Shape-Out 2 Documentation

Release 2.11.3

Paul Müller and Christoph Herold

Jun 10, 2022

CONTENTS

1 Getting started **3**
- 1.1 Installation . . . 3
- 1.2 Update . . . 3
- 1.3 How to cite . . . 3

2 RT-DC Basics **5**
- 2.1 Working Principle . . . 5
- 2.2 Measured Features . . . 6

3 User Interface **11**
- 3.1 Differences to Shape-Out 1 . . . 11
- 3.2 Terminology . . . 11
- 3.3 Basic usage . . . 12

4 Quick Guides **21**
- 4.1 Checking experimental data for consistency . . . 21
- 4.2 Excluding invalid events . . . 22
- 4.3 Accessing DCOR data . . . 24
- 4.4 Young's Modulus . . . 25
- 4.5 Statistical significance testing (R and lme4) . . . 26
- 4.6 Loading Extensions (machine-learning and plugin features) . . . 35
- 4.7 Export statistical data . . . 37

5 Changelog **39**
- 5.1 version 2.11.3 . . . 39
- 5.2 version 2.11.2 . . . 39
- 5.3 version 2.11.1 . . . 39
- 5.4 version 2.11.0 . . . 39
- 5.5 version 2.10.0 . . . 40
- 5.6 version 2.9.6 . . . 40
- 5.7 version 2.9.5 . . . 40
- 5.8 version 2.9.4 . . . 40
- 5.9 version 2.9.3 . . . 40
- 5.10 version 2.9.2 . . . 41
- 5.11 version 2.9.1 . . . 41
- 5.12 version 2.9.0 . . . 41
- 5.13 version 2.8.1 . . . 41
- 5.14 version 2.8.0 . . . 41
- 5.15 version 2.7.4 . . . 42
- 5.16 version 2.7.3 . . . 42

5.17 version 2.7.2 . . . 42
5.18 version 2.7.1 . . . 42
5.19 version 2.7.0 . . . 42
5.20 version 2.6.9 . . . 42
5.21 version 2.6.8 . . . 42
5.22 version 2.6.7 . . . 42
5.23 version 2.6.6 . . . 43
5.24 version 2.6.5 . . . 43
5.25 version 2.6.4 . . . 43
5.26 version 2.6.3 . . . 43
5.27 version 2.6.2 . . . 43
5.28 version 2.6.1 . . . 44
5.29 version 2.6.0 . . . 44
5.30 version 2.5.2 . . . 44
5.31 version 2.5.1 . . . 44
5.32 version 2.5.0 . . . 44
5.33 version 2.4.15 . . . 44
5.34 version 2.4.14 . . . 45
5.35 version 2.4.13 . . . 45
5.36 version 2.4.12 . . . 45
5.37 version 2.4.11 . . . 45
5.38 version 2.4.10 . . . 45
5.39 version 2.4.9 . . . 45
5.40 version 2.4.8 . . . 45
5.41 version 2.4.7 . . . 45
5.42 version 2.4.6 . . . 46
5.43 version 2.4.5 . . . 46
5.44 version 2.4.4 . . . 46
5.45 version 2.4.3 . . . 46
5.46 version 2.4.2 . . . 46
5.47 version 2.4.1 . . . 46
5.48 version 2.4.0 . . . 47
5.49 version 2.3.0 . . . 47
5.50 version 2.2.6 . . . 47
5.51 version 2.2.5 . . . 47
5.52 version 2.2.4 . . . 47
5.53 version 2.2.3 . . . 47
5.54 version 2.2.2 . . . 48
5.55 version 2.2.1 . . . 48
5.56 version 2.2.0 . . . 48
5.57 version 2.1.6 . . . 48
5.58 version 2.1.5 . . . 48
5.59 version 2.1.4 . . . 49
5.60 version 2.1.3 . . . 49
5.61 version 2.1.2 . . . 49
5.62 version 2.1.1 . . . 49
5.63 version 2.1.0 . . . 49
5.64 version 2.0.3 . . . 50
5.65 version 2.0.2 . . . 50
5.66 version 2.0.1 . . . 50
5.67 version 2.0.0 . . . 50

6 Bilbliography **51**

7 Imprint/Impressum **53**
7.1 Imprint and disclaimer 53
7.2 Privacy policy 53

8 Indices and tables **55**

Bibliography **57**

Shape-Out 2

Shape-Out 2 is the successor of Shape-Out, a user interface for data analysis in real-time deformability cytometry (RT-DC). This is the documentation of Shape-Out version 2.11.3.

This documentation is also available as a PDF.

CHAPTER

ONE

GETTING STARTED

1.1 Installation

Shape-Out 2 can be installed via multiple channels.

1. **Windows installer:** Download the latest version for your architecture (i.e. `Shape-Out_X.Y.Z_win_64bit_setup.exe`) from the official release page.
2. **macOS:** Download the latest version (`Shape-Out_X.Y.Z.dmg` or `Shape-Out_X.Y.Z.pkg`) from the official release page.
3. **Python 3.8 with pip:** Shape-Out can easily be installed with pip:

   ```
   python3 -m pip install shapeout2
   ```

   To start Shape-Out, simply run `python3 -m shapeout2` or `shapeout2` in a command shell.

1.2 Update

Shape-Out automatically searches for updates (you may opt-out via the Help menu) and notifies the user when a new version is available.

1. **Windows installer:** The older version of Shape-Out will be automatically uninstalled when installing a new version.
2. **macOS:** The older version of Shape-Out will be automatically uninstalled when installing a new version.
3. **Python 3.8 and pip:**

   ```
   pip install --upgrade shapeout2
   ```

1.3 How to cite

If you use Shape-Out in a scientific publication, please cite it with:

> Paul Müller and others (2019), Shape-Out version 2.X.X: Graphical user interface for analysis and visualization of RT-DC data sets [Software]. Available at https://github.com/ZELLMECHANIK-DRESDEN/ShapeOut2.

If the journal does not accept `and others`, you can fill in the missing names from the credits file.

CHAPTER

TWO

RT-DC BASICS

This section conveys the basic understanding necessary for analyzing and interpreting RT-DC data. If you have the feeling that something is not covered here, please create an issue on GitHub.

2.1 Working Principle

In Real-time deformability cytometry (RT-DC), small objects, such as cells or beads, are flushed through a narrow channel at high speeds. The flow profile inside the channel causes hydrodynamic stresses that deform these objects [MOG+15], [MMM+17]. For each event (detected object), an image is recorded and analyzed in real-time [ORM+15]. The analysis consists of the extraction of several features from the event images, such as deformation, projected area, or average brightness. These features can then be used to identify and characterize sample populations; For instance, major blood cells and their pathological changes in disease conditions [THO+17].

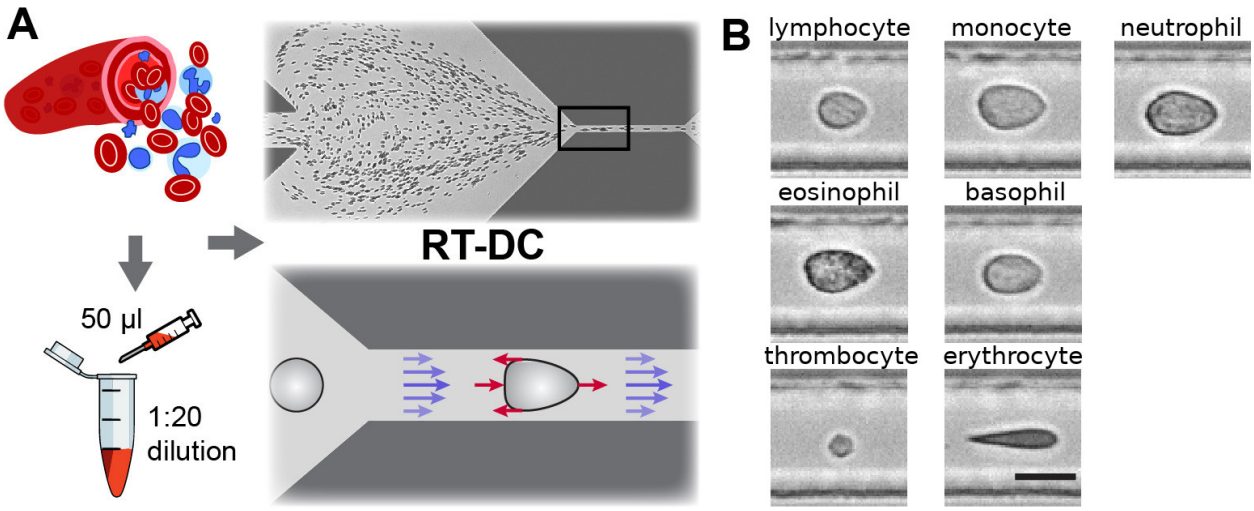


Fig. 2.1: Single-cell, morpho-rheological phenotyping of blood. (A) Analysis of whole, diluted blood. Hydrodynamic shear forces (red arrows) induce deformation of cells passing through a microfluidic channel (20x20 µm^2) at speeds of more than 30 cm/s (blue arrows). (B) Representative images of blood cell types acquired. Scale bar is 10 µm. Figure and caption adapted from Toepfner et al.[1].

[1] *Detection Of Human Disease Conditions By Single-Cell Morpho-Rheological Phenotyping Of Whole Blood* by Toepfner et al., licensed under CC BY 4.0 [THO+17].

2.2 Measured Features

A multitude of features can be extracted from the data recorded during an RT-DC measurement. These features are mostly computed live during data acquisition and are stored alongside the raw image data. Here, only the most important features are discussed. A full list of the features available in Shape-Out is maintained in the dclab documentation.

2.2.1 Area and porosity

The area is the projected object area which is determined via the contour of the binarized event image. Shape-Out differentiates between two types of area, area of the measured contour ("Measured area [px]") and area of the convex contour ("Convex area [px]" and "Area [µm^2]"). The convex contour is the convex hull of the measured contour and enables a quantification of porosity (convex to measured area ratio). The porosity is often used for filtering, e.g. to *remove high-porosity dirt particles in a preprocessing step*.

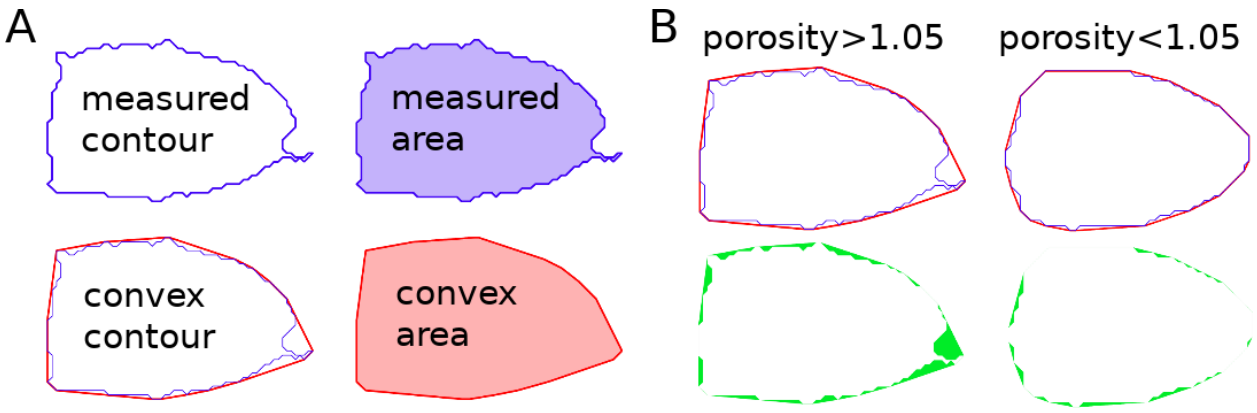


Fig. 2.2: Visualization of porosity. (A) The measured contour (blue line) defines the measured area (blue shade). The convex contour (red line) results in an area (red shade) that is usually larger than the measured area. (B) The porosity is the ratio between measured and convex contour. The difference (the "pores") between the measured and convex areas is indicated in green. Porosity is often used to remove events with non-physical contours, e.g. for cells all events with a porosity above 1.05.

A porosity of 1 means that the measured contour is convex. Note that the porosity can only assume values larger than 1. Also note that the convex contour/area is computed on the same pixel grid as the measured contour/area and is, as such, subject to pixelation artifacts.

2.2.2 Bounding box

The bounding box of an event image is the smallest rectangle (with its sides parallel to the x and y axes) that can hold the event contour. The aspect ratio of the bounding box is the rectangle's side length along x divided by the side length along y. The size of the bounding box along x and y as well as its aspect ratio are often used for filtering.

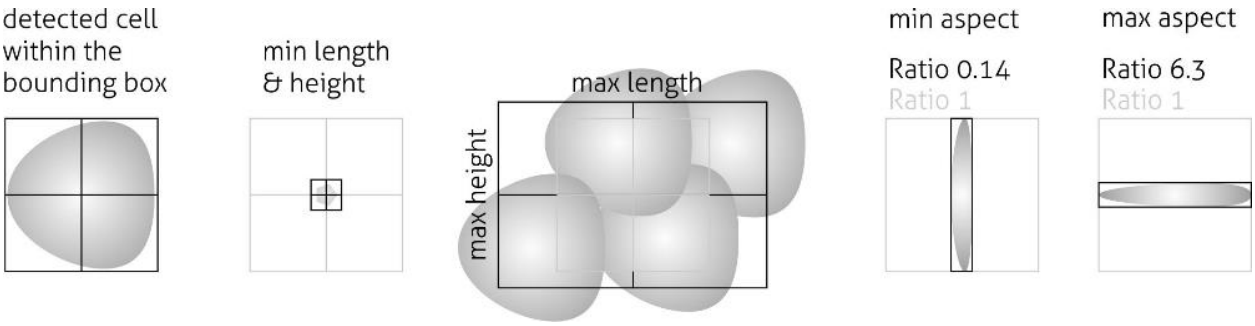


Fig. 2.3: Illustration of the event bounding box and its use cases. From left to right: definition of the bounding box, exclusion of small objects (e.g. debris) via the bounding box size, exclusion of clusters via the bounding box size, exclusion of objects elongated perpendicular to the channel axis, exclusion of objects elongated along the channel axis.

2.2.3 Brightness within contour

Quantifying the brightness values within the image contour yields information on object properties such as homogeneity or density. For instance, it has been shown that the quantities "mean brightness" and "convex area" are sufficient to identify (and count) all major blood cells in a drop of blood [THO+17].

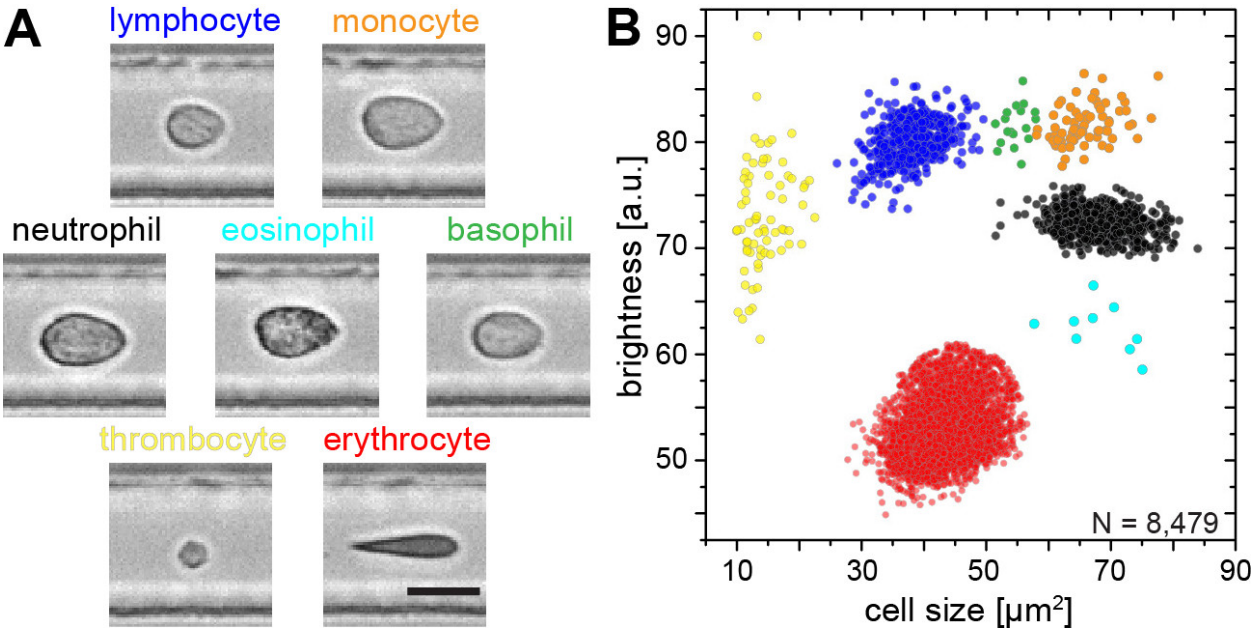


Fig. 2.4: Blood-classification with event brightness and cell size. (A) Representative images of blood cell types acquired. Scale bar is 10 µm. (B) Brightness versus cell size (area) scatter plot with cell types labeled according to the color scheme in (A). Figure and caption adapted from Toepfner et al.[Page 5, 1].

In addition to the average brightness values, Shape-Out also has access to the standard deviation of the brightness in each image.

2.2.4 Deformation and elasticity

The deformation describes how much an event image deviates from a circular shape. It is defined via the circularity:

$$\begin{aligned} \text{deformation} &= 1 - \text{circularity} \\ &= 1 - 2\sqrt{\pi A}/l \end{aligned}$$

with the projected area A and the contour length of the convex hull of the event image l. The contour length is computed from the *convex* hull to prevent an overestimation due to irregular, non-convex event shapes. It has been shown that the knowledge of deformation and area allows to derive a value for elasticity in RT-DC [MOG+15] [MMM+17]. As a convenient measure for elasticity, isoelasticity lines are often employed to visualize stiffness.

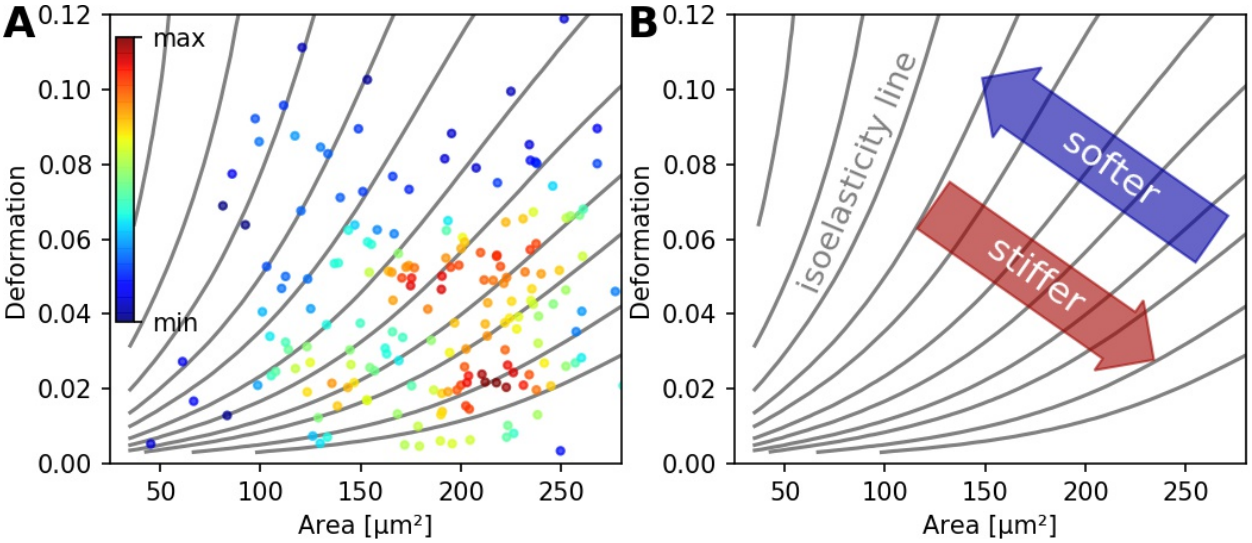


Fig. 2.5: (A) Typical deformation versus cell size scatter plot. The color scale indicates event density. (B) Isoelasticity lines derived from numerical simulations indicate trends in stiffness.

Note that it is also possible to directly *access the Young's modulus in Shape-Out*.

2.2.5 Fluorescence

Real-time fluorescence and deformability cytometry (RT-FDC) records, in addition to the event images, the fluorescence signal of each event [RPJ+18]. The raw fluorescence data consists of the one-dimensional fluorescence intensity trace from which features such as peak fluorescence or peak width can be computed. For more advanced applications, RT-FDC also supports multiple fluorescence channels.

2.2.6 Inertia ratio

The inertia ratio is the ratio of the second order central moments along x and y computed for the event contour. Thus, the inertia ratio is a measure of deformation. In contrast to deformation (which is computed from the contour), inertia ratio is based on the integral over the cell area and thus less prone to noise. In comparison to deformation, the inertia ratio has a low correlation to porosity. Shape-Out also allows to compute the principal inertia ratio which is the maximal inertia ratio that can be obtained by rotating the contour. Thus, the principal inertia ratio is rotation-invariant which makes it applicable to reservoir measurements where e.g. cells are not aligned with the channel. To quantify the alignment of the measured objects with the measurement channel, Shape-Out can additionally quantify the tilt of the contour relative to the channel axis.

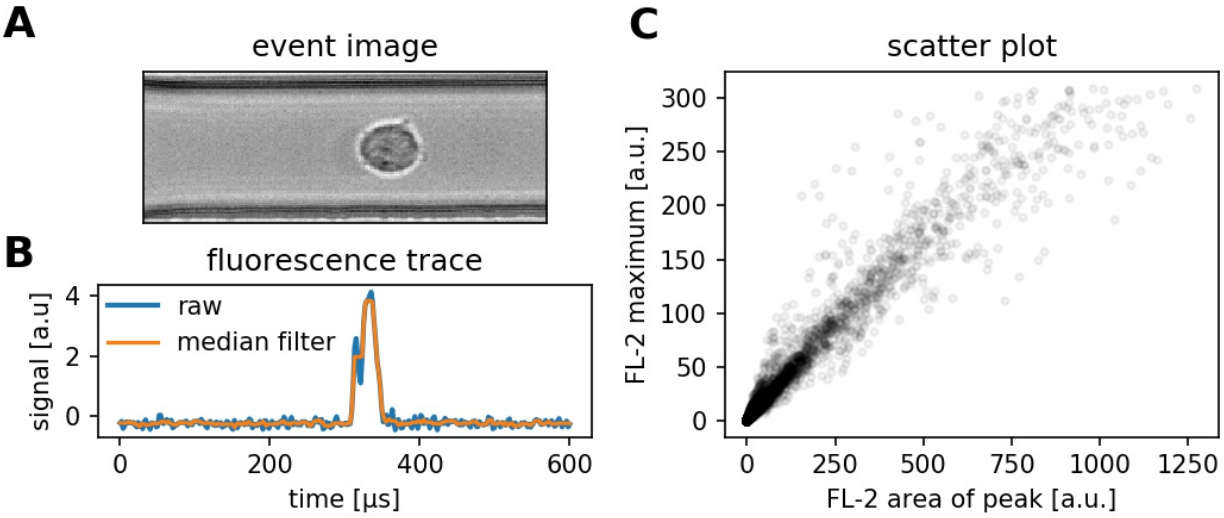


Fig. 2.6: (A) Event brightfield image. (B) Fluorescence trace of the event. The raw fluorescence data and the fluorescence data filtered with a rolling median filter (from which the parameters are computed) is shown. (C) Scatter plot of two parameters extracted from the fluorescence trace.

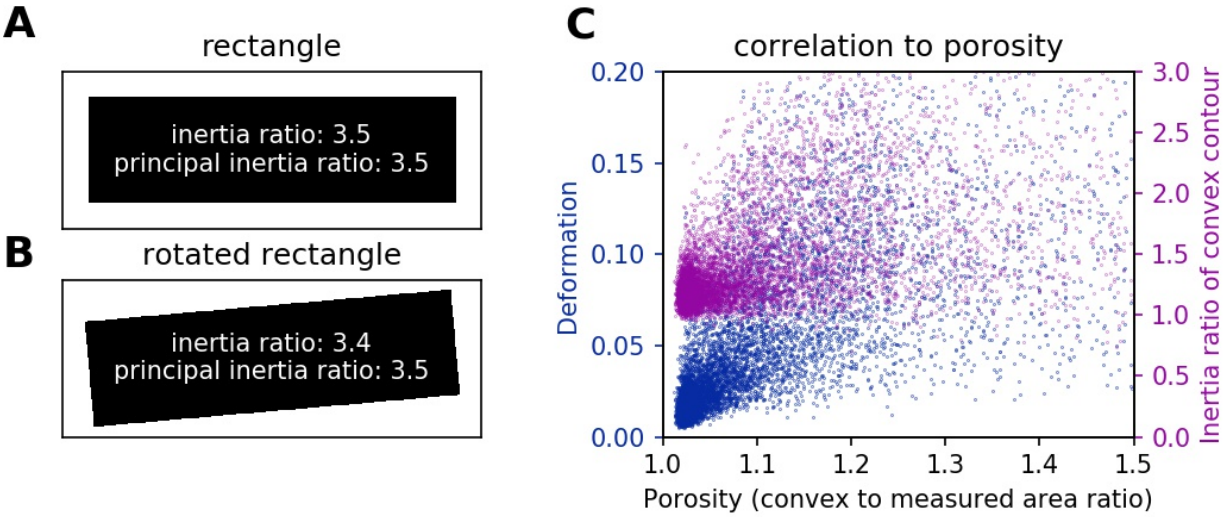


Fig. 2.7: (A) For a rectangle that is aligned with the coordinate axes, the inertia ratio and the principal inertia ratio are identical. (B) If the rectangle is rotated, the inertia ratio changes, but the principal inertia ratio does not. (C) Comparison between deformation and inertia ratio when plotted against porosity for an exemplary RT-DC experiment. Deformation exhibits a higher correlation to porisity.

2.2.7 Volume

Shape-Out can compute the volume from the event contour under the assumption of rotational symmetry, i.e. it is assumed that the projection of the cell volume onto the detector plane does not change when the cell is rotated, with a rotational axis parallel to the flow direction. The computation of the volume is based on a full rotation of the upper and the lower halves of the contour from which the average is then used [HWT02]. Volume has the advantage to be less correlated to deformation when compared to the projected area and it is therefore a better measure of cell size in the channel.

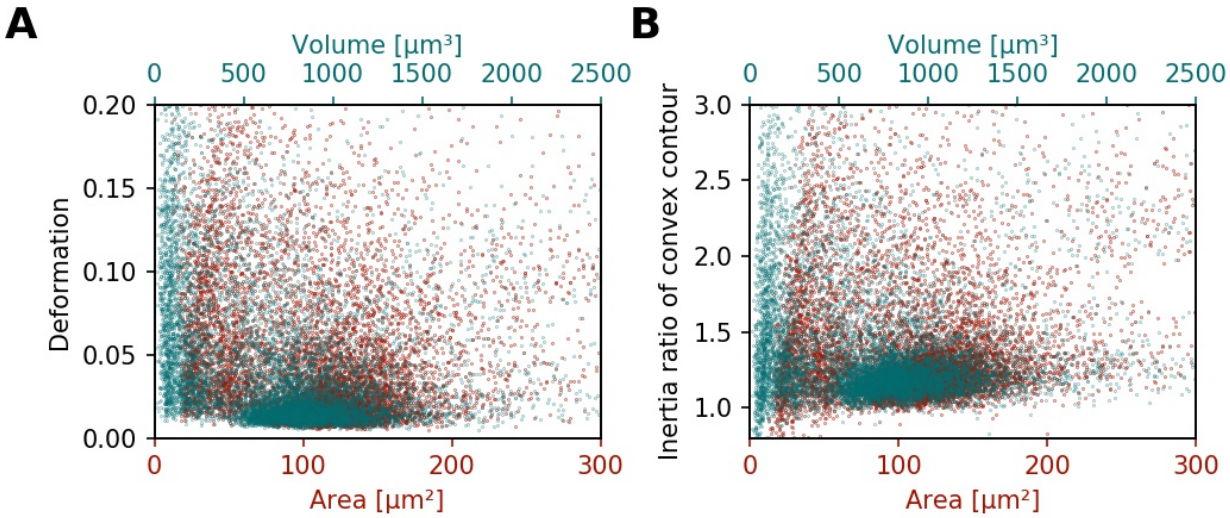


Fig. 2.8: (A) Deformation versus area (red) and volume (cyan) for an exemplary RT-DC dataset. There is a correlation between area and deformation, at least for small (<50µm^2) areas. (B) Inertia ratio versus area and volume. A correlation between inertia ratio and area is visible as well.

CHAPTER

THREE

USER INTERFACE

3.1 Differences to Shape-Out 1

Shape-Out 2 is a complete rewrite of the user interface from scratch. The overall user experience is much better compared to Shape-Out 1. The main reasons for the development of Shape-Out 2 were:

- The graphical user interface (GUI) of Shape-Out 1 was designed to be flexible with regard to new configuration parameters. While this played well for the beginning of RT-DC development, the GUI now appears not very thought-out for a mature measuring technique. The workflow of Shape-Out 2 has been designed to meet most of its users needs.
- Shape-Out 1 is condemned to Python 2 (which reached its end of life January 1st 2020), because it relies on the combination of chaco (for plotting) with wxPython 3 (for the GUI). Chaco is now focusing on PyQt and thus does not support the new wxPython 4. However, wxPython 3 does not support Python 3, so we are stuck with Python 2.
- For Shape-Out 1, there appeared a weird issue which is probably related to the old wxPython library and some other software installed on some Windows systems.

Both versions of Shape-Out rely on dclab which provides the core functionalities for RT-DC data analysis. There are, however, a few breaking changes that might prevent you from moving your data analysis from Shape-Out 1 to Shape-Out 2:

- Shape-Out 2 does not anymore support the .tdms file format. If you would like to use .tdms data in Shape-Out 2, you have to convert those data to the .rtdc file format first. You can do just that with DCKit, which provides a convenient GUI for several other RT-DC data management tasks as well. The .rtdc file format is faster, occupies less space on disk, and consists of only one file per measurement (no more para.ini, etc.).
- Shape-Out 1 sessions cannot be opened in Shape-Out 2. This is mostly caused by the different approach to data analysis. In principle, it could be possible to convert sessions (including the corresponding .tdms files), but the effort in doing so would probably exceed the effort required to just rebuild a clean analysis session in Shape-Out 2.

3.2 Terminology

Shape-Out 2 introduces several terms in the user interface that also play a role in data analysis and are consequently used in the entire code base.

block matrix The *Block Matrix* is a visualization of the analysis pipeline. It is divided into data matrix (filters) and plot matrix (plots).

dataslot or slot A slot holds all information about a measurement: the path to the .rtdc file, fluorescence labels, display color, etc. (see the *Dataset* tab in the *Analysis View*).

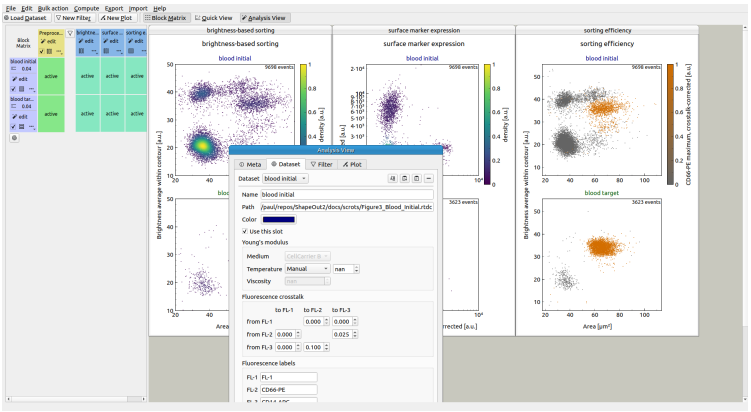
filter A filter is a set of filtering options (box filter, polygon filter, downsampling, etc.) that can be applied to a slot (see the *Filter* tab in the *Analysis View*). Filters can be exported and imported again in Shape-Out 2 (.sof file format).

filter ray A filter ray is a list of filters that can be applied to a slot. In Shape-Out 2, each row in the *Block Matrix* contains one filter ray.

pipeline A pipeline consists of all filters, slots, plots, and the filter rays (the filter selection) applied to each slot. A Shape-Out 2 session file (.so2) stores all information necessary to rebuild a pipeline.

plot A plot is a user-defined visualization of a slot/filter ray combination.

3.3 Basic usage

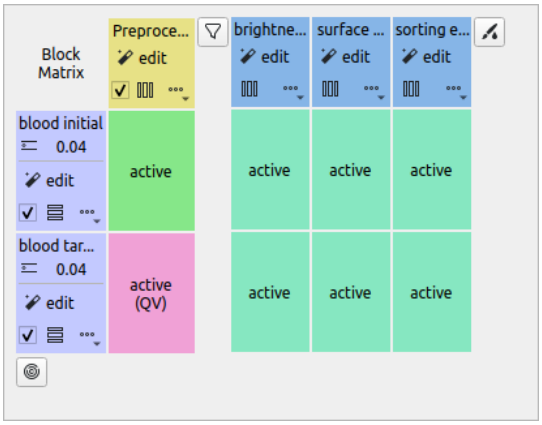


The user interface is split into several parts: the menu bar and the tool bar at the top, the Block Matrix on the left, and the Workspace on the right (example data taken from [NUH+20], [NUH+19]).

3.3.1 Menu bar and tool bar

The menu bar is used for session management (File menu) and makes additional functionalities available, such as batch processing, data statistics, data export, plot export, or filter import/export. The tool bar contains shortcuts for adding new datasets, filters, or plots (left part) and allows to hide/show the *Block Matrix* as well as the *Quick View* and *Analysis View* windows (right part).

3.3.2 Block Matrix



The Block Matrix gives an overview of the current analysis pipeline. Each row represents a dataset (purple). The columns represent either filters (yellow) or plots (blue) of your pipeline. You can change the order of datasets via the *Edit*|*Change dataset order* menu bar entry.

You can perform dataset operations in the purple rectangular area at the beginning of each row: To modify a dataset, click on the *edit* button. You can duplicate, insert anew (unmodified), or remove datasets using the dropdown menu. You can also exclude a dataset from an analysis via the check box.

Filters can also be modified, copied, removed and disabled. By default, all filters are disabled when they are created. To apply a filter to a dataset, click on the corresponding matrix element. The element changes its color from gray (incactive) to green (active). In Shape-Out, all filters that are applied to a dataset are called a **filter ray**. In the above example, the filter ray only consists of a single filter for each dataset. Filter rays may be different for each dataset.

By holding down the *Shift* key while clicking on a matrix element, you can activate the *Quick View* for the specific dataset (with filters applied up until the selected column). The block matrix element is then colored pink.

To add a plot, click on the *New Plot* button in the tool bar. This adds a plot column with a blue header to the Block Matrix and creates an empty plot window. You can add datasets to your plot by clicking on the corresponding matrix elements. In the above example, both datasets are being used in all three plots.

The modification of datasets, filters, and plots is discussed below.

3.3.3 Workspace

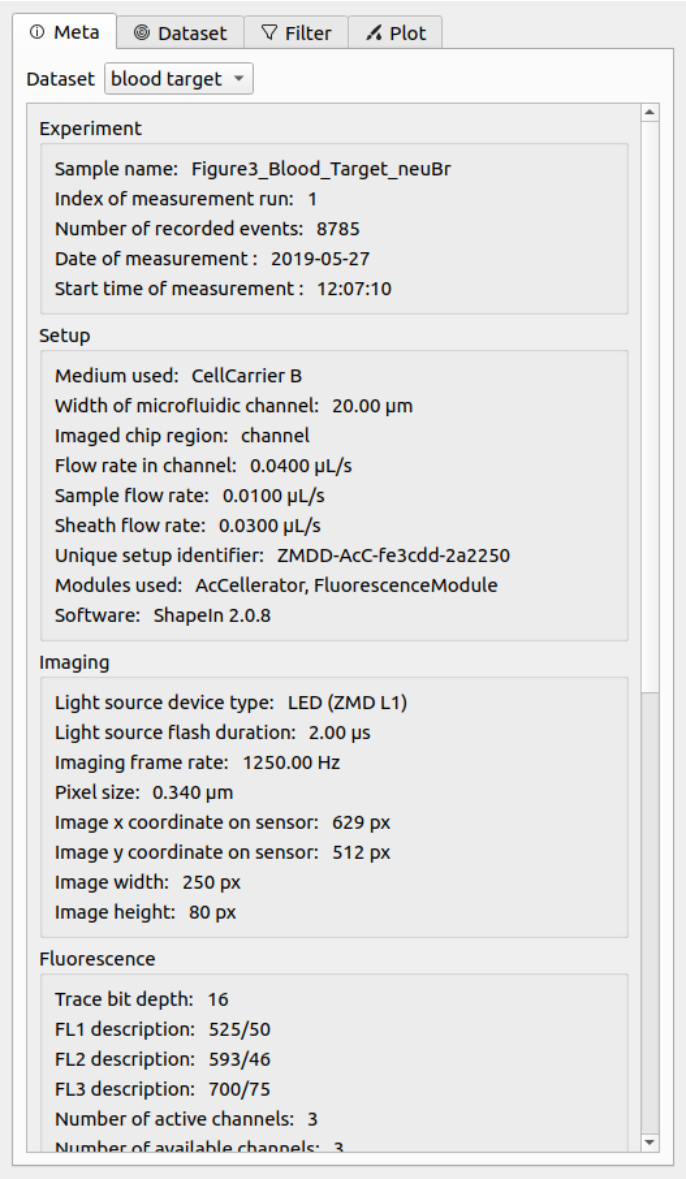
The Workspace is designed as an infinite scrollable area and contains all plot windows as well as the *Quick View* and Analysis View windows.

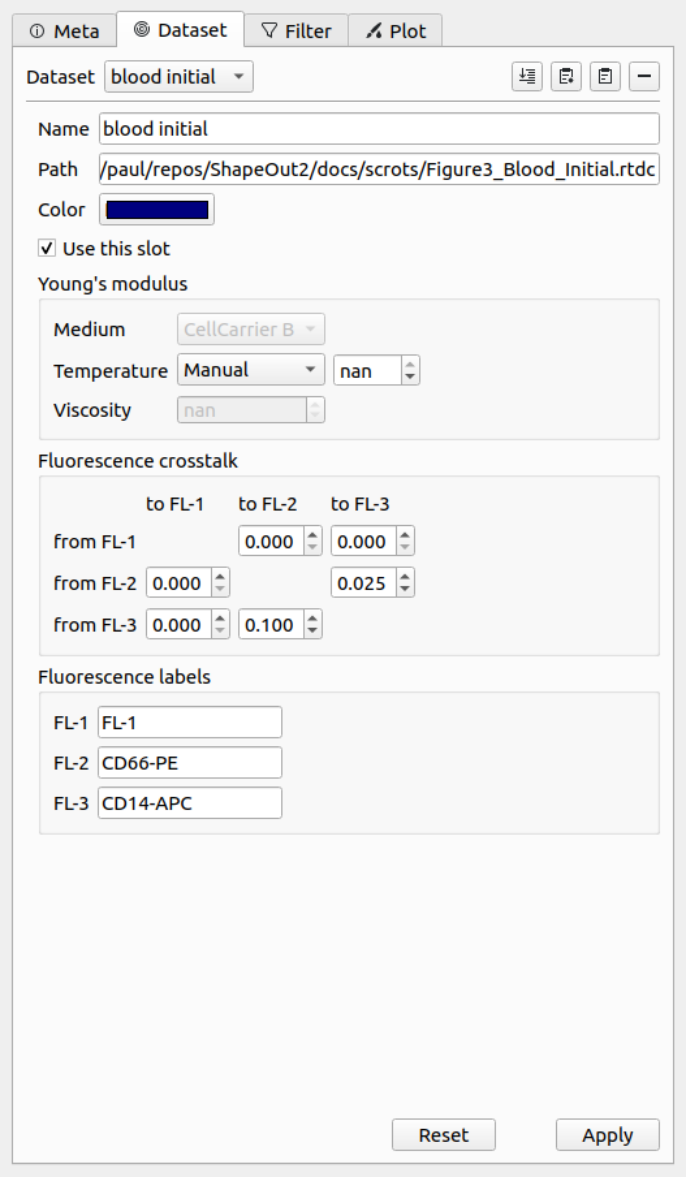
3.3.4 Analysis View

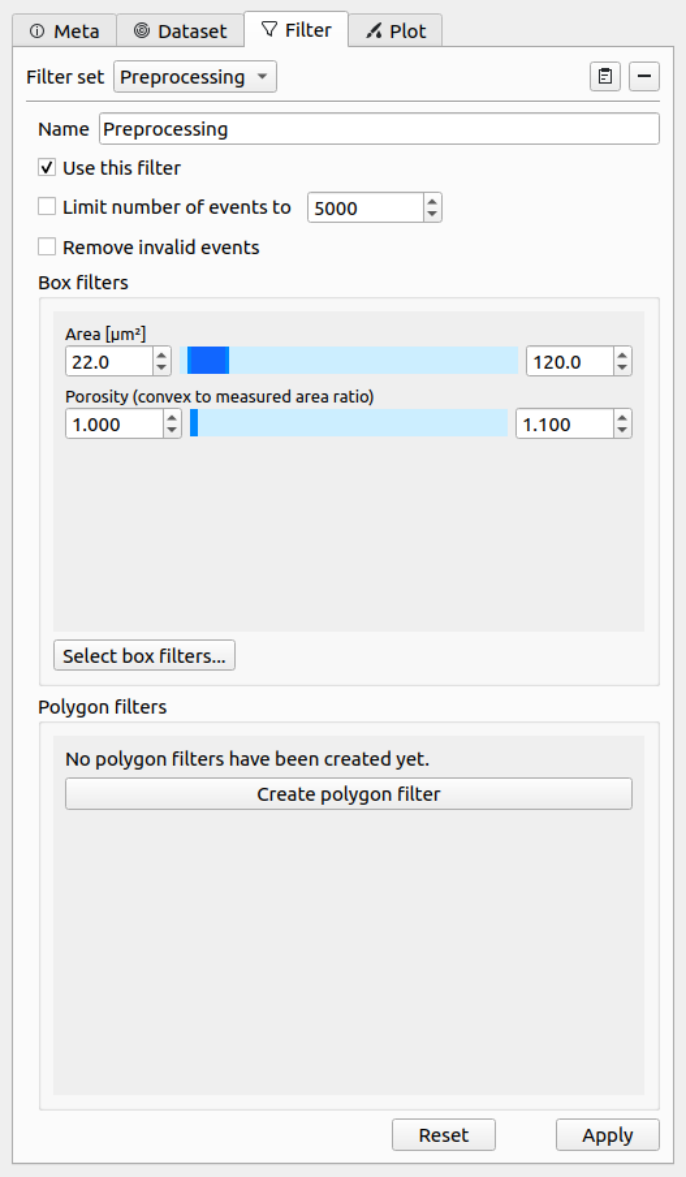
The analysis view is separated into four tabs (see screenshots below).

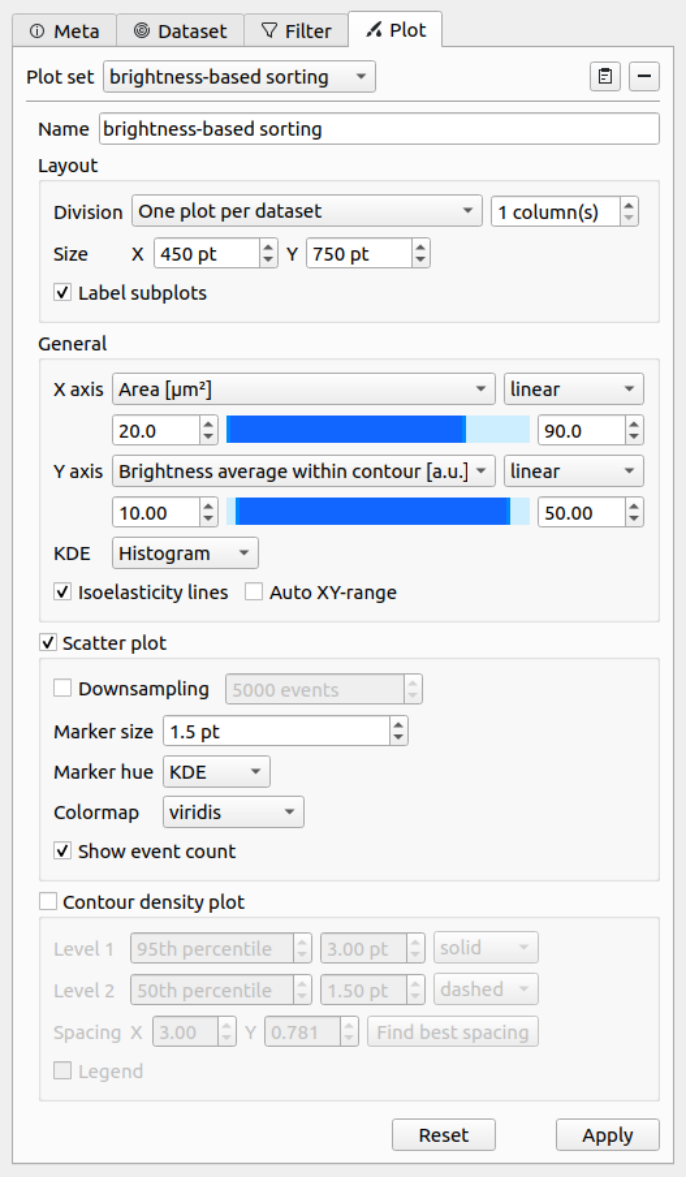
- The **Meta** tab displays all metadata of the selected dataset that are stored in the original .rtdc file.
- The **Dataset** tab allows to specify additional metadata, such as unique colors used for plotting and additional metadata for computing the Young's modulus or correcting for fluorescence cross-talk. It also allows to specify fluorescence channel labels that will then be used for labeling the axes of plots.
- The **Filter** tab is used to modify the filters of the pipeline. New box filters can be added by selecting *Choose box filters...*. Polygon filters are created in the *Quick View* window.

- The **Plot** tab allows to specify all plotting parameters. Please take special note of the *Division* option in the *Layout* section (defines the arrangement of the subplots) and the *Marker hue* option in the *Scatter plot* section (allows you the specify whether the scatter data points are colored according to a kernel density estimate (KDE), another feature dimension, or the dataset color specified in the *Dataset* tab). In this example, contour plots are not used.




Meta
Dataset
Filter
Plot
Dataset
blood initial
Name
blood initial
Path
/paul/repos/ShapeOut2/docs/scrots/Figure3_Blood_Initial.rtdc
Color
Use this slot
Young's modulus
Medium
CellCarrier B
Temperature
Manual
nan
Viscosity
nan
Fluorescence crosstalk
to FL-1
to FL-2
to FL-3
from FL-1
0.000
0.000
from FL-2
0.000
0.025
from FL-3
0.000
0.100
Fluorescence labels
FL-1
FL-1
FL-2
CD66-PE
FL-3
CD14-APC
Reset
Apply

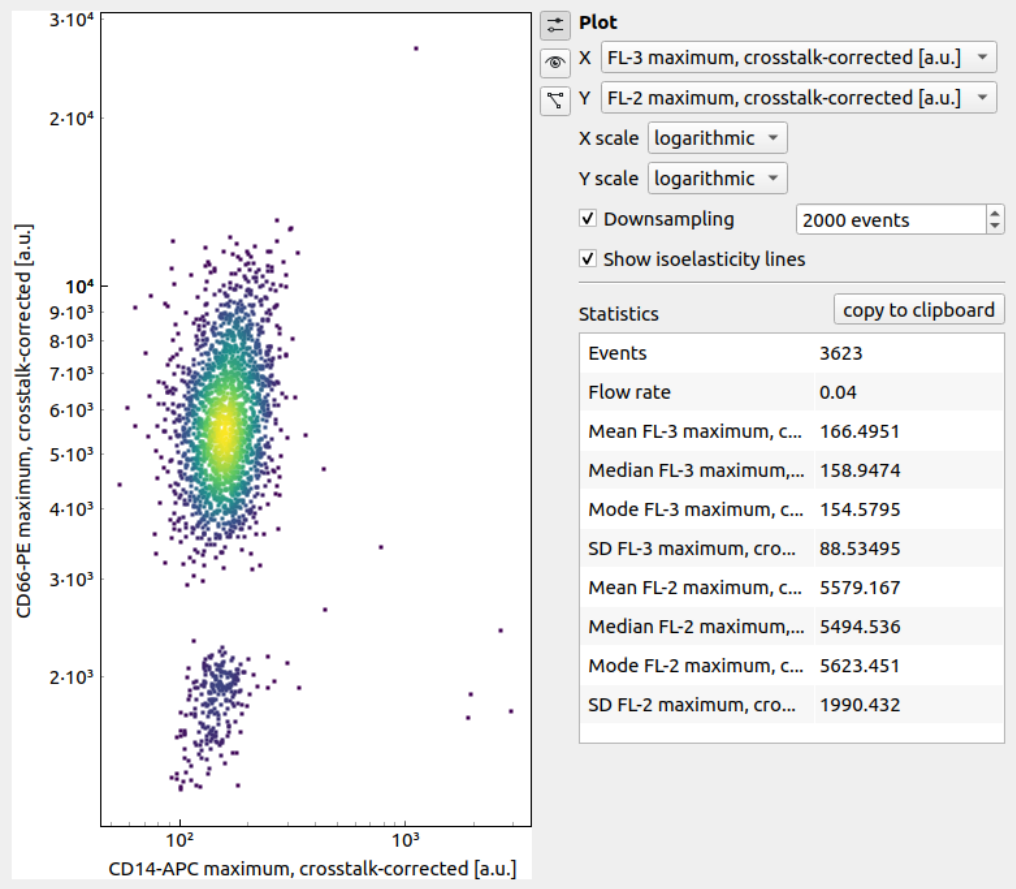

Meta
Dataset
Filter
Plot
Filter set
Preprocessing
Name
Preprocessing
Use this filter
Limit number of events to
5000
Remove invalid events
Box filters
Area [µm²]
22.0
120.0
Porosity (convex to measured area ratio)
1.000
1.100
Select box filters...
Polygon filters
No polygon filters have been created yet.
Create polygon filter
Reset
Apply



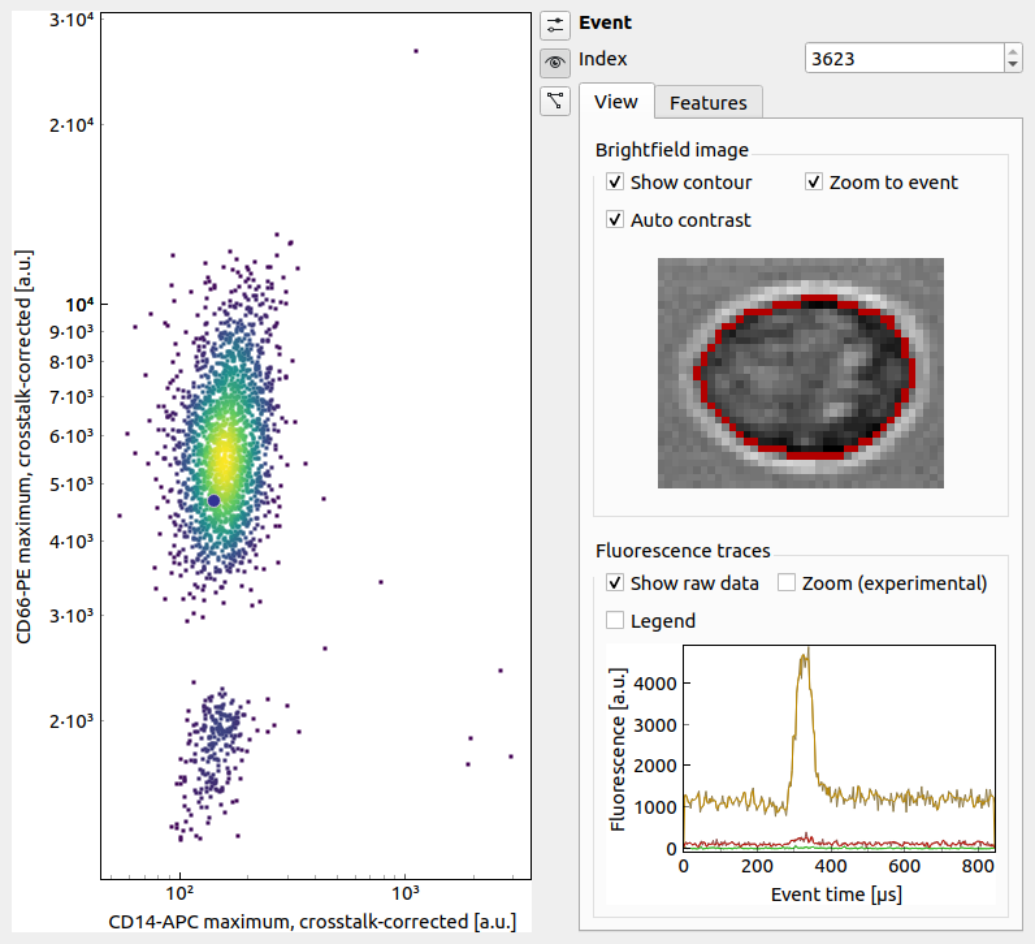
3.3.5 Quick View

The Quick View is meant for dataset exploration. It consists of a scatter plot on the left (left click for panning and right-click for zooming) and a set of tool panels that are accessible via the corresponding tool buttons on the right.

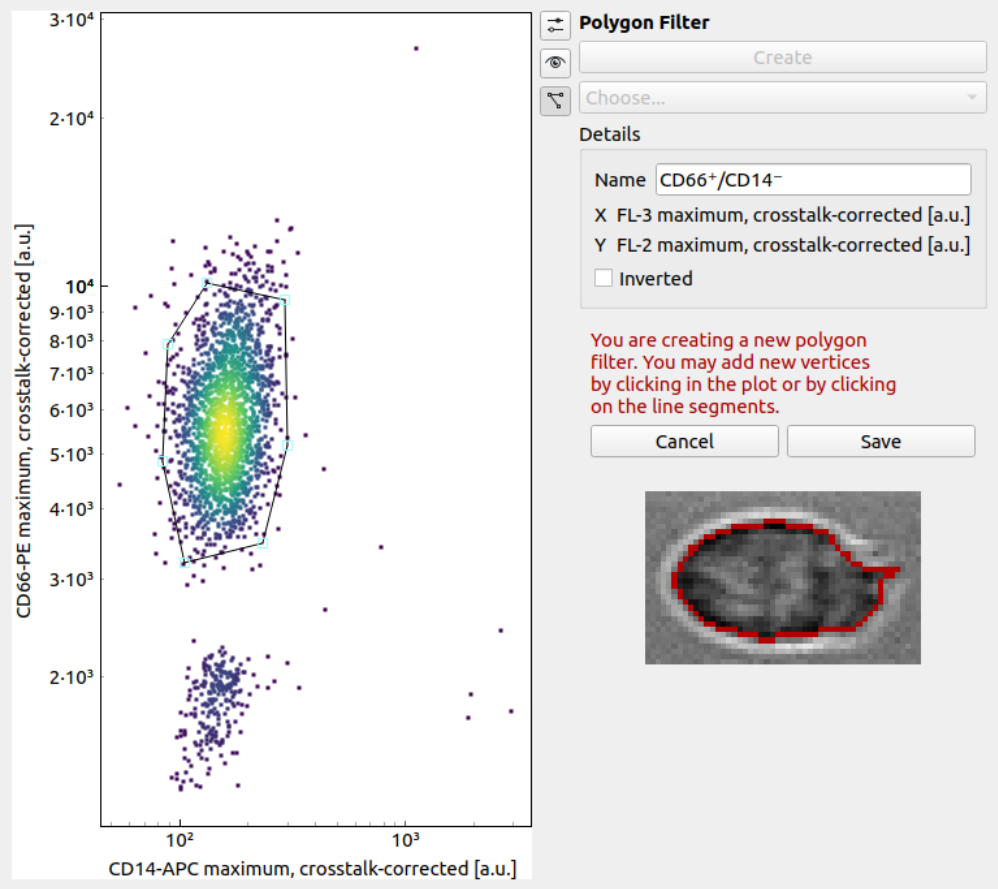
Use the **Plot** panel to define all plot parameters. It also displays common statistics of the two features plotted.



The **Event** panel displays all parameters of an individual event. You can select single events by clicking in the scatter plot or by scrolling through the *Index* spin control. If available, the event image is shown alongside the fluorescence trace of the event. All features of the event are listed in a separate tab.



The **Polygon Filter** panel allows you to create and modify polygon filters. When the panel is active you can move the mouse pointer across the scatter plot and the image of the event closest to the mouse pointer is displayed.


3·10⁴
2·10⁴
10⁴
9·10³
8·10³
7·10³
6·10³
5·10³
4·10³
3·10³
2·10³
CD66-PE maximum, crosstalk-corrected [a.u.]
10²
10³
CD14-APC maximum, crosstalk-corrected [a.u.]
Polygon Filter
Create
Choose...
Details
Name CD66⁺/CD14⁻
X FL-3 maximum, crosstalk-corrected [a.u.]
Y FL-2 maximum, crosstalk-corrected [a.u.]
Inverted
You are creating a new polygon filter. You may add new vertices by clicking in the plot or by clicking on the line segments.
Cancel
Save

CHAPTER

FOUR

QUICK GUIDES

4.1 Checking experimental data for consistency

- **Trends over time:**

  To verify that the sample "behaves" stationary during the measurement, it is helpful to plot a few selected features versus time. No systematic deviations should be observable, as in the figure below.

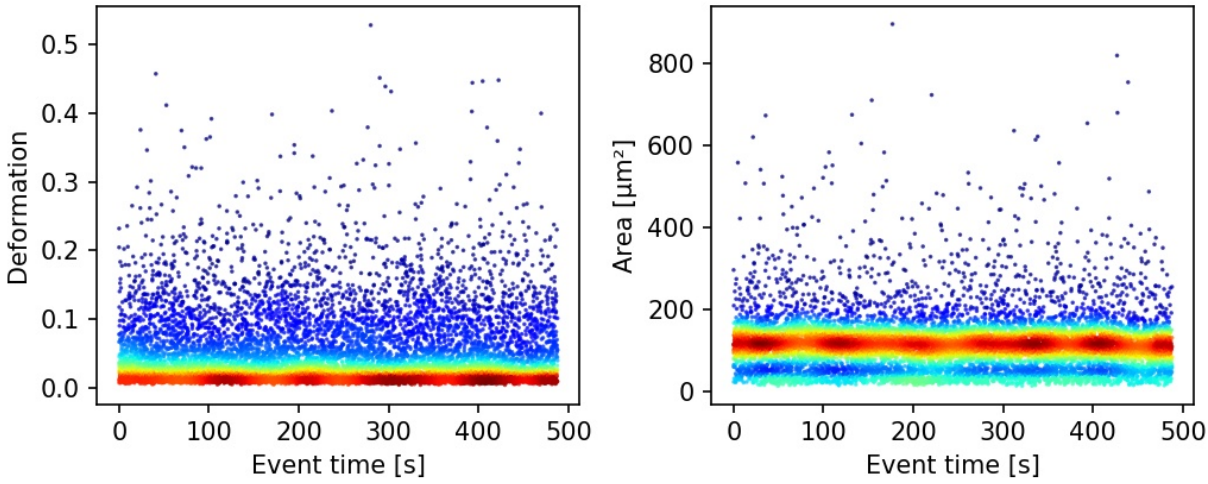


Fig. 4.1: Deformation and event area versus time. There is no systematic trend over time, indicating stable measurement conditions.

- **Countrate:**

  Trends over time do not directly visualize the count rate (measured events per second). The count rate is the slope of the event index versus event time plot and should remain constant.

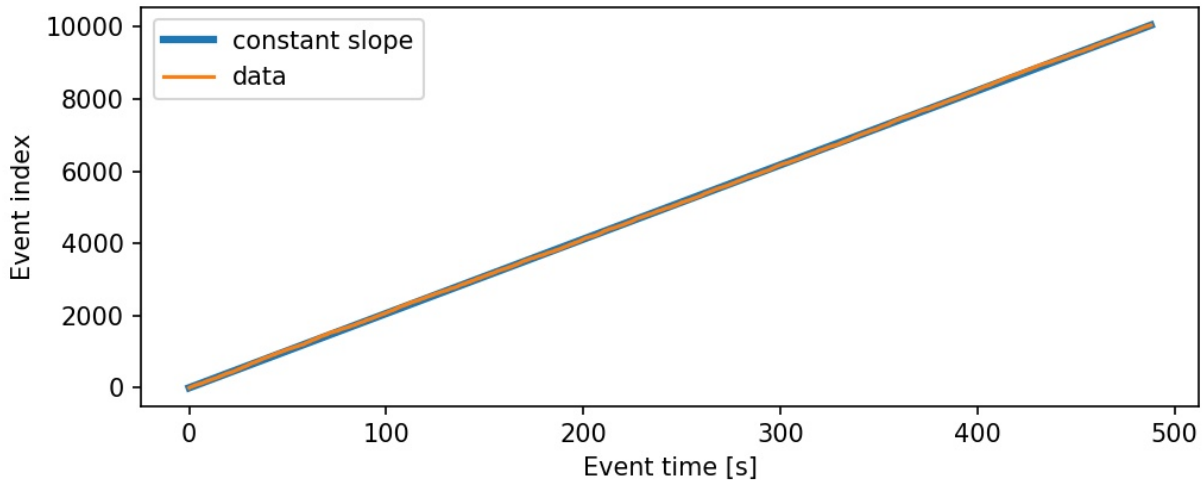


Fig. 4.2: The event index versus event time plot shows a constant slope, an indicator for stable measurement conditions.

4.2 Excluding invalid events

Filters can be used to exclude invalid events, such as cell debris, cell doublets or larger aggregates, or cells that are not intact, from an analysis.

- **Area:**

  The event area (the area defined by the event contour) can be used to filter cell aggregates (larger area), dead or degenerate cells, as well as cell debris (small area). Use the deformation versus area plot to identify exclusion candidates by clicking on an event and visualizing it in the event area.

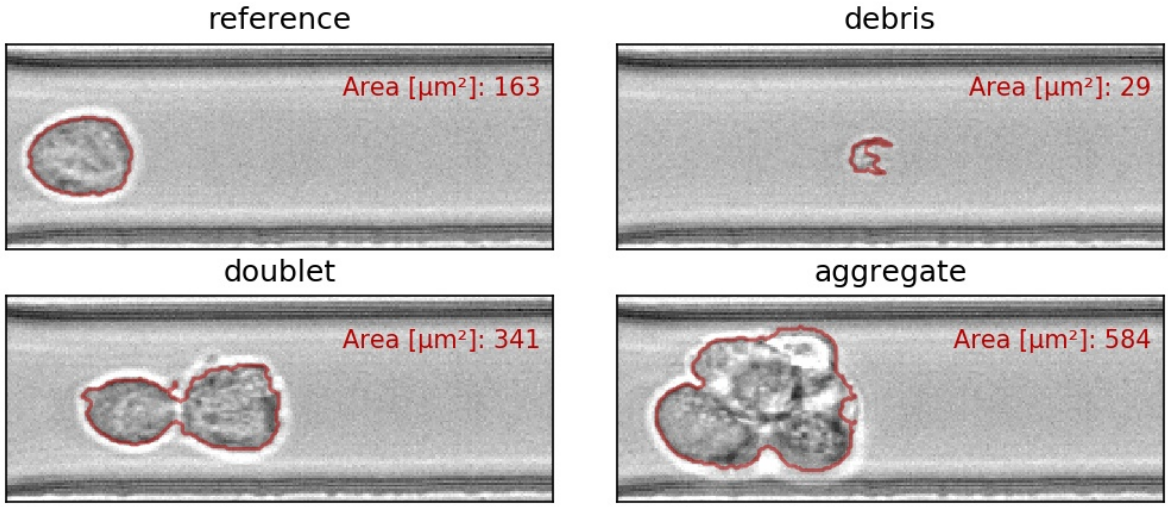


Fig. 4.3: Examples of events with small or large areas.

- **Aspect and inertia ratio:**

  The aspect ratio of the bounding box and the inertia ratio of the contour can be used to filter cell aggregates and otherwise invalid events. For instance, an aspect ratio below 1 (elongation perpendicular to the channel axis) is

most-likely debris and can be excluded from the analysis. An inertia ratio below 1 also indicates invalid events.

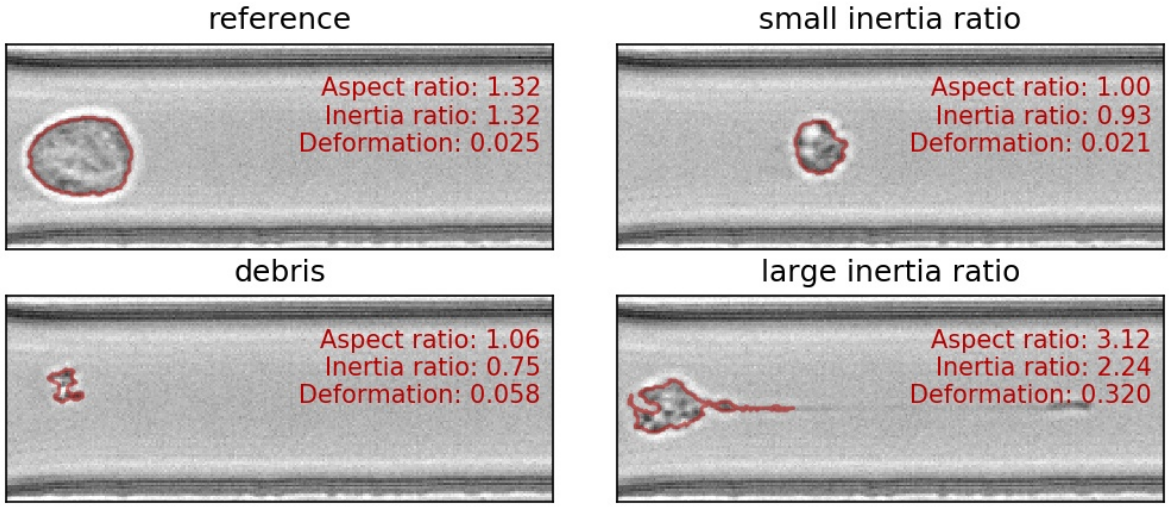


Fig. 4.4: Examples of events with various aspect and inertia ratios.

- **Porosity:**

  The porosity is the ratio between measured contour and the convex contour. Porosity is commonly used to remove events with non-physical contours, e.g. for cells, all events with a porosity above 1.05.

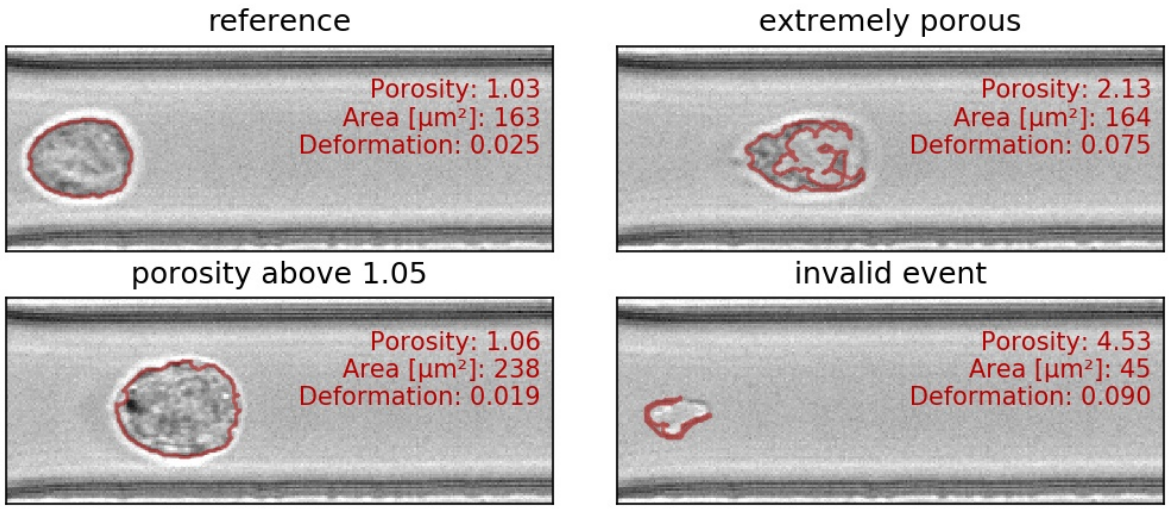


Fig. 4.5: Examples of events with various porosities.

4.3 Accessing DCOR data

The deformability cytometry open repository (DCOR) allows you to upload and access RT-DC datasets online (internet connection required). In Shape-Out 2, you can access parts of the dataset (e.g. just two features) without downloading the entire data file (which includes image, contour, and traces information). This saves disk space and time when accessing large datasets.

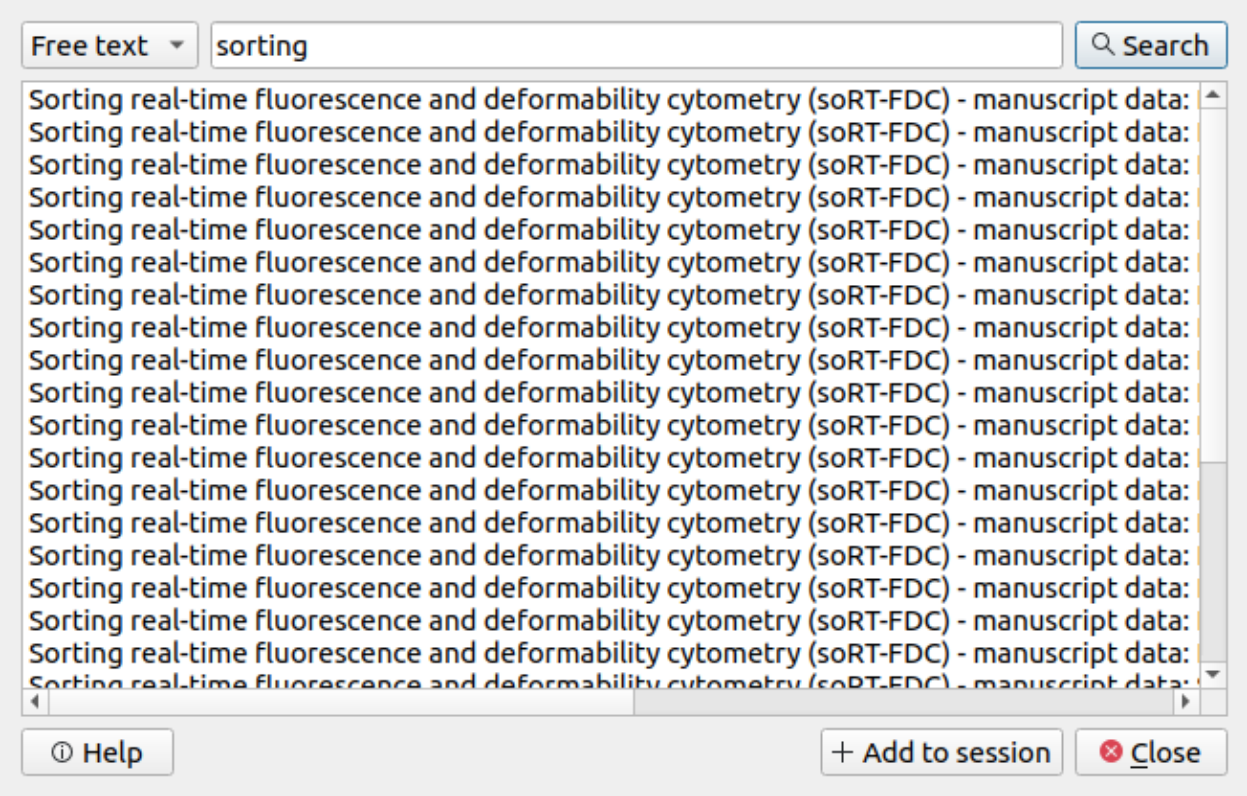


Fig. 4.6: Dialog for loading DCOR data.

In the *File* menu, choose *Load DCOR data*. A dialog (Fig. 4.6) lets you search data in the DCOR repository.

Note: If you need to access private data, you have to specify your personal API key in the preferences dialog (via the *Edit* menu). There, you can also specify which server you wish to connect to. You can find your personal API Key in your user pofile page after you logged in (e.g. at https://dcor.mpl.mpg.de).

You can choose between a free text search (dataset and resource names, tags, etc.) and an identifier search (dataset and resource identifiers). After clicking the *Search* button, you may select the resources you wish to load into your sesssion in the list view below. Then, click *Add to session* to load the data.

4.4 Young's Modulus

With Shape-Out, it is possible to translate the measured area and deformation to the Young's modulus based on numerical simulation for fully elastic spheres according to Mokbel et al. [MMM+17].

Note: For reservoir data, the Young's modulus cannot be computed. The corresponding options are hidden.

4.4.1 Parameter settings

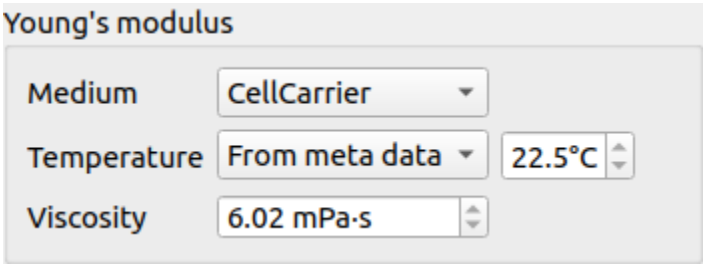


Fig. 4.7: Temperature taken from the meta data.

The parameters for computing the Young's modulus can be set in the *Dataset* tab of the *Analysis View*. The Young's modulus is computed using a precomputed look-up table and additionally depends on channel width, flow rate, pixel size (pixelation correction), and viscosity. For known media, such as CellCarrier, the viscosity can be derived from channel width, flow rate, and temperature. In some RT-DC setups, the chip temperature is recorded during the measurement. For instance, in Fig. 4.7, the average chip temperature of 22.5°C from the [setup] meta data section is used. The value of the resulting viscosity is shown below. If the chip temperature is recorded for each event, then the user may select the *From feature* option (Fig. 4.8). In this case, the Young's modulus is computed from the individual viscosities for each event.

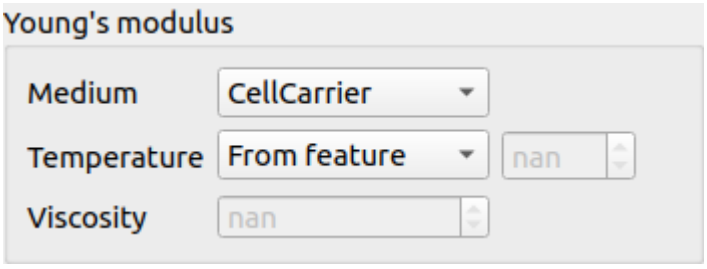


Fig. 4.8: Use the temperature (temp) feature.

If the temperature is not given as a feature or as meta data, then you may select the temperature manually. This case is visualized in Fig. 4.9. Here, the temperature is purposely set outside of the known range defined in [Her17], which is highlighted by coloring the viscosity red.

You may also set the viscosity manually by selecting *other* as a medium (Fig. **??**). In this case, the values for temperature are irrelevant. Please only use this option if you know what you are doing (e.g. you have considered shear-thinning [Her17]).

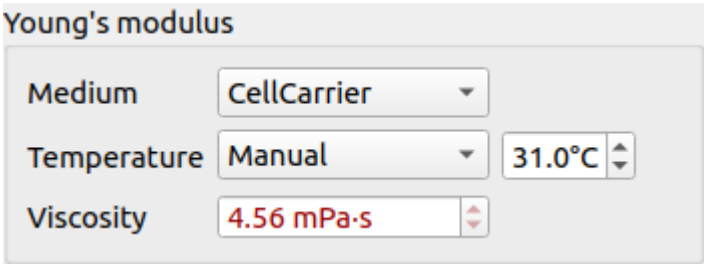


Fig. 4.9: Temperature outside of known range.

Click *Apply* for any changes to take effect. The Young's modulus is then available for the selected dataset.

4.4.2 Bulk actions

The *Bulk action* menu has an entry for Young's modulus computation. In the associated dialog you can set the corresponding parameters for all datasets of the current session. Note that the options in this dialog do not necessarily reflect the options available for the individual datasets. Only valid options are adopted. For instance, you will not be able to change the medium for a dataset if a medium is already given in its meta data. To verify the options set, you can always check the current setting via the *Analysis View* (see above).

4.4.3 Validity

The computation of the Young's modulus is valid only for objects that initially have a spherical shape. In addition, the deformation and size values must be in a "valid region". Events outside this region will have a *nan*-valued (not a number) Young's modulus. Note that as a result, these events will be excluded from all plots when *remove invalid events* is checked in the *Filter* configuration tab.

The invalid regions (white in the figure above) include objects that are very small or objects with very low deformation values. The reason for that is a very steep increase of the Young's modulus with little decrease in deformation that could potentially result in very large simulation errors. In addition, regions with high deformation are invalid as well, because the simulations do not converge (objects simulated with lower Young's moduli become more and more elongated until they rupture). In practice, this means that the channel size has to be selected carefully to match the object sizes. Larger object sizes require wider channels if meaningful values for the Young's modulus are to be computed.

In the following, additional visualizations for other commonly used channel sizes and flow rates are shown:

4.4.4 Implementation

As described above, the Young's modulus can be derived in multiple ways, for known media and global or event-based temperature values. The underlying implementation is described in the dclab docs.

4.5 Statistical significance testing (R and lme4)

Since version 2.5.0, Shape-Out 2 allows you to quantify the statistical significance (p-value) when comparing two measurement groups. For this to work, you need to have R installed (tested with R 3.6.3). The analysis uses linear mixed effects models (LMM) as described in [HMMO18] and yields results identical to those obtained with Shape-Out 1.

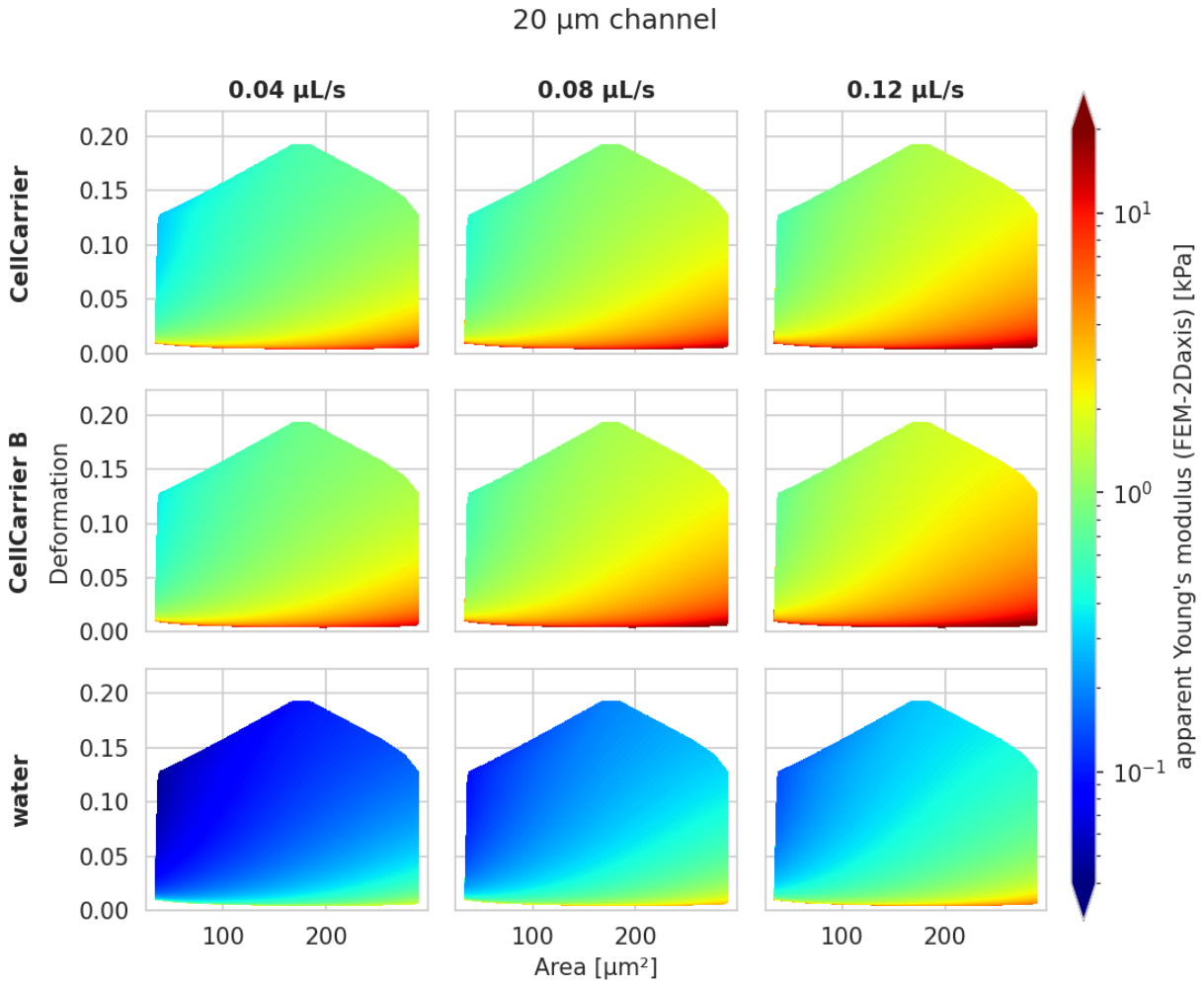
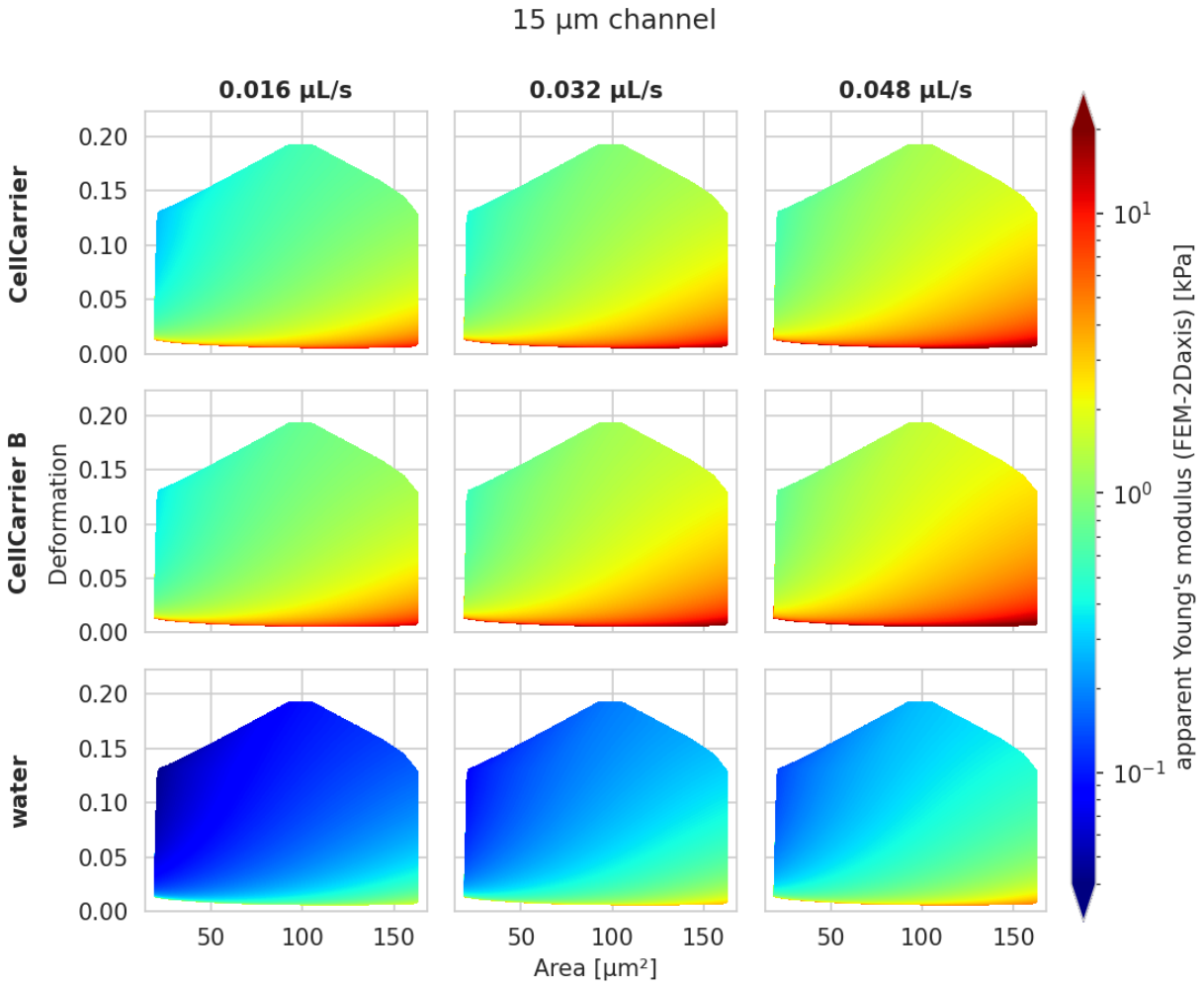
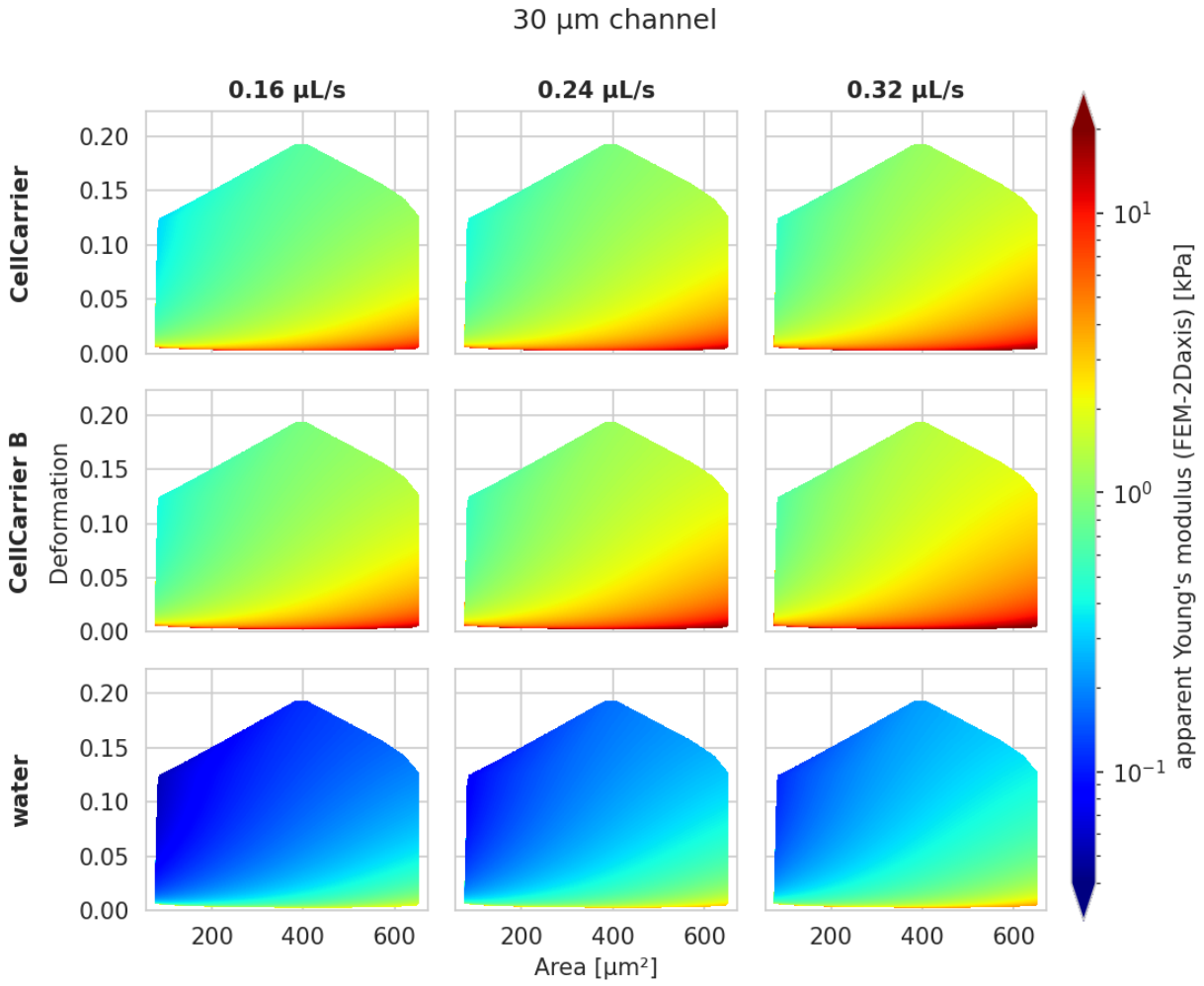
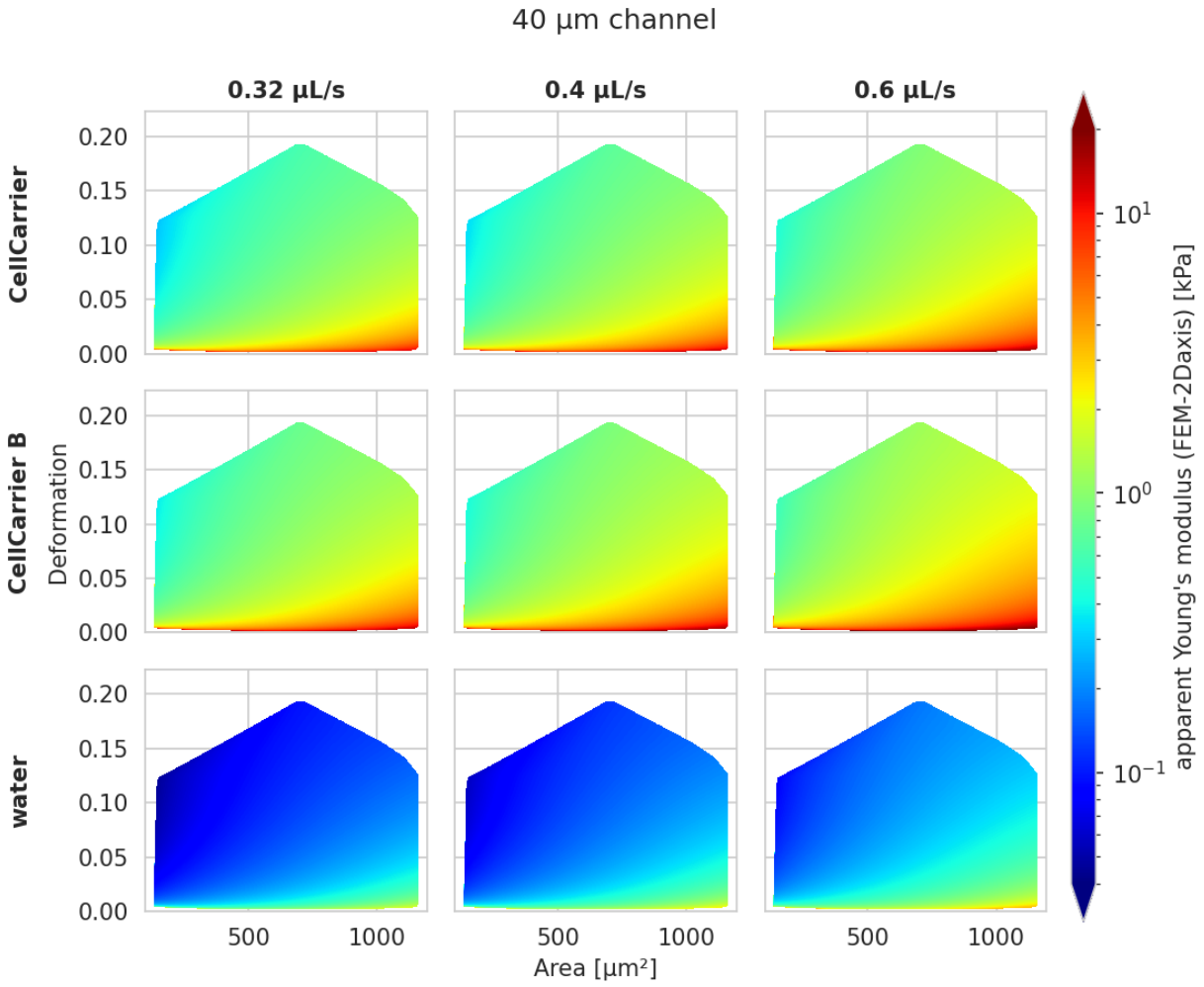


Fig. 4.11: Visualizations of the support and the values of the look-up table (LUT) used for determining the Young's modulus from deformation and cell area. The values of the Young's moduli in the regions shown depend on the channel size, the flow rate, the temperature, and the viscosity of the medium [MOG+15]. Here, they are computed for a 20 µm wide channel at 23°C with an effective pixel size of 0.34 µm. The data are corrected for pixelation effects according to [Her17].


15 µm channel
0.016 µL/s
0.032 µL/s
0.048 µL/s
CellCarrier
CellCarrier B
water
Deformation
0.20
0.15
0.10
0.05
0.00
50
100
150
Area [µm²]
10^1
10^0
10^{-1}
apparent Young's modulus (FEM-2Daxis) [kPa]


30 µm channel
0.16 µL/s
0.24 µL/s
0.32 µL/s
CellCarrier
CellCarrier B
water
Deformation
0.20
0.15
0.10
0.05
0.00
200
400
600
Area [µm²]
10^1
10^0
10^{-1}
apparent Young's modulus (FEM-2Daxis) [kPa]


40 µm channel
0.32 µL/s
0.4 µL/s
0.6 µL/s
CellCarrier
CellCarrier B
water
Deformation
0.20
0.15
0.10
0.05
0.00
500
1000
Area [µm²]
10^1
10^0
10^{-1}
apparent Young's modulus (FEM-2Daxis) [kPa]

Warning: Unfortunately, LMM analysis currently does not work with the .app or .pkg bundle on macOS. This problem is tracked in issue #73. If you are using macOS and you would like to use LMM analysis, you have to install Shape-Out 2 as a Python package (see *Installation*).

The LMM analysis is performed using the lme4 R package and is based on the implementation in dclab (please read for more information on the options available) which uses rpy2 to connect to R. Here, we make use of the dataset [XRM+20] to illustrate this functionality.

4.5.1 Basic linear mixed models

We would like to quantify the difference between human skeletal stem cells (SSC) and the human osteosarcoma cell line MG-63 (which is often used as a model system for SSCs) using a likelihood ratio test based on LMM.

Please load `this session` into Shape-Out (via *File* | *Open Session*). The experimental data are accessed from DCOR, so you will need a working internet connection. The session contains five datasets:

- SSC_16uls_rep1_20150611.rtdc (SSC repetition 1)
- SSC_16uls_rep2_20150611.rtdc (SSC repetition 2)
- MG63_pure_16uls_rep1_20150421.rtdc (MG-63 repetition 1)
- MG63_pure_16uls_rep2_20150422.rtdc (MG-63 repetition 2)
- MG63_pure_16uls_rep3_20150422.rtdc (MG-63 repetition 3)

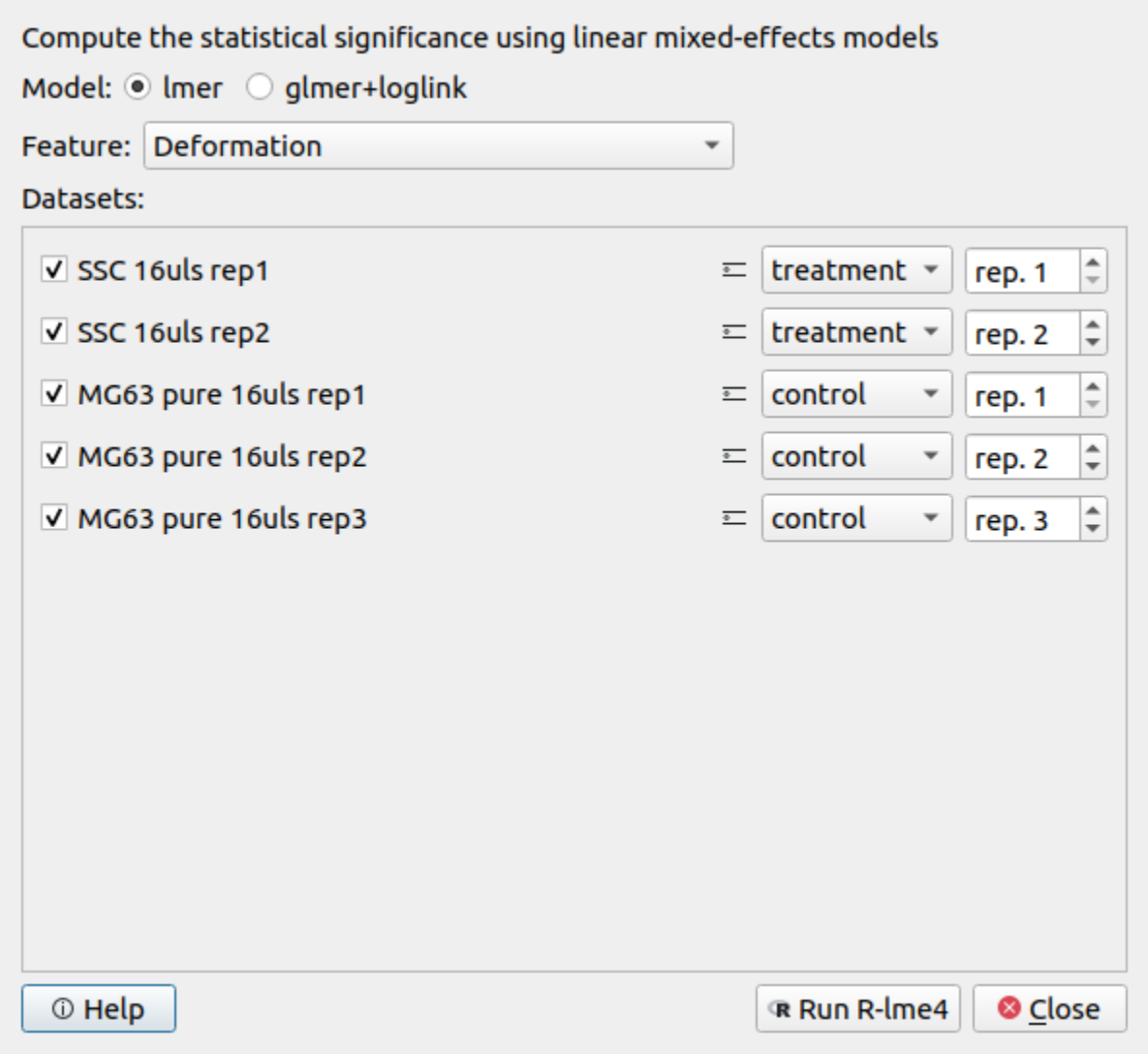
Note: The data sets are ordered according to run index. In an ideal case, the run index would resemble the repetition of an experiment. Here, however, the run index is only an internal lab book reference.

In this example, we treat SSC as our "treatment" and MG-63 as our "control". These are just group names that remind us that we are comparing one type of sample against another type.

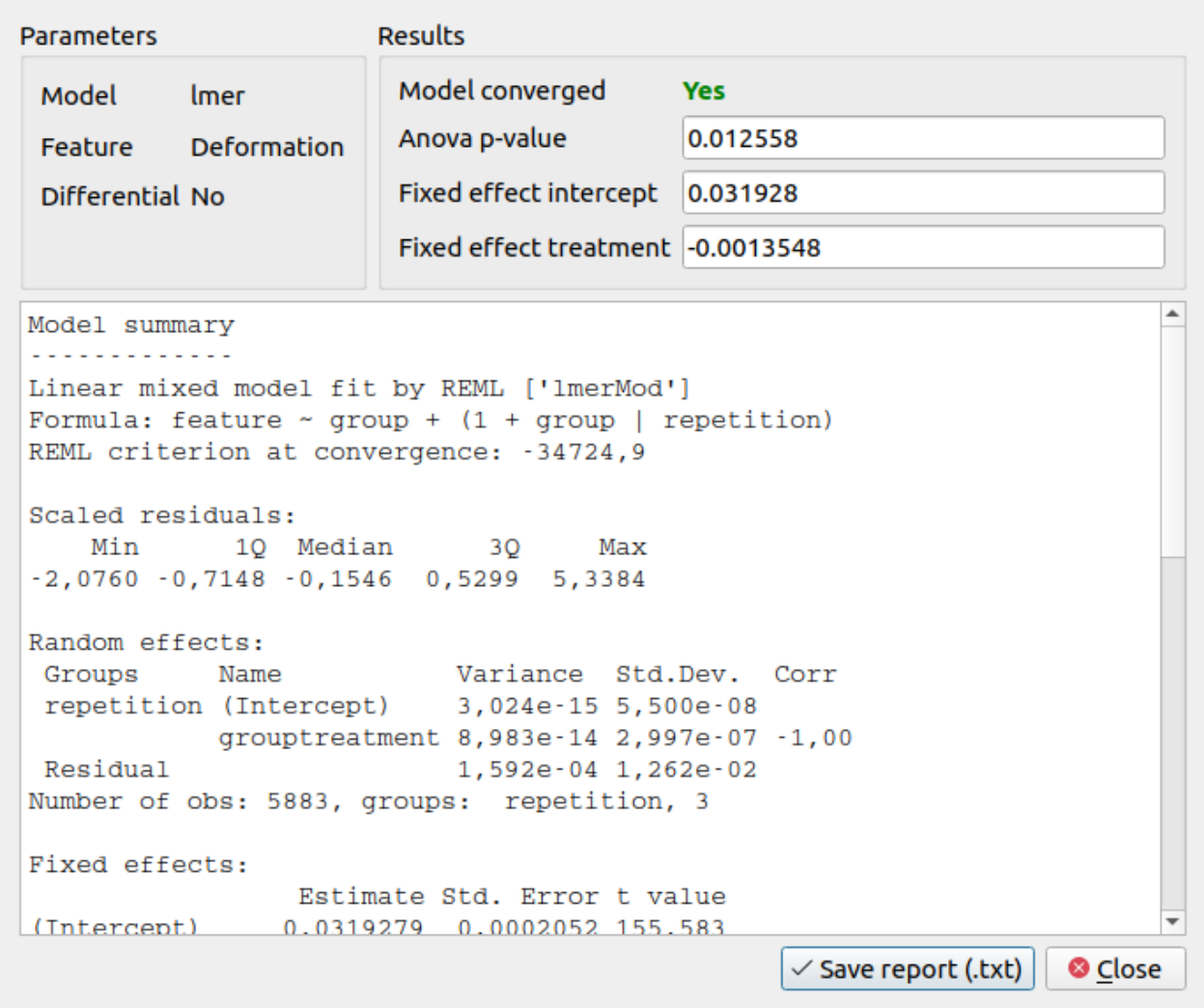
Note that the settings for Young's modulus computation have already been defined and that the datasets are already filtered with these box filters:

- Porosity (*area_ratio*): 0 to 1.05
- Area [µm^2] (*area_um*): 120 to 550
- Deformation (*deform*): 0 to 0.1

To perform an LMM analysis, go to *Compute* | *Statistical significance* and set the groups (SSC: treatment; MG63: control) and repetitions (enumeration by date-time).



Make sure that the feature "Deformation" is selected and hit *Run R-lme4*. The results of the LMM analysis are shown in a new dialog.



The most important numbers are extracted from the report and are shown at the top:

- **Fixed effect intercept:** The mean of the parameter chosen for all controls.
- **Fixed effect treatment:** The effect size of the parameter chosen between the mean of all controls and the mean of all treatments.
- **Anova p-value:** The p-value of the ANOVA test.

In the report, you will also find the **Full coefficient table**, which shows the effect size of the parameter chosen between control and treatment for every single experiment.

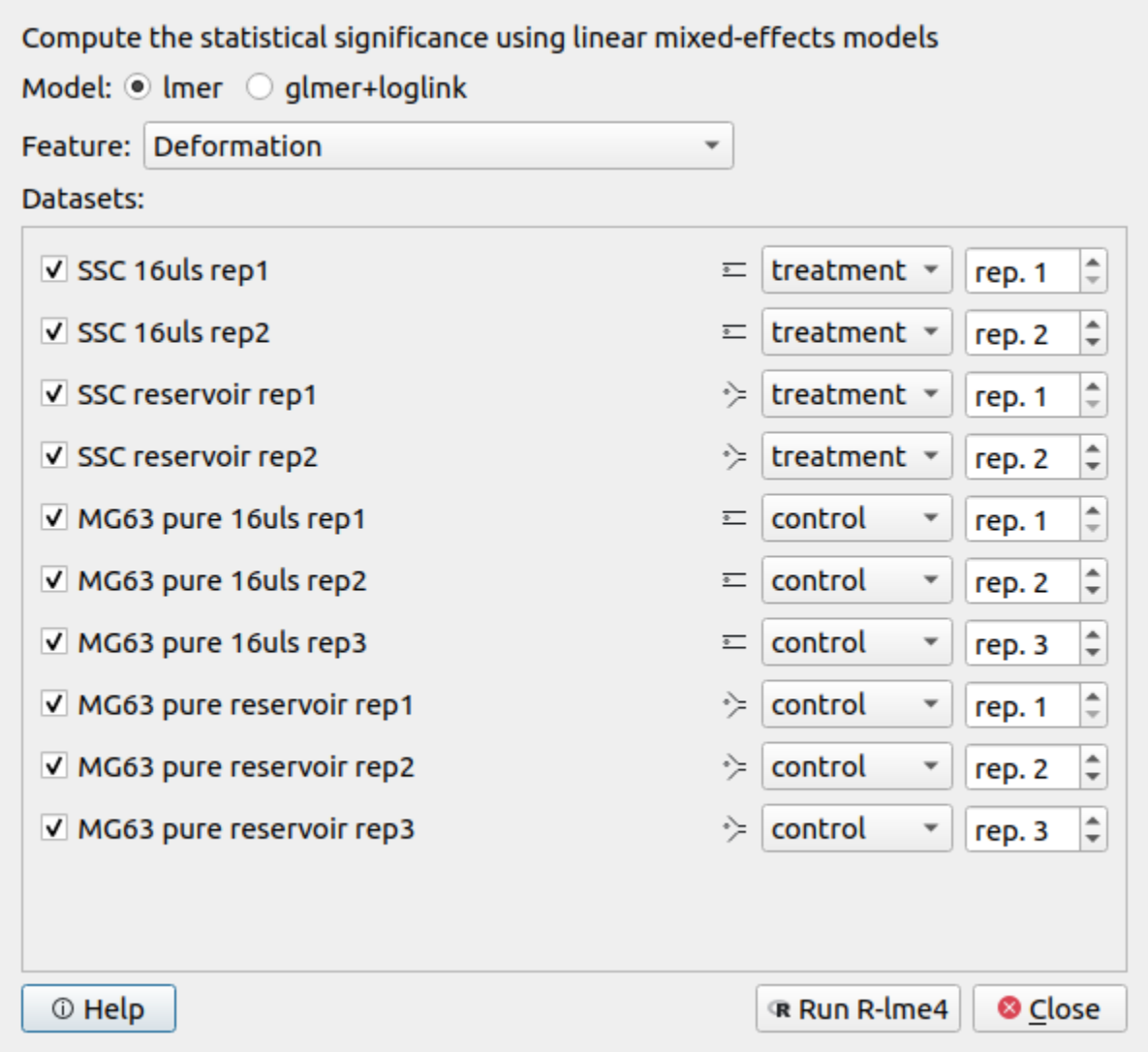
We are interested in the p-value, which is 0.01256 for Deformation. We repeat the analysis with Area (0.0002183) and Young's modulus (0.0002777). The p-values indicate that MG-63 (mean elastic modulus 1.26 kPa) cells are softer than SSCs (mean elastic modulus 1.54 kPa) [HMMO18].

4.5.2 LMM analysis of differential deformation

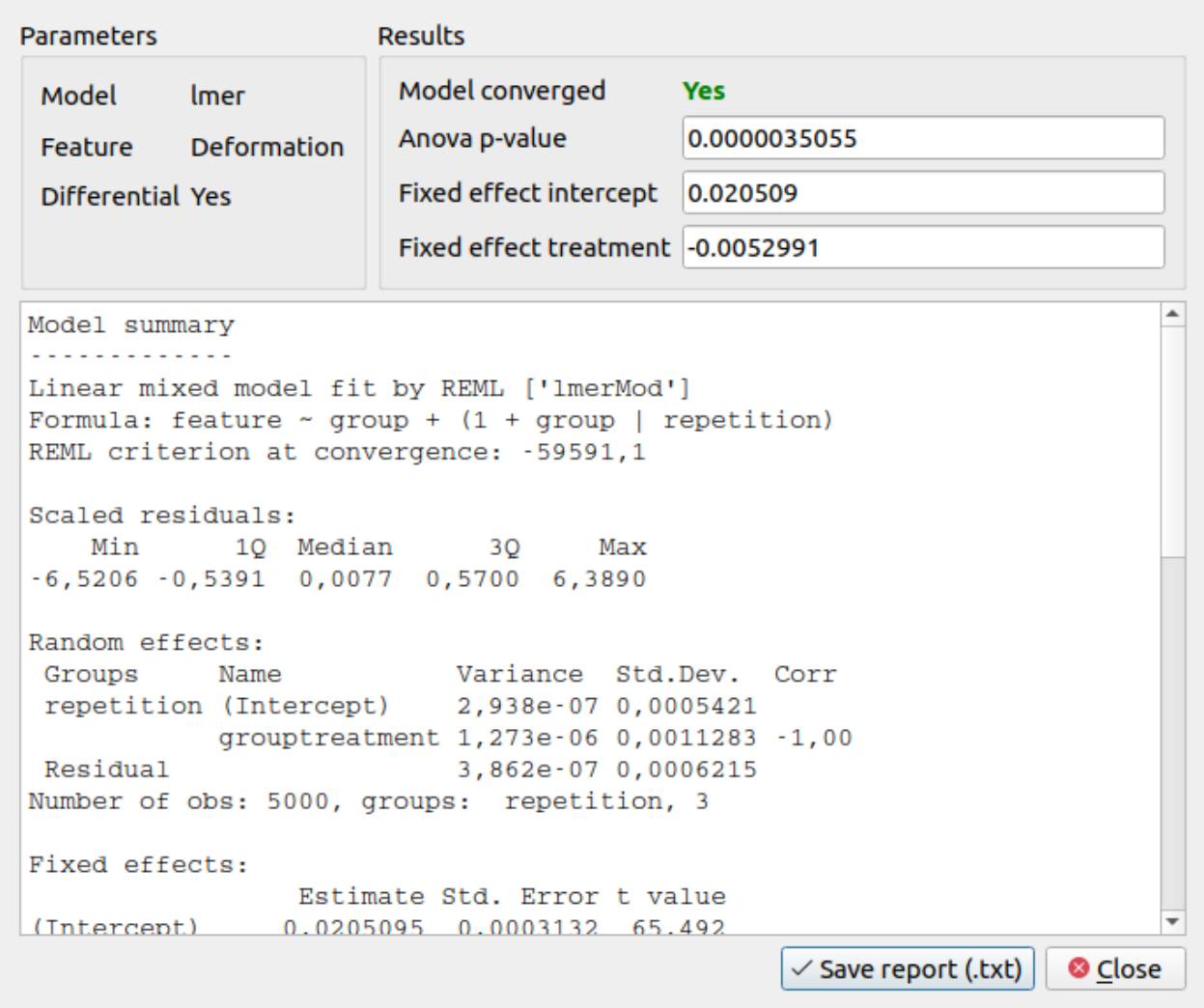
To check whether the deformation we see is not just an inherent deformation (visible in the reservoir data), we would like to perform a differential feature analysis.

Please load `this session` into Shape-Out (via *File* | *Open Session*). The experimental data are accessed from DCOR, so you will need a working internet connection. The session contains ten datasets (reservoir measurements in addition to the measurements above).

As in the example above, go to *Compute* | *Statistical significance* and set the groups and repetitions.



The results of the LMM analysis for differential deformation are:



The p-value for the differential deformation is magnitudes lower than the p-value for the (non-differential) deformation in the previous example. This indicates that there is a non-negligible initial deformation of the cells in the reservoir.

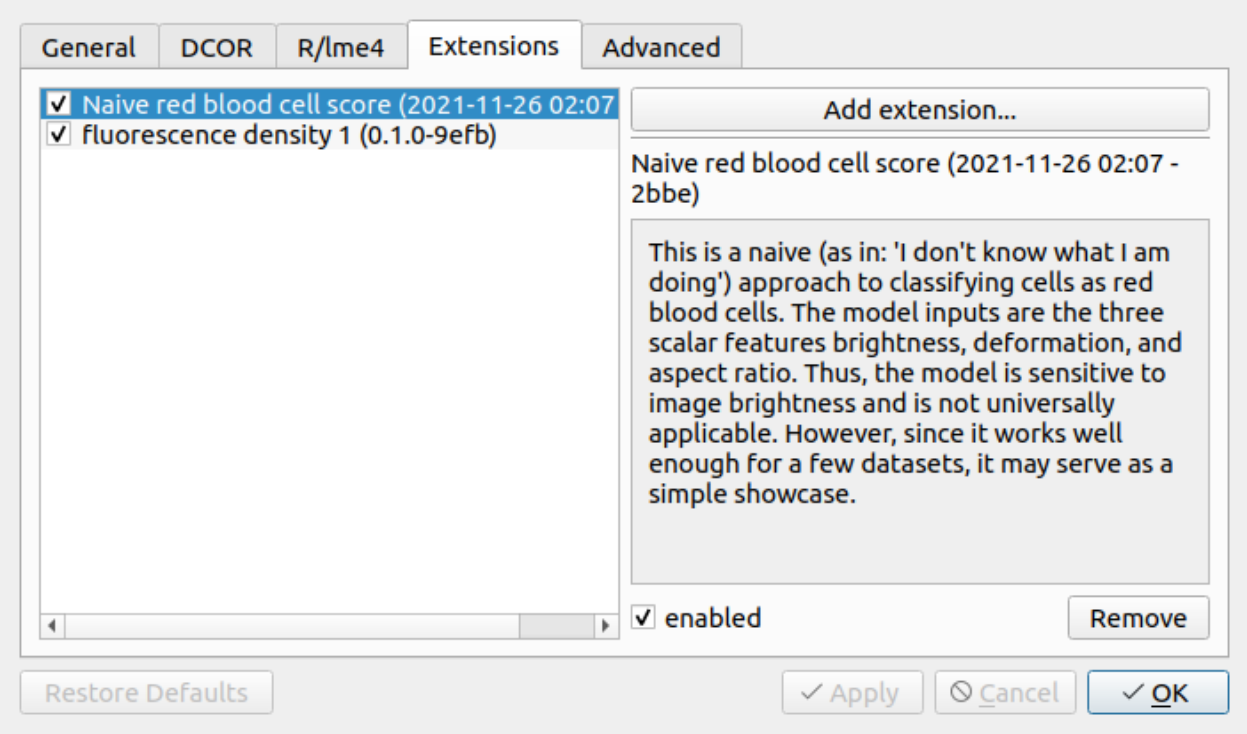
4.6 Loading Extensions (machine-learning and plugin features)

Since version 2.9.0, Shape-Out 2 allows loading dclab plugin features and machine-learning features. You may need those if you need to quantify anything that is not covered by the default features.

Note: If you installed Shape-Out via installer (not via *pip*), then many extensions might not work due to software dependencies that those extensions might have. If this happens, please create an issue in the Shape-Out 2 repository so we can find a solution.

Warning: Extensions can be harmful. Please only load extensions that you received first-hand from people you trust.

You can load and manage extensions via the *Edit* | *Preferences* dialog in the *Extensions* tab.



4.6.1 Example: Fluorescence density

Download the `extension_fl1_density.py` extension and add it to Shape-Out. You will see a new scalar feature named "FL-1 density [a.u.]" that quantifies the collected fluorescence signal per object volume for the fluorescence channel 1. Using this extension as a template, you could create the density features for the other fluorescence channels as well.

4.6.2 Example: RBC-detection with machine-learning

Note: For this example you will need to have tensorflow installed.

Download the `extension_naive_rbc_score.modc` extension and add it to Shape-Out. You will see a new scalar feature named "RBC score (naive)". This showcase model only consists of one dense layer of size six, but already does a better-than-random job in identifying red blood cells.

4.7 Export statistical data

The *Quick View window* quantifies the statistics for the features shown there. If you need access to statistics for more features of a dataset or if you need to quantify the statistics for multiple datasets on disk, you can use the *Statistics* tool available via the *Compute* menu.

There are two modes of operation. The first mode exports statistical data from the current session (Fig. 4.12). Statistics are always computed after the filter ray has been applied. You may select the features in the table on the left and the desired statistics in the table on the right.

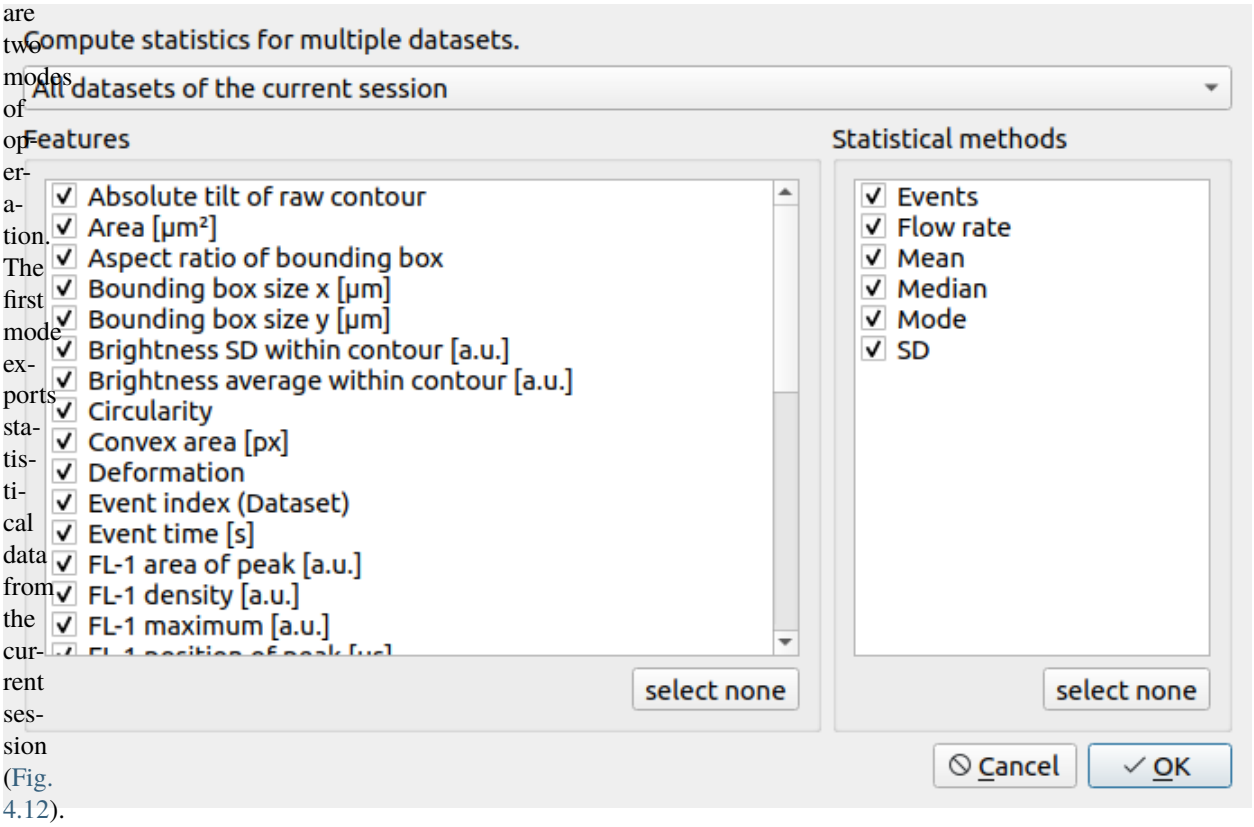
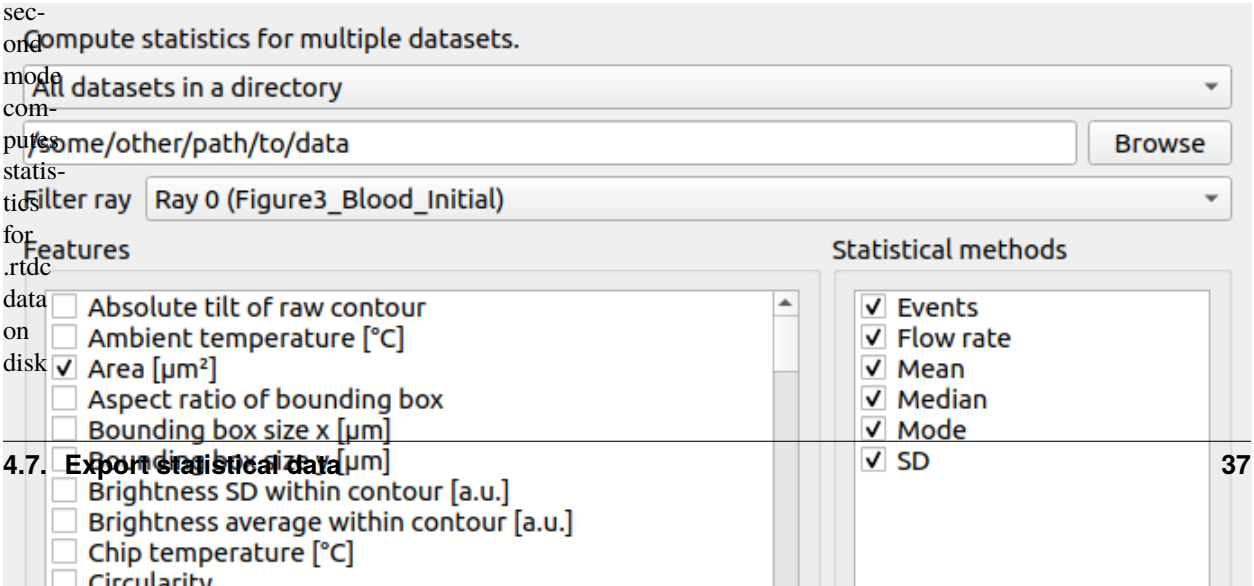


Fig. 4.12: Statistics export of current session

The second mode computes statistics for .rtdc data on disk



(Fig. 4.13). Select the *All datasets in a directory* option and (optionally) a filter ray from the current session that you would like to apply before computing the statistics.

Note that statistics may not be available for all selected features in all datasets. In such a case, statistics are exported as *nan* values in the resulting .tsv file.

CHAPTER

FIVE

CHANGELOG

List of changes in-between Shape-Out releases.

5.1 version 2.11.3

- setup: bump dclab from 0.42.1 to 0.42.2 (lazy-load .rtdc data which increases initial opening times)
- setup: bump pyqtgraph from 0.12.3 to 0.12.4

5.2 version 2.11.2

- setup: bump dclab from 0.42.0 to 0.42.1 (support background image access from DCOR)

5.3 version 2.11.1

- fix: allow to specify 0.5 increments when filtering integer features (#123)
- setup: bump dclab from 0.41.0 to 0.42.0

5.4 version 2.11.0

- feat: allow to load encrypted dcor access tokens (#122)
- setup: bump dclab from 0.39.17 to 0.41.0
- setup: bump scipy to 1.8.0
- build: bump Python version to 3.9

5.5 version 2.10.0

- feat: export filter ray for each dataset via shortcut (#119)
- setup: bump dclab from 0.39.15 to 0.39.17 (#118)

5.6 version 2.9.6

- fix: extensions not loaded during startup on Windows (#117)

5.7 version 2.9.5

- fix: polygon translation via drag-n-drop in polygon edit mode was not taken into account (#115)
- enh: add (de-)select all button for lists in ComputeStatistics (#116)
- enh: convert all selectable lists in dialogs to new BulkList class
- setup: bump dclab from 0.39.13 to 0.39.15

5.8 version 2.9.4

- enh: display session file name in window title
- setup: drop support for older numpy versions
- setup: bump dclab from 0.39.9 to 0.39.13 (fix visualization of the "ml_class" score when the ml_score features contain nans)

5.9 version 2.9.3

- fix: implement closeEvent in main and ask user before exiting
- fix: fix possible failure during update check
- fix: show a message box when the R version is incompatible (#114)
- ref: only reload R/lme4 information in preferences when the corresponding tab is selected to avoid preferences lock-up (#114)
- tests: use hidden settings variable to skip asking the user to clear the session
- setup: bump dclab from 0.39.5 to 0.39.9

5.10 version 2.9.2

- setup: bump dclab from 0.39.3 to 0.39.5 (fix opening files with some online filter settings)

5.11 version 2.9.1

- enh: add button "Find best spacing" in Plot tab in Analysis View that iteratively finds sane spacing values (#110)
- enh: notify user in plot if contour spacing is too large (#110)
- enh: add action to remove all datasets from a session (#97)
- fix: mysterious behavior of subclassed pg.ColorBarItem (#109)
- setup: bump dclab from 0.39.1 to 0.39.3 (increased verbosity, renamed feature labels, refactored DCOR access)

5.12 version 2.9.0

- feat: add support for extensions in the form of loading dclab plugin and machine-learning features (#85)
- setup: drop support for Python 3.7
- setup: bump dclab from 0.37.2 to 0.39.1

5.13 version 2.8.1

- fix: wrong scale for viscosity in bulk emodulus computation (#104)
- fix: save unused polygon filters in sessions (#101)
- fix: allow to delete polygon filters (#100)
- setup: bump dclab from 0.37.0 to 0.37.2 (enhanced speed for operations involving contour computation)

5.14 version 2.8.0

- BREAKING CHANGE: The volume feature in all previous versions was overestimated by about 2µm^3. Please re-run your analysis pipeline.
- setup: bump pyqtgraph from 0.12.2 to 0.12.3
- setup: bump dclab from 0.36.1 to 0.37.0 (faulty volume computation)

5.15 version 2.7.4

- setup: bump dclab from 0.35.6 to 0.36.1 (improved export speed)

5.16 version 2.7.3

- build: remove console window (#102)

5.17 version 2.7.2

- setup: bump dclab from 0.35.4 to 0.35.6 (Young's modulus could not be computed in some cases)

5.18 version 2.7.1

- enh: display online polygon filters in metadata tab (#94)
- enh: show user configuration section in metadata tab (#86)
- setup: bump dclab from 0.34.3 to 0.35.4
- tests: change test data to avoid warnings in dclab 0.35.0

5.19 version 2.7.0

- feat: accept DCOR drag events from DCOR-Aid data browser

5.20 version 2.6.9

- fix: don't reset color and name of dataset during duplicate matrix action (#96)

5.21 version 2.6.8

- fix: adapt color scheme to work in dark mode (#92)

5.22 version 2.6.7

- fix: correctly handle missing features on data export (#80)
- ref: cleanup (default arguments and PyQt5 object references)
- setup: bump dclab from 0.34.2 to 0.34.3 (set R_HOME)
- build: bump rpy2 from 3.4.2 to 3.4.5
- build: bump pyqtgraph from 0.12.1 to 0.12.2 and remove colorbar export workarounds

5.23 version 2.6.6

- fix: false alarms about wrong pyqtgraph version on startup (removed the responsible code, because there are now more regular releases of pyqtgraph)
- enh: allow passing session files, datasets, and filters via the command line (#88)
- enh: associate the .so2 file format with Shape-Out on Windows (#88)
- build: migrate to Inno Setup 6
- ref: avoid pyqtgraph DeprecationWarning due to *translate*

5.24 version 2.6.5

- fix: migrate to pg.ColorBarItem (with some manual patches from upstream) to resolve issues with SVG export (#87)
- fix: units of viscosity is mPa*s (#91)
- setup: bump pyqtgraph from 0.12.0 to 0.12.1
- setup: bump dclab from 0.33.3 to 0.34.2
- ci: fix build issues for macOS

5.25 version 2.6.4

- fix: HTML-escape plot titles and axis labels (#82)

5.26 version 2.6.3

- build: pin rpy2 to 3.4.2 on Windows (#84)

5.27 version 2.6.2

- setup: migrate to pyqtgraph 0.12.0
- setup: bump dclab from 0.33.1 to 0.33.3
- docs: fix rtd build

5.28 version 2.6.1

- setup: remove deprecated setup.py test
- setup: bump dclab to 0.33.1 (filters were ignored when exporting trace data to hdf5, see https://github.com/ZELLMECHANIK-DRESDEN/DCKit/issues/17)

5.29 version 2.6.0

- feat: add background correction in quickview (#54)
- fix: workaround for different plot widths due to title label (#33)

5.30 version 2.5.2

- build: add pyinstaller hook for pyqtgraph

5.31 version 2.5.1

- fix: preferences dialog did not open in macOS App bundle (#76)
- enh: disable R settings in preferences dialog if rpy2 is missing
- build: bump dclab from 0.29.1 to 0.32.4 (this should have happened in the 2.5.0 release which introduced R-lme4 analysis)
- build: bump pyqtgraph to commit c1900 (post 0.11.1) which mainly improves plot export (#59)
- tests: introduce manual tests that require a human in "tests-manual"
- ref: key [calculation]: "emodulus model" is deprecated in dclab
- ref: use "lut_identifier" instead of deprecated "method" keyword when retrieving isoelasticity lines

5.32 version 2.5.0

- feat: implement linear-mixed effects models with R/lme4 (#55)
- ci: overhauled build process

5.33 version 2.4.15

- ci: fix rtd build
- fix: typo in infobox in analysis view (#69)
- fix: macOS builds did not use our pyqtgraph fork (reported in #70)

5.34 version 2.4.14

- maintenance release

5.35 version 2.4.13

- maintenance release

5.36 version 2.4.12

- maintenance release

5.37 version 2.4.11

- maintenance release

5.38 version 2.4.10

- maintenance release

5.39 version 2.4.9

- maintenance release

5.40 version 2.4.8

- maintenance release

5.41 version 2.4.7

- maintenance release

5.42 version 2.4.6

- maintenance release

5.43 version 2.4.5

- maintenance release

5.44 version 2.4.4

- fix: box filters selections bled to other filters (#67)
- build: do not use development version of dclab
- ci: move to GitHub Actions

5.45 version 2.4.3

- enh: improve UI (tooltips, icons)
- build: Windows build won't start due to breaking change in Windows ucrtbase.dll (#64)
- build: Migrate Windows build to PyInstaller 4.1

5.46 version 2.4.2

- fix: a regression introduced during refactoring in 2.4.0 made it impossible to select the temperature for known media
- fix: allow to reorder the dataset list (#46)
- enh: iconify buttons at top of Analysis View

5.47 version 2.4.1

- fix: strip white-space characters from line edits in the preferences to avoid line breaks introduced in the API key via copy-paste
- fix: not possible to create plots for datasets without the bright_avg feature; added a check for feature availability before updating the interface (#62)

5.48 version 2.4.0

- feat: allow to set Young's modulus parameters for all open datasets via the "Bulk action" menu (#60)
- fix: force temperature for Young's modulus computation to the value stored in the dataset for the "config" scenario
- enh: update menu keyboard shortcuts

5.49 version 2.3.0

- enh: added preferences dialog
- enh: improved user experience for searching DCOR data
- ref: migrate settings to QSettings
- setup: remove appdirs dependency, because we let QSettings determine where to store settings

5.50 version 2.2.6

- fix: support user-defined media introduced in dclab 0.29.1 (#53)
- fix: porosity box filter precision at lower limit (#58)
- docs: update bibliography
- bump dclab from 0.29.0 to 0.29.1

5.51 version 2.2.5

- setup: correct installation directory on Windows
- setup: bump dclab from 0.27.11 to 0.29.0
- fix: main window not focused after startup (#52)

5.52 version 2.2.4

- fix: UI slot issues (probably a racing condition) in the analysis view; editing the viscosity for "other" medium not possible on frozen Windows and macOS application (#49)

5.53 version 2.2.3

- fix: update Quick View plot when values for the Young's modulus computation are changed (#44)
- fix: use fixed point size in Block Matrix
- enh: improve plot matrix column/row visualization and show flow rate (#45)
- docs: update visualization of Young's modulus look-up tables using new figure script from dclab
- build: bump Python to 3.8 for Windows builds

- setup: bump dclab from 0.27.5 to 0.27.11 (improvement of hierarchy child handling)

5.54 version 2.2.2

- fix: correct box filter ranges when feature contains inf values
- enh: add pretty tool icons from KDE breeze theme

5.55 version 2.2.1

- fix: zero-division error when setting "one plot per dataset" for an empty plot (#41)
- build: pin PyQt5 to 5.14.2 on Linux/macOS

5.56 version 2.2.0

- setup: bump dclab from 0.26.2 to 0.27.5 (support for machine learning features)
- fix: modify the GUI to enable box filtering for ml_score_??? features
- ref: pipeline.get_features now only returns features that are available (previously all features were returned when no datasets were in the pipeline)

5.57 version 2.1.6

- fix: export file names were allowed to have invalid characters
- fix: Quick View showed filtered events that were not plotted (statistics were computed correctly though)
- fix: Quick View was not very informative when all events were filtered away - a message is now shown
- setup: bump dclab from 0.26.1 to 0.26.2 (fix bin num computation using Doane's formula)

5.58 version 2.1.5

- fix: marker hue feature data min/max was always computed even if not selected; for datasets without bright_avg (standard hue) this resulted in long waiting times
- fix: avoid pyqtgraph error messages in pipeline plots by disabling hovering events (not used anyway)
- fix: plot export did not work for single plots
- fix: exported event images sometimes had wrong contrast

5.59 version 2.1.4

- enh: add busy cursors wherever it seems fitting
- setup: bump dclab from 0.24.7 to 0.26.1 (volume-deformation isoelasticity lines, improved DCOR caching)

5.60 version 2.1.3

- fix: opening sessions did not work due to missing slot decorator
- fix: in some cases the temperature/viscosity setting was not correctly written to the UI for individual slots when switching between slots with fundamentally different settings (e.g. channel and reservoir)
- fix: KeyError when switching between slots with different temperature options
- setup: bump dclab from 0.23.0 to 0.24.7 (equidistant isoelastics, faster polygon filtering, crosstalk correction failed due to improper configuration update since Shape-Out 2.1.1)

5.61 version 2.1.2

- fix: ValueError when trying to compute contour spacing, because no datasets are selected for a plot
- fix: coloring scatter plot by feature did not work when the feature contains nan values
- fix: search path in session should not be filename but its directory
- setup: bump dclab from 0.22.5 to 0.23.0 (possible HDF5 export issue)

5.62 version 2.1.1

- fix: include private datasets when searching with DCOR API Key
- fix: remove UI option to compute Young's modulus for reservoir data (new "emodulus enabled" key in pipeline slot)
- fix: *Pipeline* object not returned by *session.open_session*
- setup: bump dclab from 0.22.1 to 0.22.5 (improve contour retrieval for mask images with artefacts, disable computation of Young's modulus for reservoir measurements)

5.63 version 2.1.0

- feat: basic DCOR support
- fix: improved session support (older pre-release sessions)
- docs: add DCOR guick guide
- enh: export event image via context menu in Quick View (#35)
- setup: bump dclab from 0.21.1 to 0.22.1 (DCOR support)

5.64 version 2.0.3

- setup: bump dclab from 0.20.3 to 0.21.1
- docs: add terminology section (#12)

5.65 version 2.0.2

- fix: IndexError when removing a plot (#36)
- fix: ValueError when filtering all events in Quick View (#37)
- fix: TypeError when removing datasets and opening Quick View (#38)
- fix: RuntimeError when removing datasets and opening Quick View (#38)
- fix: ValueError when removing activated datasets and opening Quick View

5.66 version 2.0.1

- fix: correctly distinguish prereleases when checking for new versions
- enh: allow loading data via drag&drop

5.67 version 2.0.0

- initial release

CHAPTER

SIX

BILBLIOGRAPHY

CHAPTER

SEVEN

IMPRINT/IMPRESSUM

7.1 Imprint and disclaimer

For more information, please refer to the imprint and disclaimer (Impressum und Haftungsausschluss) at https://www.zellmechanik.com/Imprint.html.

7.2 Privacy policy

This documentation is hosted on https://readthedocs.org/ whose privacy policy applies.

CHAPTER

EIGHT

INDICES AND TABLES

- genindex
- modindex
- search

BIBLIOGRAPHY

[HWT02] David Halpern, Howard B. Wilson, and Louis H. Turcotte. Gauss integration with geometric property applications. In *Advanced Mathematics and Mechanics Applications Using MATLAB, Third Edition*. Chapman & Hall, sep 2002. doi:10.1201/9781420035445.ch5.

[HMMO18] M. Herbig, A. Mietke, P. Müller, and O. Otto. Statistics for real-time deformability cytometry: Clustering, dimensionality reduction, and significance testing. *Biomicrofluidics*, 12(4):042214, 2018. doi:10.1063/1.5027197.

[Her17] Christoph Herold. Mapping of Deformation to Apparent Young's Modulus in Real-Time Deformability Cytometry. *ArXiv e-prints 1704.00572 [cond-mat.soft]*, 2017. arXiv:1704.00572v1.

[MOG+15] Alexander Mietke, Oliver Otto, Salvatore Girardo, Philipp Rosendahl, Anna Taubenberger, Stefan Golfier, Elke Ulbricht, Sebastian Aland, Jochen Guck, and Elisabeth Fischer-Friedrich. Extracting Cell Stiffness from Real-Time Deformability Cytometry: Theory and Experiment. *Biophysical Journal*, 109(10):2023–2036, nov 2015. doi:10.1016/j.bpj.2015.09.006.

[MMM+17] M. Mokbel, D. Mokbel, A. Mietke, N. Träber, S. Girardo, O. Otto, J. Guck, and S. Aland. Numerical Simulation of Real-Time Deformability Cytometry To Extract Cell Mechanical Properties. *ACS Biomaterials Science & Engineering*, 3(11):2962–2973, jan 2017. doi:10.1021/acsbiomaterials.6b00558.

[NUH+20] Ahmad Ahsan Nawaz, Marta Urbanska, Maik Herbig, Martin Nötzel, Martin Kräter, Philipp Rosendahl, Christoph Herold, Nicole Toepfner, Marketa Kubankova, Ruchi Goswami, Shada Abuhattum, Felix Reichel, Paul Müller, Anna Taubenberger, Salvatore Girardo, Angela Jacobi, and Jochen Guck. Intelligent image-based deformation-assisted cell sorting with molecular specificity. *Nature Methods*, 2020. doi:10.1101/862227v2.

[NUH+19] Ahmad Ahsan Nawaz, Marta Urbanska, Maik Herbig, Martin Nötzel, Martin Kräter, Philipp Rosendahl, Christoph Herold, Nicole Töpfner, Marketa Kubankova, Ruchi Goswami, and et al. Sorting real-time fluorescence and deformability cytometry (sort-fdc) - manuscript data. Dec 2019. doi:10.6084/m9.figshare.11302595.v1.

[ORM+15] Oliver Otto, Philipp Rosendahl, Alexander Mietke, Stefan Golfier, Christoph Herold, Daniel Klaue, Salvatore Girardo, Stefano Pagliara, Andrew Ekpenyong, Angela Jacobi, Manja Wobus, Nicole Töpfner, Ulrich F Keyser, Jörg Mansfeld, Elisabeth Fischer-Friedrich, and Jochen Guck. Real-time deformability cytometry: on-the-fly cell mechanical phenotyping. *Nature Methods*, 12(3):199–202, feb 2015. doi:10.1038/nmeth.3281.

[RPJ+18] Philipp Rosendahl, Katarzyna Plak, Angela Jacobi, Martin Kraeter, Nicole Toepfner, Oliver Otto, Christoph Herold, Maria Winzi, Maik Herbig, Yan Ge, Salvatore Girardo, Katrin Wagner, Buzz Baum, and Jochen Guck. Real-time fluorescence and deformability cytometry. *Nature Methods*, 15(5):355–358, apr 2018. doi:10.1038/nmeth.4639.

[THO+17] Nicole Toepfner, Christoph Herold, Oliver Otto, Philipp Rosendahl, Angela Jacobi, Martin Krater, Julia Stachele, Leonhard Menschner, Maik Herbig, Laura Ciuffreda, Lisa Ranford-Cartwright, Michal Grzy-

bek, Unal Coskun, Elisabeth Reithuber, Genevieve Garriss, Peter Mellroth, Birgitta Henriques-Normark, Nicola Tregay, Meinolf Suttorp, Martin Bornhauser, Edwin R. Chilvers, Reinhard Berner, and Jochen Guck. Detection of human disease conditions by single-cell morpho-rheological phenotyping of whole blood. *eLife*, 7:e29213, jun 2017. doi:10.1101/145078.

[XRM+20] Miguel Xavier, Philipp Rosendahl, Paul Müller, Maik Herbig, and Jochen Guck. Real-time deformability cytometry data of primary human skeletal stem cells and the human osteosarcoma cell line mg-63. Feb 2020. doi:10.6084/m9.figshare.11662773.v2.