

COMSTAT 2.1 MANUAL

Release date July 1 2015

DISCLAIMER:

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Authors

Software designed, written and implemented by Dr. Arne Heydorn (currently Glostrup Hospital, Denmark) and Prof. Dr. Bjarne Ersbøll (DTU Compute, Technical University of Denmark) (Comstat1), and Martin Vorregaard (currently SEAS-NVE A/S, Denmark) and Prof. Dr. Bjarne Ersbøll (Comstat2).

Manual: Dr. Claus Sternberg*, Department of Systems Biology, Technical University of Denmark, Building 301, 2800 Kongens Lyngby, Denmark.
E-mail: cst@bio.dtu.dk

*To whom correspondence should be addressed.

Manual revision 1.1

Copyright

This manual is licensed under a Creative Commons license 4.0:
<http://creativecommons.org/licenses/by-nc/4.0/>



Table of Contents

DISCLAIMER:	1
Authors	1
Copyright	1
Table of Contents	2
Definitions and terms	5
Nomenclature	5
Preface	6
Chapter 1 – Installation	7
Automated installation.....	7
Manual installation.....	7
Chapter 2 - Running Comstat2 – The basics	9
Input and output files.....	9
Pixels and real life.....	9
Supported file types	10
Running Comstat2.....	10
Selecting source folder(s)	11
The Images in Directories window	11
Selecting basic analysis conditions	12
Selecting basic analysis parameters.....	12
Performing the analysis	12
Chapter 3 - Running Comstat2 - Advanced use	13
Setting advanced analysis conditions	13
Automatic or manual thresholding	13
Connected Volume filtering	14
Smacking.....	14
Minimal colony size.....	14
Selecting Analyses to be performed	15
Customizing output	15
Decimal separator.....	15
File name.....	15
Appending data or make new output files	15
The log window	15
Chapter 4 – Analysis Functions in Comstat2	16
Biomass	16
Maximum Thickness.....	16
Thickness distribution.....	16
Dimensionless Roughness Coefficient (<i>Ra</i> *)	17
Area Occupied in Layers	17
Surface Area (includes Surface/Area calculation)	18
Microcolonies at Substratum	18
Volume of Microcolonies at Substratum	18
Average Run Length of Colonies (vert/horiz)	18

Diffusion Distance.....	18
Fractal Dimension.....	18
Local Thickness	19
Chapter 5 – What’s next?	19
APPENDIX	20
A – Using the OME-TIFF file format.....	20
Converting to OME-TIFF using the LOCI Tools (installed as part of Comstat2) ..	20
Using a macro to convert images	21
Converting multiple TIFF images to an OME-TIFF stack.....	21
Converting to OME-TIFF using Imaris 8.0.....	22
B - Obtaining Comstat2	22
C – Terms for use (how to reference Comstat2).....	22
D – Troubleshooting	22
Installation issues	22
ImageJ does not start (MacOSX)	22
comstat2.ini missing.....	23
Java is not found (Windows).....	23
File selection issues	23
Images opened in ImageJ are not analyzed	23
No files are shown when selecting directories	23
When the directory is selected and it contains compatible files, still no stack are loaded in the program	23
When I added a directory correctly, still no stacks are loaded	23
Nothing is analyzed even if the files are shown in the Images in Directories box ...	24
Input data quality issues	24
Wrong setting of the substratum layer when recording stacks.....	24
Dealing with “noisy” stacks.....	24
E – Output files.....	25
Text files	25
AreaOcc.txt.....	26
AvgRunLength.txt.....	26
Biomass.txt	26
DiffuDist.txt	26
FracDim.txt.....	26
MaxThick.txt.....	27
MicAtSubst.txt.....	27
RaStar.txt	27
SurfArea.txt.....	27
ThickDist.txt.....	28
VolAtSubst.txt.....	28
Image Files	28
3D_Euclidean_Distance_Transform_of_[stack-channel].tif	28
Detected_Blobs_in_[stack-channel]_Colonies_leq_[xxx]_px.tif.....	28
Local_Thickness_of_[stack-channel].tif	29
Height_Matrix_of_[stack-channel].tif.....	29
Output_substratum_of_[stack-channel]_Colonies_leq_[xxx]_px.tif	29

F – Contents of Comstat2 system files	29
comstat2.ini	29
Comstat2.bat.....	30
ImageJ.cfg	30
Convert-to-ometiff.ijm	30
G – Known bugs	31
Version 2.1 1-July-2015.....	31
H – Acknowledgements.....	31
I – Trademarks	31
References.....	31

Definitions and terms

This manual uses several terms to describe certain features. Some of these terms have definitions in the context of Comstat2 that differ from the definitions in biology. Terms defined elsewhere in the list are highlighted by *italics*.

3D Volume: Physical 3-dimensional space: can be the entire confocal recorded space or down to a single *voxel*.

Biofilm: The collected *voxels* containing *biomass*.

Biomass: *Voxels* with a *grey level* above the *threshold*.

Bottom: The position closest to the physical substratum.

Connected Volume: *Biomass* containing *voxels* that have a physical connection to the *substratum*.

Grey Level: Each *pixel* in a confocal *image* has a signal intensity value. Since most *images* are recorded as 8-bit pictures the signal intensity value is between 0 and $2^8=256$. The actual value of a *pixel* is called the grey level.

Image: A flat representation of a picture, made up of *pixels*. An image has a certain number of pixels in the x- and y-direction.

Layer: See *Slice*

Noise: Noise in an image generally means that it contains false signals, which could arise from excessive amplification of a weak fluorescence in the sample (electronic noise).

Microcolony: A defined part of the *biomass*, which is characterized by pixels on the substratum having a given minimal number of, connected neighbor *pixels* containing *biomass*. The microcolonies further consist of connected *pixels* (*voxels*) in the *slices* above the *substratum*.

Pixel: The smallest 2D picture element. Each pixel has a physical extent in the x- and y-direction. See also *image*.

Slice: One *image* recorded a defined height relative to the *substratum*

Slice distance: indicates the vertical spacing of the *slices*. The slice distance is also referred to as the z-step size.

Slice Thickness: Per definition, there is no such thing as a slice thickness as each *slice* is simply a flat *image*. See *Slice distance*.

Stack: A group of ordered *slices* making up the recorded representation of a *3D volume*.

Substratum: The substratum is defined in Comstat2 as the bottom *image* of a *stack*.

Threshold: A given value for the *grey level*, for which Comstat2 assumes that *voxels* with a higher or equal *grey level* contain *biomass*, while those with a value below are empty.

Voxel: The smallest 3D picture element, it has the base of a *pixel* (x and y) and a height of the slice distance.

Z-step size: See *slice distance*.

Nomenclature

Filenames are written with **Arial Bold**, e.g. **Biomass.txt**

Text input or output is written with *Courier*

Menu items, function names and text in program dialogs are written with *italics*

Preface

Comstat is a program that analyzes image stacks of biofilms recorded by confocal microscopes. It can extract a number of quantitative parameters from biomass to fractal dimensions. The program does not contain any statistical calculation methods. A full list of functions is found in Chapter 4.

Comstat was developed by Arne Heydorn and Prof. Bjarne Ersbøll in 2000 [1] as a MATLAB® script and has for many years been one of the preferred tools for biofilm quantification. In 2008 Martin Vorregaard ported and expanded the software [2] so it would run independently from MATLAB® but instead to utilize the public domain software Java® and the Image processing environment ImageJ [3] with the add-on package LOCI Tools [4]. Comstat v. 2.1 is the latest build of this ImageJ plugin and this version was compiled and released in June 2015.

A word of caution: Comstat2 has many limitations and you should never rely on data from Comstat2 alone. You need always to use common sense when data look wrong and check your original files. Comstat2 cannot tell you if your images have undesired properties that can give misleading results, such as bright pixels or low signal intensity.

Although Comstat2 can analyze several channels from a single stack of images, the program does no correlation between the channels. This can lead to contradictory results, e.g. if some biomass is recorded in more than one channel due to fluorescence cross-talk then the total number biomass filled pixels in some slices may exceed the number of pixels available if you add up all the channels. Comstat2 is designed to operate on single channel grey scale confocal images.

Chapter 1 – Installation

We offer two solutions for installation: A do-it-yourself version where you download and install components yourself and one where we have done it for you and packed it all in a single zip (PC) or dmg (mac) file. If you intend to run Comstat2 under Linux that should be possible – follow the directions in the *Manual installation* section below (Linux has not been tested and is unsupported)

The Comstat2 software is composed of two files, **comstat2.jar** and an accompanying configuration file, **comstat2.ini**

Automated installation

1. Download the package appropriate for your platform, PC or Mac (see appendix B).
1. *PC Version:* Download **comstat2xx.zip** and unpack to a place convenient, e.g. the desktop. xx may be 32 or 64 depending on your operating system (32 or 64 bit).
Mac version: Download **comstat2.dmg** and double click it. From the image either drag the Comstat2 folder to the provided shortcut (Applications folder), or simply drag the folder from the image to the destination of your choice, e.g. to the desktop.
2. Verify the installation by starting ImageJ from the Comstat2 folder. The first time ImageJ may create some initialization files – just ignore that. After starting go to the menu point *Plugins* and select *Comstat2* from the drop down menu. Comstat2 should now open.
3. See appendix D – Troubleshooting, if you get an error message when doing any of the steps here.

Note: The pre-installed versions of Comstat2 include a runtime version of java, so you do not need to install it separately.

Manual installation

If you intend to manually install Comstat2 you will also need the following three software components: Java runtime environment, ImageJ and the image processing toolbox LOCI Tools. All three are in the public domain and can be freely downloaded (see box)

Where to find required third party software components:

1. JAVA: <http://www.java.com> *
2. ImageJ: <http://rsb.info.nih.gov/ij/download.html>
3. LOCI Tools: <http://downloads.openmicroscopy.org/bio-formats/5.0.8/>

(* You most likely already have java on your computer – see the java website for instructions on how to find out)

A copy of ImageJ and LOCI Tools are also stored at www.comstat.dk/software

1. Start by downloading all the components, Comstat2 files in the archive **Comstat2-components.zip** – see appendix B.
2. Install java if you don't have it already. Must be version 1.6 or above.
3. Unpack the ImageJ to a folder where you can find it – a common place is the desktop on your computer. Rename the ImageJ folder to "Comstat2"
4. Move the **loci_tools.jar**, **comstat2.jar** and **comstat2.ini** files to the subfolder "plugins" in the Comstat2 folder you renamed in step 3.
5. Optional: If you have a *Windows computer* you may also want to install the **comstat2.bat** file. This bat file contains start-up commands for ImageJ that allocates more memory than what is standard. If your java files are not in the execution path of your computer you must edit the bat file to include the full path of the **javaw.exe** file. See also the note below.
6. Optional: If you have a *mac computer* you might want to allocate more memory for ImageJ. To do so Right-click on the **ImageJ64.app** file and click *Show contents of package*. A Finder window will open and show a folder called "Contents". Enter the folder and double click the file **Info.plist**. Go to the bottom of the file and after the third but last `</dict>` enter this text:


```
<key>VMOptions</key>
<string>-Xms256m -Xmx768m -Xss32m</string>
```
7. Verify the installation by starting ImageJ from the Comstat2 folder. The first time ImageJ will create some initialization files – just ignore that. After starting go to the menu point *Plugins* and select *Comstat2* from the drop down menu. Comstat2 should now open.

Note: If you already have ImageJ on your computer, you could be tempted to install Comstat2 into that installation. We recommend that you don't do that unless you have a clean installation of ImageJ as many plugins from other developers have proved to conflict with Comstat2. In particular the popular version of ImageJ called Fiji (a.k.a. "Fiji is just ImageJ") has been shown to be incompatible in the past. Newer versions of Fiji may be compliant, but we do not recommend using it with Comstat2.

Note: Comstat2.bat is obsolete if you install a version of ImageJ which includes a run-time version of java. In that case configure ImageJ.cfg instead (see appendix F)

Note: This version requires LOCI Tools version 5.0.0 to 5.0.8 – newer or older versions will not work.

Chapter 2 - Running Comstat2 – The basics

Comstat2 takes image stacks created with confocal microscopes as source data and produces one or more text files as output. This is core functionality of Comstat2. Optionally Comstat2 can also generate one or more diagnostic images which can be examined in ImageJ.

The workflow can be summarized as follows: 1) select and load image stack files, 2) select or deselect individual stacks from the loaded files, 3) select analysis parameters and desired output data, 4) select automatic thresholding or manually threshold the image stacks, 5) run analysis, 6) examine output data.

Prior to Comstat2 analysis suitable images should be obtained. Thresholding is necessary to set the limit between biomass and noise. In practice, this is done by determining the gray-level that is the minimum value (threshold) for presence of biomass. In a selection matrix all gray-level values below the threshold are discarded (set to zero) and all above included in the analysis. After running Comstat2 it is recommended to subject the resulting data to statistical analysis [5].

Note: It is important to make sure that images are recorded in such a way that the substratum is recorded in image plane one or two, or output data may be flawed.

Input and output files

Before going on make sure you have files for analysis available in a supported format (see below). We recommend that you place all files for an experiment in a folder, from now on called the source folder. Since Comstat2 will load all images at the start of the run, we recommend that you limit the number of stacks/images in the source folders. You can select more than one folder at a time.

Comstat2 will generate output files in text format that will be placed in the folder containing the source image files. The text files are designed to be compatible with Microsoft Excel®.

Pixels and real life

Confocal as other digital images are made of pixels – voxels actually. A pixel or Picture Element is the smallest square part of the picture which has a finite grey value – a pixel cannot be partially black and partially white at the same time, it is either black or white. All pixels together make up the image, for a confocal image the layers are typically made up of 512 x 512 pixels or 262144 pixels. Knowing the microscope lens etc., each pixel will represent a corresponding area in the observed object. All the layers, or slices, in the confocal image are placed on top of each other and for each pixel on the substratum layer, there will be one or more pixels above it in the stack of layers. Each such volume element is referred to as voxels. The width (x) and length (y) of the voxel is the side lengths of the pixel and the height (z) is the distance between the slices. Together all voxels represent the 3D volume image.

Comstat2 measures physical extents of the biofilm in x-, y- and z-directions. Since the 3D volume is represented by voxels certain approximations and assumptions are necessary. In Fig. 1 are two cross sections (x-z) of two slightly

different biofilms, A and B. Marked with red is the cross section of the actual extent of the biofilm. The black squares represent the voxels recognized by Comstat2 as filled with biomass. From the pictures, it is evident that the two real biofilms have different heights. However, since we can only work with voxels Comstat2 sees them as identical. Also on the figure it is shown how heights are reported. In this experiment, the maximum height will be 4 μm in both biofilm A

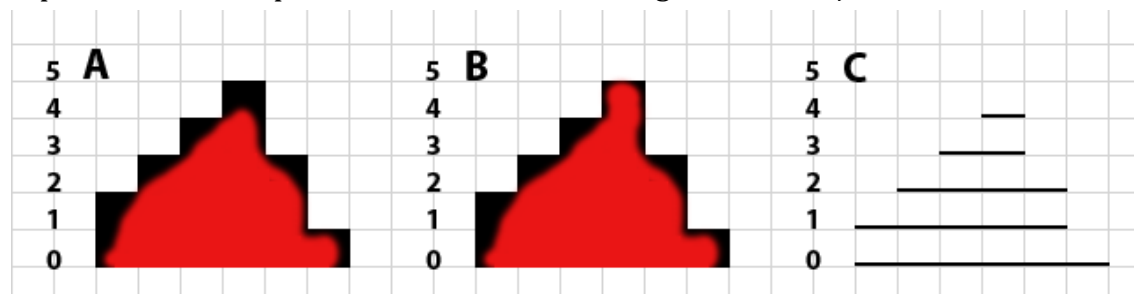


Fig. 1. Digital representation of a real biofilm (red). Numbers are height in μm . Seen from the side.

and B (Fig. 1). A confocal image stack consists of a number of individual flat images. Therefore, Comstat2 looks at data as represented in panel C in Fig 1. For the same reason, pixels, which are only occupied by biomass on the substratum layer and not in layers above, will have a reported height of 0.

Supported file types

Comstat2 was designed to support confocal images obtained from Leica TCD4D (a legacy format supported partially to ensure backwards compatibility with Comstat1), Leica LIF format and Zeiss LSM format. Due to the continuing development from the hardware vendors Comstat2 may not support newer implementations of the LIF and LSM formats. The Leica TCS4D has been discontinued a decade ago, and the file format is not directly supported by Comstat2 version 2.1 and later. To future proof Comstat2 and to allow a wide variety of other confocal file formats the latest version (version 2.1) of Comstat2 supports the Open Microscopy Environment file format OME-TIFF [4]. See Appendix A for how to convert and import to OME-TIFF.

Running Comstat2

Start by opening the folder containing ImageJ and double click the **Comstat2.bat** file (PC) or the **ImageJ64.app**. When ImageJ has launched, go to the menu *Plugins* and find the menu item *Comstat2* and click it (Fig 2). On Windows machines, the menu is included in the ImageJ dialog box. On Mac computers, the menu bar is located at the top of the computer screen, detached from the ImageJ window. If *Comstat2* is not seen in the menu, it has not been installed properly, - see Appendix D - troubleshooting.

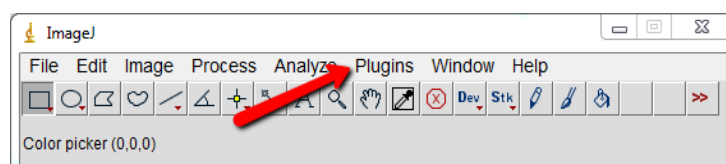


Fig 2. ImageJ main window (MS Windows)

You should now see a “Splash Screen” showing version info and three pop-up windows: *Comstat2*, *Observed Directories* and *Log*. You can remove the splash

screen by clicking *OK*. If the Comstat2 window does not look like Fig 3, the comstat2.ini file is not installed properly - see appendix D for troubleshooting.

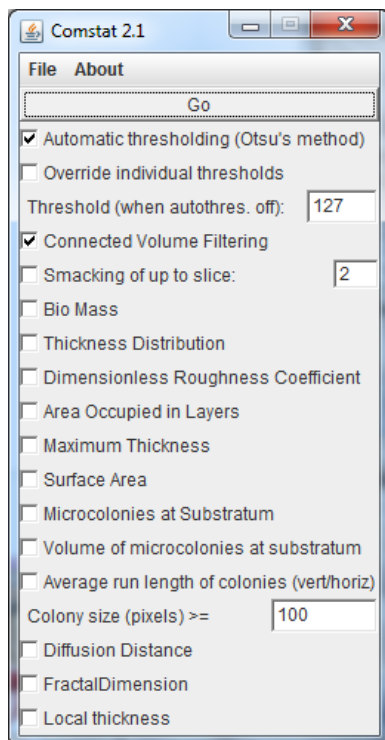


Fig 3. Comstat2 main menu (Control Panel)

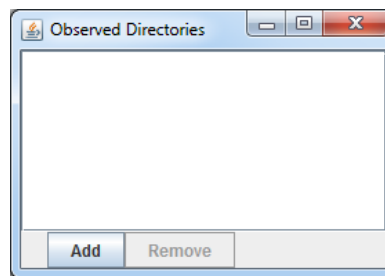


Fig 4. Observed Directories

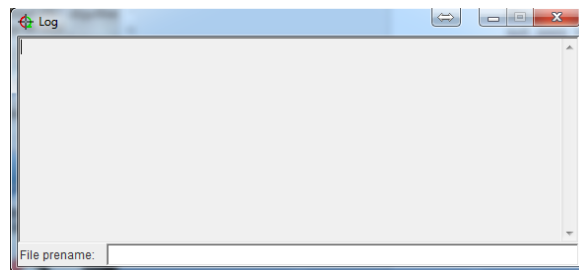


Fig 5. Log window

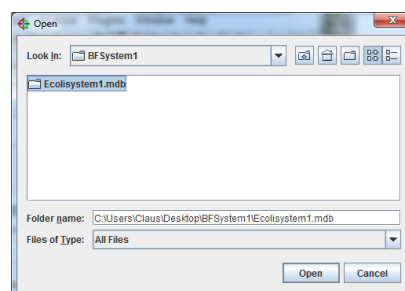


Fig 6. Selecting source folder (PC)

Selecting source folder(s)

Next tell Comstat2 where you have put the source files for analysis. Use the *Add* button in the Observed Directories menu to navigate to the folder containing the source files. Highlight the folder but do not double click it to enter it – see Fig. 6 for an example, where the source files are located in the folder **Ecolisystem1.mdb**. After clicking *Open* Comstat2 will load all stacks and display the stacks in the window *Images in Directories* (Fig. 7).

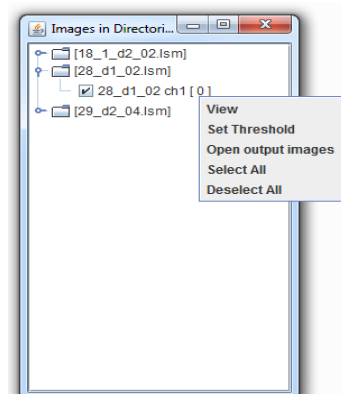


Fig. 7. Images in Directories (PC)
The pop up menu is shown.

The Images in Directories window

The Images in Directories window (Fig. 7) lists all the files that have been loaded. If a file contains more stacks and/or than one channel they are collapsed under the file name. Clicking the key (PC)/triangle(Mac) symbol expands the list to display all contained stacks and channels. A check mark indicates that a channel is scheduled for analysis. Right click on the name of the channel to display the pop-up menu. From the pop-up

menu a number of options are available: 1) View: this menu opens a new window displaying the channel (in grey scale). A slider below the image allows for browsing the slices in the channel. 2) Set threshold: This menu allows you to set manual threshold for a channel (see below). 3) Open output images: This menu is only active after analysis. It will open the generated output images. The output images are mainly intermediate images for data analysis internally in Comstat2 and is usually of little interest to the end user – see also Appendix E – Output files. 4) Select All and Deselect All: These menu points will select or deselect all channels in all stacks in the window.

Selecting basic analysis conditions

For a standard analysis of images of good quality, you do not need to change any of the parameters. However, you should check that the correct decimal separator has been set. Per default, Comstat2 will check the system settings and adjust the decimal separator accordingly. To check and if needed change the decimal separator from “.” To “,” or vice versa, click *Edit* in the Comstat2 control panel (Fig. 3). Leave *Automatic Threshold (Otsu’s method)* and *Connected Volume Filtering* checked (see also Chapter 3 for explanation).

Selecting basic analysis parameters.

You can choose any parameters you like and leave all other options unchecked. The most basic would be just to select *Biomass*.

Performing the analysis

When you are ready, click *Go*. Comstat2 will now analyze all images loaded in the *Images in Directories* window. After Comstat2 completes its run, the message *Done with selected images* is displayed in the log window. The results are saved in individual text files in the same directory as the image stacks. The files are formatted to enable easy loading into Microsoft Excel® by right-clicking the output file to be examined and select *Open with... -> Microsoft Excel*

Chapter 3 - Running Comstat2 - Advanced use

Setting advanced analysis conditions

While most parameters may be sufficient for a quick analysis usually you will want to change some settings. This section describes the things you can change to make the Comstat2 analysis better suit your needs. The settings are described in detail in Chapter 4 – Functions in Comstat2.

Automatic or manual thresholding

Comstat2 offers two ways to set the threshold in biofilm images, automatic (Otsu) thresholding [6] and manual.

The automatic threshold is set by selecting the option *Automatic Threshold (Otsu's method)* in the Comstat2 control panel.

Note: The automatic thresholding is selected per default.

To manually set a threshold value, there are two options: A global threshold, for all images in the analysis queue, and individual thresholds for each stack. For any manual threshold to be in effect, you need to first deselect *Automatic Threshold (Otsu's Method)*.

The global threshold is set in the Comstat2 control panel by specifying a value in the *Threshold (when autothres. off)* input box (Fig 3). This can be combined with individual thresholding of individual images. To set the threshold of each stack, click on the triangle symbol (mac) or key symbol (Windows) next to the stack you want to threshold in the *Images in Directories* dialog box (Fig. 7). Then, right click on the channel name and a new pop-up dialog is displayed. Click *Set Threshold*. Two image windows and a slider box are displayed. The left window shows the normal image, the right window shows the thresholded image. To set the threshold start by moving the *Slice* slider to a slice where you want to set the threshold for the image. Then move the *Adjust Threshold* slider (Fig. 8) to a position where you think a suitable threshold is applied. On the right (thresholded) image you should see an image showing only the biomass and no noise. When you are satisfied, click *Save*. On the bottom of each of the images there is also a slider bar. Do not use it for the thresholding procedure, it is there because Comstat2 uses the ImageJ build-in image viewer for displaying the two picture windows – use the slider shown in the threshold pop-up (Fig XXX). You will need to manually close the three dialog boxes (the slider box and the two image boxes). Any picture, which you have not specified an individual threshold for will be thresholded with the global value set on the Comstat2 control panel.

Note: Compared to Comstat1 the threshold in Comstat2 is 1 grey scale unit lower. E.g. when a threshold of 100 is set in Comstat1 the same analysis results are found in Comstat2 with a threshold of 101. This change is implemented to allow for an accurate use of the automatic thresholding algorithm. Using manual threshold the difference will be less than what is introduced due to operator bias.



Fig. 8. The threshold slider

Connected Volume filtering

By selecting the “Connected Volume filtering” option (CVF) all biomass must be connected to the substratum. Fig. 9 illustrates the meaning of the connected volume filtering. The red and blue biofilm is connected to the substratum in the field of view. The grey parts are not, here illustrated by a free-floating clump, and a part of the biofilm that is connected to the substratum but outside of the field of view. The CVF is useful for preventing unrelated or detached biomass from being included in the analysis. Occasionally, the first couple of slices are not containing biomass at all. In this case the CVF will result in nothing at all being measured since nothing is connected to the substratum (as it is defined as the bottom slices in a stack). A simple remedy is to uncheck CVF.

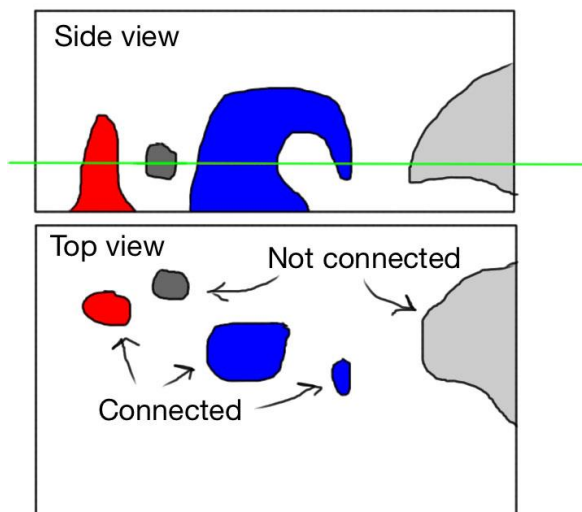


Fig. 9. Connected volume filtering.

Smacking

Sometimes the substratum is not completely level, resulting in a skewed image stack. In this case the occurrence of the substratum will happen first in one part of the first image, then in another part of the second image etc (See Fig 10). This will result in faulty analysis. To resolve this problem you can use *Smacking*. This function takes the specified number of image layers and for each pixel on the substratum determines the maximum value. This value is saved in the same pixel position on the substratum (first) layer. No other layers are affected.

Note: If you experience a skewed substratum it is preferable to re-record the stacks with a level substratum if possible.

Note: It is not recommended to smack more than two layers, or unpredictable results may occur.

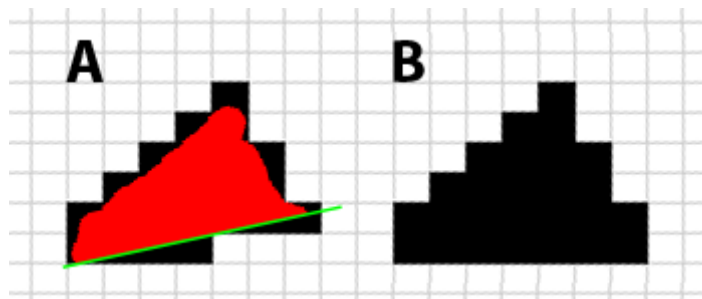


Fig. 10. Effect of smacking. Red: Real biofilm. Green: substratum. A: before smacking, Black: pixels recorded. B: After smacking two layers, Black: smacked pixels.

Minimal colony size

The minimal colony size is the minimal number of connected pixels that are required to be recognized as a microcolony. A value of 100 means that an area of 10x10 or 5x20 pixels, etc., is the minimal area that needs to be covered by biomass before Comstat2 will record the area as belonging to a microcolony.

Selecting Analyses to be performed

In Comstat2 it is possible to select many different types of data to be calculated from the source files. Which ones you select depend on what you want to analyze. Some parameters are inter-connected, meaning that some parameters in reality provide redundant data, just presented in different ways. For instance, substratum coverage is the same as biomass at substratum, just expressed with different units. All output types are described in detail in Chapter 4 – Functions in Comstat2.

Customizing output

Decimal separator

Per default Comstat2 will attempt to detect the decimal format of the machine on which it is running. E.g. in the US the decimal separator is a point ("."), while in most European countries it is a comma (","),. If you wish to change the decimal format, go to the Comstat2 control panel, select *File* and check *Use Decimal comma* for comma format and uncheck it for point format.

File name

Per default Comstat2 will name all files with standard names, such as **Biomass.txt**, etc. You can define a new file name by specifying a text that will be prepended to the output file names. It is done in the input field under the log window. Remember to add a dash after the text if you want a visible separation of the standard name and the text to be prepended. Example: If you specify **Run2-** in the input field, the biomass output file will be named **Run2-Biomass.txt**, etc.

Appending data or make new output files

Per default Comstat2 will append all analyses to existing output files. If you want a clean output file you can tell Comstat2 to create new output files. To do this, go to the Comstat2 control panel, select *File* and uncheck *Append to output files*. **Note:** This will delete existing output files unless you specify new file names (see *File name* above)

The log window

The log window contains all calculated data generation during a session. The contents of the log window is saved in the file log.txt in the user home directory (In a Windows machine that is typically *C:\Users\Username* on a Mac it would be *~Username*, where username is the actual username of the logged in user). When Comstat2 does not have any jobs in queue the log window will output *Done with Selected Images*.

At the bottom of the log window it is possible to define a text which will be prepended to all text output files (See the paragraph *File name* above).

Chapter 4 – Analysis Functions in Comstat2

Biomass

The *Biomass* is determined as the volume of all voxels that contain biomass divided by the area. Comstat2 counts all voxels above threshold. The number of voxels is multiplied by the height (z) and width (x) and length (y) in μm of a voxel (=Biovolume). The biovolume is divided by the area of the substratum in order to obtain a value independent of the size of the observed area.

The unit is $\mu\text{m}^3/\mu\text{m}^2$

Maximum Thickness

The *Maximum thickness* is defined as the thickest piece of the biofilm taking holes and cavities in the structure into account. In other words, the maximum thickness is the highest point of the biofilm relative to the substratum, minus the empty space. Fig. 11 illustrates how maximum thickness is calculated. In both biofilms the maximum thickness reported by Comstat2 will be 4 as the thickness is calculated as the highest number of biomass containing pixels in a column.

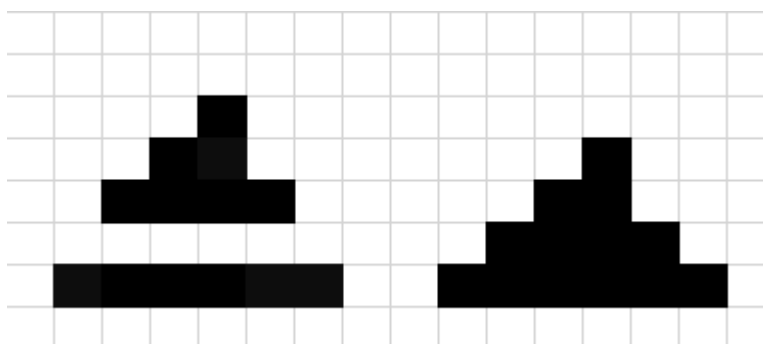


Fig. 11. Maximum thickness. Side view.

Note: If *Connected Volume Filtering* is selected, the reported maximum thickness will be 1 in the left biofilm and 4 in the right biofilm (Fig. 11)

For a representation of the average biomass thickness, see *Thickness distribution*. The unit is μm .

Thickness distribution

The *Thickness Distribution* (TD) will report the distribution of heights of the biofilm, i.e. how large a part has a maximum height between two set boundaries.

Visualize the observed space as a cube. The cube consists of several layers, one for each confocal slice. Each layer is divided into smaller squares, pixels. For each pixel in a layer there is one above or below or both. These vertically connected pixels we can call a column.

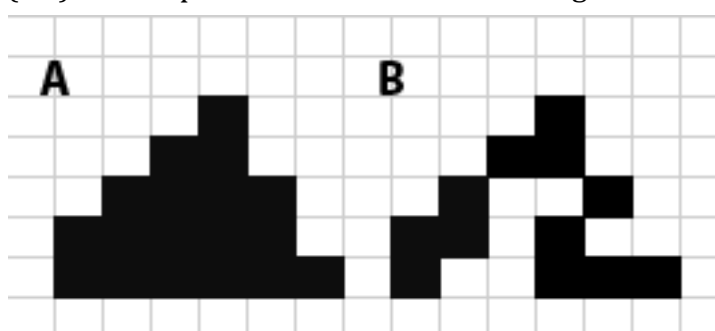


Fig. 12. Thickness distribution (TD). Biofilm A and B have identical TD. Side view.

For each layer, starting from the top, TD determines if the pixels contain biomass or not. If a pixel is determined to contain biomass (i.e. above the set threshold), that pixel and all below are marked as containing biomass. If there is a void below it is ignored (therefore biofilm A and B in Fig. 12 are giving the same TD). In other words, TD scans down the cube and for each column it determines the topmost biomass-containing pixel. Since the confocal cube contains a finite number of layers TD can determine the height of that layer. If the pixel height (z) is say 2 μm and we have five slices, we will have heights 0, 2, 4, 6 and 8 μm (the bottom of each pixel above the substratum). TD will then count all the heights and put them in boxes corresponding to the, in this case five, possible heights. In real life, of course, the height distribution is dynamic and can take any value between 0 and the height of the cube, but due to the slices in the confocal stacks Comstat2 outputs the height distribution in a number of finite heights corresponding to the z-position of each slice.

The output of TD: The first line contains the (bottom) height of each layer. The second line ("spots measured") gives the number of columns having the corresponding height (i. e. the "top of the biomass" determined in the first step). The % is the fraction of spots having the height compared to all spots (the total number of pixels in each layer). The average thickness is the average height of all columns taken together. The average thickness is presented for the entire observed area and for biomass containing columns (not including empty columns). The average height unit is μm . Spots are in pixels or % of the total number of pixels at the base.

Note: Due to the nature of the way data are represented in confocal slices (see Chapter 2, and Fig. 1) any biomass detected in layer 0 (substratum) is assigned a height of 0 (zero). Therefore a stack with some signal in slice 0 alone will have the same TD as a stack without any signal at all.

Dimensionless Roughness Coefficient (R_a^*)

Roughness coefficient (R_a^*) is a measure for the variability in the height of the biofilm. The formula to calculate the R_a^* is very simple [7, 8]:

$$R_a^* = \frac{1}{N} \sum_{i=1}^N \frac{|L_{fi} - \overline{L_f}|}{\overline{L_f}},$$

where L_{fi} is the i 'th individual thickness measurement, $\overline{L_f}$ is the average thickness and N is the number of thickness measurements. In Comstat2, the number of thickness measurements equals the number of pixels in the substratum, as the thickness is determined for each pixel in the observed area. The *Roughness coefficient* has no dimension.

Area Occupied in Layers

This function measures the biomass pixels in each layer (confocal slice) and for each layer reports the position of the lower boundary of the layer, area occupied by biomass and the number of pixels containing biomass.

The unit for area is μm^2

Surface Area (includes Surface/Area calculation)

The surface area and surface area calculation is performed by examining all voxels in the stack with and count the surfaces which do not: have a biomass filled neighbor, face the substratum, or face a boundary of the confocal stack. All such surfaces are added up to give the total surface facing the void. The volume is derived from the *Biomass* function. The units are μm^2 for surface area and $\mu\text{m}^2/\mu\text{m}^3$ for surface/volume.

Note: The surface calculation includes any enclosed surfaces inside microcolonies.

Microcolonies at Substratum

The *Microcolonies at Substratum* function finds all connected areas on the substratum layer of a size equal to or larger than the number of pixels defined in the box *Colony Size (pixels)* in the Comstat2 control panel (Fig. 3). Each microcolony is registered and numbered internally. The individual marked microcolonies are used for volume and run length calculations. Output reports Minimum defined colony size in pixels and μm^2 , the number of detected microcolonies, and for each micro colony the internal colony label (number), the size in μm^2 , the number of colonies of that size, and the average micro colony size in μm^2 , and the average pixel intensity of the microcolonies.

Note: The individual microcolonies are marked with different shades of grey in the Output image **Detected_Blobs_[stack_channel]_leq_xxx_px.tif** (xxx being the minimum selected colony size (default **100**), which can be observed by selecting *Open output images* from the drop down menu on each stack in the *Images in Directories* window. It can be beneficial to choose another look-up table (LUT) to see the differences. This is done in the menu *ImageJ -> Image -> Lookup Tables* with the Detected Blobs image active.

Volume of Microcolonies at Substratum

In this function, the microcolonies detected by the *Microcolonies at Substratum* function are measured with respect to volume. The sum of all volumes equals the biomass volume (as is also reported in the *Biomass* function). The unit is μm^3 .

Average Run Length of Colonies (vert/horiz)

The *Average Run Length* function will determine the longest and the average distance covered by biomass in the x- and y-directions on the substratum. The unit is μm .

Note: Works only on the substratum layer

Diffusion Distance

The diffusion distance is the length of the shortest path from a voxel inside the biomass to the surface. The *Diffusion Distance* function will determine the average diffusion distance for all voxels of the biomass, and the longest diffusion distance. The diffusion distance is determined in 3D. The unit is μm .

Fractal Dimension

The fractal dimension is a measure of the complexity of the circumference of the Microcolonies/biomass at the substratum. The parameter stems from the observation by Mandelbrot that a complex border line, like the coast line of a piece of land can be described as a number of short straight “yard sticks” put

together. The shorter the yard sticks the more are required. Mandelbrot predicted that the number needed to accurately measure the coast line is indefinitely as is the actual length of the coastline [9]. If you imagine the circumference described as a polygon with a side length of η then the polygon would have $\lambda\eta^{-D}$ sides, where λ and D are constants. The constant D is called the fractal dimension. The fractal dimension has turned out to be characteristic for the shape of the polygon it describes [9]. While D can be difficult to calculate, 1-D is not, and the algorithm in Comstat2 calculates D this way by an algorithm described by Flook [10]. For a 2D shape $1 \leq D \leq 2$. A perfect circle has a $D = 1$. The fractal dimension has no unit.

Note: Works only on the substratum layer

Local Thickness

Local Thickness is a function that tries to fit a sphere inside the biofilm, and then move the sphere around to be positioned in turn, at all voxels as the centrum. The output is an image showing the accumulated overlay of all these spheres. For each sphere, the radius gives the longest distance from the given voxel to the nearest surface. For this reason *Local Thickness*, like *Diffusion Distance* gives a better estimate of local biofilm density than the other thickness parameters in Comstat2.

Note: What can *Local Thickness* be used for? Nothing, really. Comstat2 was developed as masters project and as part of it the *Local Thickness* algorithm was included as an exercise.

Note: To see the output file after running Comstat2, right click the stack you want to see in the menu *Images in Directores* and select *Open Output Images*. The Local Thickness output image is named **Local_Thickness_of_[stack_channel].tif**

Chapter 5 – What's next?

After having run the analysis with Comstat2 it is important to judge the data being output. If something looks odd, it is highly recommended that you open the source images and verify that they look as expected and does not have obvious irregularities that could be the course for the discrepancy from what you expected. For instance, particular noise e.g. stemming from stray dye crystals could be destroying the data. Also, like in any other biological experiment one run can not stand alone. Preferentially, a thorough statistical analysis should be performed on multiple repetitions of experiments carried out under identical conditions. An early example demonstrating how to perform statistical analysis can be found in Heydorn et al 2000b [11]

APPENDIX

A – Using the OME-TIFF file format

To set a new standard file format that could be read by not only Comstat2 but also by a wide range of the other software packages available, Goldberg and coworkers devised a proposal for a new common microscopy file format, The Open Microscopy Environment [12]. Using this OME format most proprietary file formats can be converted to the common format, making it much easier for software developers to read and process files from diverse vendors.

Comstat2 now supports ome-tiff files, which are read natively. This ability relies on the LOCI tools [4] that are installed as a prerequisite of Comstat2. As an added bonus LOCI tools also enables tools for conversion of third-party proprietary microscopy file formats to ome-tiff. The current version of Comstat2 natively reads OME-TIFF, Zeiss LSM and Leica LIF files. Leica TCS4D format is now supported via OME-TIFF (see below).

Note: Comstat1 only directly supported Leica TCS4D formatted image stacks [1], and Comstat2 was originally designed to support Leica TCS4D, Leica Liff and Zeiss LSM formatted files [2].

In this section we describe the steps needed to convert a foreign file format to ome-tiff using the LOCI tools.

Converting to OME-TIFF using the LOCI Tools (installed as part of Comstat2)

To use confocal images of a format not directly supported by Comstat2, you need to convert them to OME-TIFF. Fortunately, the LOCI tools installed with Comstat2 makes it very easy to do so. Unfortunately, in most cases you will need to manually rename each stack you want to use in the process.

The process is straightforward:

- 1) Make a folder to hold your converted images (optional)
- 2) Copy the source files to the folder
- 3) Go to *plugins -> Bio-Formats -> Bio-Formats Importer* (see note)
- 4) Navigate to the file you want to convert and click it
- 5) The *Bio-Formats Import Options* menu is displayed. Just use default settings (see Fig XX)
- 6) Click OK
- 7) If more than one stack ("Series") exists in the file, a list of the available stacks is displayed. Select all that you want to convert. Then click OK. (*Series* " is selected per default, unselect it if you don't need it)
- 8) A new window is opened for each stack (This is part of the LOCI tools behavior).

Note: if the program prompts you to check for an updated version we recommend you ignore this and proceed. We cannot guarantee that other versions of LOCI Tools than the one that was delivered with Comstat2 works

Next step is to save the stacks in OME-TIFF format. For each stack do the following:

- 9) Select the window with the stack.
- 10) Go to *Plugins -> Bio-Formats -> Bio-Formats Exporter*
- 11) Type in a file name (this part is a bit tedious, but it is due to the LOCI tools and outside of our control). A good name could be the title of the stack (see the bar above the stack image). Click OK.
- 12) If there is more than one channel in the stack, select Save each channel separately, in the next dialog box.
- 13) Select "Uncompressed" in the final dialog box.
- 14) Close the stack image you just converted.
- 15) Repeat steps 9-14 for the remaining stacks.

The files are now saved in the same folder where you placed the source file and ready for Comstat2.

Using a macro to convert images

We provide a simple macro (*convert-to-ometiff*) for batch conversion of image stacks to OME-TIFF, courtesy of Dr. Melissa Linkert – see Appendix F. If your installation was automatic (Chapter 1) then the macro is already installed and available from the *ImageJ -> Plugins -> Macros -> convert-to-ometiff*.

If you have manually installed Comstat2 you can install the macro by going to *ImageJ -> Plugins -> Macros -> Install Macro...* and point to the file **convert-to-ometiff.ijm**.

When you run the macro you have to specify the source file, and then the macro will convert all stacks and channels in the file to one ome-tiff file per channel. The files are saved in the same directory as the source file.

Converting multiple TIFF images to an OME-TIFF stack

If you do not have a confocal image file, but just a number of individual TIFF images which you want to combine into a 3D stack for Comstat2 analysis, that is also possible. The prerequisite is that you have uncompressed TIFF images, in grey-scale (single channel) and they are ordered consecutively with the image closest to the substratum having the lowest alphabetic name or number*. All images must have a **tif** or **tiff** file name extension. Finally, you need to know the pixel dimensions in μm .

The following steps should be performed:

- 1) Place all files to be combined in a folder. Make sure no other image files are present in the folder.
- 2) In ImageJ choose: *File -> Import -> Image Sequence*
- 3) Select the folder you defined in step 1
- 4) Files are per default ordered alphabetically. If you want to use the numbering scheme e.g. used by the old Leica TCS4D format, select sort names numerically*.
- 5) Click OK
- 6) At this point it is important to define the image properties: *Image -> Properties*
- 7) Specify unit of length (μm), pixel width and pixel height (x and y dimension of one pixel, in μm) and pixel depth (z dimension).

- 8) Export the stack to an OME-TIFF file. See steps 9-14 in the paragraph above about converting stacks using LOCI tools (page XXX)

* Numerical ordering means that files are ordered e.g. **abc8.tif**, **abc9.tif**, **abc10.tif**, **abc11.tif**. ... in contrast to the alphabetical ordering which with the same files would be **abc10.tif**, **abc11.tif**, **abc8.tif**, **abc9.tif**.

Note: If you use this procedure for converting Leica TCS4D images, you can find the pixel dimensions in the accompanying **.info** file.

Converting to OME-TIFF using Imaris 8.0

Imaris has an export function that allows you to export confocal stacks and save them in the OME-TIFF format. At the time of writing this manual, the Imaris (version 8.0.3 and lower) implementation is not 100% compliant to the OME-TIFF specification and exported images will not load in Comstat2.

B - Obtaining Comstat2

Comstat2 is public domain software. This means that the software is free for you to obtain and use (no license fee or royalties required). However, for us to keep track of who gets the software, and to be able to alert you of future updates, we ask that any end-user that obtains the software will contact us via e-mail to comstat@comstat.dk. Also, we require that in the e-mail you acknowledge that you do not pass on the software, but ask any new user to go to www.comstat.dk and read the conditions and obtain the software from there. The conditions mentioned here are the same as written on the web-page.

C – Terms for use (how to reference Comstat2)

If you use Comstat2 in any work, such as but not limited to peer reviewed papers, we require that you acknowledge the use of the program. At the time of writing this manual we are preparing a paper describing Comstat2, but no such paper exist at this time. Until that paper is published we ask that use these references: Heydorn et al (2000) [1], Vorregaard 2008 [2], and <http://www.comstat.dk>.

After publication of the paper describing the Comstat2 program, we will put the reference on the comstat2 web site (www.comstat.dk) along with an updated manual and instructions for referencing the software. *Please note that the instructions on www.comstat.dk have precedence over this manual.*

D – Troubleshooting

Installation issues

ImageJ does not start (MacOSX)

Due to the application sandboxing in Apple computers with MacOSX 10.9 or higher, you may not be able to run ImageJ directly (this is particularly true if you choose the automated installation (Chapter 1). If you get a warning that the *image is corrupted*, just right-click **ImageJ64.app** and select *Open*. Then you will

be prompted to confirm that you allow the safety feature to be overridden. From then on you should be able to run ImageJ without this problem. If you do not get the prompt to override, you must temporarily disable the sandboxing feature: Go to the Apple menu (very top left of the screen) and select *System Preferences*. Then click *Security and Privacy*. Then select *Allow apps downloaded from: Anywhere*. Start ImageJ – it should now start. After this, go back to the *System Preferences* and reset it to the previous setting,

comstat2.ini missing

If you get the error message “Oops. Comstat2.ini could not be found” and cosmtat2 control panel only shows the “Go” button, you need to check for the presence of the file. For Comstat2 to correctly display and enable all features the file **comstat2.ini** must be present in the plugins folder. The file is a simple text file containing the names of the functions that should be enabled. Common causes of this error is if you downloaded the file directly from a web site with an Apple computer – in some cases the file is renamed behind the scenes to **comstat2.ini.txt**. To check, right-click the file and select *Show Info*. In the dialog box check the “Name and Extension” which should read “comstat2.ini”

Java is not found (Windows)

If the ImageJ program does not launch when you double click the Comstat2.bat file but you just see a short flash of a black (command) window, the most probable cause is the path to the java installation (required for ImageJ) is not defined. To resolve this problem, you need to add the path to the system variable %PATH% or to add the path to the **Comstat2.bat** file (Appendix F). Follow this link to get information on how to set the path:

<http://www.java.com/en/download/help/path.xml>

File selection issues

Images opened in ImageJ are not analyzed

If you open an image stack in ImageJ it may show on the screen. However, when the *Go* button is clicked in Comstat2, nothing is analyzed. This is because you need to add images via the *Observed Directories* dialog box. Otherwise Comstat2 will not know what images to analyze.

No files are shown when selecting directories

Due to the way the selection dialog box (*Observed Directories*) is designed in Java, you will only see the names of the directories, not the contents of the directories (which will appear empty). This is a Java bug which we can not do anything about.

When the directory is selected and it contains compatible files, still no stack are loaded in the program

It is important that you do not *enter* the directory when you navigate with the *Observed Directories* dialog box. Just go to the directory and highlight it – then press *Add*.

When I added a directory correctly, still no stacks are loaded

Comstat2 will only load files which are of the type **lsm**, **lif** or **ome.tif**.

Nothing is analyzed even if the files are shown in the Images in Directories box

When adding a directory, which contains lif, lsm or ome.tif files you can see the file name in the *Images in Directories* dialog box, but nothing is analyzed when you run the program. If you click the triangle (Mac) or key symbol (Windows) next to the file name, the symbol disappears where you would expect the stack list for that file. This problem can happen if the file contents is not acceptable by Comstat2: For instance, newer versions of the Leica LIF format do not comply with Comstat2 while older do. Likewise some **ome.tif** files cannot load correctly, if they are generated with a program that does not 100% adhere to the OME-TIFF standard (an example is Imaris prior to version 8.0.3). Another situation resulting in the same problem is if the file contains “stacks” without a z-dimension, such as xy-snapshots (single planes) or 3D-renderings (pertains primarily to Leica LIF files).

Input data quality issues

Sometimes your source data does not have sufficient quality to allow for efficient Comstat2 analysis. In most cases, unfortunately, there is little else to do than try to obtain new data. However, in some cases you can actually save your data.

Wrong setting of the substratum layer when recording stacks

Comstat2 relies on the first slice in a stack to contain the substratum layer. If the layer is recorded below the actual substratum layer, the first couple of slices may be empty. For some parameters like *Biomass* this problem can be surpassed by deselecting *Connected Volume Filtering*. But some parameters cannot be calculated without biomass in the first slice (or couple of slices if *Smacking* is selected), this includes *Microcolonies at Substratum*, volume and area calculations, *Fractal dimension* and *Run length*.

To resolve the problem, simply delete the slices under the substratum from the image stack and save it as OME-TIFF. ImageJ contains functions for stack manipulation and this can be utilized for this purpose.

- 1) Follow the procedure listed in the paragraph *Converting to OME-TIFF using LOCI-Tools* (Appendix A) to open the stacks you want to edit
- 2) Click the stack you want to modify and inspect it using the slider under the image.
- 3) Put the slider at the slice just below the substratum (i.e. the first slice you want to delete)
- 4) Go to *Image -> Stacks -> Delete*
- 5) Repeat until all slices below the substratum are deleted
- 6) Save the edited stack using the procedure listed in the paragraph *Converting to OME-TIFF using LOCI-Tools* (Appendix A), steps 9-14.
- 7) Now you can use the modified stack for analysis.

Dealing with “noisy” stacks

Sometimes when the fluorescence signal is low it is necessary to increase amplification during image acquisition. In such a scenario a higher level of noise is recorded, which could give rise to false signal. Since noise normally is in form of single bright pixels it can to a certain extent be eliminated by median filtering. Median filtering is a process where each pixel grey scale value in an x-y slice is recalculated as the average of the surrounding pixels. For instance with a radius

of 1 pixel each pixel will be given the average of the pixel its self and the neighboring pixels. Fig. 13 illustrates median filtering on an area with one noise pixel (A) and an area with a cell (B).

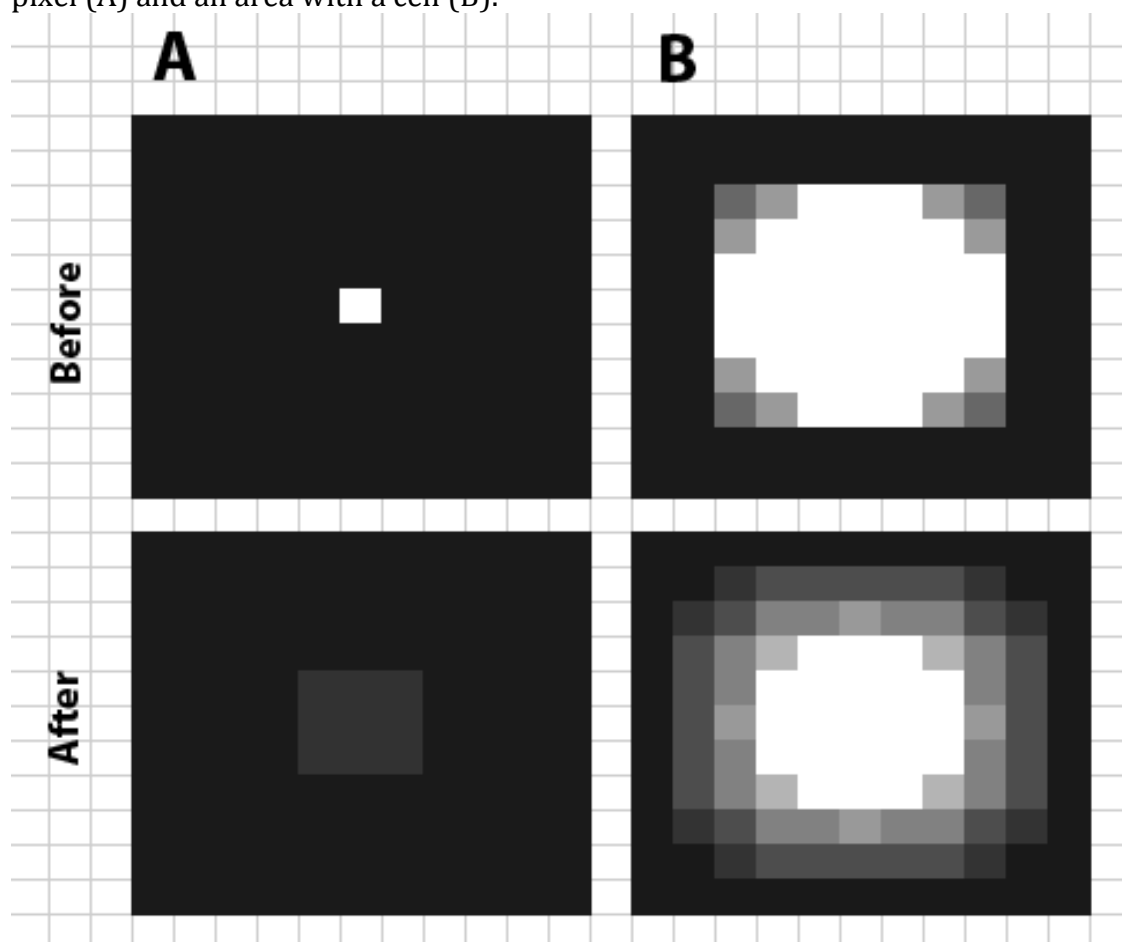


Fig. 13 Effect of median filtering. A: Single pixel noise, B: Bacterial cell. The images are shown before filtering and after applying a 1-pixel median filter. Top View.

E – Output files

Comstat generates a number of text and image files. In general, the text files contain numerical output of the analyses while the image files contain intermediate data needed for Comstat internal calculations.

Text files

In general, please first refer to chapter 4 for a description of each analysis function.

All text files has the parameter description in the first row. Then all files will contain the creation date and time, and for each data point the time for the addition of that particular point. This can be useful if you do not used the prepend-filename feature to differentiate the runs of the program. See Chapter 3 – Filename for details on how to change the names of the output files.

AreaOcc.txt

Calculation of the area that is filled with biomass in each layer of the confocal stack. Generated when selecting *Area Occupied in Layers*

Field	Contents
Layer	Layer number (counted from substratum)
Distance (μm)	Distance from substratum in μm
Percent	Percent of the layer filled with biomass
Area (μm^2)	Area of the filled part of the layer
Pixels	Number of pixels filled in the layer

AvgRunLength.txt

Calculation of the average vertical or horizontal cross section of each micro colony on the substratum. Generated when selecting *Average Run Length of Colonies*.

Field	Contents
Colony Id	Colony ID (Comstat2 internally numbers all micro colonies it detects on the substratum)
Average horizontal run length of colonies (μm)	Run length in the x direction in μm
Average vertical run length of colonies (μm)	Run length in the y direction in μm

Biomass.txt

Calculation of the biomass in the entire stack. Generated when selecting *Biomass*.

Field	Content
Biomass ($\mu\text{m}^3/\mu\text{m}^2$)	Volume of biomass divided by the area of view

DiffuDist.txt

Calculation of the longest and the average distance from any biomass voxel to the surface of the biomass. Generated when selecting *Diffusion Distance*.

Field	Contents
Maximum diffusion distance (μm)	The longest distance from any voxel in the biomass to a surface
Average diffusion distance (μm)	The average distance from any voxel in the biomass to a surface

FracDim.txt

Calculation of the fractal dimension of the biomass on the substratum. Generated when selecting *Fractal Dimension*.

Field	Contents
Average fractal dimension	Fractal dimension (2D)

Note: If the shape of the biomass on the substratum is very complex the Fractal Dimension function may run out of Stack Space for the calculation. Then the calculation may be inaccurate. You can increase stack space by teaking the -Xss parameter for ImageJ (See chapter 1 – manual installation).

MaxThick.txt

Calculation of the maximum thickness of the biofilm. Generated when selecting *Maximum Thickness*.

Field	Contents
Maximum thickness (μm)	Maximum thickness detected

MicAtSubst.txt

Calculation of basic parameters about detected Microcolonies on the substratum. Generated when selecting *Microcolonies at Substratum*.

Field	Contents
Minimum colonysize at the substratum (in pixels)	User setting for minimum size (default 100)
Minimum colonysize at the substratum (in μm^2)	User setting for minimum size, in μm^2
Number of microcolonies detected at the substratum	The number of separate biomass areas (microcolonies) found on the substratum layer
Colony Id	Colony ID (Comstat2 internally numbers all micro colonies it detects on the substratum)
Area of microcolonies at the substratum (μm^2)	Area covered by each microcolony
Number of colonies with this size at the substratum	The number of microcolonies with the above area
Average colony size at the substratum (μm^2)	Average substratum coverage by microcolonies
Average pixel intensity of detected microcolonies at the substratum	The average pixel intensity (grey level) of biomass on the substratum.

RaStar.txt

Calculation of the roughness coefficient of the biomass. Generated when selecting *Dimensionless Roughness Coefficient*.

Field	Contents
Roughness Coefficient (Ra^*)	Dimensionless Roughness Coefficient

SurfArea.txt

Calculation of the area of all surfaces exposed to the void, and the surface to volume ratio. Generated when selecting *Surface Area*.

Field	Contents
Surface Area (μm^2):	Surface area
Surface to biovolume ratio ($\mu\text{m}^2/\mu\text{m}^3$)	Surface to volume ratio

ThickDist.txt

Calculation of the distribution of thickness (height) of the biofilm, per pixel column. Generated when selecting *Thickness Distribution* and/or *Roughness Coefficient*.

Field	Contents
Layer	The layer in the confocal stack. 1=substratum
Thickness (µm)	Height corresponding to the layer
Spots Measured (abs)	How many pixels in the field of view (x-y area) has this height as maximum
Spots Measured (%)	How many percent of the pixels in the field of view (x-y area) has this height as maximum
Average thickness (Entire area) (µm)	Average thickness of the stack
Average thickness (Biomass) (µm)	Average thickness only of the biomass (not counting the uncovered area)

Note: the number of Spots in Layer 1 corresponds to the area that do NOT have any biofilm (height = 0).

VolAtSubst.txt

Calculation of the volume of each detected microcolony on the substratum. Generated when selecting *Volume of microcolonies at substratum*.

Field	Contents
Colony Id	Colony ID (Comstat2 internally numbers all micro colonies it detects on the substratum)
Individual colony volumes at the substratum (µm ³)	For each microcolony the volume is calculated
Average volume of colonies at substratum (µm ³)	Average of all microcolony volumes.

Image Files

The output image files are generally not useful for the end user (with one exception). The output files are placed in a subfolder of the folder containing the source images. They can be opened from Comstat2 by right-clicking on the stack in the *Images in Directories* window, and select *Open Output Images* (Fig. 7). Each file name will refer to the stack and channel – this is shown as [stack-channel] in the file names below. Here is a short description of the origin of these files.

3D_Euclidean_Distance_Transform_of_[stack-channel].tif

This file is generated when Comstat2 calculates *Diffusion Distance*. There is no practical use of this file for the end user.

Detected_Blobs_in_[stack-channel].Colonies_leq_[xxx]_px.tif

This file is generated when calculating *Average Runlength*, *Microcolonies at Substratum* and/or *Volume of microcolonies at substratum*. The [xxx] represents the user selected minimum size of Microcolonies.

This file is useful for visualizing the detected microcolonies. Open the image in ImageJ (see Fig.7). Make sure the Detected Blobs image is active (click it). Then

change the lookup table, (Image]: *Image->Lookup Tables*). When a suitable lookup table is selected each detected microcolony will have a different color.

Local_Thickness_of_[stack-channel].tif

This file is generated when Comstat2 calculates the *Local Thickness*. This is the only output from the Local Thickness function. The image can be visualized by changing the lookup table (see the description of the **Detected_Blobs_in_[stack-channel]._Colonies_leq_[xxx]_px.tif** file above). Usually there is no practical use of this file for the end user.

Height_Matrix_of_[stack-channel].tif

This file is generated when Comstat2 calculates *Roughness Coefficient* and/or *Thickness Distribution*. There is no practical use of this file for the end user.

Output_substratum_of_[stack-channel]._Colonies_leq_[xxx]_px.tif

This file is generated when Comstat2 calculates *Average Runlength*, *Microcolonies at the substratum* and/or *Volume of Microcolonies at Substratum*. The [xxx] represents the user selected minimum size of Microcolonies. There is no practical use of this file for the end user.

F – Contents of Comstat2 system files

comstat2.ini, Comstat2.bat and convert-to-ome-tiff.ijm are system files which contain text-format configuration information.

comstat2.ini

```
BasicThresholding
ConnectedVolumeFiltering
Smacking
BioMass
MaxThickness
ThicknessDistribution
AreaOccupied
SurfaceArea
MicroColoniesAtSubstratum
DiffusionDistance3D
FractalDimension
LocalThickness3D
```

This file determines which functions and setting are displayed in the Comstat2 control panel (Fig. 3)

This file controls what features are available in the main control panel (Fig. 3). If you prefer another order of the items or e.g. if you wish certain features are not displayed you can simply reorder the text lines or delete some of them.

Note: Do not enter any other text in the file than what is shown here, or Comstat2 will crash.

Comstat2.bat

```
@echo off
start javaw -Xms64m -Xmx768m -Xss32m -jar ij.jar
```

This bat file applies to Windows systems only and serves to allocate more memory to ImageJ than what is done automatically. For versions of ImageJ with a built-in Java run-time module, configure ImageJ.cfg instead (see next section).

Note: If javaw.exe is not in the system path, it can be necessary to add the full path to javaw.exe in front of javaw. For example:

C:\progra~1\Java\jre1.8.0_40\bin\javaw

The actual path depends on the system configuration – see also Appendix D – troubleshooting.

ImageJ.cfg

```
.
jre\bin\javaw.exe
-Xms64m -Xmx768m -Xss32m -cp ij.jar ij.ImageJ
```

Not strictly a comstat system file, the ImageJ.cfg file serves as a means to allocate more memory to the ImageJ environment. This file must reside in the same directory as ImageJ.exe (Windows systems only)

Note: The single point "." in the first line of the file is important and must be present.

Convert-to-ometiff.ijm

```
// Macro created by Melissa Linkert
// open the file, splitting stacks and channels

file = File.openDialog("Choose a file to open...");

run("Bio-Formats Importer", "open='" + file + "' autoscale
color_mode=Default open_all_series split_channels
view=Hyperstack stack_order=XYZCT");

outputDir = File.directory;

// iterate through each open window and save it to OME-TIFF

imageCount = nImages;

for (i=1; i<=imageCount; i++) {
    selectImage(1);
    title = getTitle();
    run("Bio-Formats Exporter", "save='" + outputDir + i + " - "
+ title + ".ome.tif' compression=Uncompressed");
    close();
}
```

This macro file should be present in the macros directory of ImageJ. It serves as an automatic tool to convert image stacks of any format (must be supported by the LOCI tools) to OME-Tiff format.

Note: The macro is included in the **startupmacro.txt** (ImageJ macro directory) on preinstalled versions of Comstat2.

G – Known bugs

As much as we would like to provide a bug free product there are unfortunately a couple of bugs left in the software. Here we list the bugs as known to us.

Version 2.1 1-July-2015

We are not aware of any bugs as of this release

If you come across other bugs, please report them to us (comstat@comstat.dk)

H – Acknowledgements

Numerous people have been involved in developing and testing Comstat2. This non-comprehensive list includes some of them (apologies if you are forgotten).

Special thanks to Assoc. Prof. Janus Haagenzen for constant feedback and suggestions, and for rigorous testing over the years.

Thanks to Profs. Drs. Søren Molin, Tim Tolker-Nielsen, Thomas Bjarnsholt, Michael Givskov, Assoc. Profs. Drs. Thomas Martini-Jørgensen, Anders Folkesson, Drs. Morten Gjermansen, Mikkel Klausen, Alex Toftgaard Nielsen, Morten Hentzer, just to mention a few. Thanks to Dr. Melissa Linkert for providing the macro convert-to-ometiff.ijm. And the hundreds of Comstat2 users all over the world! The development was partially funded by The Danish Research Councils.

I – Trademarks

Imaris® is a trademark of Bitplane AG, Zurich, Switzerland. Apple®, Macintosh® and MacOSX® are trademarks of Apple Inc., Cupertino, CA. Microsoft®, Windows®, Excel® are trademarks of Microsoft Corp., Redmond, WA. MATLAB® and Image Processing Toolbox are trademarks of MathWorks, Inc., Natick, MA. Java® is a trademark of Oracle Corp, Redwood Shores, CA. Leica®, and TCS4D are trademarks of Leica Microsystems GmbH, Germany. Zeiss® is a trademark of Carl Zeiss AG, Germany.

References

1. Heydorn, A., et al., *Quantification of biofilm structures by the novel computer program COMSTAT*. Microbiology, 2000. **146 (Pt 10)**: p. 2395-407.
2. Vorregaard, M., *Comstat2 - a modern 3D image analysis environment for biofilms*, in *Informatics and Mathematical Modelling*. 2008, Technical University of Denmark: Kongens Lyngby, Denmark. p. 85.
3. Abramoff, M.D., P.J. Magalhães, and S.J. Ram, *Image Processing with ImageJ*. Biophotonics International, 2004. **11(7)**: p. 36-41.
4. Linkert, M., et al., *Metadata matters: access to image data in the real world*. J Cell Biol, 2010. **189(5)**: p. 777-82.
5. Heydorn, A., et al., *Experimental reproducibility in flow-chamber biofilms*. Microbiology, 2000. **146 (Pt 10)**: p. 2409-15.
6. Otsu, N., *A Threshold Selection Method from Gray-Level Histograms*. IEEE Transactions on Systems, Man, and Cybernetics, 1979. **9(1)**: p. 62-66.

7. Nowicki, B., *Multiparameter Representation of Surface-Roughness*. Wear, 1985. **102**(3): p. 161-176.
8. Murga, R., P.S. Stewart, and D. Daly, *Quantitative analysis of biofilm thickness variability*. Biotechnol Bioeng, 1995. **45**(6): p. 503-10.
9. Mandelbrot, B.B., *Fractals : form, chance, and dimension*. 1977, San Francisco: W. H. Freeman. xvi, 365 p.
10. Flook, A.G., *The use of dilation logic on the quantimet to achieve fractal dimension characterisation of textured and structured profiles*. Powder Technology, 1978. **21**(2): p. 295-298.
11. Heydorn, A., et al., *Statistical analysis of Pseudomonas aeruginosa biofilm development: impact of mutations in genes involved in twitching motility, cell-to-cell signaling, and stationary-phase sigma factor expression*. Appl Environ Microbiol, 2002. **68**(4): p. 2008-17.
12. Goldberg, I.G., et al., *The Open Microscopy Environment (OME) Data Model and XML file: open tools for informatics and quantitative analysis in biological imaging*. Genome Biol, 2005. **6**(5): p. R47.