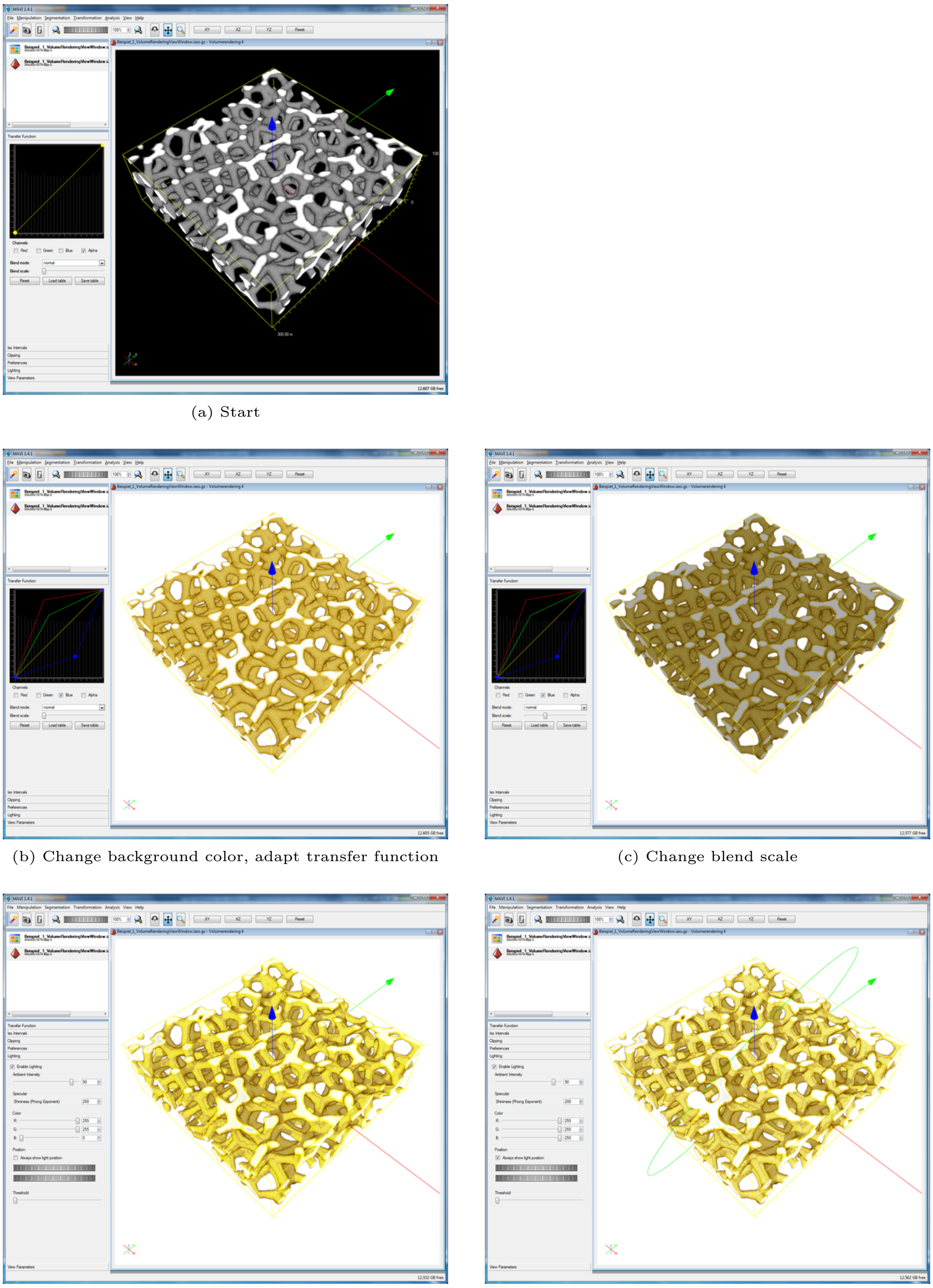
8.6.4 Volume Rendering View - Selection

This function offers a choice of render engines to be employed in the rendering process. Before opening the Volume Rendering View, a dialog showing the render engines available for your graphics card is opened.

The 3D_T_CG render engine only uses memory of the same size as the image data. It is the best choice if light effects are nor required.

For GRAY8 and label images, Pre-Classification or Post-Classification can be chosen. Pre-Classification means that the RGBA value of a pixel in the visualization is obtained by first applying the transfer function, which assigns RGBA color values to the image pixels, and then interpolating the resulting RGBA values of the surrounding image pixels. With Post-Classification, the gray value of the visualization pixel is interpolated from the gray values of the surrounding image pixels and then the transfer function is applied to obtain the corresponding RGBA value.

Usually, post-classification creates more realistic, smooth renderings. However, for label images, post-classification causes artifacts along the borders of the labeled objects due to the intermediate colors obtained by interpolation, see Figure 52. Thus pre-classification should be chosen in cases where crisp jumps between colors are desired.



(a) Start

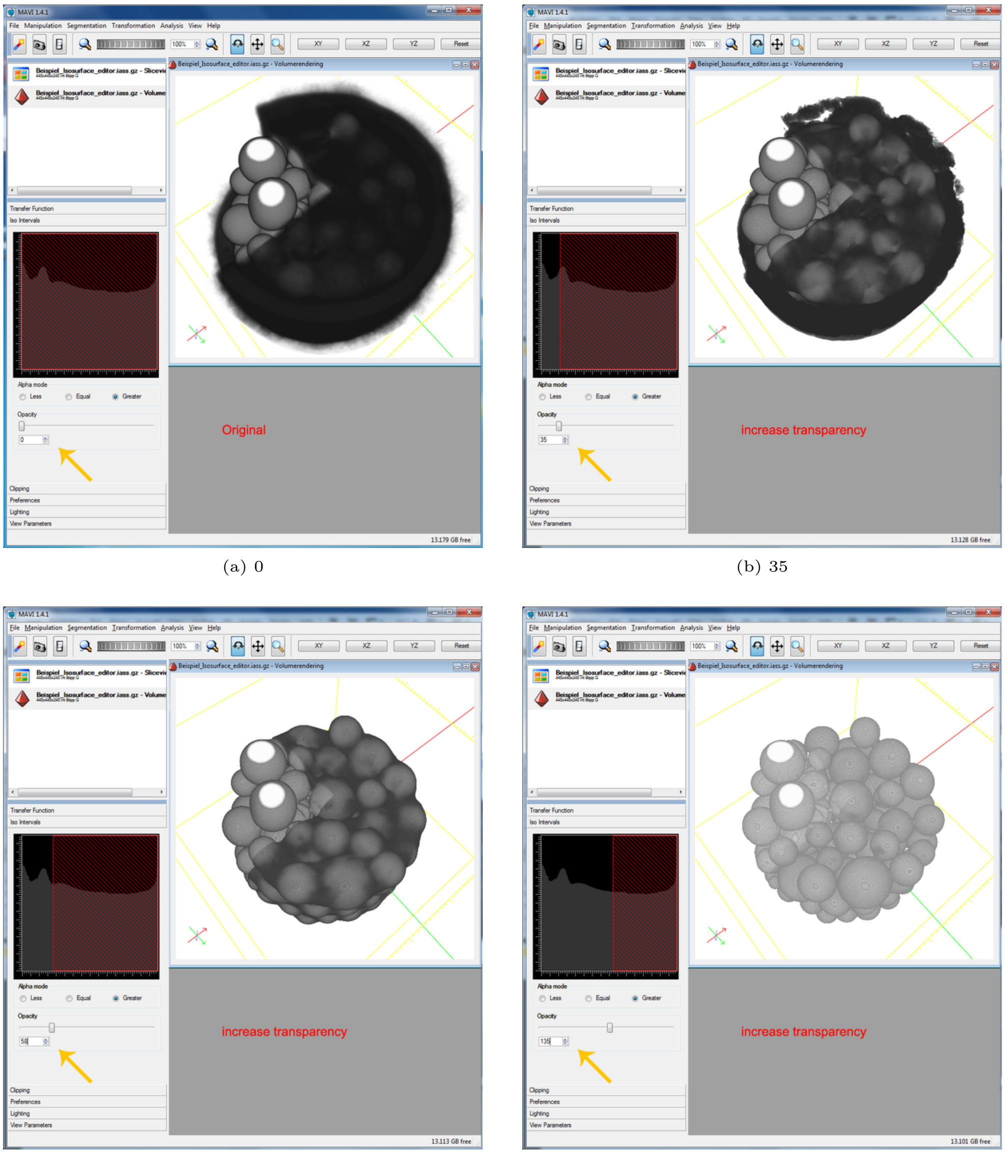
(b) Change background color, adapt transfer function

(c) Change blend scale

(d) Add light

(e) Fine tune light

Figure 48 Step-by-step volume rendering.



(a) 0

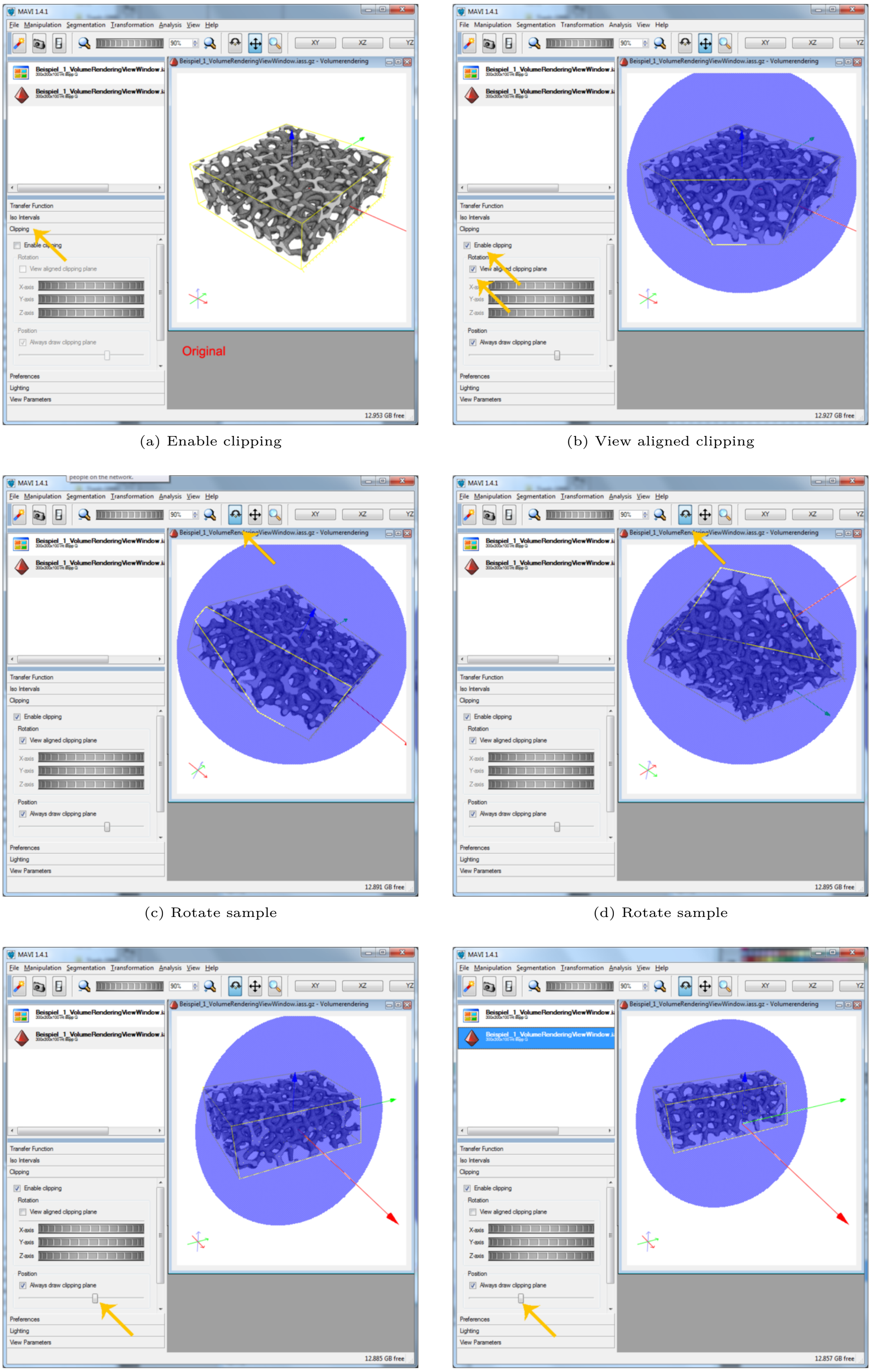
(b) 35

(c) 58

(d) 135

Figure 49 Example effect of isosurface threshold. Pixels with gray value below the threshold are completely transparent.

Volume Rendering View - Selection


MAVI 1.4.1
File Manipulation Segmentation Transformation Analysis View Help
XY XZ YZ
Beispiel_1_VolumeRenderingViewWindow.iass.gz - Volumerendering
Transfer Function
Iso Intervals
Clipping
Enable clipping
Rotation
View aligned clipping plane
X-axis
Y-axis
Z-axis
Position
Always draw clipping plane
Preferences
Lighting
View Parameters
Original

(a) Enable clipping

(b) View aligned clipping

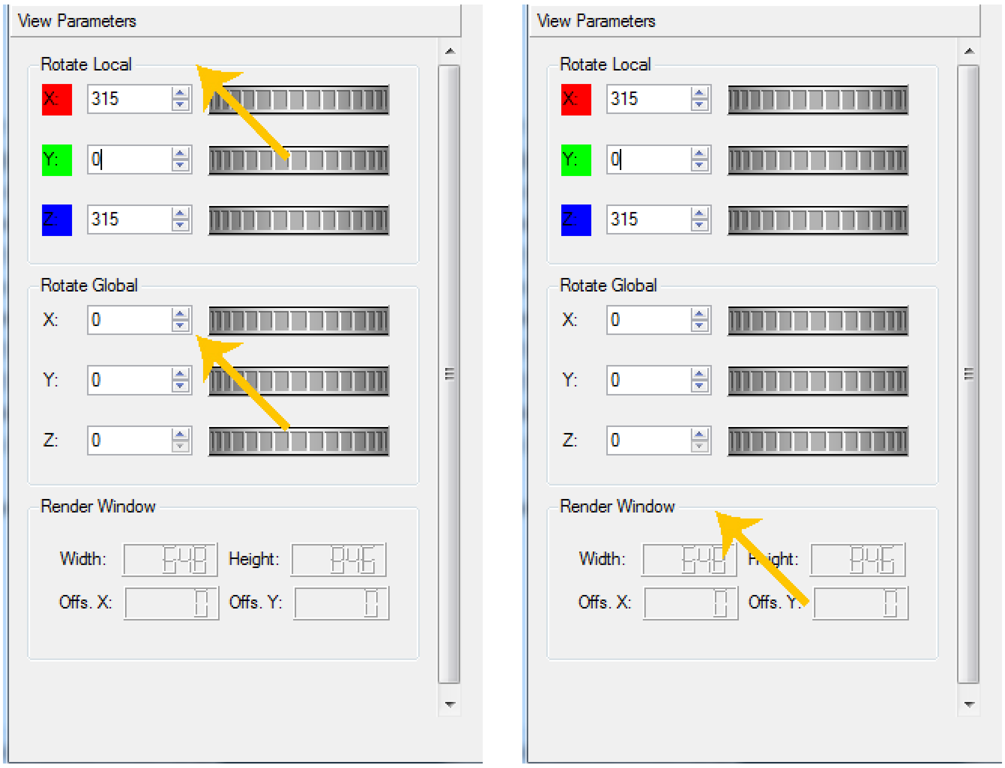
(c) Rotate sample

(d) Rotate sample

(e) Move plane

(f) Move plane

Figure 50 Volume Rendering View: enabling and positioning of clipping plane.



View Parameters

Rotate Local

X: 315

Y: 0

Z: 315

Rotate Global

X: 0

Y: 0

Z: 0

Render Window

Width: 648 Height: 846

Offs. X: 0 Offs. Y: 0

View Parameters

Rotate Local

X: 315

Y: 0

Z: 315

Rotate Global

X: 0

Y: 0

Z: 0

Render Window

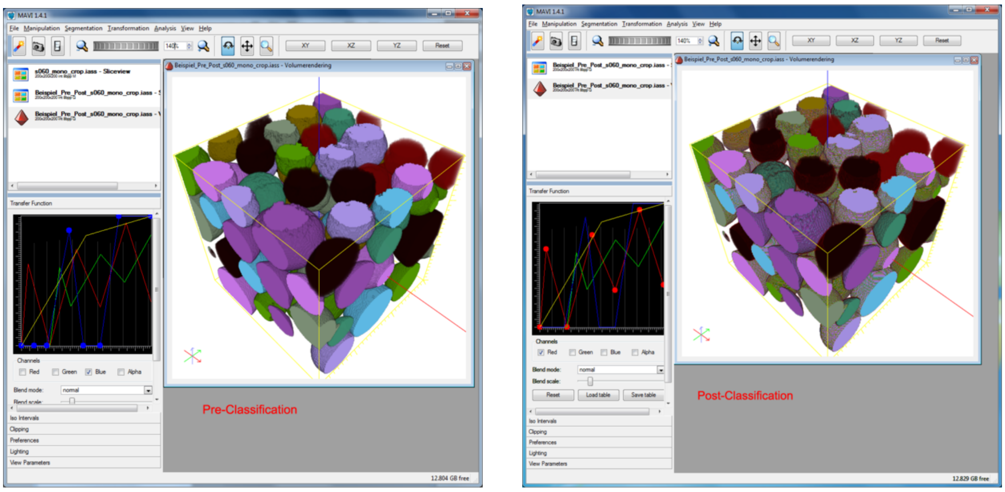
Width: 648 Height: 846

Offs. X: 0 Offs. Y: 0

(a) Rotate parameters

(b) Render window

Figure 51 Volume Rendering View view parameter settings.



(a) Pre-classification

(b) Post-classification

Figure 52 Volume Rendering View, effect of choosing pre- or post-classification when rendering a label image. Post-classification generates unwanted grayish color artifacts at the surfaces of the spheres.

8.7 Help

8.7.1 System Info

Pops up a window with essential information on

- operating system (OS)
- processor(s) (CPU)
- memory (RAM)
- graphics card (GPU).

Please provide this information e. g. via a screen shot when asking for e-mail support at mavi@itwm.fraunhofer.de.

8.7.2 Help Assistant

The Help Assistant provides the complete **MAVI** user documentation handbook (this document) online. Clicking the respective button opens a separate window enabling navigation via content tree and index as well as a key word search.

8.7.3 About

Pops up a window with essential information on the version of **MAVI** you're running. Please provide the full information in the "Version Info" tab e. g. via a screen shot when asking for e-mail support at mavi@itwm.fraunhofer.de.

9 Key board short cuts

Keys	action
CTRL + O	Open image
CTRL + S	Save image
CTRL + Q	Quit **MAVI**
CTRL + W	Close view
CTRL + 1	Focus window 1
CTRL + 2	Focus window 2
CTRL + 3	Focus window 3
CTRL + 4	Focus window 4
CTRL + Mouse wheel	Scroll up: Zoom in Scroll down: Zoom out
CTRL + +	Zoom in
CTRL + -	Zoom out
Mouse wheel	Scroll up: Next slide of volume image Scroll down: Previous slide of volume image
CTRL + 0	In 4-square view-mode: Reset to window 1 If focused to one slice: Reset to 4-square-view
CTRL + F4	Minimize **MAVI**
CTRL + Shift + W	Close view
CTRL + “–>” or “–>”	Next slice
CTRL + “<–” or “<–”	Previous slice
CTRL + Shift + O	Open image
dot	zoom to fit
comma	zoom reset (100%)
CTRL+TAB	switch to next view (in order of having been active)
Shift+CTRL+TAB	switch to prev view (in order of having been active)

Table 20 Key board short cuts to **MAVI** functions.

10 Frequently Asked Questions

Where do I report problems with MAVI ? Please report all bugs, problems, and feature wishes to mavi@itwm.fraunhofer.de. Please always attach the Version Information from the plugin-tab in the **Help**:About dialog, and the logfile mavi.log in \$HOME/.ITWM/ (linux) or ApplicationData\ITWM (windows).

The slice view is black, but there should be objects/structure. If your image is a labeled one, obtained by labeling or the watershed transformation, try setting the color filter manually. If your image is of type GRAY16, try using the 12 Bit Gray Scale view filter. More general, **Manipulation**:Equalize the gray value histogram.

I tested the FieldFeatures/ObjectFeatures using a cube parallel to the coordinate directions. Why is the surface area measured by MAVI too small? This is due to **MAVI** minimizing the mean error in general position. See the Remark at the end of Section 8.5.8 for details.

I measured both the FieldFeatures and the ObjectFeatures for my MONO image. Why do the results for the surface area and the integral of mean curvature differ? This is due to the different treatment of edge effects. See the Remark at the end of Section 8.5.8 for details.

When I'm connected to a machine running SUSE Linux Enterprise 11 using ssh, MAVI crashes with a segmentation fault. This is an OpenGL library/driver problem on machines running SUSE Linux Enterprise 11 with servicepack 0 and servicepack 1. It is fixed with servicepack 2, so please update the remote machine to servicepack 2. An easy workaround if you can not update: set LANG to en_ENutf8.

11 Credits for Third Party Libraries

MAVI uses a number of third party libraries, each released under its individual license terms, see below. The copyright of these libraries remains with their original authors, see below. Note, however, that this does not affect the license terms of **MAVI** itself in any way. If you would like to obtain the source code of any of the open source libraries used within this software, please contact Fraunhofer ITWM either by mail or e-mail. Please state specifically for which of the open source third party libraries below you require the source code and where it should be sent to.

Fraunhofer ITWM
Katja Schladitz
Fraunhofer Platz 1
67663 Kaiserslautern
Germany
mavi@itwm.fraunhofer.de

The following open source third party libraries are used, and their respective authors are acknowledged for their work:

- Cg Toolkit under the License For Customer Use of NVIDIA Software, `http://developer.download.nvidia.com/cg/Cg_3.0/license.pdf`.
- FreeImage under the FreeImage Public License v1, `http://freeimage.sourceforge.net/`.
- FreeType under The FreeType Project License, `http://git.savannah.gnu.org/cgit/freetype/freetype2.git/tree/docs/FTL.TXT`.
- libpng under the libpng license, `http://www.libpng.org/pub/png/src/libpng-LICENSE.txt`.
- libusb under the GNU Lesser General Public License 2.1, `http://www.gnu.org/licenses/lgpl-2.1.html`.
- OpenGL Extension Wrangler Library (GLEW) under a modified BSD license, `http://glew.sourceforge.net/glew.txt`.
- zlib under the zlib license, `http://zlib.net/`.

For the original terms and conditions of each of these libraries, please read the contents of the file

```
  %ITWMDIR%\share\license-thirdparty.txt
```

.

12 Optional modules

The following modules are not part of **MAVI** under the standard full license. They can be added on user's demand. In that case, they are integrated completely within **MAVI** 's framework.

12.1 Particle Features

This module adds analysis options for image objects exceeding the range of functionality of the Object Features. As the latter, the Particle Features work on label images as generated e. g. by Labeling, Particle Separation or Cell Reconstruction. Some of the particle features can not be measured reliably for small particles. Thus the Particle Features actually derived depend on the particle's typoe according to the following classification:

small: particles constituted of 64 pixels or less,
medium: particles constituted of more than 64 pixels but less than 400 pixels,
large: particles constituted of more than 400 pixels.
reference: the biggest particle (if set), only volume and mean gray value are computed for this object

The threshold values can be adapted.

The following characteristics are provided by the Particle Features:

- mean gray value – average of the gray values of the pixels of which the object consists
- length, width, and thickness derived from the sorted edge lengths of the minimal volume bounding box (MVBB, cuboid)
- maximal Feret diameter
- maximal local thickness
- elongation and elongation index.

The results can be exported as CSV.

The maximal Feret diameter is calculated for 'medium' and 'large' particles. It is the largest (Euclidean) distance between two points in the particle (in pixels and micrometers). It is larger or equal to the length defined by the MVBB, being equal in the case of a sphere for example, but larger for a cube or a parallelepiped, where it corresponds to the diagonal.

The elongation is calculated for 'medium' and 'large' particles. It is the geodesic length of the particle, that is the maximal geodesic path connecting two points within the particle. It is an estimate of the length of the medial

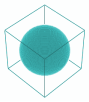
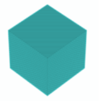
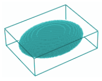
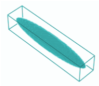
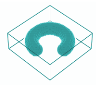
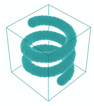
						
volume [px]	179 579	91 125	49 977	24 941	22 773	216 862
Euler number	1	1	1	1	1	1
length [px]	70.00	45.00	79.60	120.00	60.92	116.78
width [px]	70.00	45.00	59.70	19.40	53.25	116.78
thickness [px]	70.00	45.00	20.00	19.40	14.61	96.23
maximal Feret diameter [px]	70.00	76.21	80.00	120.00	61.38	150.24
shape factor f_1	0.992	0.827	0.651	0.583	0.529	0.236
shape factor f_2	0.989	0.733	0.511	0.188	0.148	0.007
shape factor f_3	1.000	0.923	0.851	0.471	0.428	0.094
elongation [px]	72.07	78.41	79.38	120.00	98.59	639.12
elongation index	1.09	2.77	5.24	36.28	21.99	630.29
maximal local thickness [px]	70.03	45.00	20.10	20.10	15.48	20.00

Table 21 Selected features for some 'large' reference particles.

axis. For fibrous particles, it can be interpreted as the real length of the fiber.

The elongation index can be used to compare particles: $IG_g = \frac{\pi L_g^3}{6V}$, where L is the elongation. It is equal to 1 for spheres and gets larger, the more elongated a particle is. Values for some sample particles can be found in Table 21.

The maximal local thickness, also known as inner diameter, is calculated for 'medium' and 'large' particles. It is defined as the diameter of the largest ball completely contained in the particle, yielding the maximal local structure thickness. This measure estimates the real thickness of a fiber or a chip. Moreover, for particles without holes, it yields the size of the "core".

Combined with the Object Features, the Particle Features enable easy classification of particles into the classes grain, chip, and fiber. See [23] for details. Examples of Object Features and Particle Features for a choice of reference objects are given in Table 21.

12.2 Point Field Statistics

On the basis of the Object Features, this module offers additional functions for analyzing the spatial arrangement of the found objects. To this end summary statistics for random point fields from spatial statistics are computed for the object centers. Suppose the object center's to form a realization of the random point field Φ with distribution P and expected point density λ.

The statistics provided in the module are

- Ripley's K-function $K(r)$ (mean number of r-close pairs of points)
$$K(r) = \frac{1}{\lambda \text{Volume}(B)} \text{Expectation} \sum_{x \in \Phi \cap B} \Phi \cap \text{Ball}(x, r) \setminus \{x\}$$
- L-function $L(r)$ (normalized version of K)
$$L(r) = \sqrt[3]{\frac{K(r)}{\text{Volume}(\text{Ball}(o, r))}}$$
- empty space function, $F(r)$ (distribution function of the distance of an arbitrary point to the next object center)
$$F(r) = P(\Phi \cap \text{Ball}(o, r) \setminus \{0\} \neq \emptyset)$$
- nearest neighbor distance distribution function , G-function $G(r)$ (distribution function of the distance of an object center to the next object center)
$$G(r) = P(\Phi \cap \text{Ball}(o, r) \setminus \{0\} \neq \emptyset | 0 \in \Phi)$$
- J-function $J(r) = (1 - G(r))/(1 - F(r))$
- paircorrelation function, $g(r)$ (roughly the probability density of K)

The user has to specify two parameters:
range the maximal distance to which the respective statistic is estimated and
step width the discretization fineness of the distance range.
The pair correlation function is a density and hence has to be estimated using a kernel, in the present case an Epanečnikov kernel whose width is user defined, too.

Pushing the suggestion button fills
range with half of the length of the shortest edge of the observation window,
step width with the smallest lattice spacing$s$, and
kernel width with the values recommended by [4]:

$$\begin{aligned} &\max\{s, 0.05\sqrt[3]{V/\#\text{points}}\} && \text{and} \\ &\max\{s, 0.1\sqrt{A/\#\text{points}}\} && \text{for 3D and 2D, respectively.} \end{aligned}$$

Here V and A denote the volume and the area of the observation window, respectively, that is the total volume and the total area as given by the Field Features.

Results can be exported as CSV.

The points of a Poisson point field are independently identically uniform distributed without interaction among each other. Their number is Poisson distributed. The Poisson point field often serves to compare observed point patterns to it. If there is interaction between the points, one usually describes it as clustering (in case of attraction) or regularity (in case of repulsion).

For 2D Poisson point fields it holds that

$$\begin{aligned} K(r) &= \pi \mathrm{r}^2 \\ L(r) &= \mathrm{r} \\ F(r) &= 1-\exp(-\lambda\pi\mathrm{r}^2) \\ G(r) &= F(r) \quad \text{and thus} \quad J(r)=1 \\ g(r) &= 1 \end{aligned}$$

For 3D Poisson point fields it holds that

$$\begin{aligned} K(r) &= \frac{4}{3}\pi \mathrm{r}^3 \\ L(r) &= \mathrm{r} \\ F(r) &= 1-\exp(-\frac{4}{3}\lambda\pi\mathrm{r}^3) \\ G(r) &= F(r) \quad \text{and thus} \quad J(r)=1 \\ g(r) &= 1 \end{aligned}$$

In a Poisson point field, the points are independent, thus counting neighbors of a point of the field is just the same as counting neighbors of an arbitrary point. Hence $G = F$ and $J = 1$.

In a clustered point field, a point of the field has more close neighbors than an arbitrary one. Thus $G > F$ and $J < 1$.

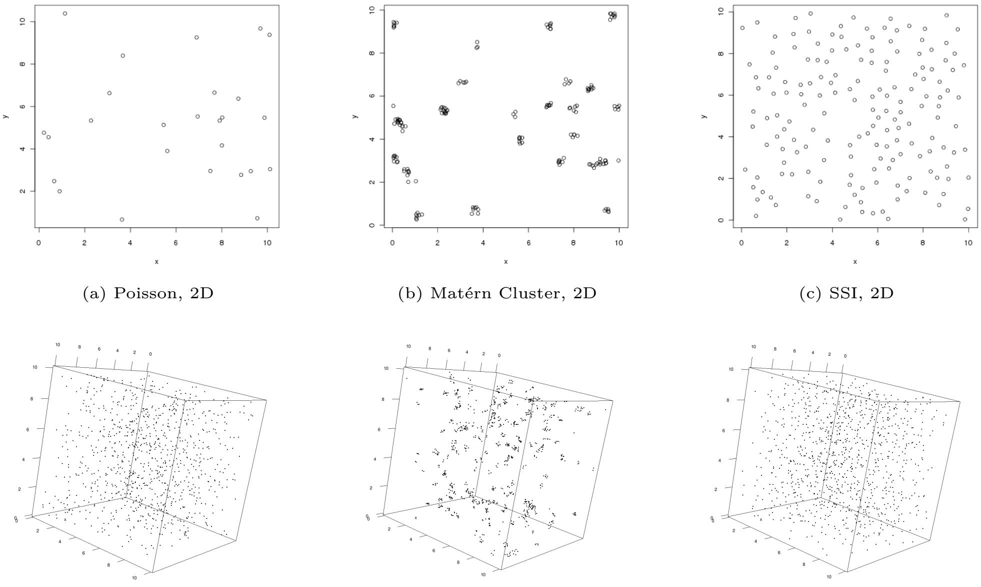
In a regular point field, a point of the field has less close neighbors than an arbitrary one. Thus $G < F$ and $J > 1$.

Note however, that a point field can be both clustered and regular. (Think of clusters centered at the vertices of a grid or small, grid-like structures forming larger clusters.) Therefore, the range or distance r has to be considered, too. Table 22 below summarizes the rules of thumb:

Point field type	F, G	J	g
Poisson	$F = G$	$J = 1$	$g = 1$
Regular	$F > G$	$J > 1$	$g < 1$
Clustered	$F < G$	$J < 1$	$g > 1$

Table 22 Point field statistics indicating clustering, regularity, or independence.

Figure 53 shows realizations Poisson point fields and typical examples of a clustered and a regular point field. Figure 54 provides the summary statistics for these realizations.



(a) Poisson, 2D

(b) Matérn Cluster, 2D

(c) SSI, 2D

(d) Poisson, 3D

(e) Matérn Cluster, 3D

(f) SSI, 3D

Figure 53 Realizations of random point fields in 2D and 3D. Matérn Cluster is clustered, SSI is regular in the sense that the points observe a minimal distance.

See [4] for more mathematical background.

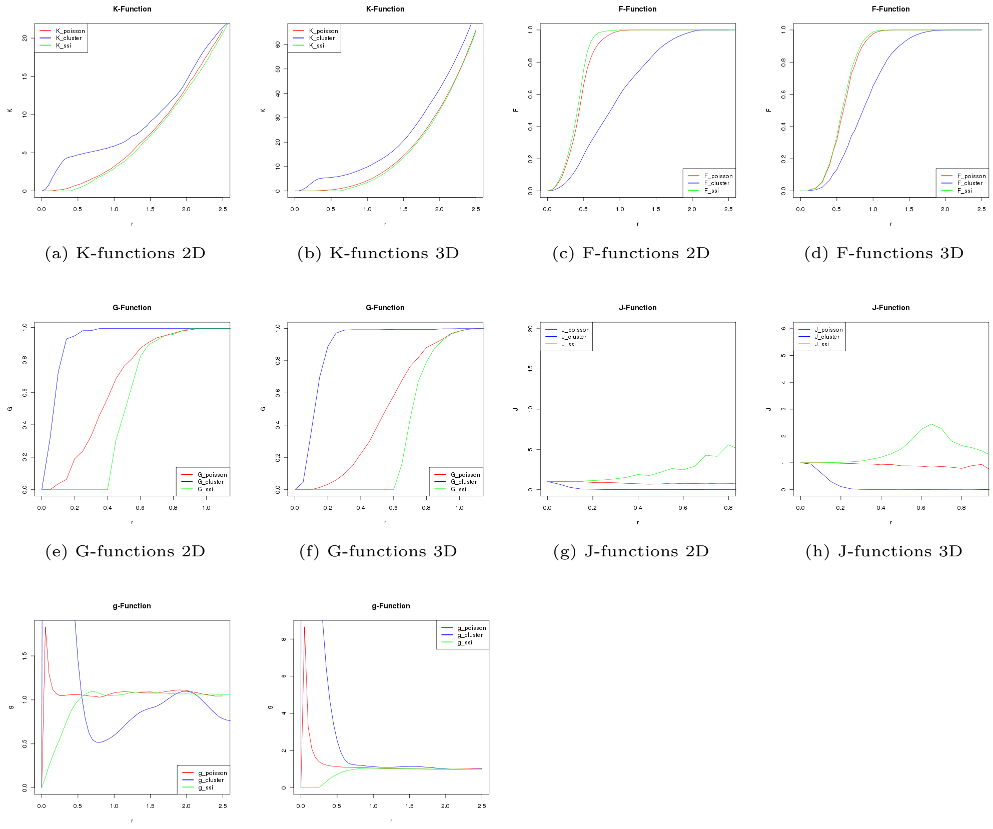
12.3 Mesh Export

For segmentedMONO images, this module offers surface meshing, mesh simplification, and export.

Meshing generates a representation of the image data by a triangulation of the surface of the foreground. The original mesh is based on the adjacency systemand subdivides each surface pixel into several triangles.

This initial mesh has to be simplified. Mesh Export unites nearly co-planar triangles to a user defined proportion of the initial triangle number while preserving the topology of the foreground.

Finally, the mesh is exported as STL file (binary, can be compressed as GZ).



(a) K-functions 2D

(b) K-functions 3D

(c) F-functions 2D

(d) F-functions 3D

(e) G-functions 2D

(f) G-functions 3D

(g) J-functions 2D

(h) J-functions 3D

(i) g-functions 2D

(j) g-functions 3D

Figure 54 Summary statistics for the point field realizations shown in Figure 53.

12.4 Mesh Vis

This function adds interactive visualization to the functionality of Mesh Export. It can only be used together with Mesh Export and offers:

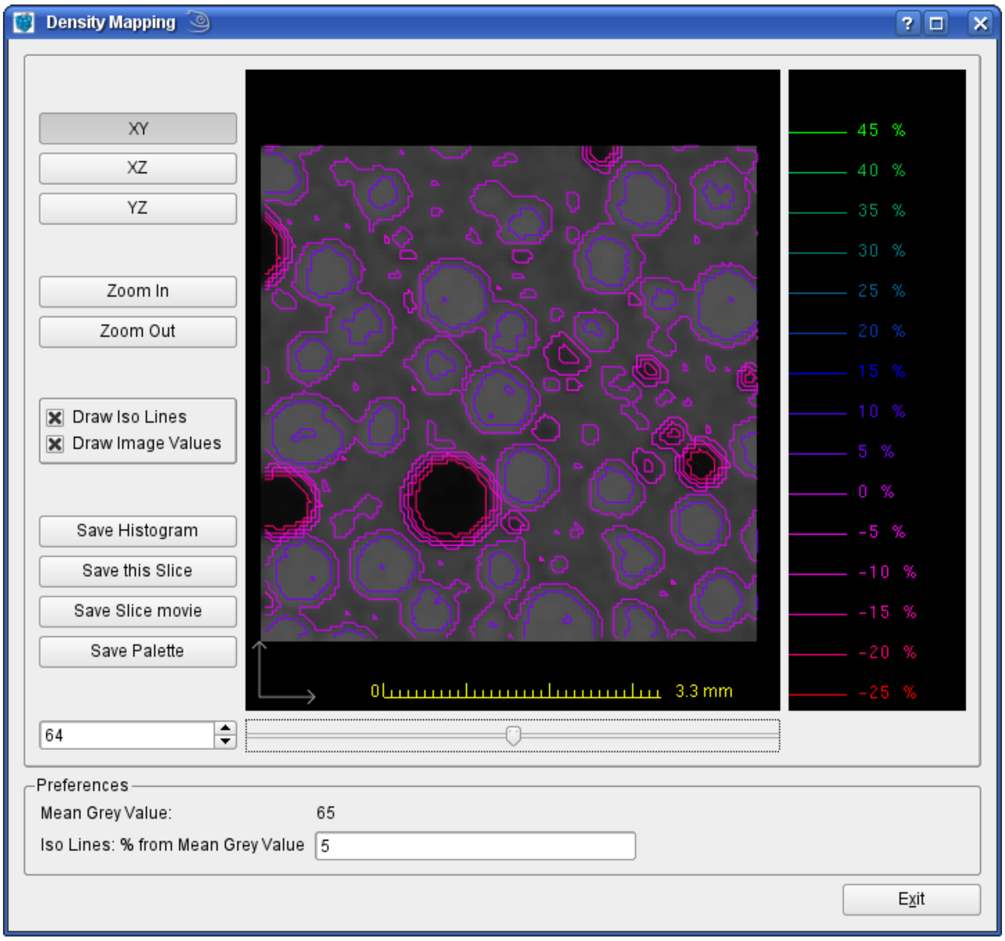
- Import a surface mesh in STL format (binary or ASCII, can be compressed as GZ)
- Visualize as surface mesh, edge mesh, or point cloud
- Rotate, Zoom, Move
- Rotate automatically, adjustable tracking shot
- Simplify the mesh in individual steps, to a user defined number of triangles, or to a user defined degree of co-planarity
- Smooth the mesh
- Export the current view (screenshot)
- Export automatic rotation or tracking shot as key frames

12.5 Gray Value Mapping

In computed tomography images high gray values correspond to a high (physical) density of the material. The Gray Value Mapping helps to check the homogeneity of a material: density variations like inclusions or condensations can be visualized.

The plugin offers a representation with iso lines, where the iso levels correspond to the deviation from the mean density. It is possible to specify the size of the filter mask and a mask image (to exclude background or certain inclusions). The view and the palette can be saved.

The filter mask should be chosen slightly bigger than the diameter of negligible structures in the image, such as small inclusions, that you do not want to visualize.



Preferences

Mean Grey Value: 65

Iso Lines: % from Mean Grey Value 5

Figure 55 Gray Value Mapping window.

References

[1] J. Canny. A computational approach to edge detection. IEEE Trans. Pattern Analysis and Machine Intelligence, 6(PAMI-8):679–698, 1986.

[2] M. Frigo and S. G. Johnson. FFTW 2.1.3 (the fastest Fourier transform of the West). http://www.fftw.org/, 1998.

[3] M. Frigo and S. G. Johnson. FFTW: An adaptive software architecture for the fft. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, volume 3, pages 1381–1384, May 1998.

[4] J. Illian, A. Penttinen, H. Stoyan, and D. Stoyan. Statistical Analysis and Modelling of Spatial Point Patterns. Statistics in Practice. Wiley, Chichester, 2008.

[5] K. Koch, J. Ohser, and K. Schladitz. Spectral theory for random closed sets and estimating the covariance via frequency space. Adv. Appl. Prob., 35:603–613, 2003.

[6] C. Lang, J. Ohser, and R. Hilfer. On the analysis of spatial binary images. Journal of Microscopy, 203:303–313, 2001.

[7] C. R. Maurer and V. Raghavan. A linear time algorithm for computing exact Euclidean distance transforms of binary images in arbitrary dimensions. IEEE Trans. Pattern Analysis and Machine Intelligence, 25(2):265–270, 2003.

[8] W. Niblack. An introduction to digital image processing. Prentice-Hall, 1986.

[9] M. Nöthe, K. Pischang, P. Ponižil, B. Kieback, and J. Ohser. Study of particle rearrangements during sintering processes by microfocus computer tomography (μct). In Proc. World Congress PM2004 Powder Metallurgy, Vienna, October 17th-21st 2004.

[10] J. Ohser, W. Nagel, and K. Schladitz. Miles formulae for Boolean models observed on lattices. Image Anal. Stereol., 28(2):77–92, 2009.

[11] J. Ohser, C. Redenbach, and K. Schladitz. Mesh free estimation of the structure model index. Image analysis and stereology, 28(3):179–186, 2009.

[12] J. Ohser and K. Schladitz. 3d Images of Materials Structures – Processing and Analysis. Wiley VCH, Weinheim, 2009.

[13] S. Osher and L. Rudin. Feature-oriented image enhancement using shock filters. SIAM J. Numer. Anal., 7:919–940, 1990.

[14] A. Rack, L. Helfen, T. Baumbach, S. Kirste, J. Banhart, K. Schladitz, and J. Ohser. Analysis of spatial cross-correlations in multi-constituent volume data. Journal of Microscopy, 232(2):282–292, 2008.

[15] C. Redenbach, A. Rack, K. Schladitz, O. Wirjadi, and M. Godehardt. Beyond imaging: on the quantitative analysis of tomographic volume data. Int J Mater Res, 2:217–227, 2012.

[16] J. Sauvola and M. Pietikäinen. Adaptive document image binarization. Pattern Recognition, 33:225–236, 2000.

[17] K. Schladitz, C. Redenbach, T. Sych, and M. Godehardt. Model based estimation of geometric characteristics of open foams. Methodology and Computing in Applied Probability, 2010. Accepted.

[18] K. Schladitz, C. Redenbach, T. Sych, and M. Godehardt. Model based estimation of geometric characteristics of open foams. Methodology and Computing in Applied Probability, pages 1–22, 2011.

[19] P. Soille. Morphological Image Analysis. Springer, Berlin, Heidelberg, New York, 1999.

[20] D. Stoyan, W. S. Kendall, and J. Mecke. Stochastic Geometry and Its Applications. Wiley, Chichester, 2nd edition, 1995.

[21] O. D. Trier and A. K. Jain. Goal-directed evaluation of binarization methods. IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(12):1191–1201, 1995.

[22] Y. F. Tsao and K. S. Fu. A parallel thinning algorithm for 3-d pictures. CVGIP, 17:315–331, 1981.

[23] I. Vecchio, K. Schladitz, G. Godehardt, and M. Heneka. 3d geometric characterization of particles applied to technical cleanliness. Image Analysis and Stereology, 31(3):163–174, 2012.

[24] L. Vincent and P. Soille. Watersheds in digital spaces: An efficient algorithm based on immersion simulation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(6):583–598, 1991.

[25] P. Zamperoni. Methoden der digitalen Bildsignalverarbeitung. Vieweg, Braunschweig, 1991.

Index