User Manual

Version 2.2
### Contents

Introduction ............................................................................................................................................................ 6

Overview: Some Key Features of daime ........................................................................................................... 6

The Main Window .............................................................................................................................................. 7

Image Comments and Action Log .................................................................................................................... 7

Standard Workflows ......................................................................................................................................... 8

How to Get Help ............................................................................................................................................... 8

General Preferences .......................................................................................................................................... 9

Loading, Saving, and Converting Images ....................................................................................................... 10

Import Images .................................................................................................................................................. 10

  Import a Single 2D Image .......................................................................................................................... 11

  Import a 2D Image Batch or a 3D z-Stack ................................................................................................ 11

Import Color Images ...................................................................................................................................... 12

Pixel / Voxel Resolution and μm Scale ........................................................................................................... 12

  Reduce Pixel (Voxel) Resolution .............................................................................................................. 13

STK Image Files: The Native File Format of daime .................................................................................. 13

Export Images ................................................................................................................................................... 14

Image Types and Conversion .......................................................................................................................... 14

Image Processing and Enhancement ............................................................................................................ 15

Clone and Copy Images .................................................................................................................................. 15

Invert Images ................................................................................................................................................... 15

Preview Dialogs for Image Modification ...................................................................................................... 16

  How to Adjust Intensity Thresholds .......................................................................................................... 17

Remove Noise or Background based on Pixel (Voxel) Intensity ................................................................... 17

The Image Calculator ..................................................................................................................................... 17

  Binary Images ............................................................................................................................................ 18

  Grayscale and color images ...................................................................................................................... 18

  Image Calculator: Summary ...................................................................................................................... 19

Filters for Image Processing and Enhancement .......................................................................................... 19

Image Segmentation ....................................................................................................................................... 21

Image Segmentation Basics .......................................................................................................................... 21

  Global and Local Thresholding .................................................................................................................. 21

  Edge Detection ......................................................................................................................................... 22

  Color Segmentation .................................................................................................................................. 23
Tools for Manual Segmentation and Object Editing................................................................. 48
Single Object Selection, Rejection, and Custom Colors.......................................................... 48
Selection, Rejection, and Editing of Multiple Objects.............................................................. 49
Object Editing and Definition Based on a Region of Interest.................................................... 51
Magic Wand Tools for Semi-Automatic Object Definition and Color-based Segmentation.................................................................................................................... 52
Tools for Two (or More) Segmented Session Members .......................................................... 54
The Multiple Stains Tool ............................................................................................................. 54
The Artifact Rejection Tool ......................................................................................................... 55
Tools for Object Classification .................................................................................................. 55
Semi-Automated Classification .................................................................................................... 56
Manual Classification .................................................................................................................. 60
Save and Restore the Current State .......................................................................................... 60
Image Analysis .............................................................................................................................. 61
Link between daime and R .......................................................................................................... 61
Display and Export of Image Analysis Results .......................................................................... 61
Isotropic Scale ............................................................................................................................. 63
Count Objects ............................................................................................................................ 63
Measure Features of Objects ..................................................................................................... 63
Biovolume Fraction (in situ Abundance Quantification of Microbial Populations)................. 66
Useful Rules to Obtain Reliable Quantification Results ............................................................ 66
The Artifact Rejection Tool of the Object Editor ...................................................................... 67
Biovolume Fraction Quantification and Evaluation of Results ................................................ 68
Evaluation of FISH Probes (Formamide Concentration Series) .......................................... 69
Useful Rules to Obtain Probe Dissociation Profiles ................................................................. 69
Image Import and Preparation ................................................................................................. 70
Analysis of Formamide Concentration Series and Evaluation of Results ............................ 71
Analysis of Spatial Arrangement Patterns ................................................................................. 72
Useful Rules for Reliable Spatial Analyses ............................................................................. 73
Spatial Arrangement Pattern Analysis by the Linear Dipole Algorithm .................................. 74
Spatial Arrangement Pattern Analysis by the Inflate Algorithm ............................................ 76
Differences Between the Two Algorithms .................................................................................. 78
Spatial Arrangement Analyses: Examples .............................................................................. 78
Reference Space Masks ............................................................................................................. 79
Introduction

daime is a digital image analysis and 2D/3D visualization software for microbial ecology, microbiome research, and related fields.

daime is free for use at institutions for academic research and for educational purposes. We ask academic users to cite the daime publication (1) in all their own publications that contain data or images obtained by using daime.

Overview: Some Key Features of daime

Import of single microscope images, image batches, and confocal z-stacks. Joint analysis of all images in batches and z-stacks.

Automatic and manual 2D and 3D image segmentation (object recognition). Object classification and quantification of 2D and 3D object features. Interactive object selection in 2D and 3D.

Stereological methods for quantifying microbial abundance and spatial arrangement patterns. Automatic analysis of stratified biofilms.

Export of analysis results in formats readable by R and other software. Seamless connection to an existing R installation for data plotting within daime. Automatic generation and export of R scripts that can be used for customizing data plots in R.

High-speed 3D visualization of z-stacks. Interactive rotation, fly-through, and stereo rendering. Export of still images and movies.
The Main Window

Figure 1. The main window of daime and the user interface elements.

The main window of daime contains the main menu and toolbar, which provide access to most features of the program. The currently loaded single images, batches, and z-stacks are listed in the central part of the main window.

An entry (image) is marked in the main window’s list by using the mouse: click with the left mouse button at the respective row of the list, and the whole row will be highlighted. Some options of the main menu and the toolbar are enabled or disabled depending on the properties of the currently marked image.

Note: The marked entry in the main window’s list can represent a single image, image batch, or z-stack. In this manual, the term “marked image” is used to simplify the text, but it always includes all these possibilities.

Many functions of daime work on the image that is currently marked in the main window. If it represents a batch or z-stack, these functions will process or analyze all images of this set.

Image Comments and Action Log

A text comment can be entered for each loaded image, batch of images, or z-stack. The comment is saved together with the images in daime’s own STK image file format.

Click this button to open the text comment area, and to view or edit the comment for the image that is currently marked in the main window’s list.
*daime* automatically logs all operations performed during an image processing and analysis workflow. This **action log** contains information needed to reproduce an analysis or to check the scientific validity of results.

Click this button to view the action log of the image that is currently marked in the main window's list. The action log can be exported in HTML and plain text formats.

*daime* logs operations that modify images or have a potential effect on analysis results. It also removes any log entries, which have become obsolete due to later operations and cannot have influenced an analysis.

The action log is saved together with the images in *daime*’s own **STK image file format**.

**Standard Workflows**

(a) Many image analysis tasks that involve one or more images, batches of images, or z-stacks

![Diagram](image)

(b) 3D Visualization of confocal z-stacks

![Diagram](image)

Figure 2. Standard image analysis workflows using features of *daime*. These workflows are only examples; they can be modified and extended to meet the specific requirements of a study.

**How to Get Help**

This user manual can be browsed from inside *daime*. Choose **Help:User manual** in the main menu.

Short help texts (**tool tips**) are available for many elements of the graphical user interface. Simply move the mouse pointer above an element and wait until the tool tip is shown. To get more information, click with the right mouse button at the element, then choose **What's This?** in the menu that will appear. Please note that these help texts may be available for enabled (active, not grayed out) user interface elements only.

Some elements of the user interface to not respond to clicks with the right mouse button, or no **What's This?** option is shown. This applies to most edit fields where words or
numbers can be entered. In this case, click with the **left mouse button** at the element (the edit field will get the cursor). Then push the **SHIFT and F1** keys to show the help text.

![Help Button](image)

Click this button to get help about main menu options or main toolbar buttons.

**General Preferences**

Choose **Extras:Preferences** to change some general settings of the program. The options include automatic checks for new *daime* versions and for news about *daime*. Other settings affect the type of decimal separator (point or comma) and the column separator (tabs or spaces) in exported data tables. **Note:** Columns in data tables, which are **exported for R**, are always separated by tabs.
Loading, Saving, and Converting Images

Importing images into daime is the first step of an image analysis or visualization workflow. daime can handle three types of image datasets:

A single 2D image can be analyzed independently from other images.

A 2D image batch is a set of 2D images that are analyzed together. The images are not spatially connected.

A z-stack consists of images that are spatially connected along the z-axis. Hence, it contains 3D information. Usually, z-stacks are acquired by using confocal microscopes or equivalent imaging technologies.

An arbitrary number of single images, batches, and z-stacks can be imported (as long as the computer has enough RAM memory to store the images simultaneously). The imported image datasets are listed in the main window of daime.

Once imported, images can be saved in daime's own STK file format. Images can also be exported from daime in the TIFF format.

Import Images

daime imports only images in the TIFF image file format. TIFF is a universal, platform-independent, standardized format and supported by most imaging programs, including microscopy software. Consult the documentation of your microscopy software for instructions on saving the acquired images in the TIFF format.

daime does not support proprietary vendor-specific image file formats, because the publicly available specifications of such formats are often incomplete or outdated. These formats may also be changed by the manufacturers without notice, breaking compatibility with third-party software.

Do not worry if you have saved all microscope images in a vendor-specific format but now want to analyze them using daime. Virtually all microscopy software packages support conversion of their proprietary image formats to TIFF.
**Import a Single 2D Image**

Click this button or choose **File:Import images** in the main menu. During image import, *daime* asks for the *µm scale* of the image.

**Import a 2D Image Batch or a 3D z-Stack**

The file names of TIFF images, which belong to the same batch or z-stack, must contain index numbers that define the order of the images. In z-stacks, images with smaller indices are on top of the stack (shown in the front in 3D visualization). All TIFF files must be in the same folder on disk.

Click this button or choose **File:Import images** in the main menu. **Select only one** of the numbered TIFF files, click **OK** or **Open** if needed (details of the file loading dialog differ among operating systems). *daime* scans the folder, which contains the selected image, for other images with index numbers and the same base file name. Then it asks for the index of the first and the last images to import. During image import, *daime* asks for the *µm scale* of the images.

⚠️ The index numbers of all images, which belong to the same batch or z-stack, must be consecutive (without gaps). Otherwise, *daime* cannot import the whole set.

<table>
<thead>
<tr>
<th>3 digits</th>
<th>≥ 2 digits</th>
<th>≥ 1 digits</th>
</tr>
</thead>
<tbody>
<tr>
<td>image000.tif</td>
<td>image00.tif</td>
<td>image0.tif</td>
</tr>
<tr>
<td>image001.tif</td>
<td>image01.tif</td>
<td>image1.tif</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>image010.tif</td>
<td>image10.tif</td>
<td>image10.tif</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>image999.tif</td>
<td>image999.tif</td>
<td>image999.tif</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Leica microscopy software uses more complex file indexing schemes when it exports z-stacks to the TIFF format. *daime* recognizes Leica indexing schemes during image import.
Table 2. Leica image indexing schemes recognized by *daime*.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>image_z(0)00_ch00</td>
<td>base name “image”, z-index 0 with 2 or 3 digits, color channel 0</td>
</tr>
<tr>
<td>image_z(0)01_ch00</td>
<td>base name “image”, z-index 1 with 2 or 3 digits, color channel 0</td>
</tr>
<tr>
<td>image_z(0)00_ch01</td>
<td>base name “image”, z-index 0 with 2 or 3 digits, color channel 1</td>
</tr>
<tr>
<td>image_t000_z(0)01_ch01</td>
<td>base name &quot;image&quot;, time-lapse index 0, z-index 1 with 2 or 3 digits, color channel 1</td>
</tr>
</tbody>
</table>

*daime* cannot automatically determine whether a set of imported images represents a 2D image batch or a 3D z-stack. The user must apply the respective 2D or 3D image analysis functions of the program accordingly.

A special file name and indexing scheme exists for formamide concentration series.

Import Color Images

When importing color images, *daime* asks whether the images should be imported as color or be converted to grayscale. When grayscale is chosen, the color channels are converted to grayscale and imported separately.

Pixel / Voxel Resolution and µm Scale

Each image in *daime* has a given **pixel** (2D) or **voxel** (3D) **resolution**. Common 2D image resolutions are 512 × 512 and 1024 × 1024 pixels. In z-stacks, the resolution in the z-dimension equals the number of images in the stack (e.g., 512 × 512 × 30 would be a z-stack of 30 images).

In addition, each image has a **micrometer (µm) scale**, which defines the size of the image in µm. The µm scale is important and affects most image analysis and visualization results. It must be indicated by the user. *daime* asks for the µm scale while importing images.

Click this button or choose **Edit:Set µm scale** in the main menu to change the µm scale of the image that is marked in the main window’s list. Always enter the size in µm of the whole image (or z-stack) in the x, y and z dimensions. Simply ignore z for 2D images and image batches. **Note:** If the µm scale of segmented images is changed, the segmentation data is lost and the images must be segmented again.
In 2D images, the origin of the pixel coordinate system is in the top left corner. The origin of the μm coordinate system is in the bottom left corner.

In 3D z-stacks, the origin of the voxel coordinate system is in the top left corner of the first image of the stack. By default, in 3D visualization this is the front image. Thus, the z-coordinate increases away from the viewer and towards the last image of the stack. The origin of the μm coordinate system is in the bottom left corner of the first (front) image.

**Reduce Pixel (Voxel) Resolution**

If the pixel (voxel) resolution of images is very high, images consume much memory and most analyses take longer without a significant improvement of the results.

To reduce the pixel (voxel) resolution of images, choose **Edit:Reduce resolution** in the main menu. An \(x \times y\)-resolution of 512 \( \times \) 512 or 1024 \( \times \) 1024 is sufficient for most analyses. The \(z\)-resolution can be changed by keeping only every \(n\)th image, discarding the others. *daime* uses an interpolation algorithm to preserve as much image data as possible when the resolution is reduced. Still, very small objects may be lost if the target resolution is much lower than the original resolution.

**STK Image Files: The Native File Format of *daime***

**STK** (an abbreviation for “stack”) is *daime*’s own image file format. An STK file can contain a single image, or a whole image batch or z-stack. Images are compressed to save disk space. STK files also contain all metadata including the \(\mu m\) scale, segmentation data, text comments, and the action log.

It is strongly recommended to save images in the STK format. This ensures that analyses can easily be reproduced.

[Image: STK icon]
**Click this button or choose File:Load in the main menu to load an STK image file from disk.**

[Image: STK icon]
**Click this button or choose File:Save in the main menu to save the currently marked image as STK file to disk.**

[Image: STK icon]
**Click this button or choose File:Save as in the main menu to save the currently marked image as STK file to disk. A new filename and folder can be chosen.**

Choose **File:Load all** in the main menu to load all STK files at once from a selected folder. Choose **File:Save all** to save all images at once as STK files into their folders, without changing the file names. Choose **File:Close** or **File:Close all** to close images.
Export Images
Choose File:Export in the main menu to export images as TIFF files. Batches and z-stacks are exported as multiple TIFF files with index numbers appended to the file name. Note: Exported TIFF files do not contain metadata, such as µm scale and segmentation data.

Image Types and Conversion
Supported TIFF images can be monochrome (8 or 16 bits per pixel) or color (RGB, 24 or 32 bits per pixel). Uncompressed and some compressed TIFF formats (e.g., LZW-compressed) can be imported.

**Monochrome (grayscale) image.** Each pixel has an intensity (brightness) value, but there is no color information.

**Color image.** Each pixel has a green, red, and blue (RGB) component. They determine the pixel color and brightness.

**Binary image.** Each pixel is either black or white. Binary images are also called object masks in daime. See also Object Layer and Object Mask.

Table 3. Commands for image type conversions.

<table>
<thead>
<tr>
<th>Task</th>
<th>Command (main menu)</th>
<th>Source image</th>
<th>Result image</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convert grayscale image to color</td>
<td>Edit:Color:Convert to color (RGB) (choose a base color; red in this example)</td>
<td><img src="image1.png" alt="Grayscale Image" /></td>
<td><img src="image2.png" alt="Color Image" /></td>
</tr>
<tr>
<td>Convert grayscale image to binary</td>
<td>Segment:Convert to object mask (choose intensity thresholds)</td>
<td><img src="image3.png" alt="Grayscale Image" /></td>
<td><img src="image4.png" alt="Binary Image" /></td>
</tr>
<tr>
<td>Convert color image to grayscale</td>
<td>Edit:Color:Extract color channels (check Convert to grayscale)</td>
<td><img src="image5.png" alt="Color Image" /></td>
<td>One to three grayscale images that contain the red, green, and blue color channels</td>
</tr>
</tbody>
</table>
Image Processing and Enhancement

daimé contains tools for processing and enhancing images. Applications of these tools include:

- Preparing images for segmentation and subsequent analysis, or for visualization, if the quality of the original images is insufficient.
- Combining images, extracting or removing overlapping regions, inflating or shrinking regions as part of an image analysis workflow.
- Increasing/decreasing image brightness, detecting and highlighting the edges of regions (e.g., for specific segmentation or visualization tasks).

Clone and Copy Images

Click this button or choose **Edit:Clone** in the main menu to clone the image that is currently marked in the main window's list. Image data, µm scale, segmentation data (if any), action log, and comments are cloned automatically.

In order to copy images into existing other images, first mark the source images in the main window's list. Then choose **Edit:Copy** in the main menu. Select the target images. **Note:** The contents of the target images will be overwritten.

Invert Images

Choose **Edit:Invert** in the main menu to invert the currently marked image. This function can help analyze bright field or phase contrast images: Cells that are brighter than the background in the inverted images are more easily detected by the segmentation tools.

<table>
<thead>
<tr>
<th>Image type</th>
<th>Original image</th>
<th>Inverted image</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary image (object mask)</td>
<td><img src="image1.png" alt="Binary Image" /></td>
<td><img src="image2.png" alt="Inverted Binary Image" /></td>
</tr>
<tr>
<td>Grayscale image</td>
<td><img src="image3.png" alt="Grayscale Image" /></td>
<td><img src="image4.png" alt="Inverted Grayscale Image" /></td>
</tr>
<tr>
<td>Color image</td>
<td><img src="image5.png" alt="Color Image" /></td>
<td><img src="image6.png" alt="Inverted Color Image" /></td>
</tr>
</tbody>
</table>
Preview Dialogs for Image Modification

Several image processing functions of daime show a preview dialog. This dialog contains the image, batch or z-stack to be processed and a user interface for adjusting any required parameters. The effects that the processing algorithm has on the images can be previewed before the changes are finally accepted (or dismissed).

Figure 3. Main parts of a preview dialog. A dialog may contain additional, feature-specific elements for parameter adjustments (not shown here).

Table 5. Mouse and keyboard commands for scrolling, zooming and panning in image preview dialogs.

<table>
<thead>
<tr>
<th>Action</th>
<th>Mouse only</th>
<th>Mouse &amp; keyboard, or keyboard only</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scroll through image batch or z-stack</td>
<td>Move the slider</td>
<td>Press TAB until slider has keyboard focus, then use up/down arrow keys for scrolling</td>
</tr>
<tr>
<td>Zoom in/out the image</td>
<td>Turn mouse wheel</td>
<td>Press CTRL (zoom in) or SHIFT (zoom out) and click the middle mouse button</td>
</tr>
<tr>
<td>Panning (image must be zoomed in)</td>
<td>Push middle mouse button and move mouse</td>
<td>-</td>
</tr>
<tr>
<td>Show whole image</td>
<td>-</td>
<td>Press ALT and click middle mouse button</td>
</tr>
</tbody>
</table>

Click Apply in a preview dialog to see the effects that the processing function has on the images. Every time Apply is clicked the function is applied again to the (already modified) images. Thus, it is possible to apply a function multiple times but with different parameters to the same images. Click Reset to undo all changes (the original images are shown again). Click OK to permanently accept the changes, or click Cancel to dismiss the dialog.
How to Adjust Intensity Thresholds

Various image processing functions take pixel (or voxel) intensity thresholds as parameters. The respective preview dialogs contain a user interface for adjusting these thresholds in different ways: by directly entering the intensity values, by choosing the thresholds in an image histogram, and by pointing/clicking at pixels in the image.

![User interface for adjusting intensity thresholds.](image)

**To set thresholds directly from an image:** Click a mouse button above the image (left button: low, right button: high threshold). The respective threshold will be set to the intensity of the pixel below the mouse cursor.

**Selection mask:** pixels below the low threshold are shown in blue, pixels above the high threshold are shown in red in the preview dialog.

Remove Noise or Background based on Pixel (Voxel) Intensity

Choose Edit:Eliminate pixels (voxels) in the main menu. Then adjust the intensity range. Then click Apply to "eliminate" (=set to black) all pixels/voxels within this intensity range. If needed, repeat this procedure to eliminate pixels (voxels) in different intensity ranges.

The Image Calculator

Click this button or choose Edit:Image calculator in the main menu to open the Image Calculator. This tool uses simple mathematical and Boolean operations to combine, add or subtract images. It modifies the image, which is currently marked in the main window's list (here: "Image 1"). For operations requiring two images, a second image (or batch, z-stack) can be selected in the Image Calculator dialog. The second image (here:
"Image 2") must have the same resolution as image 1. Batches/z-stacks must contain the same number of images. Note: Only suitable images, which meet these conditions, are offered for selection. **Image 2 is never modified.**

**Binary Images**

The Image Calculator is primarily designed for processing [binary images](#) (object masks) by Boolean operations.

Table 6. Effects of Image Calculator operations on binary images. Values in the table are the states (1=white, 0=black) of four adjacent pixels.

<table>
<thead>
<tr>
<th></th>
<th>AND</th>
<th>OR</th>
<th>XOR</th>
<th>Addition</th>
<th>Subtraction</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Image 1</strong></td>
<td>1010</td>
<td>1010</td>
<td>1010</td>
<td>1010</td>
<td>1010</td>
</tr>
<tr>
<td><strong>Image 2</strong></td>
<td>1100</td>
<td>1100</td>
<td>1100</td>
<td>1100</td>
<td>1100</td>
</tr>
<tr>
<td><strong>Result (in image 1)</strong></td>
<td>1000</td>
<td>1110</td>
<td>0110</td>
<td>1110</td>
<td>0010</td>
</tr>
</tbody>
</table>

The “Multiplication” operation is not designed to be used on binary images.

**Grayscale and color images**

**Addition, Subtraction, and Multiplication** are the main operations of the Image Calculator to be used with grayscale and color images. The first two operations add/subtract the pixel intensities or RGB color values of two images: “Image 1” + “Image 2” or “Image 1” – “Image 2”. Multiplication multiplies the pixel intensities or RGB values of “Image 1” with the indicated factor. Use multiplication to increase or decrease the overall brightness or color intensity of an image.

The Boolean operations can also be applied to grayscale and color images, but this is usually not very useful. The pixel intensities or RGB color values are changed by bit-wise Boolean operations.
Image Calculator: Summary

Table 7. Illustration of the main operations of the Image Calculator.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Input</th>
<th>Image 1</th>
<th>Image 2</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary Image 1</td>
<td>Image 1 AND Image 2</td>
<td>Image 1 XOR Image 2</td>
<td>Original grayscale image</td>
<td></td>
</tr>
<tr>
<td>Binary Image 2</td>
<td>Image 1 OR/plus Image 2</td>
<td>Image 1 minus Image 2</td>
<td>Image multiplied by factor 3</td>
<td></td>
</tr>
</tbody>
</table>

Filters for Image Processing and Enhancement

Several algorithms ("filters") are available to prepare images for subsequent steps, such as image segmentation and analysis. In most cases, the aim is to reduce noise or autofluorescent background, improve contrast, or visualize specific features (e.g., edges).

Most filters exist as 2D and 3D versions. The 3D versions take all three dimensions into account, thus processing voxels instead of pixels. Use the 3D versions for z-stacks and the 2D versions for single images and 2D image batches.

Filters can be fine-tuned by adjusting specific parameters. In a filter preview dialog, click this icon ![Icon](image) and then click at a parameter to get an explanation.
<table>
<thead>
<tr>
<th>Input image</th>
<th>Filter</th>
<th>Output image</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Noise reduction" /></td>
<td>Edit:2D/3D filters: Noise reduction</td>
<td><img src="image" alt="Noise reduction" /></td>
<td>Removes spurious pixels or voxels (random noise). Images are not smoothed.</td>
</tr>
<tr>
<td><img src="image" alt="Median filter" /></td>
<td>Edit:2D/3D filters: Median filter</td>
<td><img src="image" alt="Median filter" /></td>
<td>Reduces noise and background, and removes spotty artifacts, grains, etc. by smoothing images. Smooths surfaces of objects.</td>
</tr>
<tr>
<td><img src="image" alt="Histogram stretching" /></td>
<td>Edit:2D/3D filters: Histogram stretching</td>
<td><img src="image" alt="Histogram stretching" /></td>
<td>Improves the contrast and can be useful for (dark autofluorescence) background reduction.</td>
</tr>
<tr>
<td><img src="image" alt="Blur and subtract" /></td>
<td>Edit:Background: Blur and subtract</td>
<td><img src="image" alt="Blur and subtract" /></td>
<td>Removes bright background (but cells must still be brighter). A similar effect is achieved by Edit:Background: Local average subtraction.</td>
</tr>
<tr>
<td><img src="image" alt="Illumination correction" /></td>
<td>Edit:Background: Illumination (vignetting) correction</td>
<td><img src="image" alt="Illumination correction" /></td>
<td>Corrects for uneven illumination in transmitted light or phase contrast images. Not for fluorescence images.</td>
</tr>
<tr>
<td><img src="image" alt="Edge detection" /></td>
<td>Edit:2D/3D filters: Edge detection</td>
<td><img src="image" alt="Edge detection" /></td>
<td>Extracts the edges of objects. Mainly useful for 2D/3D visualization purposes.</td>
</tr>
</tbody>
</table>

**Edit:2D/3D filters: Morphology** provides mathematical morphology operations (dilation, erosion, open, close, etc.) with different structuring elements. For background information, refer to the general literature on image analysis. For binary images, an additional operation (skeleton by influence zones, SKIZ) is available. This is one way to split touching objects. First, erode the images (with a suitable structuring element) until touching objects are separated. Then select SKIZ and make sure that **Use original images** as mask for SKIZ is checked. After SKIZ, original object sizes are restored and touching objects should be separated. Success depends on various factors including the chosen structuring element.
Image Segmentation

Image segmentation is an essential and also the most critical step of almost every image analysis workflow. It comprises (1) the general differentiation of objects from "non-object" image regions (usually the background), and (2) the subsequent detection of individual objects and their differentiation from each other.

All downstream analyses rely on the successful object recognition during segmentation. Thus, it cannot be overemphasized that segmentation is an extremely important task. Its success depends on the image quality, image content, and the applied segmentation algorithm. There is no silver bullet approach that would segment all images equally well. Instead, one must often test different segmentation methods to find the one that works best with a given set of images.

daime provides different methods for the segmentation of 2D images and 3D z-stacks. In 2D segmentation mode, objects extend only in the x and y dimensions but are flat in the z dimension. In 3D segmentation mode, objects can span multiple images of a z-stack and thus extend in all three spatial dimensions.

2D objects in a 2D-segmented image. The objects are marked by orange edges.

A 3D object in a 3D-segmented z-stack. The object is marked by an orange bounding box.

Once images are segmented, the actual image analysis can be performed. For example, the objects can be counted, their size or brightness be measured, their spatial arrangement patterns be analyzed, and so on. daime also provides an interface to work interactively with segmented images: The Object Editor, which is part of the Visualizer.

Image Segmentation Basics

Global and Local Thresholding

Thresholding is an image segmentation approach that identifies objects based on the brightness (intensity) of their pixels (voxels). This works only if the intensity of objects is sufficiently different from the background intensity.
A pixel intensity threshold ($t$) and the intensity of a pixel ($I$) determine whether a pixel is background or biomass.

If $t$ is chosen well, objects are correctly detected (yellow line). If $t$ is not optimal, objects are too large (blue line) or small (red line).

To find a suitable threshold $t$ automatically, 	extit{daime} offers these thresholding algorithms:

- **RATS**: Optimized to segment objects of medium to high brightness and to exclude darker background noise (2).
- **RATS-L**: Modified RATS that performs better on images containing also darker objects. But is affected more by noise.
- **Isodata**: Similar to RATS, but can work better if background is relatively bright (3).

In 	extbf{global thresholding}, one $t$ is used to segment the whole image. This can fail if images contain both bright and dark objects (arrow). In this case, no global $t$ can be found for correct segmentation.

**Local thresholding** solves this problem by subdividing an image in regions and finding a suitable local $t$ for each region separately. This method can correctly segment dark and bright objects in the same image. However, it is also more sensitive to noise.

	extit{daime} offers local thresholding in addition to the global thresholding methods (4).

**Edge Detection**

Segmentation by 	extbf{edge detection} aims to find the edges of objects without intensity thresholding. It detects brighter objects on dark background and vice versa. This method can segment fluorescence, bright field, and phase contrast micrographs. 	extit{daime} offers edge detection as an alternative to thresholding.
Fluorescence image of bacterial cells. Original image (left) and detected object edges (right).

Transmitted light image of bacterial cells. Original image (left) and detected object edges (right).

**Color Segmentation**

Color can also be used as a criterion for segmentation. *daime* offers functionality for image segmentation based on color in the Object Editor.

A color image of microbial cells that were labeled by rRNA-targeted fluorescence *in situ* hybridization with different probes.

Assume as an example that in the color image above, the yellow-orange cells have been detected by color-based segmentation. A binary image of the identified objects is shown here to illustrate this.

**Object Splitting**

After initial segmentation, touching objects (e.g., cells) are often treated as one big object (shown here in green).

Object splitting algorithms try to improve this and detect the individual objects, which are shown here in different colors to illustrate this.
**daime** offers a tool for object splitting based on the *morphological multiscale algorithm* (5) in the **Object Editor**.

**Object Layer and Object Mask**

In **daime**, all segmented images consist of two “layers”: a visible **image layer** that contains all the graphical data (pixels or voxels), and an invisible (virtual) **object layer** that contains all the information about individual objects (=the segmentation data). Unsegmented images lack the object layer.

The object layer is mainly used internally by **daime**. However, it can be extracted as a new, binary image that is called an **object mask**. In the mask, object pixels are 1 and non-object pixels are 0.

Click this button or choose **Segment:Extract object layer** in the main menu to extract the object layer of a segmented image.

![Object Layer and Object Mask](image)

Figure 5. A segmented image has an object layer (depicted as a red frame here; the object layer is actually invisible to the user). The extracted object mask is a binary image that represents the objects, which are defined in the object layer.
An object layer can be **transferred** (copied) from a segmented source image to an unsegmented target image. This will effectively segment the target image (*Figure 6*). The target image must have the same resolution as the source image.

![Click this button or choose Segment:Transfer object layer in the main menu to copy the object layer of a segmented image to another, unsegmented image.](image)

Click this button or choose **Segment:Delete object layer** in the main menu to remove the object layer from a segmented image. This will delete all segmentation data, leaving the image unsegmented.

![Click this button or choose Segment:Delete object layer in the main menu to remove the object layer from a segmented image. This will delete all segmentation data, leaving the image unsegmented.](image)

*Figure 6.* The object layer of a segmented image can be transferred (copied) to another, still unsegmented image.

Transferring object layers is a simple and efficient approach to solve problems that would be much harder to address by other means. For example, one can process and enhance an image to prepare it for segmentation but do the actual image analysis with the original, unmodified image.

This is illustrated by the following example (*Figure 7*): segmentation of a noisy image, which is hard to segment, without modifying the original image (preserving all image data for scientific accuracy).

![Figure 7. Segmentation of a noisy image without modifying the original image data.](image)

Being ordinary binary images, object masks can be manipulated, for example by the Boolean operations of the **Image Calculator**. An object mask is a separate image and has no further link to the original segmented image it was extracted from. However, an object mask can also be segmented itself (then it gets its own object layer). This object layer can be transferred to other images, including the original image. **Example:** two coexisting microbial populations have been labeled with different fluorescent dyes. In images, the fluorescent signals of these two organisms should never overlap (because cells can obviously not overlap). In reality, this happens due to imaging (microscopy) biases. To
correct this error, overlapping regions can be removed from the object definitions by the following workflow.

Figure 8. Workflow to correct overlapping object definitions. For the sake of simplicity, the microbial populations are illustrated as circles, which are surrogates for cells, cell aggregates, etc.
Automatic Image Segmentation

Click this button or choose **Segment:Automatic segmentation** in the main menu to automatically segment the image that is currently marked in the main window's list.

The automatic segmentation feature treats color images as grayscale for thresholding or edge detection. The colors are ignored. Semi-automatic tools for color-based segmentation are part of the **Object Editor**.

The automatic segmentation feature can also be used to segment object masks. In this case, no thresholding or edge detection is needed.

Several parameters (**Table 9**) can be adjusted prior to automatic segmentation. Click **Segment!** to start segmentation. When segmentation is completed, the segmentation mode (2D or 3D) and the total number of objects are shown for the segmented image in the main window's list (see **The Main Window**).

**Table 9. Options for automatic image segmentation.**

<table>
<thead>
<tr>
<th><strong>Segmentation mode (2D or 3D)</strong></th>
<th>Determines whether the image will be segmented in 2D or 3D mode. Use 3D mode to analyze three-dimensional objects in z-stacks. Use 2D mode for all other images.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Biomass detection (Edge detection or Thresholding)</strong></td>
<td>Determines how biomass will be distinguished from image background. For details, see <strong>Global and Local Thresholding</strong> and <strong>Edge Detection</strong>.</td>
</tr>
<tr>
<td><strong>Thresholding algorithms (RATS, RATS-L, Isodata, Local, Custom)</strong></td>
<td>Only relevant if biomass detection method is <strong>thresholding</strong>. Selects the algorithm for intensity thresholding. For details, see <strong>Global and Local Thresholding</strong>. If <strong>Custom</strong> is selected, the intensity range for thresholding can be <strong>manually defined</strong> (only pixels/voxels within this range will be part of objects).</td>
</tr>
<tr>
<td><strong>Edge detection parameters</strong></td>
<td>Only relevant if biomass detection method is <strong>edge detection</strong>. The edges of objects usually enclose bright areas (such as fluorescent biomass), but objects may also contain darker regions (&quot;holes&quot;). By default, internal dark regions are not included as parts of the detected objects. The <strong>Dark threshold</strong> is used to define internal dark regions: all pixels, which are located within the object borders and have an intensity below this threshold, are considered to be &quot;dark&quot; and are not part of the object. To change the default behavior, check <strong>Incl. internal dark regions</strong>. Dark regions will then be included as parts of the objects.</td>
</tr>
<tr>
<td><strong>General options</strong></td>
<td>Check <strong>Ignore obj. up to</strong> if the images contain artifacts or noise that is smaller (in pixel/voxel size) than the biomass objects. <strong>daime</strong> will ignore all objects with a size up to the indicated number of pixels (or voxels in 3D mode). A default value is provided based on the image resolution and the segmentation mode (2D or 3D).</td>
</tr>
</tbody>
</table>
**Dump Segmentation Data to Disk**

The fundamental segmentation data (which pixels/voxels belong to which object) can be dumped to a text (ASCII) file. This file can be imported into third-party software, such as spreadsheet programs or R, for object-based analyses outside *daime*.

Choose **Segment:Dump segmentation data** in the main menu to export the segmentation data from the 2D- or 3D-segmented image that is currently marked in the main window's list. Subsequently, choose a disk folder and a file name for the text file.

The resulting text file contains a tab-delimited table. For every object pixel (or voxel) there is one table row with the following columns (from left to right):

- **2D-segmented images**: the index number of the image (in a batch), the internal object identifier number, the x-coordinate, and the y-coordinate (both in the pixel coordinate system). The last column(s) contain the pixel intensity for grayscale images and the red, green, and blue intensities for color images. No intensities are exported for binary images.

- **3D-segmented images**: the internal object identifier number, the x-coordinate, the y-coordinate, and the z-coordinate (all in the voxel coordinate system). The last column(s) contain the voxel intensity for grayscale images and the red, green, and blue intensities for color images. No intensities are exported for binary images.

ℹ️ Only the data of **selected** objects are exported, whereas **rejected** objects are ignored. Use the **Object Editor** to select and reject objects.
The Visualizer: Overview and Basics

dai me is software for image analysis and for 2D and 3D visualization. The visualization functionality is tightly linked to the image processing and analysis tools. The Visualizer is the part of dai me for 2D and 3D visualization. The Visualizer also contains the Object Editor, which allows the user to work interactively with objects in segmented images.

Open the Visualizer

Click this button or choose View: New visualizer in the main menu to open a new Visualizer window. The new Visualizer will not yet show any images. Alternatively, double-click at an image in the main window's list. This will open a new Visualizer that already shows this image.

To close a Visualizer, click the standard close button in the title bar of the Visualizer window.

Multiple Visualizer windows can be open at the same time. The number of simultaneously open Visualizers is limited only by the available computer memory (RAM).

Note: 3D visualization may consume much memory.

Simultaneously open Visualizers can even show the same images. For example, one Visualizer may show a z-stack in 2D, whereas another Visualizer shows the same z-stack in 3D. Or two Visualizers show the same z-stack in 3D but from different viewing perspectives. Of course, the Visualizers can also show different images.

The µm scale of an image must be defined prior to visualization. The Visualizer relies on the µm scale for correct 2D and 3D rendering.

The Visualization Session

A visualization session contains all information that is needed for 2D or 3D visualization: which images are shown, how these images are rendered, and whether additional features such as coordinate systems are drawn. A visualization session can be saved to disk and can later be loaded again. Thus, visualization work can be reproduced easily.

There are three ways to create a new session:

1. By double-clicking an image in the main window's list (see above).
2. Drag and drop. Click with the left mouse button at an image in the main window's list. Keep the mouse button pushed down, “drag” the image into the black display field of an empty Visualizer, and release the mouse button. This will create a new session with the “dragged and dropped” image as the first member. The image will be shown in 2D visualization mode. Up to two more images can be added to the session in the same way. Grayscale and binary images will be shown in red, green...
and blue (the standard colors used to visualize microbial cells labeled by rRNA-targeted FISH). Color images will be shown in their original colors.

3. By choosing **Session:New session** in the Visualizer menu. This opens the **Session Editor**.

> Pixels (or voxels) that occur at the same position in different images are automatically shown in mixed colors (e.g., yellow if a red and a green pixel are at the same position).

> If a session already contains three members, it is not possible to add more images by the drag and drop method. Instead, choose **Session:Edit session** in the Visualizer menu to open the **Session Editor** and add additional images to the session.

> All images in the same session must have identical pixel (or voxel) resolution and µm scale. Image batches or z-stacks must also contain the same number of images.

### The Session Editor

Click this button or choose **Session:New session** or **Session:Edit session** in the Visualizer menu to open the Session Editor.

The Session Editor can be used to create a new visualization session and to add or remove session members. It contains the following elements:

1. A list of **available images** that can be added to the new or current session. If the Session Editor is used to create a new session, this list contains all loaded images. If the Session Editor is used to modify an existing session, this list contains only images whose pixel (voxel) resolution, µm scale, and number of images (in batches and z-stacks) match the other members of the session.

2. **Arrow buttons.** Click these buttons to add or remove images to/from the session.

3. A list of the **current session members**.

### Combinations of Session Members

The list of current session members contains a checkbox **Comb.** next to each image. Check this box to **combine** the respective session members. This has some advantages:

1. Mixed colors (e.g., caused by multiple FISH probes bound to the same microbial cells) are correctly shown in all visualization modes. If the respective session members are not combined, then mixed colors may not be visible in some 3D visualization modes.

2. Visualization parameters, such as brightness and opacity, can be adjusted once for the whole combination. Adjustments for each member are not needed.

3. Much less memory is needed for the 3D visualization of combined members than for the same number of individual session members.
4. 3D visualization is faster for combinations than for the same number of individual session members.

5. During 3D visualization of z-stacks, shadows can be rendered more realistically. This is, because objects in one z-stack cast shadows onto objects in another z-stack only if the z-stacks have been combined in the session.

A visualization session can contain a maximum of one combination. However, it can contain additional members outside this combination. Session members, which need individual adjustments of visualization parameters, should not be part of a combination.

The Object Editor cannot be used if the session contains a combination.

Save and Load a Visualization Session

A visualization session (including all rendering settings, the viewer position and scene rotation angles, clipping objects, coordinate systems, etc.) can be saved to disk. Visualization work can easily and quickly be reproduced by loading a previously saved session from disk.

Click this button or choose Session:Save session in the Visualizer menu to save the current session to disk. The extension of session files is .vis.

The member images themselves are not saved to disk when the session is saved. Instead, the session file contains links to the image (STK) files on disk. Therefore, save all member images of a session to disk before the session itself is saved (the session can be saved into another folder than the images). This ensures that the session file contains valid links to the images. The separation of images from session files saves disk space: The same images can be member of many visualization sessions, but need to be stored only once on disk.

Click this button or choose Session:Load session in the Visualizer menu to load a session from disk. daime loads the session and also the images that are members of this session. The program asks for the location of these images on disk if they cannot be found automatically (this happens if the images have been renamed or moved to another folder after the session was saved). Member images, which have already been loaded into memory before loading the session, are usually not loaded again.
The Visualizer Window

Use the options in the **Size** menu to change the size of the Visualizer's image display area.

![Visualizer window](image)

Figure 9. A Visualizer window that already contains a visualization session and shows a biofilm sample in 2D visualization mode.

<table>
<thead>
<tr>
<th>Button Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Session:Edit session</strong></td>
<td>Open the <strong>Session Editor</strong>.</td>
</tr>
<tr>
<td><strong>Session:New session</strong></td>
<td>Open the <strong>Session Editor</strong>.</td>
</tr>
<tr>
<td><strong>Session:Load session</strong></td>
<td><strong>Load a visualization session</strong> from disk.</td>
</tr>
<tr>
<td><strong>Session:Save session</strong></td>
<td><strong>Save the current visualization session</strong> to disk.</td>
</tr>
<tr>
<td><strong>Render:2D images</strong></td>
<td>Switch to <strong>2D visualization mode</strong>.</td>
</tr>
<tr>
<td><strong>Render:3D volume rendering</strong></td>
<td>Switch to <strong>3D visualization mode</strong>. This button is only available if the computer hardware supports 3D rendering.</td>
</tr>
<tr>
<td><strong>Render:Snapshot</strong></td>
<td>Take a <strong>snapshot</strong> of the current scene.</td>
</tr>
<tr>
<td><strong>Render:Clipping</strong></td>
<td>Show the tools for <strong>3D clipping</strong> (available in 3D visualization mode only).</td>
</tr>
<tr>
<td><strong>Render:Keyframes and animation</strong></td>
<td>Show the <strong>Keyframe Editor</strong> for creating animations.</td>
</tr>
<tr>
<td><strong>Render:Object editor</strong></td>
<td>Show the <strong>Object Editor</strong>.</td>
</tr>
</tbody>
</table>
Snapshots

Click this button or choose **Render:Snapshot** in the Visualizer menu to take a snapshot image of the currently rendered scene. The snapshot is shown in a new window. When a Visualizer is closed, all snapshot windows belonging to this Visualizer are closed, too.

A snapshot image can be saved to disk, copied to the clipboard, and printed. Snapshots are saved to disk as color (RGB) TIFF images. The size of a snapshot image (in pixels) equals the current size of the Visualizer's image display area.

The Visualizer: 2D Visualization

When a new visualization session is started, the Visualizer is by default in **2D visualization mode**.

Click this button or choose **Render:2D images** in the Visualizer menu to switch to 2D visualization mode.

The Visualizer offers the following tools for 2D visualization.

“Nav” tab of the tool area

- **Image slider**: Use the vertical slider next to the display area for scrolling through an image batch or z-stack. The current image is shown in 2D. Its index number is displayed at the bottom of the slider.

- **Split view**: If a session contains more than one member, the resulting overlay image is shown by default. Overlapping objects are shown in mixed colors (e.g., yellow is shown where red and green objects overlap). To separate the images, check the **Split view** box.

“Scene” tab of the tool area

- **Coordinate systems**: Use these switches to show coordinate systems. The origin can be in the center (**Center**) or in the bottom left corner (**Bottom**) of the image. The coordinate units are µm. Click **Settings** to change the tick intervals and colors of coordinate systems.

“Members” tab of the tool area

- A list of all session members is shown. Select one of the members to adjust specific settings by using the other tools of the “Members” tab.

- Click at the **color bar** or click the **color button** to change the color of the selected session member. **Note**: Only grayscale and binary images are displayed in the chosen color. Color images are always displayed in their original colors.

- **Intensity factor**: Change this value to display the selected session member brighter (values > 1) or darker (values < 1). Value 1 means that the displayed brightness equals the original pixel intensities.
- **Intensity cutoff**: Pixels that are darker than this value are not rendered. This option can be useful to suppress dim noise or background.
- **Render**: Check/uncheck this box to show/hide the selected session member.

*Zooming and panning* of 2D images are possible by using the mouse or the keyboard as described in *Table 5*.

The **Visualizer: 3D Visualization**

The 3D visualization mode is used for exploring z-stacks. A z-stack can be projected, rotated, zoomed, and shifted interactively on screen. The user can virtually “fly through” the z-stack, turn around, and inspect the image data from any viewing perspective and position in 3D space.

The 3D visualization technique used by *daime* is called **volume rendering**. The complete information (all voxels) of a z-stack is taken into account for rendering the 3D scene. This approach offers a high flexibility for fine-tuning the appearance. For example, objects can be rendered semi-transparent in order to visualize their surfaces and inner structures at the same time.

*Interactive volume rendering uses the graphics hardware of the computer. The 3D visualization mode is only available if the hardware supports this.*

**Rendering and Exploring a 3D Scene**

Click this button or choose **Render: 3D volume rendering** to switch to the 3D visualization mode. *daime* analyzes the z-stacks of the current session and prepares internal data structures for 3D visualization. The 3D scene is shown in the display field of the Visualizer.

**Rotating the 3D Scene**

“Nav” tab of the tool area, “Scene rotation”

- Use the **wireframe sphere** for rotating the 3D scene. Click with the left mouse button at the sphere, push the button, and move the mouse to rotate the sphere. The 3D scene follows the movement. Release the mouse button to stop the rotation.
- Alternatively, use the **X/Y/Z dials** for rotating the scene around the respective axis of a 3D Cartesian coordinate system. **Turn the dials with the mouse.** To rotate the scene in exact steps of 90°, first click with the left mouse button at a dial, and then use the PAGE UP and PAGE DOWN keys to rotate the 3D scene.
- Alternatively, **click with the left mouse button at any point in the display field** of the Visualizer. Keep the mouse button pushed down and move the mouse to rotate the scene. Release the mouse button to stop the rotation.
- Click **Reset** to restore the original, non-rotated state of the 3D scene.
Rotating and Moving the Viewer

“Nav” tab of the tool area, “Viewer rotation”

- Use the wireframe sphere for rotating the viewer. Click with the left mouse button at the sphere, push the button, and move the mouse to rotate the sphere. The viewer will follow the movement. Release the mouse button to stop the rotation.

- Alternatively, use the X/Y/Z dials for rotating the viewer around the respective axis of a 3D Cartesian coordinate system. Turn the dials with the mouse. To rotate the viewer in exact steps of 90°, first click with the left mouse button at a dial, and then use the PAGE UP and PAGE DOWN keys to rotate the 3D scene.

- Alternatively, push down the SHIFT key and click with the left mouse button at any point in the display field of the Visualizer. Keep the key and mouse button pushed down and move the mouse to rotate the viewer. Release the mouse button to stop the rotation.

- Click Reset to restore the original, non-rotated state of the viewer.

- To move the viewer sideward (panning), click with the middle mouse button at any point in the display field of the Visualizer. Keep the mouse button pushed down and move the mouse to move the viewer sideward. Release the mouse button to stop the movement.

- To move the viewer backward/forward, ensure that the mouse pointer is above the display field of the Visualizer, then turn the mouse wheel. For smaller steps, push down the SHIFT key and turn the mouse wheel. If the mouse has no wheel, push down either CTRL or SHIFT and click the middle mouse button.

- Click Reset Viewer to move the viewer to the default position and to restore the original, non-rotated state of viewer.

Defining the Global Center of Rotation

(Advanced topic)

The whole 3D scene is rotated around the global center of rotation, which is a single point in 3D space. By default, this point is the center of the 3D scene. It can make sense to move the global center of rotation to another place. For example, if an object should rotate around itself (to show it from all sides), the rotation center should be close to the center of that object.

“Nav” tab of the tool area

- If Show is checked, the global center of rotation is shown as a 3D star. The star is yellow if the global center of rotation is located within the volume of the z-stack(s). It is orange if the global center of rotation is outside this volume. Both situations work well, but the resulting rotations are quite different.

- To move the global center of rotation to another point, first move the viewer close to the desired location in 3D space and ensure that the viewer is looking at this point (i.e., the point should be in the center of the display field). Then click button Set. The orange or yellow star appears in front of the viewer.
- Now **push the ALT key and turn the mouse wheel** to move the star away from or towards the viewer, until the star is placed at the intended new center of rotation. For smaller steps, push **both SHIFT and ALT** and **turn the mouse wheel**. Note: The color of the star changes when it moves out of or back into the z-stack volume in 3D space.
- When done, uncheck **Show** to hide the global center of rotation.
- Click **Reset** to move the global center of rotation back to its default position.

**Rendering Settings for the 3D Scene**

The tools described in this section are found in the **“Scene” tab** of the Visualizer tool area.

**Coordinate Systems**

Click **Center** or **Border** to show or hide 3D coordinate systems. The origin can be in the center of the 3D scene (**Center**) or at its bottom left and front corner (**Bottom**). The coordinate unit is µm. Click **Settings** to change the tick intervals and colors of coordinate systems.

![3D Coordinate systems: center (left) and border (right)](image)

**3D Settings (Quality, Projection, Depth Cue, Stereo, Background)**

- **Quality**: Choose a higher number of slices for a better display quality. Rendering is faster with a lower number of slices, but quality may suffer. **Rule of thumb**: This number should be at least two times the number of images in a rendered z-stack. A higher quality may be needed to reduce rendering artifacts in close-up images (**Figure 11**).
Figure 11. Detail of a z-stack visualized by volume rendering. **A.** Quality set to 25 slices; so-called woodgrain artifacts are clearly visible. **B.** Quality set to 100 slices; woodgrain artifacts have disappeared. This z-stack contained 40 images.

- **Parallel projection:** 3D rendering uses one of two principal projection modes, perspective or parallel projection. With **perspective projection**, objects far away from the viewer appear smaller than objects close to the viewer. With **parallel projection**, the relative size of objects does never change. Parallel projection clearly differs from real-world optics but is more suitable for scientific visualization, because all objects are shown true to scale. The default projection mode in daime is parallel projection. Uncheck this box (i.e., switch to perspective projection) for a better 3D impression and for “fly-through” by moving the viewer.

- **Depth cue:** This means that those parts of the 3D scene, which are far away from the viewer, are displayed darker than the parts close to the viewer. This trick can improve the 3D impression of the rendered scene. Check this box to enable depth cueing.

  Depth cueing is highly useful in combination with **maximum intensity projection (MIP)**. Usually, MIP images lack depth information. Depth cueing solves this problem.

- **Stereo:** daime supports real-time stereo image (anaglyph) rendering. When stereo rendering is enabled, use **red/green glasses** to view the scene (the red filter must be in front of your right eye). Note that the rendering speed will decrease. Adjust the **Angle** to optimize the stereo effect: larger angles result in a stronger stereo effect.

- **Background:** Click the **color button** to change the background color of the 3D scene.
Lighting and Shadows

The settings described here affect the entire 3D scene. Additional parameters for lighting and shadows can be adjusted for each member of a visualization session.

- **Local light:** Local lighting uses one light source that shines from a defined direction. To enable local lighting, check Local light. To change the position of the local light source, use the mouse (left button pushed down) to move the bright spot on the red sphere. Use slider Light source intensity to adjust the brightness of the light source, and use slider Ambient light intensity to change the brightness of the ambient light.

- **Hemisphere light:** Hemisphere lighting virtually splits the 3D scene into “sky” and “ground”, where the sky is brightly illuminated and the ground is darker. To enable hemisphere lighting, check Hemisphere light. To move the illuminated area (the “sky”), use the mouse (left button pushed down) to move the illuminated region on the blue sphere. Use sliders Sky light intensity and Ground light intensity to adjust the brightness of the “sky” and the “ground”, respectively.

⚠️ Lighting can slow down 3D rendering.

- **Shadows:** Choose Cast shadows to switch on shadows that affect all objects in the scene, depending on the current position of the light source. Choose Cast shadows (background only) to switch on shadows that fall onto the background but not onto objects. The background color must be bright enough in order to see these shadows. Choose Shadows with local light or Shadows with hemisphere light to determine whether the shadows depend on the local light source or the hemisphere light source, respectively. Choose No shadows to switch off shadows (the default).

⚠️ Objects cast shadows only onto other objects in the same z-stack. If two (or more) z-stacks are visualized together, this limitation is overcome by combining these z-stacks in the session. Within a combination, objects also cast shadows onto other objects in a different z-stack.

ℹ️ Shadows are shown even if the local and the hemisphere lights are both switched off (but shadows are enabled). Thus, the better “3D effect” of shadows can be utilized with or without additional lighting effects.

⚠️ Shadows slow down 3D rendering.
Figure 12. Lighting effects and shadows. A. Local lighting. B. Hemisphere lighting. C. Like (A), but with shadows enabled. D. Shadows on the background only.

Rendering Settings for Session Members

Various 3D rendering settings can be adjusted individually for each member of a visualization session.

The settings are found in the “Members” tab of the Visualizer tool area. This tab contains a list of all members of the current session. Select a member to change settings by using the other tools of the “Members” tab.

Color, Intensity, and Opacity

- Click at the color bar or click the color button to change the color of the selected session member. Note: Only grayscale and binary images are displayed in the chosen color. Color images are always displayed in their original colors.
- **Intensity factor**: Change this value to display a session member brighter (values > 1) or darker (values < 1). Value 1 means that the displayed brightness equals the original voxel intensities.

- **Intensity cutoff**: Voxels that are darker than this value are not rendered. This option can be useful to suppress dim noise or background. The cutoff can also be used to mimic isosurface rendering, with the intensity cutoff resembling the isovalue (refer to literature on 3D visualization for more information).

- **Opacity**: 100% opacity means that all voxels appear “solid”. A lower opacity means that voxels are rendered semi-transparent and objects, which are behind these voxels, can shine through. Changes of this setting affect all voxels to the same extent.

- **Opacity cutoff**: A value <100% results in semi-transparent rendering, but in contrast to option **Opacity**, brighter voxels are rendered more transparent than darker voxels. This approach can create transparency effects where only the surfaces of objects are displayed and other objects shine through.

- **Render**: Check or uncheck this box to show or hide the currently selected session member in the 3D scene.

- **Apply clipping**: This switch determines whether the selected session member is subject to clipping.

**Transfer Functions**

Volume rendering uses mathematical operations to modulate the input image data for visualization. These **transfer functions** are an essential component of the volume rendering pipeline. The visual appearance of a 3D scene can change dramatically when the transfer function is changed.

- **Transfer function**: Choose the transfer function for the selected session member.

- **Additional parameters** can be adjusted for some transfer functions. A description of these parameters would have to cover transfer function mathematics and the implementation in daime. This would be beyond the scope of this manual. Just play with these settings and observe the effects.

### Table 11. Transfer functions offered by daime for 3D volume rendering.

<table>
<thead>
<tr>
<th>Transfer function</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opacity mapping</td>
<td>A voxel’s brightness determines its opacity in the 3D scene. This is the default transfer function.</td>
</tr>
<tr>
<td>Maximum</td>
<td><strong>Maximum intensity projection</strong> shows only the brightest voxels in the current viewing direction. Depth information is lost but can be restored by depth cueing.</td>
</tr>
<tr>
<td>Hull</td>
<td>Objects are rendered semi-transparent. Outer and inner surfaces (&quot;hulls&quot;) are rendered more opaque than non-surface regions. Hence, implicit surfaces of objects become visible. Use this to visualize both outer and inner surfaces of objects.</td>
</tr>
<tr>
<td>Raw</td>
<td>This means that no transfer function is used. The result can look ugly! However, this allows one to see how transfer functions modulate the raw image data. It is a control and can be useful to spot any visualization biases caused by the transfer functions.</td>
</tr>
</tbody>
</table>
Lighting and Shading

In addition to the lighting options for the whole scene, settings for lighting and shading can be adjusted for individual session members.

- **Local lighting** and **Hemisphere lighting**: These switches determine whether local and hemisphere lighting are applied to the selected session member. **Note**: The respective lighting mode must first be switched on for the entire scene, or these switches do not have any visible effect.

- **Shininess** and **Highlight size**: These settings affect only local lighting. Increase shininess and the surfaces of objects will look smoother (or more “slimy”, can be nice for rendering biofilms). The highlight size determines the size of the reflected light spots on the surfaces of objects. Local lighting must be switched on for these settings to have a visible effect.

- **Shadow darkness**: This setting has an effect only if shadows are enabled. It controls how dark the shadows appear on objects and on the background.

Clipping

Internal regions of complex structures can be visualized by removing the obstructing outer parts. This is called **clipping**.

daime offers two different clipping modes:

1. **Quick, interactive clipping**. Up to six clipping planes can be defined. Each clipping plane divides the 3D space into two regions, and the content of one region is clipped away. The result is shown immediately (Figure 13). The clipping planes can be moved and rotated. Although parts of the 3D scene are clipped away, the z-stacks (i.e., the actual image data) are not modified.

2. **Permanent, non-interactive clipping**. An arbitrary number of clipping planes or clipping boxes can be defined. These clipping objects can be moved and rotated interactively, but the results of the clipping process are not shown immediately. Instead, clipping must be manually triggered. The z-stacks are modified: image regions, which are clipped away, are permanently removed.
Figure 13. A. Bacterial cell aggregate, which was stained by a fluorescent marker, visualized in 3D. B. The same aggregate “cut open” by a clipping plane (shown as a yellow frame). Cavities inside the cell aggregate are displayed in green. C. The same cell aggregate clipped by two clipping planes. The second clipping plane is orthogonal to the first one.

Click this button or choose Render:Clipping in the Visualizer menu to use the clipping tools. The tools will appear in the Visualizer tool area. Similar tools are used to control interactive clipping (I-Clip) and non-interactive clipping (NI-Clip).

- **New:** Click this button to create a new clipping object. All created clipping objects are shown in the 3D scene and appear in list Object. Most of the clipping tools work with the clipping object that is currently selected in this list. The currently selected clipping object has a **yellow frame**.
- **Kill:** Click this button to remove the currently selected clipping object.
- **Clip:** Click this button to clip the image data. This button is needed only for non-interactive clipping. In the following dialog, select the session members to be clipped. **Note:** This will modify the image data.
- Use the wireframe sphere to rotate the currently selected clipping object. Click with the **left mouse button** at the sphere, keep the button pushed down, and move the mouse to rotate the sphere. The clipping object will follow this movement. Release the mouse button to stop the rotation.
- Alternatively, use the **X/Y/Z dials** to rotate the currently selected clipping object around the respective axis of a 3D Cartesian coordinate system. **Turn the dials with the mouse.** To rotate the clipping object in exact **steps of 90°**, first click with the **left mouse button** at a dial, and then use the PAGE UP and PAGE DOWN keys to rotate the object.
- Use the **X/Y/Z position sliders** to move the currently selected clipping object along the respective axis of a 3D Cartesian coordinate system. Use the **X/Y/Z size sliders** to change the size of the clipping object in the respective direction. **Note:** As clipping planes are by definition infinite, these sliders are disabled if the currently selected clipping object is a plane.
- **Reverse orientation:** This switch flips the clipped part of the 3D scene.
- Check/uncheck **Object clips** to enable or disable the currently selected clipping object.
Clipping objects are shown in the 3D scene only when the clipping tools are visible. When these tools are not visible, the defined clipping objects are not shown but still exist. Interactive clipping is still effective, too.

Clipping can be enabled or disabled for individual members of the visualization session.

Non-interactive clipping only: Each clipping object has two sides. One side faces the parts of the scene that will be clipped away (this side is displayed in red). The other side faces the parts of the scene that will remain (this side is displayed in green). By default, all outer sides of a clipping box are green and all inner sides are red. Thus, the part of the scene that is enclosed by the box will be removed. Check Reverse orientation to flip the sides of the currently selected clipping object (green sides will turn red and vice versa). For a clipping box this means that the part of the scene, which is not enclosed by the box, will be removed and everything inside the box will remain. Use more than one clipping object to define complex clipping geometries. Those parts of the 3D scene, which face only red sides of clipping objects, will be removed. All other parts, which face at least one green side of any clipping object, will remain.

Keyframes and Animations

The Visualizer contains a convenient tool, the Keyframe Editor, for creating animations. These animations can be played back in the Visualizer, or they can be exported to disk as movie files.

A keyframe stores the current state of a visualization session (e.g., the rotation of the scene, the viewer’s position, all rendering settings, clipping objects, etc.). A sequence of two or more different keyframes defines an animation sequence.

Even a complex animation can be defined with just a few keyframes. When playing back an animation or exporting a movie file, daime interpolates all animation steps between the keyframes. This applies not only to the rotation of the 3D scene and the viewer position, but also to many other visualization settings. For example, different colors of a session member in adjacent keyframes are interpolated as a gradual color change in the resulting animation.

The Keyframe Editor can be used in both 2D and 3D visualization modes.

Click this button or choose Render:Keyframes and animation in the Visualizer menu to open the Keyframe Editor. It will appear in the Visualizer tool area.

- **Keyframes:** This list contains all currently defined keyframes.
- **Add:** Creates a new keyframe and appends it to the list. First, make all necessary adjustments in the Visualizer to render the 2D or 3D scene. Then click Add.
• **Arrow up/down buttons:** Move the currently selected keyframe up or down in the list. **Note:** The order of the keyframes in the list determines the animation sequence.

• **Restore:** Click this button to restore the currently selected keyframe. The Visualizer will then render this keyframe. **Note:** The previously shown scene is lost, unless it has also been added as a keyframe (or saved as a session file to disk).

• **Replace:** Replaces the keyframe, which is currently selected in the list, with the current scene displayed by the Visualizer.

• **Delete:** Removes the currently selected keyframe from the list.

• **Rename:** Renames the currently selected keyframe. The new name must first be entered in the edit field next to the button.

• **Save and Load:** Saves all keyframes to disk, or loads them from disk. Click **Save**, then choose an **empty folder** on disk. All keyframes will be saved into this folder. To load keyframes from disk, click **Load** and choose a folder on disk that contains the keyframes to be loaded. **Note:** Loading keyframes from disk will overwrite all currently defined keyframes.

• **First/last keyframe:** Defines the first (=start) and last (=end) keyframe for the animation sequence.

• **Interpolation:** Options to modify how the animation between two keyframes is interpolated. Default is linear interpolation. The selected option will affect the currently selected and the following keyframe.

• **Frames/sec:** Frames per second in the exported movie. Higher values yield smoother animations but larger movie files.

• **Current to next keyframe sec:** Time (in seconds) between the currently selected and the following keyframe. Can be different between pairs of keyframes to accelerate or slow down the animation. **Note:** This setting will apply to an exported movie, but not if the animation is played from within **daime**.

• **All:** Sets the same entered time (in seconds) between all keyframes. The resulting duration in **Total sec** is calculated automatically.

• **Total sec:** The duration (in seconds) of the whole exported movie.

• **Set:** Adjusts the time between all keyframes such that the sum of all seconds equals the duration entered in field **Total sec**.

• **Play/Pause/Stop:** Use these buttons to play back the animation in **daime**. Check **Slow-motion** to slow down the animation. **Note:** The time needed to render and play the whole animation sequence will likely differ from the value in **Total sec**. That value applies to an exported movie, not to playback within **daime**.

• **Movie:** Exports the animation as a movie file to disk. The resulting movie can be watched with external movie player software.

⚠️ It is not possible to save two different sets of keyframes in the same folder. Always make sure that keyframes are saved into an empty folder. The file names of keyframes should never be changed manually. Otherwise, **daime** cannot load them again.
When exporting a movie, daime renders the animation sequence and saves it as a sequence of TIFF images into the selected folder. Subsequently, the external program ffmpeg is used to create the movie file. Button Movie is disabled if ffmpeg was not found. On Windows, the daime installation package already contains ffmpeg. On Linux, ffmpeg must be installed separately.

Advanced Rendering Settings and Troubleshooting

Advanced Rendering Settings

daime uses special data structures (3D textures) to prepare image data for volume rendering. The internal format of these textures can be changed by choosing Render:Advanced rendering settings in the Visualizer menu. In 99% of all cases, the default settings will be OK. In principle, 3D texture formats with more bits per voxel provide more colors or levels of gray. Whether choosing a texture format with more bits per voxel really improves the quality of the rendered images depends largely on the graphics hardware; not all graphics cards can handle all 3D texture formats (in such a case, daime silently overrides the user’s choice and selects a compatible format). Consider also that 3D textures with, for example, 32 bits per voxel need much more video memory than textures with only 8 bits per voxel. If the size of a 3D texture exceeds the available video memory, daime will silently scale down the texture until it fits.

Compress textures should save video memory, but this does not work with all graphics hardware. Uncheck this option if it causes problems, such as wrong or no colors displayed in the 3D scene.

Troubleshooting

- **Problem:** Snapshots appear upside down. **Solution:** Choose Render:Preferences in the Visualizer menu and check Vertically flip snapshots.
- **Problem:** Snapshots do not work. They are black, contain only noise, or show only parts of the 3D scene. **Solution:** First take a new snapshot. If the problem remains, choose Render:Preferences in the Visualizer menu and check Take snapshots as screenshots. **Note:** Snapshots may now be incomplete if the Visualizer window extends the borders of the screen.
- **Problem:** The display area updates very slowly (or not at all) when the Visualizer was covered by other windows. Or the 3D scene appears only in a part of the display area, while the remaining parts of the display area are black or show noise. **Solution:** These problems are unspecific and can be hard to solve. Make sure that you have installed the latest driver for your graphics hardware, and that this driver is compatible with your operating system (e.g., do not use an old Windows 7 graphics driver on a Windows 10 system).
- **Problem:** The rendered 3D scene seems to lack detail, although the z-stack has a high voxel resolution. **Solution:** Choose Render:Preferences and uncheck Texture
size limit. This can improve image quality, but 3D rendering might also fail, because the larger 3D textures need more video memory.

- **Problem:** The option for switching to 3D visualization mode is not available in the Visualizer menu. **Solution:** The hardware does not support 3D volume rendering as it is implemented in daime. Use another computer that meets the system requirements.

### Simple Projections of z-Stacks

In addition to the Visualizer, daime contains a simple feature to quickly create 2D projections from 3D z-stacks. These projections have a fixed viewer perspective and are not rotated. However, they can be created even on computers that lack any 3D graphics hardware.

Mark a z-stack in the main window’s list, then choose **View:2D Projection** in the main menu. The following dialog contains two image display areas. The left area shows the z-stack, whereas the right area (initially empty) shows the projection. Next to the right area, a menu offers different projection modes.

Choose a projection mode, then click **Apply**. The projection will appear in the right display area. To test different projection modes, just select another mode and click **Apply** again. Click **OK** to accept the projection. The projection will appear as a new entry in the main window’s list. The name of this new image is the name of the original z-stack plus a suffix, which indicates the applied projection mode: _over for Overlay, _max for Maximum intensity, _min for Minimum intensity, and _extfoc for Extended focus.

A projection can be **saved to disk as an STK file**, or as a TIFF file by **exporting the image**.

<table>
<thead>
<tr>
<th>Overlay</th>
<th>The images of the z-stack are superimposed and blended. Darker voxels are drawn more transparent than brighter voxels.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum intensity</td>
<td>At every (x,y) coordinate of the projection, only the brightest voxel (along the z-axis of the stack) is shown.</td>
</tr>
<tr>
<td>Minimum intensity</td>
<td>Similar to Maximum intensity, but here only the darkest voxels are shown. This can be useful for projecting z-stacks that were recorded by brightfield or phase contrast microscopy (e.g., the MAR signals in FISH-MAR experiments).</td>
</tr>
<tr>
<td>Extended focus</td>
<td>z-stacks, which were recorded by using non-confocal brightfield or phase contrast microscopy, often contain out-of-focus blur. This mode tries to eliminate blur by projecting only the sharp regions of each image. It is not suitable for fluorescence images.</td>
</tr>
</tbody>
</table>
The Object Editor

After image segmentation and prior to image analysis, the success of segmentation must be checked. Moreover, in many cases only a subset of the objects in the images should be analyzed. Thus, it must be possible to select objects for analysis (and reject others). In *daime*, these and other tasks can be accomplished by using the tools of the **Object Editor**. Its main features are:

- Works with objects detected by [automatic image segmentation](#), and also contains tools for defining new objects manually.
- Semi-automatic image segmentation based on object colors.
- Interactive selection of objects to be analyzed (and rejection of other objects).
- Interactive analysis of individual objects: measurement of object features such as size, brightness, shape descriptors, etc.
- Classification of objects based on selected features. Selection and rejection of whole object classes for further analysis.
- Tools for refining image segmentation (e.g., to reject very small or dark objects, split touching objects, fill holes in objects).
- All options work with 2D and 3D objects.
- Seamless integration in the [Visualizer](#).

Please read the sections on the [Visualizer and 2D/3D visualization](#). Much of this information is needed for using the Object Editor efficiently.

The Object Editor provides access to segmented objects in all members of a Visualization session.

How to Open the Object Editor

Click this button or choose **Render:Object Editor** in the Visualizer menu to open the Object Editor. It will appear in the Visualizer tool area.

If none of the images in the current visualization session is already segmented, a dialog asks for the segmentation mode (2D or 3D). The chosen segmentation mode will be applied when tools of the Object Editor are used to define objects manually.

The Object Editor is only available if all session members are segmented in the same mode (2D or 3D). If 2D- and 3D-segmented images are mixed, the Object Editor is disabled.

Some tools of the Object Editor are only available if the session contains at least two segmented members.
Tools for Manual Segmentation and Object Editing

Single Object Selection, Rejection, and Custom Colors

When the Object Editor is visible, the Visualizer marks segmented objects with colored borders (2D mode) or boxes (3D mode). Objects surrounded by an **orange** border (or box) are *selected* for image analysis. Objects surrounded by a **purple** border (or box) are *rejected* and will be ignored by the image analysis functions. **Non-object regions** are not surrounded by a border (or box). See *Figure 14.*

*i* By default, all objects in newly segmented images are selected.

---

**Figure 14.** A. Selected and rejected objects (2D visualization mode). B. Selected and rejected objects in a 3D-segmented z-stack (3D visualization mode). C. One 3D object is highlighted, and a context menu for this object is shown.

- **To select or reject an object,** click with the **left mouse button** at this object (2D mode) or push the **CTRL key** down and click with the **left mouse button** at the object (3D mode). Repeat this to toggle the selection/rejection state of the object.

- **To pick an object and open a context menu for the object,** click with the **right mouse button** at the object. The object will be highlighted (only in 3D mode) and a context menu will appear (in 2D and 3D modes, *Figure 14*). This menu contains options for measuring various features of the object and for editing attributes of the object (e.g., its color).

*i* In 3D mode, objects can be selected/rejected or picked (for the context menu) from any viewing perspective. However, **perspective projection** is not compatible with this. Always switch to parallel projection to select, reject, or pick 3D objects.

- **Object counts and size:** This field shows the total number of objects, as well as the numbers of selected and rejected objects in the currently displayed image (2D mode) or whole z-stack (3D mode). It also shows the size (in pixels or voxels) of the object that is currently below the mouse cursor (2D mode). A size of zero is displayed when the mouse cursor is not pointing at any object.
- **Object colors:** Each object can be displayed in a user-defined custom color. Click this button to choose a custom color, which will then be assigned to all currently selected objects. Click this button to automatically assign different custom colors to all objects. This tool is useful for checking the results of image segmentation, especially whether touching objects have been split correctly. Click this button to reset the custom colors of all objects to white. Check/uncheck **Show object colors** to toggle the display of custom object colors by the Visualizer. Check/uncheck **Show object borders** to toggle the display of orange or purple borders and boxes around objects.

To set the custom color of a single object, pick the object (see above) and choose **Edit object attributes** in the context menu. Then select a color and check option **Custom color** (the custom color of the object will not be displayed if this option is unchecked – use it to toggle the display of custom colors for individual objects).

**Selection, Rejection, and Editing of Multiple Objects**

- If images are 2D-segmented, the tools described here affect only the objects in the currently displayed image (of a batch) by default. Check **In all images** to apply the tools simultaneously to all images of a 2D-segmented batch. In the same way, the **In all images** option changes the behavior the **custom color tools** and the **magic wand tools**. In contrast, this option is not available (and would be irrelevant) for 3D-segmented z-stacks, because 3D objects span multiple images in the z dimension.

**Table 13. Tools of the Object Editor for handling multiple objects.**

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Select tool]</td>
<td>Selects all objects for image analysis.</td>
</tr>
<tr>
<td>![Reject tool]</td>
<td>Rejects all objects for image analysis. All image analysis functions ignore rejected objects.</td>
</tr>
<tr>
<td>![Toggles tool]</td>
<td>Toggles the selection state of all objects.</td>
</tr>
<tr>
<td>![Reject borders tool]</td>
<td>Rejects all objects that touch the borders of an image (in 2D segmentation mode) or of the z-stack (in 3D segmentation mode).</td>
</tr>
<tr>
<td>![Smallest size range tool]</td>
<td>Switches to mode: select the smallest object of a size range (see description below).</td>
</tr>
<tr>
<td>![Largest size range tool]</td>
<td>Switches to mode: select the largest object of a size range (see description below).</td>
</tr>
<tr>
<td>![Smallest object tool]</td>
<td>Finds and selects the smallest object (does not change the selection state of all other objects). See description below.</td>
</tr>
<tr>
<td>![Largest object tool]</td>
<td>Finds and selects the largest object (does not change the selection state of all other objects). See description below.</td>
</tr>
<tr>
<td>Tool</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td><img src="image.png" alt="Unregister rejected objects" /></td>
<td>Unregisters all rejected objects (= removes their segmentation data). These objects will become unsegmented regions. Use to exclude noise and artifacts from analyses. No data are deleted from the image layer (primary image data is not modified).</td>
</tr>
<tr>
<td><img src="image.png" alt="Unregister very dark objects" /></td>
<td>Unregisters all very dark objects (= removes their segmentation data). These objects will become unsegmented regions. Use to exclude dark objects (e.g., noise) from analyses. No data are deleted from the image layer (primary image data is not modified).</td>
</tr>
<tr>
<td><img src="image.png" alt="Unregister very small objects" /></td>
<td>Unregisters all very small objects (= removes their segmentation data). These objects will become unsegmented regions. Use to exclude small noise and artifacts from analyses. No data are deleted from the image layer (primary image data is not modified).</td>
</tr>
<tr>
<td><img src="image.png" alt="Permanently delete rejected objects" /></td>
<td>Permanently deletes all rejected objects. <strong>Attention: This will selectively change parts of the images and modify primary image data.</strong> Use only for removing noise or artifacts and if scientifically justified. A warning message pops up when this button is clicked. This tool affects both the image layer and the object layer. It is highlighted red in the action log. In case of doubt, use one of the other tools that do not modify primary image data.</td>
</tr>
<tr>
<td><img src="image.png" alt="Permanently delete pixels" /></td>
<td>Permanently deletes all pixels (voxels) that do not belong to any segmented object. <strong>Attention: This will selectively change parts of the images and modify primary image data.</strong> Use only for removing noise or artifacts and if scientifically justified. A warning message pops up when this button is clicked. This action is highlighted red in the action log. In case of doubt, use one of the other tools that do not modify primary image data.</td>
</tr>
<tr>
<td><img src="image.png" alt="Merge objects" /></td>
<td>Merges the segmentation data of all selected objects to form one larger object.</td>
</tr>
<tr>
<td><img src="image.png" alt="Fill holes" /></td>
<td>Fills internal holes in all selected objects (see description below).</td>
</tr>
<tr>
<td><img src="image.png" alt="Split objects" /></td>
<td>Splits all selected touching objects and defines new objects when necessary. Use this feature to separate adjacent cells or cell clusters, which have erroneously been segmented as one object. See description below.</td>
</tr>
</tbody>
</table>

### How to select objects based on their size

1. Reject all objects in the image. If needed, check **In all images** first.
2. ![Select smallest object](image.png) Click this button, then click at the smallest object that should be selected. Alternatively, click ![Select smallest object](image.png) and let the program find the smallest object.
3. ![Select largest object](image.png) This button should now **have automatically been pushed down** (= active mode for selecting the largest object). Click at the largest object that should be selected. Alternatively, click ![Select largest object](image.png) and let the program find the largest object.
4. *daime* will select all objects whose size is between the chosen smallest and largest objects.
5. **Note:** The order can be switched (first select the largest, then the smallest object).

*For example, these tools can be used to select only single microbial cells and to reject larger cell aggregates.*
How to fill internal holes in objects

By default, the segmentation functions exclude internal holes (= dark regions inside of objects) from the object definitions. Click this button to include internal holes in the definitions of the currently selected objects. If needed, check In all images first. The image analysis functions will treat included holes as if they were solid material. Note: Filling holes in large 3D objects can take some time. Cavities that extend to the surface of an object are invaginations, not internal holes, and are ignored by this tool.

How to split touching objects

1. Click this button. If needed, check In all images first.
2. A parameter dialog opens. In most cases, the default values work well. However, if the objects are not correctly split then try to improve the result by changing these values:
   a. Increase the numbers of Erosions or Iterations if large objects have not been split with the default values. Decrease these numbers if small objects have not been split.
   b. Increase the Diameter only if the objects to be split are relatively large (compared to the total image size).
3. Click this button to check the results of this function by displaying all objects in different colors.

This function is based on the Morphological Multiscale Algorithm described in (5). Please note that object splitting can be a time-consuming process and may not always yield the expected results.

Object Editing and Definition Based on a Region of Interest

A region of interest (ROI) is a user-defined region of an image. Objects located within an ROI can be selected or rejected. It is also possible to define the content of an ROI as a new object. In this version of daima, an ROI is always 2D even if the images are 3D-segmented. An image can contain only one ROI at any time.

Table 14. Tools of the Object Editor for working with regions of interest.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Rectangle" /></td>
<td>Defines a rectangular ROI. Click this button, then click the left mouse button in the image to define a corner of the ROI. Release the mouse button. Move the mouse to adjust the size of the ROI. Finally, click the right mouse button to define the ROI.</td>
</tr>
<tr>
<td><img src="image" alt="Circle" /></td>
<td>Defines a circular ROI. Click this button, then click the left mouse button in the image to define the center of the ROI. Release the mouse button. Move the mouse to adjust the size of the ROI. Finally, click the right mouse button to define the ROI.</td>
</tr>
</tbody>
</table>
| | Defines a polygonal ROI. Click this button, then **click the left mouse button** in the image to define a corner of the ROI. **Release the mouse button.** Move the mouse to drag a line to the next intended corner of the ROI, and then **click the left mouse button** to define this corner. **Click the right mouse button** to finish and define the ROI (this will draw a straight line from the current mouse cursor position to the first defined corner).

| | Defines an ROI of arbitrary shape. Click this button, then **click the left mouse button** in the image to define a point of the ROI border. **Release the mouse button.** Move the mouse to draw the border of the ROI. **Click the right mouse button** to finish and define the ROI (this will draw a straight line from the current mouse cursor position to the first defined point).

| | Deletes the ROI in the currently displayed image.

| | Toggles the selection state of all objects inside (or outside) the ROI. Click this button, then **click with the left mouse button** at any point either inside or outside the ROI.

| | Use this tool to detect and define new objects inside (or outside) the ROI. First, **select an intensity thresholding method** or "None" from the menu next to the button. Now click the tool button. Finally, **click with the left mouse button** at any point either inside or outside the ROI. The chosen thresholding method will be used to detect objects located inside or outside the ROI, respectively. If "None" was chosen, the whole ROI or the whole region outside the ROI will become one object (including background). **Note:** If the ROI contains multiple objects (e.g., cells), this tool will not distinguish them. It will merge them in one defined object. Use the tool for splitting objects (see above) to separate them.

---

The border of an ROI will never split previously defined objects. Any already existing segmented objects, which are located on an ROI border, will become part of the ROI border.

**Magic Wand Tools for Semi-Automatic Object Definition and Color-based Segmentation**

The **magic wand tools** are used to manually or semi-automatically define new objects. *daime* contains two magic wand tools, one that uses pixel intensities (brightness) and another one that uses color information to detect objects.

The magic wand tools are not available if a visualization session contains more than one member, which are displayed by the Visualizer. In this case, **display only one session member** (the one to be edited) in order to use the magic wand tools.

Do not click at the image background when a magic wand tool is being used in 3D segmentation mode. This would define (almost) the whole volume of the z-stack as one huge object. This can take a long time, and *daime* may even run out of memory and crash.

**How to use the intensity-based magic wand tool**

**Note:** This tool can be used with grayscale and color images. It cannot be used with binary images (**object masks**).

1. **Click this button (it will remain pushed down).**
2. **Click with the left mouse button** at any representative point that belongs to the new object to be defined. In 3D mode, **push the CTRL key down and click with the left mouse button** at the point.

3. **daime** will try to identify all pixels (voxels) that belong to the object. Depending on the current segmentation mode, this will be done in 2D or in 3D.

4. If the result is not satisfying, **unregister the new object** and adjust the **Tolerance** (see below). Then try again.

Object pixels (voxels) are identified based on their location relative to the first selected point and based on their intensity (brightness). Only pixels (voxels), whose intensity is similar to that of the first selected point, will belong to the new object. This behavior can be adjusted by changing the **Tolerance** parameter. A higher tolerance allows larger intensity differences among pixels (voxels) of the same object.

If this tool is applied to color images, it ignores the colors of the processed pixels – only their brightness is taken into account.

**How to use the color-based magic wand tool**

**Note:** This is the primary tool **daime** offers for image segmentation based on colors. It can only be used with color images.

1. ![Click this button (it will remain pushed down).](image)
2. **Click with the left mouse button** at any representative point that belongs to the new object to be defined. In 3D mode, **push the CTRL key down and click with the left mouse button** at the point.
3. **daime** will try to identify all pixels (voxels) that belong to the object. Depending on the current segmentation mode, this will be done in 2D or in 3D.
4. If the result is not satisfying, **unregister the new object** and adjust the **Tolerance** (see below). Then try again.

Object pixels (voxels) are identified based on their location relative to the first selected point and based on their color (but not on their intensity). For instance, bright red and dark red pixels can belong to the same object. Adjust the **Tolerance** parameter to determine how strictly **daime** evaluates colors. A higher tolerance allows larger color differences among pixels (voxels) of the same object.

**How to use the magic wand tools to define multiple objects (in multiple images)**

The procedures described above will define only one object each time. However, the magic wand tools can be used to detect all objects with a similar brightness or color at once.

1. Check option **In whole image.** The magic wand tools will now detect similar objects in the currently displayed image (2D mode) or in the whole z-stack (3D mode).
2. In 2D mode: if needed, check also option **In all images**. The magic wand tools will then detect similar objects not only in the currently displayed but also in all other images of a batch. **Note:** This option is not available (and would not make sense) in 3D mode.

3. Proceed as described above for the intensity-based and the color-based magic wand tools. **Note:** In both cases, only one point must be selected. All objects will be defined based on the intensity or color of that point and the **Tolerance** parameter.

**Tools for Two (or More) Segmented Session Members**

If a **visualization session** contains two (or more) segmented members, the tools of the Object Editor work on all displayed session members simultaneously. For instance, click ✪ to select all objects in all session members. An exception is the **magic wand tools**, which are not available if more than one session member is being displayed. When the display area of the Visualizer is split in **2D visualization mode**, the Object Editor tools still work on all displayed session members simultaneously.

The tools of the Object Editor, which are described in this section, are available only if the visualization session contains at least two segmented members (which are also being displayed).

**The Multiple Stains Tool**

Imagine two different, 2D-segmented images. One image shows bacterial cells stained by DAPI (or another unspecific dye), whereas the other image shows cells stained by a specific FISH probe in the same field of view. Not all DAPI-stained cells are labeled by the specific FISH probe, but all cells labeled by FISH are also stained by DAPI.

Click this button to select only the cells that were labeled by both markers. This tool inspects all segmented objects in all displayed members of the visualization session. Objects that occur in all members at the same position are selected, whereas all other objects are rejected. In our example, this tool would select cells that occur in both the DAPI and the FISH images. Check **In all images** to apply this tool to all images in 2D-segmented batches simultaneously.

After using this tool, it is also very easy to select only the mono-labeled cells. Simply click ✪ this button.

This tool works also with three or more session members. Objects to be selected must occur in all images at the same position, but do not need to have exactly the same size and shape. It is sufficient that the objects overlap partially.

The tool works with 3D-segmented images, too. 3D objects in the different session members must spatially overlap to be selected.
The Artifact Rejection Tool
This tool is explained in the section about biovolume fraction quantification.

Tools for Object Classification
The Object Editor contains tools for classifying objects based on features such as size, perimeter, diameter, brightness, etc. Classification is used to group objects for subsequent analyses. For example, if cocci and rod-shaped bacteria must be separately analyzed in a large number of images, classification can make this task much easier.

Each object in a segmented image belongs to one class. The default class of all objects is called unclassified, indicating that the objects do not yet belong to any user-defined class. daim offers automatic classification tools that create new classes and assign objects to these classes. Alternatively, individual objects can be manually assigned to classes. Object classes can be manipulated or analyzed by selecting (or rejecting) all objects that belong to the same class.

To use the classification tools, switch to the “Classes” tab of the Object Editor.

In 2D segmentation mode, the classification tools work by default only with classes and objects in the currently displayed image. Check In all images to apply the tools simultaneously to all images of a 2D-segmented batch. In 3D segmentation mode, the tools always work with classes and objects in the whole z-stack.

- Class list: This list shows all classes defined in the images.
- Class name: This field contains the name of the class that is currently selected in the class list. The field is used to enter a new class name when the New from sel. or the Rename class buttons are pushed (see below).
- Classify objects: Click this button to start semi-automated object classification.
- Delete all classes: Click this button to delete all current class definitions. Note: The actual objects are not deleted. Merely their classification state is reset to unclassified.
- Class manipulation tools: These tools work with already defined classes. First select a class in the class list, then use a class manipulation tool.

Table 15. Class manipulation tools of the Object Editor.

<table>
<thead>
<tr>
<th>Tool (Button)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select</td>
<td>Selects all objects that belong to the class, which has been chosen in the class list. Does not change the selection state of other objects.</td>
</tr>
<tr>
<td>Select (excl.)</td>
<td>Selects all objects that belong to the class, which has been chosen in the class list. Rejects all other objects.</td>
</tr>
<tr>
<td>Reject</td>
<td>Rejects all objects that belong to the class, which has been chosen in the class list. Does not change the selection state of other objects.</td>
</tr>
<tr>
<td>Reject (excl.)</td>
<td>Rejects all objects that belong to the class, which has been chosen in the class list. Selects all other objects.</td>
</tr>
<tr>
<td><strong>Delete class</strong></td>
<td>Deletes the class, which has been chosen in the class list. Does not delete any objects, but only removes the class definition. The objects are set to <em>unclassified</em>.</td>
</tr>
<tr>
<td><strong>Rename class</strong></td>
<td>Renames the class, which has been chosen in the class list. First enter the new name in field <strong>Class name</strong>, then click this button.</td>
</tr>
<tr>
<td><strong>New from sel.</strong></td>
<td>Use this tool to manually create a new class. Select all objects that should belong to the new class, reject all others, then enter the name of the new class in field <strong>Class name</strong>. Now click this button. A new class will be created that contains the selected objects.</td>
</tr>
</tbody>
</table>

### Semi-Automated Classification

Push button **Classify objects** of the “Classes” **tab** to access the tools for semi-automated classification.

*daime* offers three methods for semi-automated object classification:

**Method 1: Based on template objects.** The user selects one (or more) objects as **templates**. *daime* finds objects, which are similar to the templates, and assigns them to one and the same new class (together with the templates). The user must also provide a name for the new class. The similar objects are found by measuring and comparing selected object features (size, brightness, etc.). The success of classification depends on the templates and on the features chosen for comparing other objects to these templates.

**Method 2: Based on user-defined values.** The user chooses the object features to be measured and evaluated for classification (size, brightness, etc.). The user also defines a range of values for each feature. All objects, which fall into the range defined for each feature, will be assigned to one and the same new class. The user must also provide a name for the new class. The success of classification depends on the chosen features and on the user-defined range for each feature.

**Method 3: Automatic.** The user chooses the object features to be measured and evaluated for classification (size, brightness, etc.). The user also indicates a number \((n)\) of new classes to be created. Finally, the user indicates whether the objects should be distributed among these classes as uniformly or as unequally as possible. Among the selected object features, *daime* will choose the one feature that satisfies this condition best. Based on this chosen feature, the objects will be grouped into \(n\) automatically generated classes. The success of classification depends on the selected features and the distribution condition.

Methods 2 and 3 consider only selected objects for classification. Rejected objects are always ignored. Method 1 considers all objects, and only the template object(s) must be selected (see below for details).

If previously classified objects are classified again, the old classification is overwritten.

### Overview of the Classification Parameters dialog

- **Use:** Check (or uncheck) the boxes in this column to select which object feature(s) should be evaluated for classification. Multiple features can be chosen.
• **Object Feature**: The object features that are available for classification. They are described in the section about measuring object features.

• **Condition**: The options in these menus are used to fine-tune the classification for each selected feature (only methods 1 and 2). An object is assigned to a class if all conditions for all selected features apply. See below for details.

• **Number 1 and Number 2**: Enter values here to indicate value ranges for the selected features. Only methods 1 and 2. See below for details.

• **Tolerance**: The value entered here is used when Condition tolerance has been chosen for a feature. Only methods 1 and 2. See below for details.

• **Classification method**: Selects which of the main classification methods 1-3 (see above and below for details) should be used.

• **Max. no. of classes**: The maximal number of new classes to be created during automatic classification. Only method 3. See below for details.

• **Object distribution**: Indicates how objects should be distributed among the new classes after automatic classification. Only method 3. See below for details.

• **Class name**: The name of the new class must be entered here. Only methods 1 and 2. Method 3 generates new class names automatically (see below for details).

**Workflow for classification based on template objects (method 1)**

1. Switch to the “**Objects**” tab of the Object Editor.
2. Only in **2D segmentation mode**: check **In all images** in order to classify objects in all images of a batch. Uncheck this option in order to classify objects only in the currently displayed image.

3. ![Click this button to reject all objects.](image)
4. **Select** one or more template object(s). The template objects should be similar to each other (e.g., similar size, brightness, shape, etc.).
5. Switch to the “**Classes**” tab of the Object Editor. Click **Classify objects**.
6. Select method **Template object(s)**.
7. **Select one (or more) object features** for classification. These features should be suitable to identify objects that are similar to the template objects.
8. **Choose a condition** for each selected object feature. The condition determines how an object is compared to the template object(s).
   a. **equal**: The feature of the object must be equal to the same feature of the template object. For example, its brightness must be the same as the brightness of the template object. If more than one template object was provided, the feature must be within the range [min..max] of the template objects.
   b. **less**: Like (a), but feature must be less than the same feature of the template object(s).
   c. **less or equal**: Like (a), but feature must be less than or equal to the same feature of the template object(s).
d. **greater**: Like (a), but feature must be greater than the same feature of the template object(s).

e. **greater or equal**: Like (a), but feature must be greater than or equal to the same feature of the template object(s).

f. **tolerance**: The feature of the object must be similar to the same feature of the template object. The comparison is “fuzzy”, that means, a certain degree of dissimilarity is allowed. A high tolerance means that an object can be quite dissimilar compared to the template and still end up in the new class. If more than one template object was provided, the fuzzy comparison is based on the range [min..max] of the template objects.

9. If condition **tolerance** has been chosen, adjust the **Tolerance** value for the respective feature (or use the default value).

10. Enter a name for the new class in field **Class name**.

11. **Click OK**. Classification can take a little while. Afterwards, all objects belonging to the new class will be selected, including the template object(s). The new class will be listed in the **Class list** of the “Classes” tab.

**Workflow for classification based on user-defined values (method 2)**

1. Switch to the “**Objects**” tab of the Object Editor.
2. Only in **2D segmentation mode**: check **In all images** in order to classify objects in all images of a batch. Uncheck this option in order to classify objects only in the currently displayed image.
3. **Select the objects** to be classified. Those objects, which should not be classified, must be rejected.
4. Switch to the “**Classes**” tab of the Object Editor. Click **Classify objects**.
5. Select method **User-defined values**.
6. **Select one (or more) object features** for classification.
7. **Choose a condition** for each selected object feature.
   a. **equal**: The feature of the object must be equal to **Number 1**. For example, its brightness must be the same as the value entered for **Number 1**.
   b. **less**: Like (a), but feature must be less than **Number 1**.
   c. **less or equal**: Like (a), but feature must be less than or equal to **Number 1**.
   d. **greater**: Like (a), but feature must be greater than **Number 1**.
   e. **greater or equal**: Like (a), but feature must be greater than or equal to **Number 1**.
   f. **tolerance**: The feature of the object must be similar to **Number 1**. The comparison is “fuzzy”, that means, a certain degree of difference is allowed. A high tolerance means that the object feature can be quite different from **Number 1** and still end up in the new class.
   g. **between**: The feature of the object must be between **Number 1** and **Number 2**.
8. For each selected object feature, enter a suitable value for **Number 1**. If condition **between** has been chosen, enter also a value for **Number 2** for the respective
feature. If condition *tolerance* has been chosen, adjust the **Tolerance** value for the respective feature (or use the default value).

9. Enter a name for the new class in field **Class name**.

10. **Click OK.** Classification can take a little while. Afterwards, all objects belonging to the new class will be selected. The new class will be listed in the **Class list** of the “Classes” tab.

**Workflow for automatic classification**

1. Switch to the “**Objects**” tab of the Object Editor.

2. Only in **2D segmentation mode**: check **In all images** in order to classify objects in all images of a batch. Uncheck this option in order to classify objects only in the currently displayed image.

3. **Select the objects** to be classified. Those objects, which should not be classified, must be rejected.

4. Switch to the “**Classes**” tab of the Object Editor. **Click Classify objects**.

5. **Select method** **Automatic**.

6. **Select one (or more) object features** for classification.

7. Enter the **Max. no. of classes** to be created (at least 2 classes). Here, we call this number \( n \).

8. Choose **Minimize variance** or **Maximize variance**. During classification, the following steps are performed for each selected object feature:
   a. The objects with the smallest and the largest values for the feature are found. These values define the range \([\text{min}..\text{max}]\) for the feature.
   b. The range is partitioned into \( n \) equal intervals. These intervals define the \( n \) classes for the feature.
   c. Every object is assigned to one of the \( n \) classes based on its value for the feature.
   d. The objects in each of the \( n \) classes are counted. The variance of the object numbers is determined.

9. **Click OK.** Classification can take a little while.

10. At the end, the program will choose one of the selected object features for final classification. If **Minimize variance** was chosen before, this will be the feature with the smallest variance of object counts in the \( n \) classes (= most uniform distribution of objects among the classes). If **Maximize variance** was chosen before, this will be the feature with the largest variance of object counts in the \( n \) classes (= most unequal distribution of objects among the classes). **Note:** “most uniform” does not mean that the distribution of objects must be truly uniform; it just means that it is more uniform than for the other features. Same for unequal.

11. Based on the chosen feature, up to \( n \) new classes will be created (any empty classes without objects are skipped, so that \( n \) is a maximal number). The names of the new classes consist of the feature name plus a trailing letter (‘A’, ‘B’, ‘C’, and so forth). They will be listed in the **Class list** of the “Classes” tab.
12. If *daime* is linked with R, a distribution plot is shown for each object feature. This plot reveals how the objects are distributed among the $n$ classes. It allows the user to compare the object distribution for the finally chosen feature to the distribution for the other features (*Figure 15*).

![Distribution plots](image)

*Figure 15. Distribution plot shown after automatic classification of FISH-labeled microbial cells. Among the three analyzed features, the object size in pixels was chosen for classification. The condition was “Minimize variance.” The plot for the chosen feature is always shown first and in a different color, with the feature name marked by arrows. The boundaries of the $n$ classes (here: $n=4$) are shown as vertical dashed lines.*

**Manual Classification**

To classify an individual object manually, pick the object with the mouse and choose **Edit object attributes** in the menu. Enter the name of the class in field **Obj. class** and click **OK**. This can be an existing or a new class. The object will become a member of the specified class. If the class did not exist previously, it will automatically be created and will appear in the **Class list** of the **“Classes” tab**.

**Save and Restore the Current State**

The Object Editor lacks a real “undo” function. All modifications of object definitions and images are permanent! However, one can transiently save the current state of the edited images in computer memory (not on disk) and restore this state at any later time, as long as the respective Visualizer window is open. Save the state, for example, prior to larger modifications with an uncertain outcome. If something did not work as expected, the state can be restored and the modifications are undone.

liğ다 Click this button to save the current state of the images and object definitions in computer memory.

liğ다 Click this button to restore the previously saved state of the images. All modifications applied between saving and restoring the state will be lost.
Image Analysis

This section describes the tools of *daime* that extract quantitative data from images. The images must be segmented prior to analysis. Use the **Object Editor** to select or reject objects in segmented images: Only selected objects are included in subsequent analyses, whereas rejected objects are ignored.

Link between *daime* and R

*daime* can be “linked” with the R language and environment for statistical computing and graphics (https://cran.r-project.org). Once R and the required R packages have been installed on the computer, the link between *daime* and R is easy to establish (the procedure is described in the appendix).

When linked with R, *daime* utilizes R for plotting data and for certain statistical analyses. The results are displayed seamlessly by the user interface of *daime*. Moreover, *daime* can export data in a format that is compatible with R. *daime* can also generate and export R scripts, which can be used to reproduce and fine-tune data plots in R (outside *daime*). Thus, users with programming skills in R can use the data and scripts exported from *daime* to perform further, customized analyses of image analysis results.

Note that *daime* functions very well without R, but then data plots are more basic and some statistical analyses are not performed. Therefore, it is strongly recommended to establish the link between *daime* and R in order to make full use of all features *daime* offers.

The descriptions of features, data plots, etc., in this manual are based on the assumption that *daime* has been linked with R.

Display and Export of Image Analysis Results

Image analysis results are presented as data tables and data plots (some functions may show only a data table or a data plot, respectively). Data and plots can also be exported for use in other software or for presentation in publications, talks, etc.

- **Export data**: This button appears below each data table that can be exported. Click this button to access the following options.
  - **Export data table**: Writes the data to a text (ASCII) file. The exported table is arranged in such a way that it can easily be imported into third-party spreadsheet or plotting software.
  - **Export data for R**: Writes the data to a text (ASCII) file. The exported table is arranged in such a way that it can easily be imported into R as a data frame. The data columns are tab-delimited. **Note**: This option is even available if daime is not linked with R.
  - **Copy data**: Copies the data table into the clipboard.
  - **Comma as decimal separator**: Check this option to export data tables with a comma instead of a point as decimal separator.
- **Tab-delimited columns**: Check this option to export data tables with tab-delimited columns. Uncheck it to export data tables with space-delimited columns.

  The decimal separator and column delimiter for data table export can be permanently chosen in the [General Preferences](#). Changes of these settings made in the data export dialog will override the general preferences but are not permanent.

- **Export/Print plot**: This button appears below each plot that can be exported. Click this button to access the following options.
  - **Export plot**: Writes the plot to an image file. The format of the image file can be chosen in the [Plot export format](#) menu:
    - **Scalable Vector Graphics (SVG)**: A vector-based format, which can be opened in illustration software such as Inkscape and Adobe Illustrator. This format is most suitable for editing a plot (adapt font sizes, colors, etc.) prior to publication.
    - **Portable Document Format (PDF)**: The best format for sharing the plot with others. As a PDF file, the plot should look identical regardless of the viewing software and computer platform used.
    - **Tagged Image File Format (TIFF)**: A raster image format, which can be viewed by virtually all graphics programs and can be inserted into slide presentations (e.g., Powerpoint). However, plots exported as TIFF images cannot be edited. Scaling to large sizes (e.g., on posters) may cause visual artifacts, too. Better use SVG images for scaling plots to fit large print media.
  - **Print plot**: Sends the plot to a printer. Printer settings can be adjusted.
  - **Copy plot**: Copies the plot as a raster image into the clipboard.
  - **Export R script+data**: This option is available if [daime is linked with R](#). It generates an R script, which contains R code to produce the plot as shown on screen. The script and the data are exported into a selected folder on disk. Together, script and data can be used to reproduce and fine-tune the plot in R. **Note**: The R script is automatically configured to (i) import the data from the chosen folder on disk, and (ii) save the plot in the format chosen in the [Plot export format](#) menu.

  If [daime is linked with R](#), a [zoom slider](#) and options to change the appearance of the plot are available. Use the slider to enlarge the plot (especially useful if a plot contains many panels, which may be displayed small by default to fit in the plot area). A plot can also be zoomed by turning the mouse wheel while the mouse cursor is above the plot. **Note**: A plot can also be enlarged by resizing the whole results window.

  Depending on the image analysis function there may be additional options below a data table or plot, respectively. For details, refer to the specific sections of this manual.
Isotropic Scale

The pixel (voxel) resolution and the µm scale of an image determine the µm size of a single pixel or voxel, respectively. Pixels are true squares in µm space if the x and y dimensions have both the same pixel resolution and µm scale. For example, if an image has a resolution of 512 x 512 pixels and a scale of 100 x 100 µm, each pixel is a small square of size 0.2 x 0.2 µm. The voxels of z-stacks are true cubes in µm space if all three dimensions have the same voxel resolution and µm scale. Such images are isotropic, because the size of their basic elements (pixels or voxels) is equal in all spatial directions. Measurements of some object features, such as volume and surface, can be slightly more accurate for isotropic images. Therefore, daime can convert anisotropic to isotropic images. The pixel (or voxel) resolution is then modified based on the µm scale. To achieve this, new pixels (voxels) are added by interpolating image data.

Choose Analysis:Isotropic scale in the main menu to convert the image, which is currently marked in the main window’s list, to its isotropic equivalent. Select whether the image resolution should be adapted in 2D or 3D. The new isotropic image gets the name of the source image plus the suffix _iso.

The interpolation of image data can cause small differences in object size, texture, and brightness. Sometimes these differences may outweigh the advantages of isotropic images. The conversion to isotropic images is optional. Most image analysis features of daime work well with anisotropic images, too.

Count Objects

Choose Analysis:Count objects to count the objects in the image that is currently marked in the main window’s list. The image must already be 2D- or 3D-segmented. The total object number, and the numbers of selected and rejected objects, are reported for each image in a 2D-segmented batch or for a whole 3D-segmented z-stack, respectively.

Measure Features of Objects

Key features of the objects in segmented images such as brightness, area, volume, and shape-related parameters can be measured. Choose Analysis:Measure objects to measure the objects in the image that is currently marked in the main window’s list. In the following dialog, select the object features to be quantified (Table 16). The precision (decimal places) of the reported results can also be adjusted.

Only selected objects are measured. Rejected objects are ignored. Objects can be selected or rejected in the Object Editor.

The features of single objects can be measured directly in the Object Editor. Pick the object with the mouse, then choose Measure object features in the menu.
Table 16. Description of the object features that can be measured. Note: Cavities, which extend to the surface of an object, are considered to be invaginations and are not treated as “internal holes”.

<table>
<thead>
<tr>
<th>2D Mode</th>
<th>Explanation</th>
<th>3D Mode</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pixels</td>
<td>The size of the object in pixels.</td>
<td>Voxels</td>
<td>The size of the object in voxels.</td>
</tr>
<tr>
<td>Pixels in holes</td>
<td>The size of internal holes in the object in pixels (0 if there are no holes).</td>
<td>Voxels in holes</td>
<td>The size of internal holes in the object in voxels (0 if there are no holes).</td>
</tr>
<tr>
<td><strong>Min., Max., Mean, Median intensity</strong></td>
<td>The minimum, maximum, mean, median intensity (brightness) of the object, measured without or with (incl.) internal holes.</td>
<td><strong>Min., Max., Mean, Median intensity</strong></td>
<td>The minimum, maximum, mean, median intensity (brightness) of the object, measured without or with (incl.) internal holes.</td>
</tr>
<tr>
<td>s.d. intensity</td>
<td>The standard deviation of the intensity of the object, measured without or with (incl.) internal holes.</td>
<td>s.d. intensity</td>
<td>The standard deviation of the intensity of the object, measured without or with (incl.) internal holes.</td>
</tr>
<tr>
<td>Hole area (µm²)</td>
<td>The estimated 2D area of the object in µm². Includes any internal holes.</td>
<td>Hole volume (µm³)</td>
<td>The volume of internal holes in the object (0 if there are no holes).</td>
</tr>
<tr>
<td>Perimeter (µm)</td>
<td>The estimated length of the outer edge of the object.</td>
<td>Surface (µm²)</td>
<td>The estimated area of the outer surface of the object.</td>
</tr>
<tr>
<td><strong>Perim. / total area</strong></td>
<td>The ratio of the perimeter to the total area of the object.</td>
<td><strong>Surface / total volume</strong></td>
<td>The ratio of the surface to the total volume of the object.</td>
</tr>
<tr>
<td>Hole perimeter (µm)</td>
<td>The estimated perimeter of internal holes of the object (0 if there are no holes).</td>
<td>Hole surface (µm²)</td>
<td>The estimated surface of internal holes of the object (0 if there are no holes).</td>
</tr>
<tr>
<td>Hole perim. / perim.</td>
<td>The ratio of the hole perimeter to the perimeter of the outer edge.</td>
<td>Hole surface / surface</td>
<td>The ratio of the hole surface to the outer surface.</td>
</tr>
<tr>
<td>Hole area / total area</td>
<td>The ratio of the hole area to the total area.</td>
<td>Hole volume / total volume</td>
<td>The ratio of the hole volume to the total volume.</td>
</tr>
<tr>
<td><strong>Circularity</strong></td>
<td>This value is approx. 1 for circular objects, &lt; 1 for non-circular objects.</td>
<td><strong>Sphericity</strong></td>
<td>This value is approx. 1 for spherical objects, &lt; 1 for non-spherical objects.</td>
</tr>
<tr>
<td>Max. diameter (µm)</td>
<td>The largest caliper diameter of the object (in 2D).</td>
<td>Max. diameter (µm)</td>
<td>The largest caliper diameter of the object (in 3D).</td>
</tr>
<tr>
<td><strong>Min. diameter (µm)</strong></td>
<td>The smallest caliper diameter of the object (in 2D).</td>
<td><strong>Min. diameter (µm)</strong></td>
<td>The smallest caliper diameter of the object (in 3D).</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------</td>
<td>------------------------</td>
<td>-----------------------------------------------------</td>
</tr>
<tr>
<td><strong>Ratio of diameters</strong></td>
<td>The ratio of the largest to the smallest diameter. This value is approx. 1 for circular objects.</td>
<td><strong>Ratio of diameters</strong></td>
<td>The ratio of the largest to the smallest diameter. This value is approx. 1 for spherical objects.</td>
</tr>
<tr>
<td><strong>Centroid</strong></td>
<td>The position of the centroid (= the central pixel) of the object, reported in pixel and in µm coordinates.</td>
<td><strong>Centroid</strong></td>
<td>The position of the centroid (= the central voxel) of the object, reported in voxel and in µm coordinates.</td>
</tr>
</tbody>
</table>

Areas, volumes, perimeters, and surfaces are only estimates. In digital images, the smooth edges and surfaces of real curved objects are approximated by rectangular pixels or block-shaped voxels. *daime* tries to estimate the real shape of an object from the arrangement of its pixels (voxels). It uses methods designed for analyzing curved object borders and surfaces (like those of most microbial cells and cell clusters). It may not precisely measure straight lines, rectangles, or 3D blocks. Such structures are rare in microbiology. However, this can limit the potential use of *daime* to analyze certain other kinds of images (e.g., technical images that contain geometric objects).

The **circularity** of 2D objects is computed by using the formula $\frac{4\pi A}{P^2}$ where $A$ is the total area and $P$ is the perimeter of the object. The **sphericity** of 3D objects is computed by using the formula $\frac{\frac{1}{\pi^3} (6V)^{\frac{2}{3}}}{S}$ where $S$ is the surface and $V$ is the total volume of the object.

**Note:** For both features we expect only values ≤1 (or slightly greater than 1 due to the inaccurate representation of curved objects in raster images). However, values considerably greater than 1 may be measured for very small and/or highly irregularly formed objects, which often depict artifacts or noise. Actually, classification by user-defined values, with circularity or sphericity ≥1.1 as criterion, can be one way to identify such spurious objects in segmented images.

The intensity of objects in color images is measured based on the highest intensity of each pixel (voxel) in any color channel.
Figure 16. Quantification of two features for rod-shaped bacteria from a pure culture that were labeled by rRNA-targeted FISH. Mean fluorescence intensity (left) and max. diameter as a measure of cell size (right). The cells were auto-classified by mean intensity prior to this analysis.

Biovolume Fraction (in situ Abundance Quantification of Microbial Populations)

daiime can quantify the relative abundances of microbial populations labeled by specific fluorescence markers. The most widely used labeling method for this purpose is fluorescence in situ hybridization (FISH) with rRNA-targeted probes, but other labeling methods such as immunofluorescence or GFP tagging are also suitable. See (4) for a methodological description of the quantification approach. The method does not quantify cell numbers but determines the biovolume fraction of a specifically labeled target population, relative to the biovolume of the total biomass. The total biomass should be labeled by a general probe or unspecific fluorescent dye. Although biovolume is a 3D parameter, this stereological approach uses 2D-segmented image batches to estimate biovolume fractions.

Useful Rules to Obtain Reliable Quantification Results

- The images should be acquired by using a confocal laser scanning microscope. Images recorded by a non-confocal microscope and processed by deconvolution software may be suitable, too. Alternatively, a sample may be physically sectioned and thin sections analyzed. Images should not contain blurred fluorescence signals from above or below the focal plane.
- The image pairs must be recorded at randomly chosen positions in the sample. Do not select the positions based on the presence or absence of the target population. Choose the positions “blindly”! A good approach is to view the sample through a filter that blocks those fluorescence wavelengths, which are emitted by the specifically labeled target population.
• The target population cells must be labeled by both the specific probe and the general probe (or stain). The same cells should be congruent in the images of each image pair (i.e., they must not appear much larger in one image than in the other one). Adjust the image acquisition settings of the (confocal) microscope accordingly. Once adjusted, these settings should not be changed until all the image pairs have been recorded.

• If more than one target population is to be quantified, always use the same microscope settings for recording the images of the total (= unspecifically labeled) biomass. If adjustments are needed to meet the congruency requirement (see above), change only the settings for recording the target population-specific signals. This rule ensures that the relative abundances of different target populations in the same sample remain comparable.

• Usually, 20 to 30 image pairs for each quantified population are sufficient to get reliable results. However, the required number of image pairs depends on the sample type and the quantified population. More image pairs may be needed if the target population is very irregularly distributed in the sample.

• Use a relatively low magnification (e.g., 400×) in order to capture much biomass per image.

To quantify microbial population(s), import image batches that show the specifically labeled target population(s) and the unspecifically labeled total biomass, respectively. Images that form an image pair must have the same index number in each batch.

The image batches must be 2D-segmented prior to quantification. Tools of the Object Editor can be used to reject artifacts that must be excluded from the analysis.

**The Artifact Rejection Tool of the Object Editor**

This tool of the Object Editor is only available if a visualization session contains two or more 2D-segmented members. It detects putative artifacts in the images of the target population and rejects them automatically. The tool assumes that every object in a target population image, which has no counterpart in the corresponding total biomass image, must be an artifact. The decision whether an object has a counterpart or not depends on congruency: for each object in the target population image, the program checks if there is an object at the same position in the total biomass image, and how much these two objects overlap (i.e., how congruent they are). The degree of overlap (congruency) is expressed in per cent, so that “total overlap” means “100% congruency”. The congruency threshold for artifact detection is specified by the user in field Congr. threshold. Objects with less than the specified overlap are considered artifacts. Hence, if this threshold is increased, the tool finds more artifacts. Once the threshold has been set, click Reject. The tool will reject all detected artifacts in the first member of the visualization session (this should be the target population image(s)). Check In all images to detect artifacts in all images of a batch at once. The rejected objects will be ignored during biovolume measurement. **Note:** This tool cannot detect artifacts in the total biomass images – reject those artifacts manually.
Biovolume Fraction Quantification and Evaluation of Results

When segmentation and object selection are done, click this button or choose Analysis:Stereology:Biovolume fraction in the main menu. In the next dialog, choose 1 to 8 target populations to be quantified (left list) and 1 image (batch) for the total biomass (right list). Note: The selected image batches must have the same pixel resolution and contain the same numbers of images. Click OK, and daime measures the biovolume fraction(s) of the target population(s).

Figure 17. Biovolume fraction quantification results for three target populations. Cumulative biovolume fraction plot (left panel) and biovolume fraction distribution plot (right panel). Note that populations 1 and 2 have nearly identical biovolume fractions.

The results are reported as data tables and plots. The table Total biovolume fraction shows the final, quantified biovolume fraction of each target population. It also shows the average congruency (overlap, in per cent) of the target population cells with their counterparts in the total biomass images. For a reliable quantification, the congruency should be at least 90%. These values are determined by evaluating all image pairs together. Table Image pair biovolume fractions lists the measured biovolume fractions for each image pair separately. The raw data (sums of object pixels; abbreviations: pop.=target, gen.=general biomass) are also listed. The mean, median, and s.d. of the biovolume fractions are listed in the bottom rows of this table.

The main quantification result is the values determined for all image pairs together (in table Total biovolume fraction). However, the list of biovolume fractions per image pair can be useful for analyzing the distribution of a target population among the image pairs (e.g., do all image pairs contain similar biovolume fractions of a target population?).

The Cumulative biovolume fraction plot (Figure 17) shows a curve that represents the total biovolume fraction, which would be obtained by analyzing only a given number of image pairs (on the x-axis). The largest number on the x-axis is the real number of image
pairs in the image batches. The final quantification result is shown as a dashed horizontal line (this number is also reported in the **Total biovolume fraction** table). If the curve has a long tail that approaches the dashed line (like the curves in Figure 17), the quantification result likely reflects the real relative abundance of the target population. If the tail of the curve is short or if there is no tail, consider repeating the analysis with more image pairs. A curve is shown for every analyzed target population. The biovolume fractions in the individual image pairs can be shown as a box plot (**Biovolume fraction distribution**). The box plot indicates how uniformly a target population is distributed among the image pairs. A statistical test indicates whether the relative abundance of at least one target population is significantly different from the others.

**Evaluation of FISH Probes (Formamide Concentration Series)**

Finding the optimal hybridization stringency is a critical step in the design of new **rRNA-targeted FISH probes**. For this purpose, a series of FISH experiments with the new probe is performed with increasing **formamide (FA)** concentrations in the hybridization buffer and with probe target and suitable non-target organisms (6, 7). The most precise way to determine the optimal FA concentration is to measure the average fluorescence intensity of probe-labeled cells at the different FA concentrations, and to plot these values against the FA concentrations. The plot shows the dissociation profile (melting curve) of the probe. It should be compared for probe target and non-target organisms.

*daime* can evaluate such **FA concentration series**. In order to use this feature, one must acquire images of cells that were hybridized to the same probe with different FA concentrations.

**Useful Rules to Obtain Probe Dissociation Profiles**

- Suitable non-target organisms have very few, ideally only one, nucleotide mismatch to the FISH probe on their 16S or 23S rRNA. The aim is to find stringent hybridization conditions that discriminate against such “one-mismatch organisms”.
- FISH with the different FA concentrations must be performed using the same reagents, probe solutions, and appropriately fixed cells of probe target and non-target organisms. The same hybridization and washing times must be applied for all FA concentrations and organisms.
- Images should be recorded with a CCD camera and an epifluorescence microscope or with a confocal laser scanning microscope. Exposure time or detector settings must be adjusted to ensure that bright cells are not overexposed and darker cells are still visible in the images. **All images of the FA series must be recorded with the same exposure time and detector settings!**
- Fluorochrome **bleaching must be minimized** during image acquisition. The cells should be exposed as shortly as possible to the excitation light. Slides with hybridized cells should be kept at a dark and cool place prior to image acquisition.
- The **pinhole of a confocal microscope** must be adjusted properly to ensure that the detector captures all light emitted by a single cell. For example, 2 µm optical sections should be suitable for bacterial cells of a size like *E. coli*. 
• If available, pure cultures of probe target and non-target organisms should be used. If pure cultures are unavailable, environmental samples that contain the respective organisms may be used. Alternatively, apply the “Clone-FISH” approach [8].
• A sufficient number of images must be recorded and analyzed. At least 100 cells should be measured for each FA concentration and organism. The file name scheme described below must be used for saving the TIFF images to disk.

Image Import and Preparation

While importing the images of a formamide series, daime must internally sort these images according to the FA concentration. Therefore, the TIFF image files must be named according to the following scheme.

Table 17. File name scheme for formamide concentration series.

<table>
<thead>
<tr>
<th>FA Concentration (%)</th>
<th>Image Index</th>
<th>File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>1st image</td>
<td>f0-1.tif</td>
</tr>
<tr>
<td>10%</td>
<td>1st image</td>
<td>F10-1.tif</td>
</tr>
<tr>
<td>0%</td>
<td>2nd image</td>
<td>f0-2.tif</td>
</tr>
<tr>
<td>10%</td>
<td>2nd image</td>
<td>F10-2.tif</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>... and so on</td>
</tr>
</tbody>
</table>

The general format is: **fx-y.tif** where x is the FA concentration (in %) and y is the image index for this FA concentration; “f” stands for “formamide”. Additional letters or words in the file names are not allowed. The FA concentrations can be values between 1 and 80, and their increment can vary within the same FA series. For example, one could import images for 5%, 10%, 15%, 20%, 30%, and 60% FA. An arbitrary (and different!) number of images can be imported for each FA concentration (therefore, the file name scheme contains an image index in addition to the FA%).

All image files of a formamide series must be located in the same folder on disk. **Note:** Different FA series cannot be mixed in one and the same folder! Import the image batch as usual, by selecting one of the files. Based on the file name, daime will detect that this image belongs to a formamide series and will automatically import all other images of this FA series. The images will automatically be sorted, in increasing order, by the FA concentration. Once imported, the image batch is automatically named **Formamide conc. series.** It can be saved under any other name in the STK format. When the image batch is loaded again from the STK file, daime will recognize it as FA series.

The imported images must be 2D-segmented. Usually, global thresholding by the RATS-L method works well for FA series. The FA series evaluation is most accurate if single cells (not cell clusters) are analyzed. To achieve this, use the tools of the Object Editor to reject objects that are larger than single cells. If this is impossible, because an organism shows a strong tendency to aggregate, try to select only small cell clumps and reject the larger aggregates. Cells/objects that touch the image borders should be rejected, too (this recommendation can be ignored for analyses of long filaments).
Analysis of Formamide Concentration Series and Evaluation of Results

After image segmentation and object selection, click this button or choose **Analysis:FISH specials:Evaluate formamide series** in the main menu. In the next dialog, choose 1 to 8 image batches (must be formamide series) to be analyzed. *daime* will then measure the mean fluorescence intensity of all selected objects in all images. Based on these values, the mean fluorescence intensity is determined for each FA concentration.

![Formamide series evaluation results](image)

**Figure 18.** Formamide series evaluation results for three rRNA-targeted FISH probes. The plot shows the probe dissociation profiles.

The results are reported as a data table and plot for all analyzed series. The table lists the formamide concentrations, mean fluorescence intensities, standard error, and 95% confidence limits. The probe dissociation plot (*Figure 18*) shows the values from the table as data points (error bars = s.e.m.), and optionally the sigmoidal probe dissociation curves obtained by nonlinear regression (only if *daime is linked with R*) and 95% confidence intervals.

The sigmoidal regression curves can only be plotted if a curve can successfully be fitted to each displayed dataset. Otherwise, an error message is shown. In such a case, disable regression curve plotting or omit the problematic dataset (if multiple formamide series are analyzed).

After the analysis, the objects in the images have automatically been assigned to new **object classes**, which are named according to the formamide concentration (fa_0, fa_05, fa_10, and so forth). To measure the fluorescence intensity of all these objects individually, **measure the object features** and choose the **mean intensity**. This will yield a table with the mean pixel intensity of each individual object and the respective plot (*Figure 19*).
Figure 19. Quantification of mean intensities of the single objects (microbial cells) in a formamide series. The images were first analyzed by formamide series evaluation. Subsequently, the feature mean intensity was measured for all objects in the images. The object classes had been automatically generated during formamide series evaluation.

**Analysis of Spatial Arrangement Patterns**

Spatial arrangement patterns of different populations are an important structural feature of microbial biofilms and flocs. They can indicate important biological features such as mutualistic symbioses, antagonistic interactions, and adaptations to specific microniches. 

daime can quantify the spatial arrangement patterns of (fluorescently labeled) microorganisms by two basic kinds of algorithms, which are both available for 2D image batches and 3D z-stacks: The **Linear Dipole Algorithm** and the **Inflate Algorithm**. These approaches are described in detail in references (1, 9–11). Please refer to this literature before performing spatial arrangement analyses, because the proper use of these features and the interpretation of the results requires some understanding of the theoretical background. In this manual we assume that the reader knows the basics of these approaches, and the respective terminology is not explained here.

daime is able to quantify the spatial arrangement pattern of cells (or cell clusters) of a single population as well as the arrangement patterns of up to nine different populations relative to each other. If more populations must be analyzed, they must be bundled in multiple runs of the respective algorithm.

Simultaneous analyses of more than three populations are facilitated by application of a suitable multicolor FISH approach (11–14).
Useful Rules for Reliable Spatial Analyses

The populations to be investigated must be labeled by specific FISH probes or by another specific fluorescent marker. In addition, the following conditions should be met to ensure reliable quantification results:

- Prior to FISH, biofilm samples should be embedded in polyacrylamide gel pads (1) or agarose (11), or in another suitable embedding medium, in order to minimize shrinkage and the disruption of original 3D structures during cell fixation, hybridization, and washing.

- Images should be recorded by using a confocal laser scanning microscope. The 2D algorithms work on batches of 2D-segmented images. These images must be taken at random positions (!!) in the sample. The 3D algorithms work on 3D-segmented z-stacks, which should also be recorded at randomly chosen positions. All images must be sharp and high-quality; in particular noise (small non-biomass objects) can strongly bias the results. Noise can be reduced by the various background removal and noise reduction tools but should be minimized already during image acquisition.

- If the arrangement pattern of two (or more) different populations is to be analyzed: At each randomly chosen position, take one image (or z-stack) of each population. Also make sure that the biomass of the different populations does not overlap in the images. Although “overlapping cells” are not physically possible, they may occur in confocal images due to sub-optimal pinhole or detector settings (e.g., due to overexposure). Recall that cells appear larger in images if too much light is captured by the detector. Such “overlapping” cells of different populations would cause a very strong bias.

- A good image resolution for spatial arrangement analyses is 512 × 512 pixels. Larger pixel resolutions increase the computation time but do usually not improve the precision of the analyses.

- The microscopic magnification should be as low as possible in order to capture much biomass per image. However, the magnification should also match the size of the structures to be analyzed. For example, a low magnification may be needed to analyze the arrangement of large cell clusters. In contrast, a high magnification is better to analyze the arrangement of single microbial cells.

- As many images as possible should be analyzed to get sufficient statistical support for the results. As a rule of thumb, the minimum number for a reliable analysis is 20 to 30 2D images. Since z-stacks usually contain more information than 2D images, their number may be lower.

At the beginning of a spatial arrangement pattern analysis, daime performs some tests and alerts the user if common sources of bias are detected (e.g., noise or “overlapping” cells of different populations). Some of these problems can be fixed or at least reduced automatically. A dialog allows the user to choose which problems should be fixed by the program.
All analyses consider only selected objects and ignore rejected objects; use the Object Editor to select and reject objects in the segmented images. All image batches or z-stacks, which should be analyzed together, must have the same pixel (voxel) resolution and µm scale, and they must contain the same number of images. The order of the images in the batches must be identical; that means, images at the same position in the batches must show the same microscopic field of view. Likewise, the z-order of the images in z-stacks must be the same.

Spatial Arrangement Pattern Analysis by the Linear Dipole Algorithm

The Linear Dipole Algorithm is one of the two stereological approaches for spatial arrangement pattern analysis in daime.

For analysis of a single population:

Choose Analysis:Stereology:Spatial arrangement (2D linear dipoles, 1 pop.) in the main menu to analyze the 2D-segmented image batch that is currently marked in the main window's list. Accordingly, choose Analysis:Stereology:Spatial arrangement (3D linear dipoles, 1 pop.) to analyze the 3D-segmented z-stack that is currently marked in the main window's list. In the following dialog, indicate whether the program should (a) analyze only distances between different objects, or (b) analyze distances within the same object (e.g., within the same cell aggregate) and also distances between different objects (e.g. between different cell aggregates). In case (b), the resulting pair correlation curve will also indicate the average size of the cell aggregates, because a very strong clustering signal will be obtained at distances between 0 µm and the average cell cluster diameter.

For analysis of multiple populations:

Choose Analysis:Stereology:Spatial arrangement (2D linear dipoles, multiple pop.) or Analysis:Stereology:Spatial arrangement (3D linear dipoles, multiple pop.) to analyze 2D-segmented image batches or 3D-segmented z-stacks, respectively. In the following dialog, select one reference population and up to eight analyzed populations.

The multiple population algorithm quantifies the spatial arrangement patterns of the analyzed populations relative to the reference population. Patterns within the same population are not analyzed (use the options for a single population for this purpose).

Overview of the distance and parameter selection dialog

- **Distance table**: This table shows the discrete distances (in µm) that can be analyzed. daime suggests possible distances based on the pixel resolution and the µm scale of the images. By default, all possible distances are selected and will be included in the analysis. Distances can be rejected (or selected again) by toggling the check boxes of the Measure column. The fewer distances are selected, the faster the analysis will be finished (but it will lack information for all the rejected distances).
• **Select distances:** Click Select All or Reject All to toggle the selection of all distances listed in the table. Change the number in box Select every to select every $n^{th}$ distance in the table.

• **Set distance range:** Enter the shortest and the longest distances to be analyzed in the From and To fields, respectively, and then click Set. This will select only the distances within the indicated range.

• **Precision of analysis:** daime contains two implementations of the Linear Dipole Algorithm. Random dipoles virtually “throws” a large number of randomly oriented dipole probes onto an image and counts the “hits” and “misses”. Scan whole ref. space scans the whole reference space pixel by pixel, and analyzes a semicircle of linear dipole probes at every pixel position (1). Random dipoles is much faster and is selected by default. The number of dipole probes analyzed per distance can be entered in field No. of random dipoles. A larger number of dipole probes improves the precision of the analysis and yields smoother pair correlation curves, but also increases the runtime of the algorithm. The default of 50,000 dipole probes per distance is a good starting point, but larger values (such as 250,000 dipole probes per distance) may yield better results. If Scan whole ref. space is selected, there is no need to adjust the number of dipole probes analyzed per distance. Note: Scan whole ref. space is available only for 2D-segmented images.

• **Reference space:** If needed, a reference space mask can be chosen here.

![Figure 20. Results plot of a spatial arrangement analysis by the Linear Dipole Algorithm (2D analysis). In this example, population 1 shows coaggregation with the reference population at distances between 0 and ~50 µm. Population 2 shows avoidance from 0 to ~10 µm and coaggregation from 10 to ~50 µm. Both populations show random distribution relative to the reference population at longer distances.](image)
The results are reported as data tables and plots (Figure 20).

Pair correlation values >1 indicate coaggregation, values <1 indicate avoidance, and values ~1 indicate random distribution of two populations at the respective distance. If only one population is analyzed, the same rules apply with respect to the arrangement of the cells (or cell clusters) of this population. Note: If the data are arsinh transformed (see below), 0.88 is the value that indicates random distribution and separates coaggregation from avoidance.

The data table shows the analyzed distances, the mean pair correlation function for each distance, and the 95% confidence limits of the pair correlation function. The pair correlation plot (Figure 20) shows the analyzed distances from the reference population (x-axis) and the mean pair correlation function for each analyzed population with the reference population (data points, y-axis). Confidence intervals are shown as ribbons. The horizontal dashed line indicates random distribution of the populations. The populations co-aggregate if the curve (including the confidence limits) is above this line, they avoid each other if the curve is below the line, or they are randomly distributed if the line is within the confidence limits.

Choose arsinh Transformation to show the data and plot after inverse hyperbolic sine transformation. Similar to a logarithmic transformation, this improves the readability of the plot in the lower range of the y-axis. Unlike the log transformation, arsinh(0) is defined.

Spatial Arrangement Pattern Analysis by the Inflate Algorithm

The Inflate Algorithm is the other method daime provides for spatial arrangement pattern analyses.

Choose Analysis:Stereology: Spatial arrangement (2D inflate, multiple pop.) or Analysis:Stereology: Spatial arrangement (3D inflate, multiple pop.) to analyze 2D-segmented image batches or 3D-segmented z-stacks, respectively. In the following dialog, select one reference population and up to eight analyzed populations.

The algorithm quantifies the spatial arrangement patterns of the analyzed populations relative to the reference population. Patterns within the same population are never analyzed (use the Linear Dipole Algorithm for a single population for this purpose).

In the next step, adjust the maximal distance (in µm) to be analyzed (the minimal distance is always 0 µm). daime suggests the maximal possible distance based on the image resolution and µm scale. Note: The longer the maximal distance, the more time is needed for the whole analysis.
Figure 21. Results plot of a spatial arrangement analysis by the Inflate Algorithm (2D analysis). Left: Normalized positional fractions. Population 1 shows coaggregation with the reference population at distances between 0 and ~18 µm. Population 2 shows avoidance from 0 to ~4 µm and coaggregation from ~10 to ~25 µm. Population 1 shows avoidance, population 2 shows random distribution at larger distances. Right: Normalized cumulative fractions, which also indicate that the two analyzed populations show different spatial arrangement patterns with the reference population.

The results are reported as data tables and plots (Figure 21).

Normalized positional or cumulative fraction values >1 indicate coaggregation, values <1 indicate avoidance, and values ~1 indicate random distribution of two populations at the respective distance. Note: If the data are arsinh transformed (see below), 0.88 is the value that indicates random distribution and separates coaggregation from avoidance.

The data table shows the analyzed distances, the mean cumulative or positional fractions (raw and normalized) for each distance, and the 95% confidence limits. It also contains these values for virtual (simulated) randomly distributed populations for comparison. Note: Normalization is done by dividing the raw fractions of a population by the fractions of a virtual randomly distributed population. See (9) for details. The data plot (Figure 21) shows the analyzed distances from the reference population (x-axis) and the mean normalized, positional or cumulative fractions for each analyzed population (data points, y-axis). Confidence intervals are shown as ribbons. The horizontal dashed line indicates random distribution of the populations. For the normalized positional fractions, the populations co-aggregate if the curve (including the confidence limits) is above this line, they avoid each other if the curve is below the line, or they are randomly distributed if the line is within the confidence limits. The normalized cumulative fractions provide less spatial resolution (are more difficult to interpret) but can also indicate differences between the spatial arrangement patterns of populations.
Choose **arsinh Transformation** to show the data and plot after inverse hyperbolic sine transformation. Similar to a logarithmic transformation, this improves the readability of the plot in the lower range of the y-axis. Unlike the log transformation, arsinh(0) is defined.

**Differences Between the Two Algorithms**

The Linear Dipole and the Inflate Algorithm are very different approaches to quantify spatial arrangement patterns (9). Briefly, the **Linear Dipole Algorithm** measures the likelihood to encounter an arbitrary pixel of one population at a distance \(d\) from an arbitrary pixel of the other population. This analysis includes all pixels of the two populations, regardless of their location. In contrast, the **Inflate Algorithm** quantifies the analyzed population's pixels at defined distances away from the surface of the reference biomass. Internal pixels in cell aggregates formed by the reference population have no contribution to the results. Understanding this difference can be important for comparing the results of the two approaches (9).

The results of the Inflate Algorithm can differ when the analyzed and reference populations are swapped. A biological explanation for such a difference could be that one population depends on the close proximity of the other population, but not vice versa. The results must always be interpreted from the “perspective” of the analyzed population (whether it prefers to grow close to the reference population or not). **This is the main difference between the two algorithms.** The Inflate Algorithm is directional, whereas the Linear Dipole Algorithm is not: The results of that algorithm are identical when analyzed and reference populations are swapped.

In addition, The Inflate Algorithm is less sensitive to strongly **anisotropic** structures than the Linear Dipole Algorithm. Anisotropic means in this context that, if we move in any specific direction through a sample, the spatial arrangement of the populations changes depending on that direction. For example, in a vertically stratified biofilm it would make a difference whether we move vertically or horizontally through the biofilm. The Inflate Algorithm may yield more reliable results when applied to images of (strongly) stratified biofilms, granules, or similar kinds of samples.

**Spatial Arrangement Analyses: Examples**

Spatial arrangement patterns can be complex and the interpretation of these analyses may sometimes be difficult. *Figure 22* shows a few examples, which use artificial images for the sake of clarity.
Figure 22. Examples for spatial arrangement analyses. A. Image containing 4 objects separated by a constant distance, $d$. Three peaks in the pair correlation plot indicate clustering of the objects over three distances ($d$, $2d$, and $3d$). The highest peak indicates strongest clustering at the shortest distance ($d$), because $d$ is the distance between any two neighboring objects. The other peaks are lower, because $2d$ and $3d$ separate fewer objects from each other than $d$. The curve is below 1 between the peaks, indicating that the objects are separated only by multiples of $d$. In biological terms, $d$ would be the optimal and most commonly observed distance between cells of this population. B. Image containing objects that form aggregates. These aggregates are separated by a certain distance. The higher peak in the pair correlation graph indicates clustering over a range of short distances (these are the distances between objects within the same aggregate). The lower peak indicates clustering at a range of larger distances. These are the distances between different aggregates. C. Image containing two “populations”, one being localized at a defined distance from the other population. The pair correlation and normalized positional fraction plots consistently report this pattern. The normalized cumulative fraction plot also suggests avoidance at short distances and strong co-aggregation around 20 µm, but since it is cumulative, it does not indicate the avoidance at larger distances. D. Image showing three populations with different localization patterns relative to each other. Red and green were the analyzed, and blue the reference population. The three plots confirm that the red population co-aggregates very closely with the blue population, whereas the green population co-aggregates at larger distances with blue.

**Reference Space Masks**

All spatial arrangement analyses are done within the so-called reference space. The default reference space is the whole image. However, the biological sample (biofilm, floc, etc.) may cover only parts of an image, whereas the other regions of the image are empty. In this case, biases occur if the whole image is used as reference space: As the microbes are physically constrained to the image regions that show the sample, a co-aggregation signal is obtained even if the analyzed populations do not co-aggregate in the sample.
To minimize this bias, a **reference space mask** can be defined in order to tell the algorithms which image regions actually contain the sample. This mask defines a new reference space, which is smaller than the whole image. A reference space mask should cover all image regions that contain sample material (not only populations labeled by specific probes, but also other biomass). The principle of the reference space mask is illustrated in *Figure 23*.

![Figure 23](image.png)

**Figure 23.** Use of a reference space mask. **A.** Image containing objects that are randomly distributed relative to each other, but occur only within a particular region of the image. This region could be a piece of biofilm or a small floc. Although the objects are randomly distributed, the pair correlation plot suggests co-aggregation. **B.** Same analysis repeated with a reference space mask (the central binary image). The pair correlation plot now correctly reports random distribution of the objects.

In practice, reference space masks should be produced already when the images of the microbial populations are recorded. Most samples (biofilms, flocs, sediments etc.) emit autofluorescence strong enough to be recorded by confocal microscope detectors. Although autofluorescence can be an annoyance in the context of image analysis, it is useful for creating reference space masks. Simply take an image of the autofluorescence emitted by the whole biomass and extracellular matrix. If there is no autofluorescence, stain the whole biomass with an unspecific dye such as DAPI and use images of this stain for creating the reference space masks. It is important that these mask images show all the biomass, not only the specific populations to be analyzed. In addition to the mask images, take the images of the specifically labeled populations to be analyzed in the same fields of view.
Once the mask images have been recorded, they can be 2D-segmented with a thresholding algorithm such as RATS-L or Local, which detects even very dark (auto)fluorescence. After segmentation, the object layer of the mask can easily be extracted as binary image. This yields a new object mask, which covers the image regions that contain sample material. It can be used as reference space mask for spatial arrangement analyses.

Only binary images (object masks) can be used as reference space mask. The mask must have the same pixel/voxel resolution and µm scale as the images to be analyzed.

If a suitable reference space mask is available, daime allows the user to select such a mask for a spatial arrangement analysis.

The reference space mask for a 3D spatial arrangement analysis must be a z-stack.

A reference space mask should always be used, unless the sample (almost) completely covers the image(s). Thus, the need to create and use a reference space mask is a rule rather than an exception. A mask can consist of several regions, which need not be adjacent, in the same image. A mask may also contain holes without any biomass or extracellular matrix. The algorithms handle these cases correctly.

**Analysis of Stratified Biofilms, Granules, and Similar Structures**

Stratification is a common feature of multispecies biofilms. In a stratified biofilm, the organisms occur in distinct layers (depth zones) that may be characterized by different environmental conditions, such as concentration gradients of oxygen and nutrients. Stratification can also occur in suspended granules and flocs. Biological interactions and metabolic activities of microbes also influence the conditions within the biofilm and may affect its spatial organization and stratification.

daime offers a special tool, the Biofilm Slicer, which can greatly facilitate the image analysis of stratified biofilms (and similar structures) (15). The Slicer can virtually cut a biofilm image into “slices” that correspond to different depth zones (Figure 24). These slices can be handled separately, without any limits, by all other image analysis and visualization tools of daime. In addition to the basic Slicer, daime provides features that facilitate common tasks, such as the abundance quantification of target populations in different depth zones of a biofilm.

**Slicer Templates**

Before images of fluorescence-labeled microbes in biofilms can be virtually sliced, a so-called Slicer template must be created. It defines the different depth zones (the “slices”) for subsequent slicing of images. A Slicer template is created automatically by the Slicer from an image that must show the whole biofilm (for example, a FISH probe targeting all bacteria, or a general nucleic acid stain) at a specific location. Subsequently, the template is used to slice images that show the same location in the biofilm but contain different FISH
probe signals (for specific populations) or other fluorescence stains. Thus, it is easy to produce virtual slices of exactly the same depth zones from images of different microbial populations.

A Slicer template can be made from a single image or from a batch of 2D images. The images in a batch should show the same biofilm sample but different microscopic fields of view. If a template was made from a batch, then whole image batches can be sliced by using this template. Of course, the order of the images must be the same in the template and in all batches. For example, the first image of the template must show the same field of view as the first image of a batch to be sliced, and so on. The batches must contain as many images as the template and must have the same pixel resolution and µm scale.

Figure 24. The principle of the Biofilm Slicer and Slicer templates. Biofilm images courtesy of Dr. Annette Moter. A. Overlay image showing a stratified biofilm with a specific population in red. B. FISH signal of all bacteria (green). C. FISH signal of the specific population (red). D. Slicer template created automatically from image (B) with slices shown in artificial colors. The slicing distance was 20 µm. Depth zones are indicated in µm. E. A single slice of image (B) after slicing with the template shown in (D). The slice at depth 40-60 µm is shown. F. A single slice of image (C) after slicing with the template shown in (D). The slice at depth 40-60 µm is shown. Images (E) and (F) may be used for further image analyses of this depth zone. For instance, the relative abundance (biovolume fraction) of the specific population may be quantified.

**How to create a Slicer Template**

To create a Slicer template, first 2D-segment the image(s) of all microbes in the biofilm (Figure 24 B). The template will be made from 2D-segmented objects that are selected, whereas rejected objects are ignored (use the Object Editor to select or reject objects, if
needed). Make sure that all relevant biofilm pieces were correctly recognized during segmentation and have been selected.

Mark these images in the main window’s list, then choose Analysis:Biofilm tools:Create Slicer template in the main menu.

Overview of the Slicer template parameter dialog

- **Slice thickness (µm):** The thickness (in µm) of the slices. For example, the thickness was 20 µm in Figure 24 D.

- **Min. biomass content (%):** At least this area fraction (in per cent) of a slice must contain biomass. If there is less biomass, the respective slice will be removed from the generated Slicer template. Hence, such a slice will not be used in subsequent analyses. Adjust this value to exclude slices with a low biomass content, which may not be representative of the microbial community composition, from image analysis.

- **Slicing direction:** Push the arrow buttons to choose the direction(s) of slicing, away from the image borders. Multiple directions can be combined. For instance, the biofilm in Fig. Figure 24 D was sliced from the bottom to the top of the image. See Figure 25 for more examples. **Note:** The slicing direction also defines the direction of increasing depth of the biofilm.

- **Noise max. size (pixels):** The slicing algorithm must find the baseline of the biofilm. Depending on the slicing direction, this can be the biofilm surface or the substratum. The baseline is the 0 µm-line (zero depth). The shape of the baseline determines the shape of every slice (Figure 24 D-F). Particles ("noise"), such as detached biofilm pieces, between the actual baseline and one of the image borders can interfere with the algorithm. The value entered here indicates the maximal size of particles to be ignored during baseline definition. The size unit is pixels, and the default value is suitable in most cases. **Note:** If needed, the pixel size of a 2D-segmented particle can be determined by using the Object Editor. Rejected objects and particles, which have not been identified during 2D segmentation, are also always ignored.

- **Baseline smoothing (%):** When defining the baseline, the slicing algorithm follows the contour of the biofilm. This parameter determines how precisely it follows invaginations and protrusions of the biofilm. More smoothing (higher values entered here) produces a straighter baseline.

When all adjustments are made, click OK and the Slicer template will be created. The template is added to the image list in the main window. It has the same name as the images used to create the template, plus the suffix “_slices_template”.

83
Figure 25. Different slicing directions. Biofilm images courtesy of Dr. Annette Moter. A. Biofilm stained by a general nucleic acid stain. The biggest piece is colored artificially in yellow. This piece will be sliced. B. Slicer template made with slicing direction from bottom to top of the image. C. Slicer template made with slicing direction from top to bottom of the image. D. Slicer template made with combined slicing from bottom to top and from top to bottom of the image. Slices are shown in artificial colors in B-D.

**Fine-Tuning Slicer Templates**

A Slicer template is a 2D-segmented image (or a batch of 2D-segmented images). Each slice is a 2D-object. The **Object Editor** can be used to modify Slicer templates like any other 2D-segmented images. In particular, slices can be selected or rejected by the respective tools of the Object Editor. When biofilm images are sliced based on a template, the slicing algorithm considers only selected slices in the template but ignores all rejected slices. For example, assume that a Slicer template contains slices with a thickness of 20 µm. However, only one depth zone of the biofilm (40-60 µm) should be studied further. One can open the template in the Object Editor and reject all slices except the slice that represents the depth zone from 40 to 60 µm (this slice must remain selected in all images of the template). Subsequently, when this template is used to slice biofilm images, only slices representing the 40-60 µm depth zone will be produced.

Another example: Assume that a Slicer template is a batch of images that represent different microscopic fields of view. However, in one field of view there is an artifact in the 40-60 µm depth zone. All other fields of view contain no artifacts in any depth zone. The single slice, which represents 40-60 µm depth, can be rejected in this particular field of view by using the Object Editor. In all other fields of view, the slices representing this depth zone remain selected. When biofilm images are sliced, the algorithm will produce slices for the 40-60 µm depth zone from all fields of view except the one that contains the artifact.
Selecting and rejecting individual slices in the Object Editor is simply done as for any other 2D objects. If the orange or purple borders of adjacent slices are hard to distinguish, zoom in to enlarge the borders.

It is easy to select or reject all slices of a specific depth zone. Open the “Classes” tab of the Object Editor. The class list will show all depth zones in the current image of the template. If needed, check **In all images** to show all depth zones in all images of a batch. Individual depth zones can be selected or rejected by using the buttons next to the class list.

**Slice Biofilm Images**

When a Slicer template has been created, the biofilm images of the specifically labeled microbial populations can be sliced. Choose **Analysis:Biofilm tools:Slice images** in the main menu. Then select the images to be sliced and the Slicer template to be used for this purpose. Subsequently, the selected images will be sliced and new images will be created for each slice (depth zone). These new images have the same name as the sliced images plus a suffix indicating the depth zone. For example, the suffix “_100.00-120.00” indicates the depth zone from 100 to 120 µm. If a batch of images was sliced, then a new batch is created for each depth zone. These images can then be used as input data for all image analysis features of **daime**.

**Biovolume Fraction Quantification in Stratified Biofilms**

**daime** contains a feature that automatically slices biofilm images and quantifies the relative abundance (biovolume fraction) of a target population in the obtained slices (depth zones). The prerequisite is that a **Slicer template** has already been made for the images to be analyzed. Examples of this kind of analysis are found in references (15, 16).

Choose **Analysis:Biofilm tools:Slice, then quantify biovolume fraction** in the main menu. Then select the images that contain the target population (Population probe) and the images that contain the total biomass (General probe or stain). Subsequently, indicate which **Slicer template** should be used for slicing these images. **daime** will internally slice the images and quantify the biovolume fractions; the intermittently created slices of the depth zones are not visible to the user.
Figure 26. Biovolume fraction quantification results for a stratified biofilm. Here, the first section (0-20 µm; median=dotted line) was chosen as the reference that is compared with all other sections (significance symbols above the box plots). The dashed line represents the expected biovolume fraction of a randomly distributed population.

The results are reported as data tables and plots. The table **Total biovolume fractions (all sections)** lists the biovolume fraction and congruency for each section (=slice, depth zone). These values are explained [here](#). The table **Image pair biovolume fractions (all sections)** lists the measured biovolume fractions for each image pair in each section separately.

The plot (Figure 26) shows the biovolume fractions for all the image pairs in each section. The **p-value**, which is shown on the top of the plot, results from a Kruskal-Wallis test of all data. A p-value < 0.05 indicates that the biovolume fractions are significantly different among any of the sections, which is an indication for stratification. For comparison, the **horizontal dashed line** indicates the biovolume fraction that would be expected in all sections if the population was randomly distributed throughout the biofilm. The user can also select a **reference** that all sections are compared with: the first section (in the direction of slicing), the central section, the last section, or the pooled data from all sections. The **horizontal dotted line** indicates the median biovolume fraction in the chosen reference. Above each box plot, **significance symbols** indicate whether the data for the respective section are significantly different from the chosen reference (Wilcoxon rank sum test; ns: p>0.05, *: p≤0.05, **: p≤0.01, ***: p≤0.001, ****: p≤0.0001).

**Measure Features of Objects in Stratified Biofilms**

daime contains a feature that automatically slices biofilm images and quantifies the **features of the objects** (cells, cell clusters) of a target population in the obtained slices
(depth zones). The prerequisite is that a Slicer template has already been made for the images to be analyzed.

Choose **Analysis:Biofilm tools:Slice, then measure objects** in the main menu. Then select the images that contain the target population. Subsequently, indicate which Slicer template should be used for slicing these images and which object features should be measured. *daime* will internally slice the images and quantify the object features; the intermittently created slices of the depth zones are not visible to the user.

The results are reported as a data table and a plot. The table contains the values for all measured objects in all sections (depth zones) for the selected feature. The plot shows these values as box plots for each section. The data are compared among sections as described for biovolume fraction quantification in stratified biofilms (see also Figure 26).

**Randomized Images (for Tests)**

The functions for analyzing spatial arrangement patterns or stratified biofilms assume the null hypothesis that a population is randomly distributed in a sample. For testing purposes and as a control, *daime* offers the option to create images of randomly distributed points. When these images are used as input data for image analysis, an output that indicates random distribution of the “population” is expected.

Choose **Extras:Make randomized images** in the main menu. The following dialog contains several options:

- **Image pixel/voxel resolution**: The *x, y, and z resolution* of the new images. **Note**: The z resolution equals the number of images in an image batch or z-stack. Enter 1 to create a single image, or any number >1 to create a batch or z-stack.
- **Image µm scale**: The *µm scale* of the new images. **Note**: The z-scale is important for 3D analyses only.
- **Pixel/voxel density**: The fraction [0..1] of the image area (or volume of a z-stack) that will be covered by the randomly distributed points.
- **Custom image name**: The name of the new images. If nothing is entered, a name will be automatically generated.
- **Segmentation**: Selects whether the new images will be 2D-, 3D-, or not segmented. **Note**: All randomly distributed points in an image will belong to just one “object”.
- **Template images**: If images have been loaded into *daime*, they are listed here. Choose an image to automatically fill the other fields of the dialog based on the properties of that image. **Note**: This option is useful to create random images as a control for image analysis functions. It allows a direct comparison of the results obtained with random images to the results obtained with real images.
- **Mask images**: If binary images (object masks) have been loaded, they are listed here. Choose a mask (and also check **Use mask images**) to create random images that contain the randomly distributed points only in the area covered by the mask. **Note**: This option is useful if a reference space mask must be used for spatial arrangement analyses, and random images should be analyzed as a control.
References


Appendix I: Installation of *daime*

System Requirements

- Personal computer (PC) with an Intel, AMD, or compatible microprocessor.
- Supported operating systems:
  - **Microsoft Windows 10** (32- or 64-bit).
  - **Linux** (64-bit). *daime* has been tested on Ubuntu (14.04 LTS or newer), Xubuntu, Linux Mint, Debian, OpenSUSE. **Note:** *daime* runs best on Ubuntu-based distributions. **Note:** Fedora, Red Hat, CentOS are not supported although *daime* may work on some systems.
  - **Note:** There is **no MacOS** version of *daime*. As outlined below, *daime* uses OpenGL. Apple has removed OpenGL support from MacOS. Mac users may run the Windows or Linux version of *daime* in a virtual machine (Parallels, VirtualBox, etc.).
- **At least 1 GB of memory (RAM).** Analyses of large image batches and interactive 3D visualization require more RAM.
- A **graphics card** (or onboard graphics chip) for hardware-accelerated 2D and 3D graphics with OpenGL. OpenGL 1.2 is needed for starting *daime* and for non-graphics intensive image analysis tasks. The 3D visualization features require hardware support for OpenGL 2.1 (or a newer 2.x version). OpenGL 3 or newer is not directly supported, but current graphics hardware and OpenGL drivers usually offer 2.x compatibility. **Recommended:** A graphics card with at least 512 MB of video RAM (more is better). *daime* has been tested with Nvidia, Intel, and AMD graphics hardware and graphics drivers.

Installation on Microsoft Windows Systems

1. If *daime* version 2.1 (or older) is already installed on your computer, use the **Uninstall** feature to remove the older version.
2. Download the current version of *daime* for Windows. The name of the file is *daime_2_2_Windows.zip*. Extract this *zip* file into an empty folder on the hard disk of your computer. **Note:** *daime* for Windows is a portable application. The *zip* file can also be extracted into a folder on a USB stick, mobile hard disk, or any other storage device. *daime* can then be started from that device. Local installation on the computer is not required but can be more comfortable (see step 4).
3. To start *daime*, enter the new folder and double-click at *daime.exe*.
4. Optionally, if *daime* has been copied to the local hard disk: right-click with the mouse at *daime.exe*, then choose **Pin to start** or **Pin to taskbar**. *daime* can then be started from the start menu or the taskbar, respectively.
5. In order to remove *daime* for Windows from your system, simply delete the folder that was created in step 2 and contains *daime.exe*. No further uninstall procedure is needed.
Installation on Linux Systems

1. Download the current version of *daime* for Linux. The name of the file is daime_2_2_Linux.
2. Copy this file to your home folder or any other convenient place.
3. Open a terminal, then enter the folder that contains the file. Enter this command: 
   \textit{chmod a+x daime_2_2_Linux}
4. To start *daime*, open a terminal, then enter the folder that contains the file. Enter: 
   ./daime_2_2_Linux &
5. Optionally, the file can be renamed after step 3.
6. Optionally, add the folder that contains the *daime* file to the PATH. *daime* can then 
   easily be started from any other folder.
7. Optionally, create a shortcut to the *daime* file on the Desktop.
8. \textbf{Important note:} in order to create movies, *daime* uses the \textbf{ffmpg} program. It is not 
   shipped with *daime* for Linux. Please use the package manager (such as apt, 
   synaptic, rpm, yast) of your distribution to install \textbf{ffmpg} separately. \textbf{Note:} \textbf{ffmpg} 
   must be in the PATH so that *daime* can find it. This is usually the case when \textbf{ffmpg} 
   has been installed by a package manager.
9. In order to remove *daime* for Linux from your system, simply delete the *daime* file. 
   No further uninstall procedure is needed.
Appendix II: How to link *daime* with R

*daime* can be “linked” with the R language and environment for statistical computing and graphics ([https://cran.r-project.org](https://cran.r-project.org)). When linked with R, *daime* utilizes R for plotting numerical data and for certain statistical analyses. More information is provided [here](#).

1. Make sure that a recent version of R is installed on the system.
   a. R can be downloaded at [https://cran.r-project.org](https://cran.r-project.org). Installation instructions are available at the same site.
2. Make sure that the following R packages are installed, including their dependencies:
   a. ggpubr
   b. reshape2
   c. svglite
   d. RColorBrewer
   e. Cairo (this one should have automatically been installed with R. Enter the command `capabilities()` in R and check that TRUE is printed below ‘cairo’).
3. Determine which disk folder contains the `Rscript` executable file.
   a. On Windows systems, this file is usually located in: C:\Program Files\R\R-VERSION\bin (with VERSION being the installed version of R, for example 3.3.2).
   b. On Linux systems, open a terminal and enter the command: `which Rscript`
4. Create a new, empty folder for reading and writing (e.g., in your home folder). Any name can be chosen for this new folder. *daime* will need this folder as a temporary file storage place for communicating with R.
5. Choose **Extras:Link *daime* with R** in the main menu of *daime*.
6. In the dialog, choose the Rscript-containing folder (see step 3) as **R executable directory**. Choose the newly created folder (from step 4) as **Temporary directory**.
7. Click **Test** to check whether *daime* can successfully communicate with R. If it works, a plot will appear in field **Test plot**. The test fails if the wrong folders have been chosen in step 6 or if not all the required R packages have been installed (see step 2).
8. If the test was successful, click **Apply** to permanently link *daime* with R. From now on, many features of *daime* will use R for data plotting and statistical analyses.

   To remove the link, click **Remove R Link** in the dialog. *daime* will then use its own routines for plotting etc. (these plots are more basic). **Note:** The link with R can be established and removed any number of times.

   If more than one version of R is installed, choose the desired one by selecting the respective R executable folder in step 6. The linked R version can be changed by selecting another folder and clicking **Apply** again.

   The link between *daime* and R does not affect or modify the R installation. This link is exclusively managed by *daime* and R does not “know” about it.