

Analysis of Multidimensional Microscopy Data Using Cell-ACDC

Francesco Padovani^{*,1}, Timon Stegmaier^{*,1}, Benedikt Mairhörmann^{1,2,3}, Kurt M. Schmöller¹

¹ Institute of Functional Epigenetics, Molecular Targets and Therapeutics Center, Helmholtz Zentrum München ² Institute of Network Biology, Molecular Targets and Therapeutics Center, Helmholtz Zentrum München ³ Institute of AI for Health, Computational Health Center, Helmholtz Zentrum München

* These authors contributed equally

Corresponding Authors

Francesco Padovani

francesco.padovani@helmholtz-munich.de

Kurt M. Schmöller

kurt.schmoller@helmholtz-munich.de

Citation

Padovani, F., Stegmaier, T., Mairhörmann, B., Schmöller, K.M. Analysis of Multidimensional Microscopy Data Using Cell-ACDC. *J. Vis. Exp.* (225), e68954, doi:10.3791/68954 (2025).

Date Published

November 7, 2025

DOI

10.3791/68954

URL

jove.com/video/68954

Abstract

Recent advances in quantitative microscopy for the life sciences have enabled experimental biologists to probe cells with unprecedented resolution and speed. At the same time, the AI revolution has dramatically increased the amount of information that can be extracted from multidimensional microscopy data. However, the large amount of data generated and the complexity of state-of-the-art AI models pose a severe bottleneck at the image analysis stage. Cell-ACDC is an open-source, user-friendly software that provides a powerful end-to-end solution for segmentation, tracking, and quantitative analysis of single cells in multidimensional microscopy data. It is tailored for experimental biologists who may lack the advanced technical expertise required to implement such models. This article shows how to utilize the framework to easily leverage the most recent models, along with many tools for smart and semi-automated data correction, to maximize the amount of biological information obtainable. Cell-ACDC supports multi-channel, time-lapse, and z-stack microscopy data, and provides a dedicated toolset tailored to each type of data dimensionality. Because of its modular design, which allows new models to be seamlessly integrated and directly accessed by biologists, Cell-ACDC has the potential to serve as a reference tool for microscopy data analysis.

Introduction

Microscopy has become fundamental to accelerate biological discovery at every stage of research and development, from basic science^{1,2,3} to drug discovery and testing^{4,5} and across a remarkable range of sizes, from cell cultures to tissues^{6,7}, organoids^{8,9}, and whole organisms¹⁰. These advanced microscopy techniques, however, come with two

main drawbacks. First, microscopy data is inherently large¹¹, especially when acquiring multiple dimensions (e.g., time and volumes). Secondly, accurate extraction of the rich biological information in the data requires the deployment of advanced image analysis frameworks that often build on AI models. This task can be daunting for experimental

biologists. Therefore, the steps of data handling, processing, and analysis can greatly slow down scientific research while reducing the potential for novel discoveries. Ideally, software frameworks for bioimage analysis should assist the user in every step of the analysis, from raw microscopy file handling through segmentation and tracking to downstream analysis for extracting biological insights. In practice, the rapid development of new software for bioimage analysis, while being a positive outcome, has had the side effect of creating a scattered landscape of tools specific to a particular analysis step or requiring advanced programming expertise. This leaves the user with the challenging task of assembling the analysis workflow, often resulting in sub-optimal pipelines where data is saved, manipulated, and converted multiple times to make it compatible with the next tool. In addition, the development of bioimage analysis is not standardized to a single programming language, with many tools being developed as ImageJ¹² or QuPath¹³ plugins (Java), Napari plugins (Python)¹⁴, or Python scripts. In some cases, this challenging environment leads scientists to opt for manual analysis, a process that is not only slow but also introduces human bias and hinders reproducibility.

To solve this, Cell-ACDC (Cell-Analysis of the Cell Division Cycle)¹⁵ was developed, an open-source GUI-based software toolset written in Python to analyze multidimensional microscopy data. Crucially, Cell-ACDC provides a multitude of pre-implemented, and automatically installed (when needed) state-of-the-art models for both segmentation (Cellpose, StarDist, Segment Anything, YeaZ, YeastMate, etc.^{16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28}) and tracking (Trackastra, Trackpy, Bayesian Tracker, Cell-ACDC, etc.^{19, 29, 30, 31, 32, 33, 34}). These models are complemented with a framework for visualizing annotated data and computer-assisted manual corrections. Additionally, within

the same framework, Cell-ACDC provides a module for each step of the image analysis pipeline, automatically taking care of data handling, processing, and saving (**Figure 1**). More specifically, it enables the user to perform instance segmentation (e.g., single cells), object tracking, annotation of cell states (e.g., cell cycle stage), and various forms of quantification, including analysis of the available fluorescence channels.

One of Cell-ACDC's main strengths is the analysis of live-cell time-lapse microscopy data, which poses significant challenges due to the need for consistency across time points. While numerous models exist for automated segmentation and tracking of single cells, manual corrections remain essential for addressing complex biological questions. Cell-ACDC offers a suite of tools specifically designed to streamline the correction process. It integrates intelligent algorithms to minimize the number of manual adjustments. Notably, corrections are automatically propagated across all relevant future and past frames, preserving data integrity throughout the analysis. Given its modular design and growing user base, the ongoing development of this software is a priority, with continued efforts focused on integrating new models and addressing the evolving needs of the community.

Protocol

NOTE: Cell-ACDC is designed to be as clear and intuitive to users with no programming background as possible. This is mainly achieved by breaking down the analysis workflow into smaller, easy-to-follow steps and providing additional information through tooltips and info buttons. Since Cell-ACDC covers a wide range of steps whose order of execution is often not sequential, a decision chart is shown in **Figure 2**. The following guide will go through a specific example. Steps

10 and 11 can be used to import other data instead of using the provided data downloaded in step 3.

1. Installing Cell-ACDC

NOTE: Cell-ACDC is currently distributed as a Python package via the Python Package Index (PyPI) and can be installed with the command `pip install "cellacdc[torch]"`.

1. Setup Miniforge

1. **Download** the installer from the Miniforge website (see the **Table of Materials** for details).
2. Install Miniforge: For **Windows**, run the downloaded installer and follow the instructions. For **Mac or Linux**, open the terminal and run the following command:


```
curl -L -O "https://github.com/conda-forge/miniforge/releases/latest/download/Miniforge3-$(uname)-$(uname -m).sh"
```
3. Once the installation is complete, open the correct terminal by following the instructions below:
 1. For **Windows**, press **Win + S**, type **Miniforge Prompt**, and press **Enter**.
 2. For **Mac/Linux**: open the Terminal app.

2. Manage the virtual environment.

1. In the prompt, type the following command to create a virtual environment. Then press **Enter** to execute the command.


```
conda create -y -n acdc python=3.12
```
2. Activate the environment by running:


```
conda activate acdc
```

3. Installation

1. With the environment active, install Cell-ACDC by running the following command:


```
pip install "cellacdc[torch]"
```
2. After installation, run `Cell-ACDC -y` to let the software finish its setup.

2. Running Cell-ACDC

1. Open the correct terminal (see step 1.1.3).
2. Activate the environment by running the command:


```
conda activate acdc
```
3. Start Cell-ACDC by running the following command:


```
Cell-ACDC
```

3. Downloading sample data

NOTE: The example data will be downloaded to "C:\Users\%USERNAME%\acdc-appdata\acdc-examples\TimeLapse_2D\Position_8" on Windows, and to "~/acdc-appdata/acdc-examples/TimeLapse_2D/Position_8" on MacOS and Linux. If the welcome guide is not displayed, select **Help > Welcome Guide (Figure 3B)** in the menu bar of the main Cell-ACDC launcher window (**Figure 1**), .

1. Select **Download and test with a time-lapse example (Figure 3B)**.
2. Wait until the download is finished.
3. Click **No, thanks** to stop loading the data directly into the GUI.

4. Data prep module

NOTE: Data can be aligned before segmentation, and **regions of interest (ROIs)** can be selected. These steps are optional but beneficial if the field of view is displaced over time or if only parts of the data are of interest, respectively.

1. Click on **Launch data prep module...** in the main window to open the data preparation module (**Figure 1A**, Data pre-processing module).
2. Select the folder containing the data.
 1. Click the **folder icon** in the toolbar of the new window to load the microscopy data (**Figure 4A**).
 2. Select the folder containing the data, then confirm the selection by pressing **Select Folder**.
3. Select a channel for the alignment process by using the dropdown menu to select the **phase_contr** channel. Then, press **Ok** to confirm the selection (**Figure 4B**).
4. Press **Ok** to confirm image properties (**Figure 4C**).

NOTE: If working with z-stack data, but only one slice should be used for segmentation, select the appropriate z-slice using the **slider**. Use the buttons in the toolbar to apply the selection to other frames if needed.
5. Run the alignment process.
 1. Click the **start** button, which is also located in the toolbar (**Figure 5A**).
 2. Click **Yes** to start the alignment process.

NOTE: Click **No** to skip alignment.
 3. Press **Ok** to confirm that the information about padding was acknowledged.

NOTE: Alignment may take time, depending on data size.
 4. Press **Ok** to finish alignment once the process is complete.
6. Set ROIs and background ROIs.
 1. Go to the last frame of the segmentation video using the frame selection slider in the bottom part of the data prep GUI.
 2. Adjust the automatically added background ROI by dragging it across the image or resizing it using the diamonds. Make sure that the background ROI has no cells. Leave the ROI as is (**Figure 5B**).

NOTE: To define additional ROIs, click the **add crop ROI** button (located in the toolbar) and select them in the viewer. Usually, the ROI should be as small as possible while still encompassing all cells of interest across relevant frames. However, to ensure reproducibility within this protocol, it is recommended not to modify it.
7. To crop the selected ROIs into new image files, click the leftmost **Crop** button.

NOTE: This step is optional. If cropped images are not needed, the window can be closed. The coordinates of all ROIs and background ROIs are saved automatically and can be used later for segmentation. If the ROI was not changed, these buttons will do nothing. Crop options include XY directions, z-slices, or specific time ranges. Each button is labelled with the dimensions that will be cropped.
8. Save the aligned data
 1. Close the window using the window close button (e.g., the **X** in the top-right corner on Windows or the red dot in the top-left corner on macOS or Linux).
 2. Press **Yes, save aligned data** to save the aligned data.
 3. Click **Yes, save aligned data** again to confirm.
9. Save cropped data if step 4.7 was not skipped and the ROI was not modified.
 1. Select **Yes, save cropped data** to save the cropped data.

2. Press **Yes, crop please** to confirm saving the crop.
3. Click **Ok** to confirm the default folder path.
NOTE: Select a different folder to keep the uncropped data.
4. Select **Yes, overwrite** to confirm if the default path was used before.
5. Press **Ok** to confirm after the process is complete.

5. Finding the best model and parameters for segmentation

NOTE: Cell-ACDC offers a GUI with real-time feedback to find the best segmentation parameters (**Figure 6**). In general, these models are provided by third parties, each with its own documentation. It is the user's responsibility to determine the best segmentation model and its optimal parameters, and users should check the documentation of the respective model they are trying for further information.

1. Click on **Launch GUI...** in the main window (**Figure 1A**, Visualize and correct module).
2. Load the sample data.
 1. Click the **folder icon** in the toolbar of the new window to open the folder selection menu.
 2. Select the folder containing the data, then press **Select Folder** to confirm selection.
3. Select a channel for visualization. Use the dropdown menu to select the channel **phase_contr** for segmentation, then select **Ok** to confirm.
4. Finish loading the data.
 1. Press **Ok** to confirm the default segmentation mask name.
2. Click **Ok for loaded Positions** to confirm image properties.
3. Select **No** to prevent loading of additional fluorescence data.
5. In the **mode selector** (**Figure 6**, "Mode selector"), select the mode **Segmentation and Tracking**.
6. Find the best preprocessing settings.
 1. Navigate to **Image > Pre-processing...** in the top menu bar to open the custom preprocessing window (**Figure 7A**).
 2. Select **Remove Hot Pixels** or another desired preprocessing step in the dropdown menu.
 3. Use the **cog icon** to initialize and change settings for a step. Use the **info button** to display information about the step and available parameters. Leave the settings unchanged.
 4. Use the **plus icon** to add one more step.
 5. Select **Rescale Intensities** and confirm settings by repeating steps 5.6.1 and 5.6.2.
 6. Check the **preview checkbox** to see the effects of the steps in real time.
 7. Click **Apply to all frames** (**Figure 7B**).
 8. Press **Save pre-processed data** to start the saving process.
 9. Press **Ok** to confirm the default name.
 10. Close the **Pre-processing recipe** window.
7. Find the best segmentation settings.
 1. Select **Segment > Segment displayed frame** in the top ribbon, then select **YeaZ_v2** (**Figure 8A**).
 2. Press **Ok** to download **YeaZ_v2** if prompted.

NOTE: The download can take several minutes. Progress is displayed in the console window. Do not close the GUI during download, even if unresponsive.

3. Leave the default parameters unchanged.

NOTE: Most parameters have **info** buttons that provide detailed guidance.

4. Enable postprocessing by checking **Post-processing segmentation parameters** (Figure 8B).

5. Accept the parameters by clicking **Ok**.

NOTE: Segmentation might take a moment, depending on model complexity, image size, and computer specifications. Especially, Cellpose version 4 can be very slow.

6. If asked about activating automatic segmentation, select **No**.
7. Verify the segmentation works well.
8. Repeat step 5.7.1 to reopen the segmentation parameters.
9. If segmentation results are as expected, press **Save all parameters to recipe file**. Otherwise, change the segmentation parameters and repeat steps 5.7.4 to 5.7.8.
10. Use the **text input** to give the segmentation recipe the name **test**.
11. Click **Ok** to accept the name.
12. Press **Ok** to finish saving the workflow.
13. **Close** the parameter selection screen.

8. Closing the GUI window

1. Close the GUI window.

2. Press **No** to not save before closing.

6. Segmenting and tracking (batch processing)

1. Click on **Launch segmentation module...** in the main window (Figure 1A, "Segment and track" module).

1. Select the sample data. Use Cell-ACDC's folder selector to choose the folder containing the data, then press **Select Folder** to confirm the selection.

2. Select the parameters for batch processing.

1. Select the channel **phase_contr_preprocessed** as the channel for segmentation. Press **Ok** to confirm the selection.

NOTE: Some models use an additional channel as input. This channel can later be selected when setting other segmentation parameters.

2. Confirm the image properties by clicking **Ok**.

3. Set the segmentation model settings.

1. Choose **YeaZ_v2** as the model that should be used for segmentation, then confirm the selection by clicking **Ok**.

2. Click the **Load saved recipe...** button to load the previously saved recipe.

3. Select **segmentation_recipe_test.ini** from the list, then confirm selection by clicking **Ok**.

4. Dismiss the successful loading message by pressing **Ok**.

5. Confirm the parameters by selecting **Ok**.

4. Confirm other segmentation settings.

1. Press **Ok** to accept the default name for the segmentation file.

2. Select **No** to confirm that the entire image should be segmented.
3. Select **Ok** to set the stop frame as the final frame of the timelapse.
5. Set the tracking settings. Select **YeaZ** as the tracking method to be used, then confirm the selection by pressing **Ok**.
6. Run the segmentation and tracking routine.
 1. Click **Run now** to run the segmentation and tracking pipeline.

NOTE: This can take several hours based on image size, model complexity, and computer specifications. In test runs, the segmentation of the test data with **YeaZ_v2** took roughly 2 min on a device without a GPU.
 2. Click **Ok** to complete the segmentation.
3. Select a channel for visualization. Use the dropdown menu to select the channel **phase_contr_preprocessed**, then select **Ok** to confirm.
4. Select segmentation mask name. Select **Load selected** to load the segmentation file created in the previous step.
5. Finish the data loading process.
 1. Confirm the image properties by clicking **Ok for loaded Positions**.
 2. Select **No** to prevent loading of additional fluorescence data.
6. Use the **mode selector** (Figure 6, "Mode selector") to select **Segmentation and Tracking** mode.

NOTE: Use the checkboxes at the bottom of the GUI to change the displayed annotations. The left and right images can display different annotations.

7. Correcting segmentation and tracking errors

NOTE: In general, cells missing in the current frame compared to the previous one are indicated by a yellow contour and ID. Newly detected cells appear with a red ID and a thick contour. Cells that have been accepted as lost are shown with a green contour and ID.

1. Click on **Launch GUI...** in the main window (Figure 1A, "Visualise and correct" module).
2. Load the sample data.
 1. Click the **folder** icon in the toolbar of the new window.
 2. Select the folder containing the data, then press **Select Folder** to confirm selection.
7. Correct the segmentation and tracking.
 1. Use the **left and right arrow keys** to navigate between frames.
 2. Navigate to frame **10**.
 3. Press the key **S** to activate the manual bud separation tool.
 4. Use **right-click** to automatically split the segmentation mask of cell 1.
 5. Navigate to frame **14**.
 6. Press the key **B** to activate the brush.

7. Draw the missing segmentation mask for the bud using the **left mouse button**.
8. Go through the subsequent frames while correcting segmentation and tracking errors. Use the tools available (**Figure 6**, "Edit toolbar"). Correct at least until frame 42.

NOTE: Most tools have a tooltip explaining their functionality. Hover over a tool to display the tooltip.

8. Cell cycle annotations

NOTE: Only move to this step after finishing the segmentation and tracking corrections for all frames of interest. Use the correct annotation mode depending on the cell division type (symmetric, e.g., mammalian cells, or asymmetric, e.g., budding yeast). For the sample data, refer to "Asymmetrically dividing cells", step 8.1. Step 8.2 describes the general workflow for symmetrical dividing cells.

1. Asymmetrically dividing cells

NOTE: For organisms like budding yeast, buds can be assigned to mothers. Both objects need to be present in the same frame. By default, the expected cell cycle stage based on the annotations for budding yeast is shown. The cell cycle stage of buds is displayed in red, while that of all other objects is displayed in white. Mothers are connected to their buds with a dashed, yellow line.

1. Activate **Cell cycle analysis** using the **mode selector** (**Figure 6**, "Mode selector").
2. Select **Yes, go to frame 1** when prompted to.
3. Use the **left and right arrow keys** to navigate between frames.

4. Navigate to frame **41**. Click **Ok** to accept the initialization of the Cell Cycle Annotation table when prompted to.
5. Right-click on **Cell 1** or its bud to separate the connection and annotate the cell division event.
6. Continue until all relevant frames have been viewed. Correct mistakes in the automatic mother bud assignments using the available tools (**Figure 6**, "Edit toolbar").
7. Available tools

NOTE: These tools, except the **Break/Rebind Mother-Bud Association** tool, can be activated using a customizable shortcut, or by pressing the associated buttons in the toolbar.

1. Assign Bud to Mother (A): Activate the **Assign Bud to Mother** tool. Press and hold the **right mouse button** on the bud. Drag to the corresponding mother cell and release the mouse button.

NOTE: Use this tool when the automatic assignment of buds to mothers is incorrect.

2. Annotate Unknown History (U): Activate the **Annotate Unknown History** tool. Use the **right mouse button** to click on the cell that should be annotated as having an unknown history.

NOTE: Use this tool to manually annotate cells with an unknown history, helping to correct lineage ambiguities where automatic inference is insufficient.

3. Reinitialize cell cycle annotation

NOTE: Use this option to re-run the cell cycle annotation algorithm from the current frame onwards. This ensures consistency with recent

corrections or updates made to segmentation/tracking data.

4. Break/rebind mother-bud association: Make sure that no other tool is selected. Right-click on an existing mother-bud pair to break the association, or Right-click again to re-establish the connection.

2. Symmetrical dividing cells

NOTE: This feature is under beta testing and will be officially released soon. It is possible to assign two or more daughter cells to a single mother cell. Potential mother cells are those present in the previous frame but absent in the current one, while potential daughter cells are newly appearing cells in the current frame.

1. Activate **Normal division: Lineage tree** using the mode selector (**Figure 6**, "Mode selector").
2. Select **Yes, go to frame 1** when prompted to.
3. Use the **left and right arrow keys** to navigate between frames.
4. Correct mistakes in the automatic mother-daughter assignments using the available tools (**Figure 6**, "Edit toolbar").
5. Propagate changes when prompted to by clicking **Propagate**.
6. Available tools

NOTE: These tools can be activated using the shortcut or the respective buttons in the toolbar.

1. Find the mother for a new Cell ID (F): Activate the **Find Mother for a New Cell ID** tool. Right-click on the new cell to cycle through candidate mother cells. **Shift + Right-click** to cycle backwards through the candidates.

NOTE: Use this tool to assign or modify the mother of a daughter cell.

2. Set unknown mother (U): Activate the **Set Unknown Mother** tool. **Right-click** on the cell whose mother is unknown.

NOTE: Use this tool to manually designate a cell's mother as unknown.

9. Saving data

1. Navigate to **File > Save** in the top ribbon.
2. Click **Set measurements...** to select the desired measurements.
3. Check the checkbox next to **mCitrine metrics**, or any other measurements desired.
4. Click **Ok** to confirm.
5. Click **Yes** to save the measurements.
6. Click **Ok** to confirm the frame up to which measurements should be saved.
7. Click **No** when prompted about data concatenation.

10. Create the data structure

NOTE: This section discusses preparing the microscopy data for segmentation and tracking. This can be skipped when demonstration data is used, which is downloaded in "Downloading sample data". The data structure that will be generated is outlined in **Figure 9**.

1. Place the microscopy file(s) into an empty folder.

NOTE: Microscopy formats such as .czi (Zeiss), .lif (Leica), and .nd2 (Nikon), as well as single .tif and .png files, are supported.

2. Click on the module **0. Create data structure...** in the main window (**Figure 1A**, "Create data structure" module).
3. Select **BioIO** or **Fiji Macro**.
 1. If **Windows** is used, click **Use BioIO**, while for **MacOS and Linux** users, select **Use Fiji Macro**.
NOTE: In the following, it is assumed that Windows is used.
 2. If prompted to install BioIO, press **Ok** to install.
 3. Select the microscopy file arrangement. Select the option that matches the microscopy files arrangement using the dropdown menu, then press **Ok** to confirm.
4. Select source and destination folders and data loading strategy
 1. Press **Done** to confirm the files are in an otherwise empty folder.
 2. Use Cell-ACDC's folder selector to choose the folder containing the file(s).
 3. Press **Select Folder** to choose a folder.
 4. Press **Select Folder** again to choose the same folder as the destination folder.
NOTE: The raw microscopy file is not deleted.
 5. Select **Yes, load entire position at once**.
5. If prompted to install a BioIO subpackage, press **Ok** to install.
6. Set the metadata for the input file.
 1. Correct the metadata.
NOTE: Double-check the "Order of dimensions" by clicking the **little eye icon** next to the **Channel name**

fields. Other metadata that could not be loaded from the raw file is highlighted.

2. Press **Ok** to confirm the metadata.
3. Press **Yes** to confirm the order of dimensions.
7. Wait for the process to finish.
8. Press **Yes** to close the window when the process is finished.
NOTE: Creating the data structure can take hours depending on the data size.

11. Data preprocessing utilities

NOTE: Several options for processing images before moving on to analysis are available in the main window. To access these tools, go to the **Utilities** dropdown menu in the menu bar of the main window (**Figure 1B**, "Utilities"), hover over **Image preprocessing**, and then select the option that should be used. Two examples are:

1. Combine channels: Use this to merge two or more image channels.
NOTE: For example, two fluorescence channels can be averaged.
2. Resize images: Use this to reduce the size of very large image datasets.
NOTE: This can significantly speed up processing time and, in many cases, has little or no impact on segmentation quality.

12. Troubleshooting

1. If Cell-ACDC crashes, create a bug report on the Cell-ACDC GitHub page by navigating to the **Issues** tab and clicking the green **New issue** button. Follow the template provided to include all relevant details needed

to diagnose and resolve the issue. Alternatively, feel free to seek out help in the **image.sc** forum using the tag **#cell-acdc** or contact one of the corresponding authors. In many cases, especially after installing a package, simply restarting the software may resolve the problem.

2. If Cell-ACDC freezes or becomes unresponsive, check the console for progress updates. Long segmentation times, especially when using computationally intensive models, such as Cellpose version 4, or processing large datasets, are normal. If Cell-ACDC remains unresponsive, refer to step 12.1 for further instructions.
3. If automatic segmentation incorrectly includes background, check whether a preprocessing step may have overly smoothed the background. Many segmentation models are trained on raw (non-pre-processed) images and may perform poorly when the background lacks sufficient texture.

Representative Results

Nuclear volume quantification in tumor spheroids in 3D

Automated 3D microscopy (z-stacks) enables scientists to visualize complex multicellular systems such as organoids. To quantify morphological and fluorescence signal properties at the single-cell level, cell segmentation in 3D is often required. The following example demonstrates how Cell-ACDC can segment the nuclei in organoids containing thousands of cells from the nuclear staining channel and quantify their volume (data from Ref.⁹). Segmentation was performed using a custom Cellpose model that was trained and published in Ref.⁹. The trained Cellpose model can be used directly in Cell-ACDC by providing the weights file path in the GUI when selecting the model parameters for Cellpose v2. The segmentation was performed using the second

module of Cell-ACDC to batch-process all images (**Figure 1A**, "Segment and track" module). Next, the result was visualized in the third module GUI (**Figure 1A**, "Visualize and correct" module). This module has been optimized to visualize thousands of single objects with multiple annotation options (e.g., contours, overlaid segmentation masks, or text IDs). After the measurements from the 3D masks were computed, all the available measurements were documented directly in the GUI. They are accessible by going to the top menu bar (**Figure 6**, "Menu bar"), selecting the **Measurements** menu, and then **Set measurements....** The pop-up dialogue lets users obtain measurement-specific information from the info buttons and select which measurements to save. For the representative results in **Figure 10**, "cell_vol_fl_3D" was used, which is the volume of each object calculated by multiplying the total voxels in the object by the pixel size squared and the voxel depth. These properties are either automatically extracted from the microscopy raw file or provided by the user. Computation of the measurements from this GUI requires loading the raw images. Thus, to streamline the process and enable batch processing, the measurements can also be computed from the **Utilities** menu (**Figure 1B**, "Utilities"), by going to the sub-menu **Measurements** and then **Compute measurements for one or more experiments**. Finally, the nuclear volume distribution was plotted as a representative result (**Figure 10**). This analysis reveals a significant fraction of small nuclei, likely due to segmentation artifacts. They could easily be removed by filtering out small objects from the segmentation mask. At the same time, the very large nuclei could be due to merged nuclei during segmentation. It is always recommended to plot the volume distribution of objects (e.g., single cells) to identify artifacts and extract additional biological information about cell size.

Quantification of time-lapse microscopy data

With time-lapse microscopy, cellular dynamics can be directly observed at the single-cell level. Besides cell segmentation, extracting temporal dynamics requires additional analysis, including cell tracking and cell pedigree annotation. Due to the interdependence of these analysis stages, errors introduced early in the pipeline can propagate to later steps. As a result, ongoing visualization and correction of segmentation, tracking, and annotation errors are necessary. The following demonstrates that Cell-ACDC is suitable for tackling these tasks. Two datasets of two different model organisms were selected: 1) budding yeast (strain DCY001-1 from Ref.³⁵, where the two histone H2B proteins, Htb1 and Htb2, are tagged with mCitrine), and 2) mouse embryonic stem cells (mESCs, data from Ref.²²).

These two datasets highlight two annotation modes available in Cell-ACDC: asymmetric and symmetric (i.e., symmetric cytokinesis, or "normal") cell division. As the mode of division differs, the two organisms require a different tracking and annotation framework.

In asymmetric division, the mother cell forms a bud that grows and eventually separates to become a daughter cell. After division, the mother cell retains its original cell ID, and its generation number increases by one, while the daughter cell is assigned a new cell ID and its generation number is set to one. The budding phase corresponds to the S/G2/M phases of the cell cycle and is annotated as such in Cell-ACDC. These annotation choices help address typical biological questions related to the cell cycle in budding yeast.

In the case of "symmetric" division (e.g., mammalian cells), the mother cell divides into two daughter cells. The mother cell, i.e., its ID, disappears upon division, and the two

daughter cells receive new IDs. Additionally, the generation number of the daughter cells is increased by one relative to the mother cell. Cell-ACDC also keeps track of the parent ID, the root ID (the original ancestor cell at the beginning of a lineage), and the sister ID.

For both annotation modes, an innovative framework was developed to correct annotation errors, where the correction is automatically propagated to all past and future relevant time points. The visualization, annotation, and correction were performed in the third module GUI (**Figure 1A**, "Visualize and correct" module and **Figure 6**). To segment and track the cells in dataset 1, the model **YeaZ_v2**²⁶ was applied to the phase contrast channel, while for the nuclear (histone) channel, the **StarDist**²⁵ model was used. Then, using the utility **Tracking and lineage > Track and/or count sub-cellular objects** (**Figure 1B**, **Utilities** menu bar), Cell-ACDC assigned each nucleus the Cell ID of its corresponding cell, ensuring consistency between the tables generated from cell and nucleus masks.

For dataset 2, the **DeepSea**²² segmentation model was used. All three models are already available in Cell-ACDC, showcasing the advantage of integrating several segmentation models into the software.

Once segmentation and tracking errors were corrected, cell pedigrees were annotated, numerical features were computed, and downstream analysis was carried out. For dataset 1, the column **TaYFP_amount_autoBkgr** and the number of nuclei segmented (**Figure 11A**) were plotted against time. "TaYFP" is the name of the nuclear channel. "amount_autoBkgr" is a proxy for the total cellular protein amount extracted from epifluorescence images³⁶. It is calculated as the difference between the mean fluorescence intensity in each cell mask and the background median,

multiplied by the cell area (in pixels). Here, the background median is calculated from all the pixels that are not segmented as cells. As expected, the H2B amount starts increasing at bud emergence (**Figure 11A-ii**) and reaches a constant value before nuclear division (**Figure 11A-iii**). This is an important quality control in histone protein homeostasis, as the amount of histone proteins is expected to be cell cycle dependent. Additionally, plotting the number of nuclei over time confirms that histone protein amounts reach the maximum roughly around nuclear division.

For dataset 2, the cell area over time for a selected cell undergoing cell division was plotted. As expected, the cell area increases until reaching a maximum value (**Figure 11B-i**). Then, it decreases until cell division (**Figure 11B-ii**) as the cell contracts. Finally, the cycle restarts for the two daughter cells. This is another recommended analysis since checking cell size changes over the cell cycle is essential to confirm that cells are growing and dividing as expected (or not in the case of specific mutants).

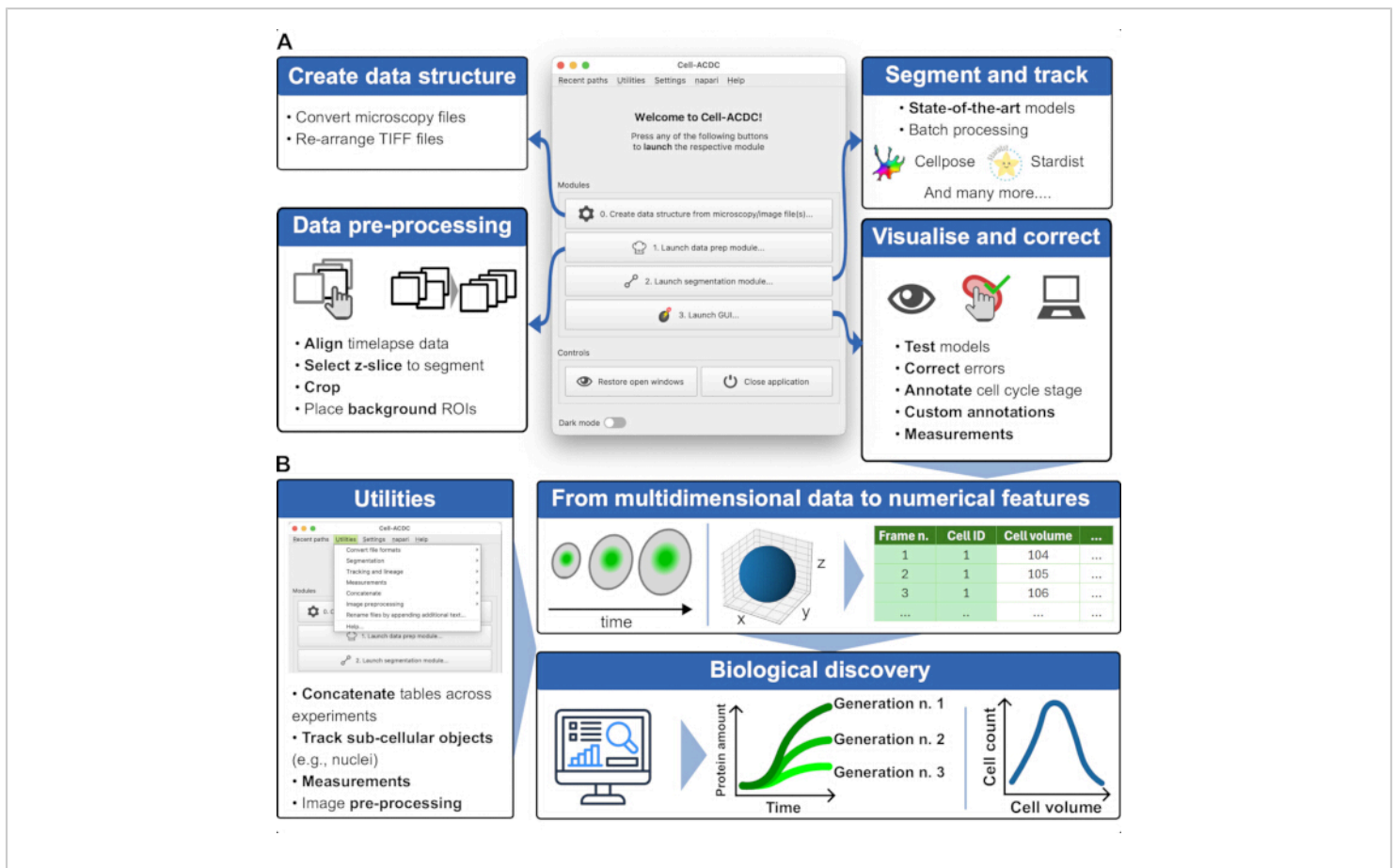


Figure 1: Cell-ACDC modules. (A) Overview of the 4 main modules that can be launched from the main Cell-ACDC launcher. After segmenting, tracking, and annotating microscopy data, numerical features can be computed from either the third module ("Visualise and correct"), or from (B) the **Utilities** menu on the top menu bar. The "Utilities" are the routines that can run automatically on multiple datasets without user input. Besides computing the measurements, other utilities include

the concatenation of multiple output tables into a single table, tracking sub-cellular objects, and image pre-processing. Cell-ACDC supports 2D, 3D (z-stack or time-lapse), and 4D data (z-stacks over time), with any number of additional channels. The output table with the numerical features can then be used for downstream analysis and biological discovery (**Figure 10** and **Figure 11**). To this purpose, Jupyter notebooks on the Cell-ACDC GitHub page are available, which include examples of plots that can be obtained from the output table. [Please click here to view a larger version of this figure.](#)

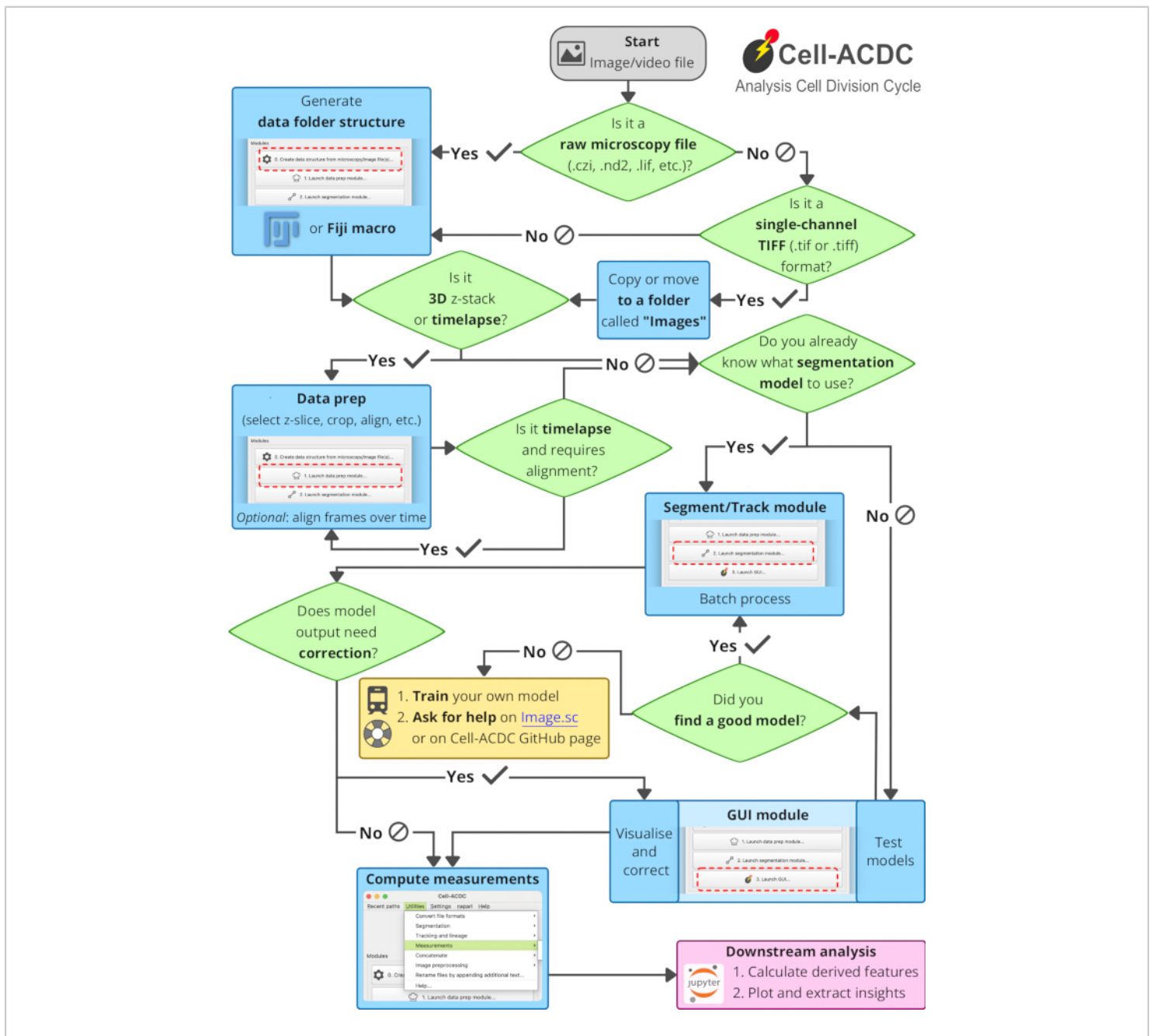


Figure 2: Cell-ACDC decision flowchart. A flowchart outlining the module to use depending on the dataset type and analysis requirements. [Please click here to view a larger version of this figure.](#)

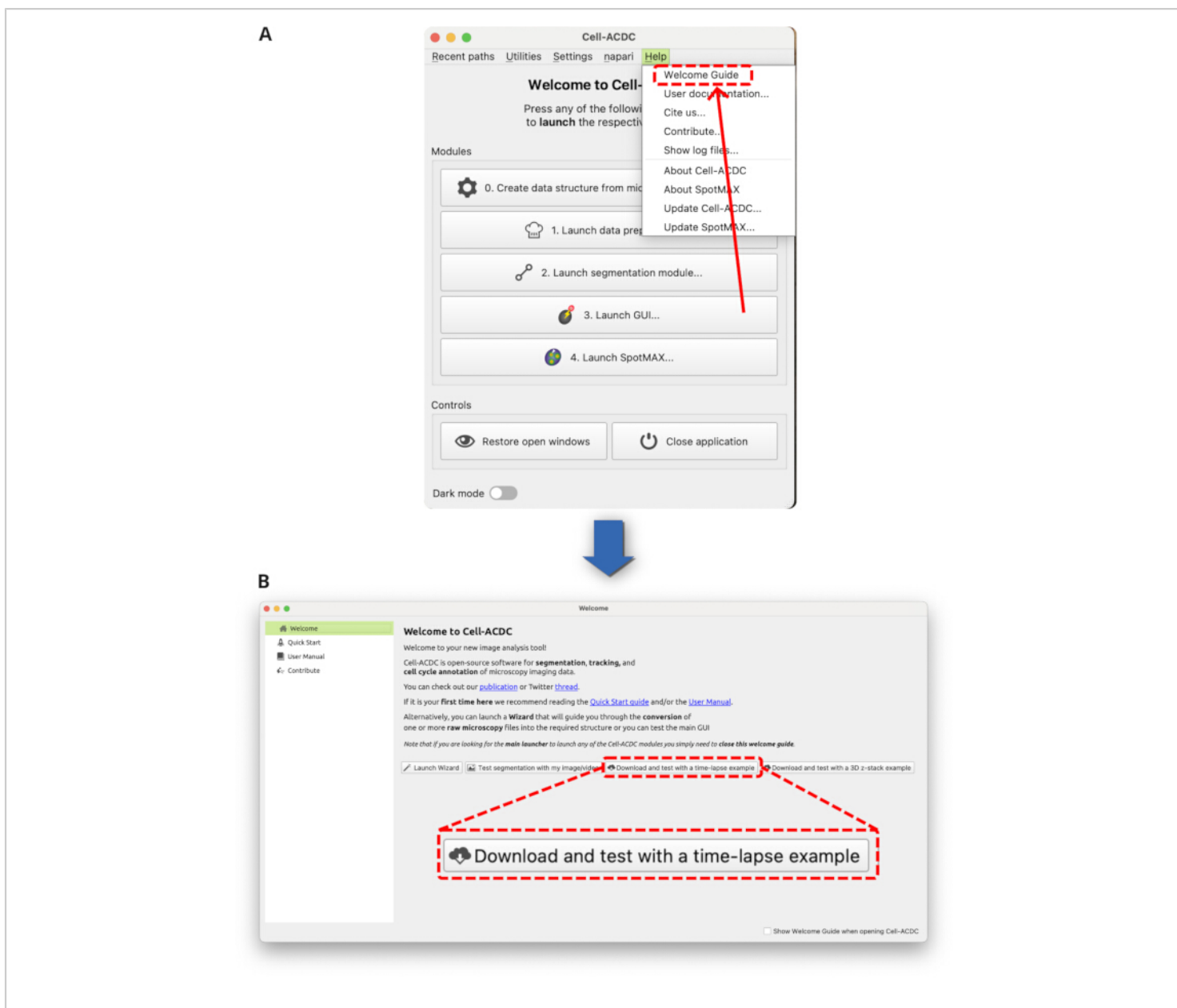


Figure 3: Download example data. (A) Open Welcome Guide. **(B)** Download example data required to replicate the protocol. [Please click here to view a larger version of this figure.](#)

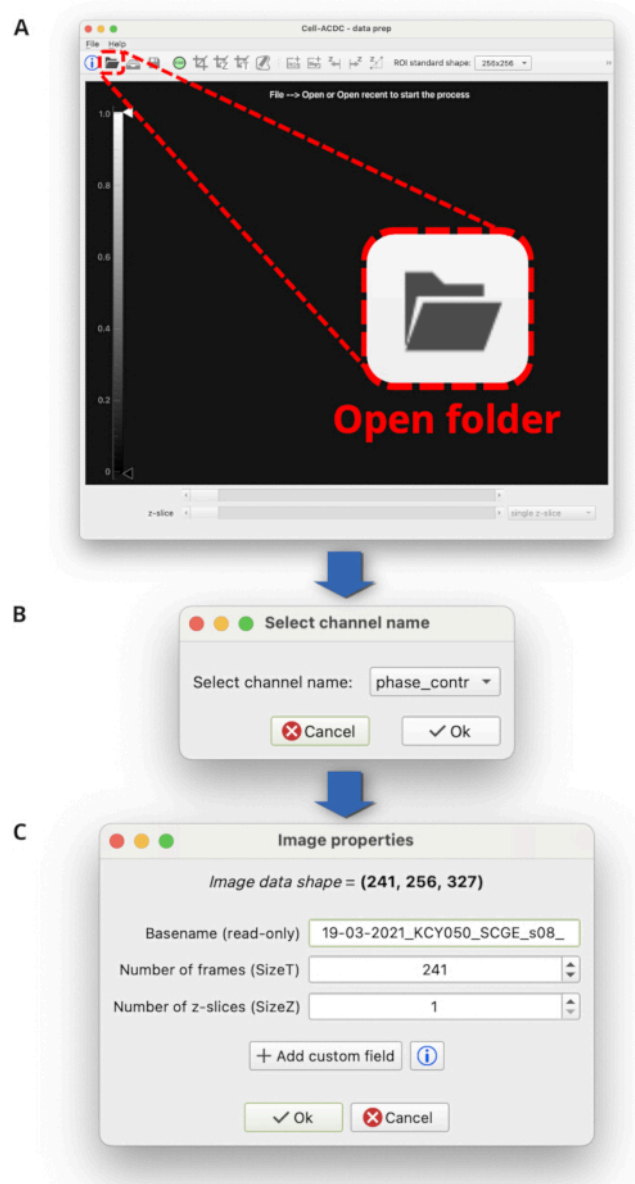


Figure 4: Load data for data prep. (A) Load data into the data prep GUI. (B) Select channel to load. (C) Edit and confirm image metadata. [Please click here to view a larger version of this figure.](#)

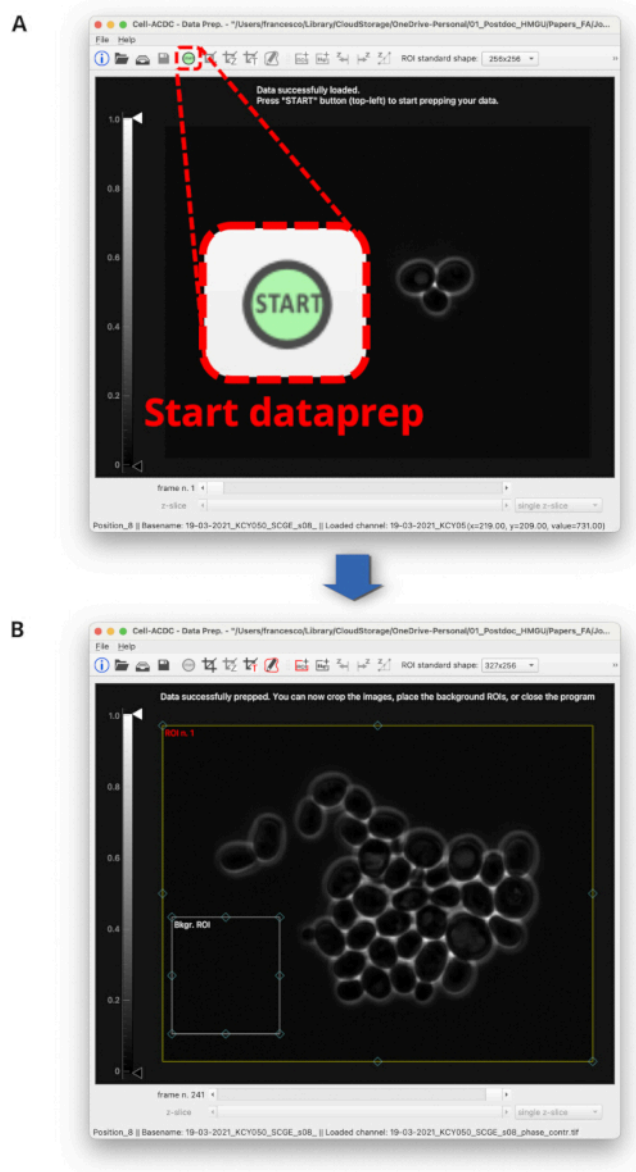


Figure 5: Run data prep process. (A) Start the process. **(B)** Position ROIs (for cropping) and background ROIs. [Please click here to view a larger version of this figure.](#)

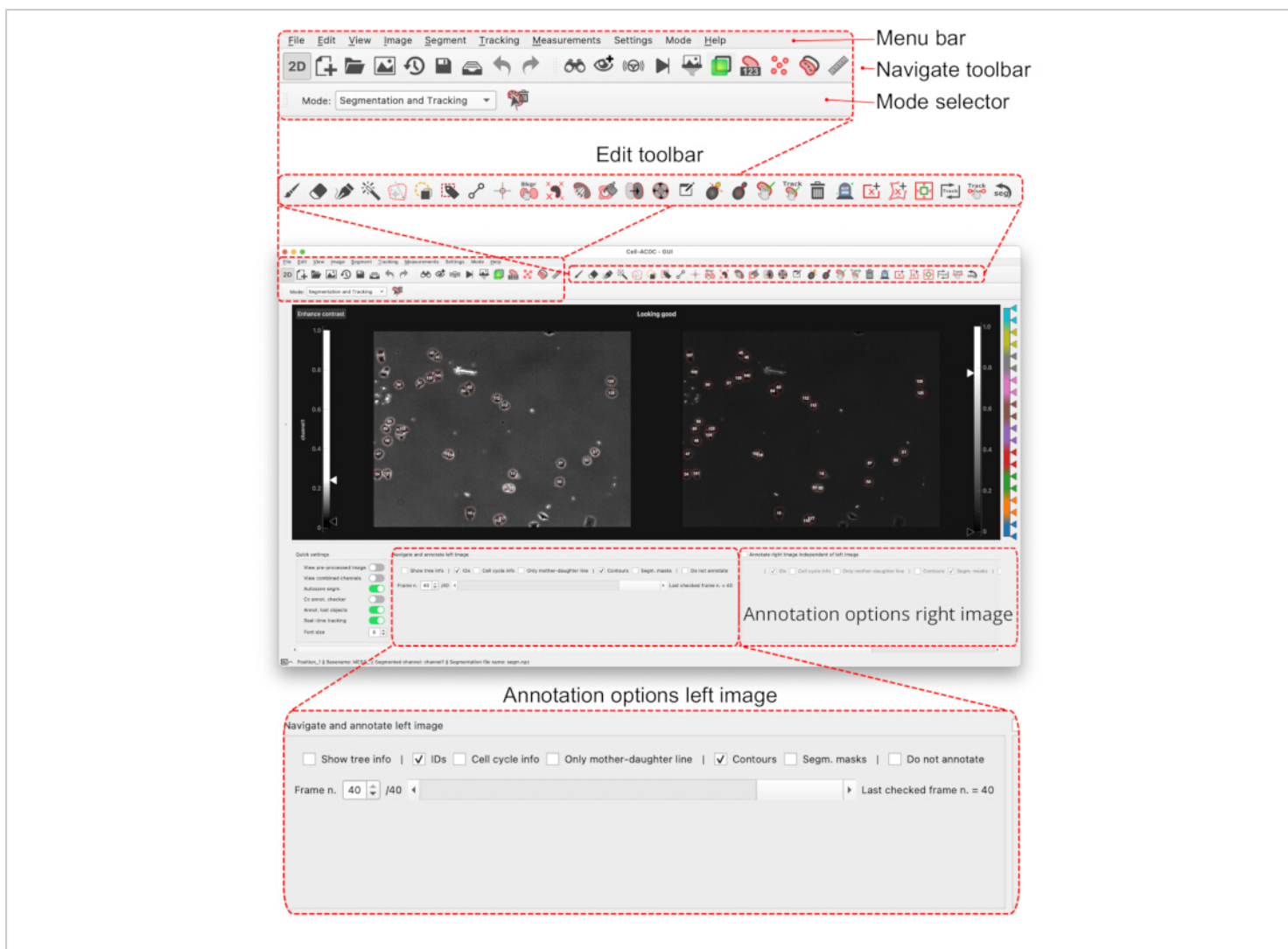


Figure 6: Third module GUI to visualize and correct results. Screenshot of the third module GUI ("Visualise and correct" in **Figure 1A**) with highlighted items. Note that most of the buttons on the toolbars have a tooltip (accessible by hovering the mouse cursor over the button) explaining how to use that specific function. The mode selector can be used to switch between 5 modes: "Viewer", "Segmentation and Tracking", "Cell cycle analysis" (for asymmetrically dividing cells), "Normal division: Lineage tree" (for symmetrically dividing cells, e.g., mammalian cells), and "Custom annotations". Note that a toolbar or menu bar is often present in the other GUIs (e.g., "Data pre-processing" module, **Figure 1**). The **Edit toolbar** contains all the functions that can be used to edit and correct segmentation and tracking errors (e.g., brush, eraser, edit ID, etc.). The loaded image is displayed in a two-panel view, which is helpful when different annotation options are needed (e.g., cell cycle info on the left image and IDs on the right image). The right image can also be toggled off (right-click on the image and de-select **Show mirrored image**). Each image panel includes a LUT slider on the side to quickly adjust intensity levels. By right-clicking on the LUT control, the user can select different color maps for the intensity images. Additionally, on the right-hand side, there is a LUT selector for the color of the segmentation labels to overlay on the intensity images (annotation option

called **Segm. masks**). On the left-side of the annotation options for the left image, there are additional toggles to control some settings, such as auto-save, font size, etc. [Please click here to view a larger version of this figure.](#)

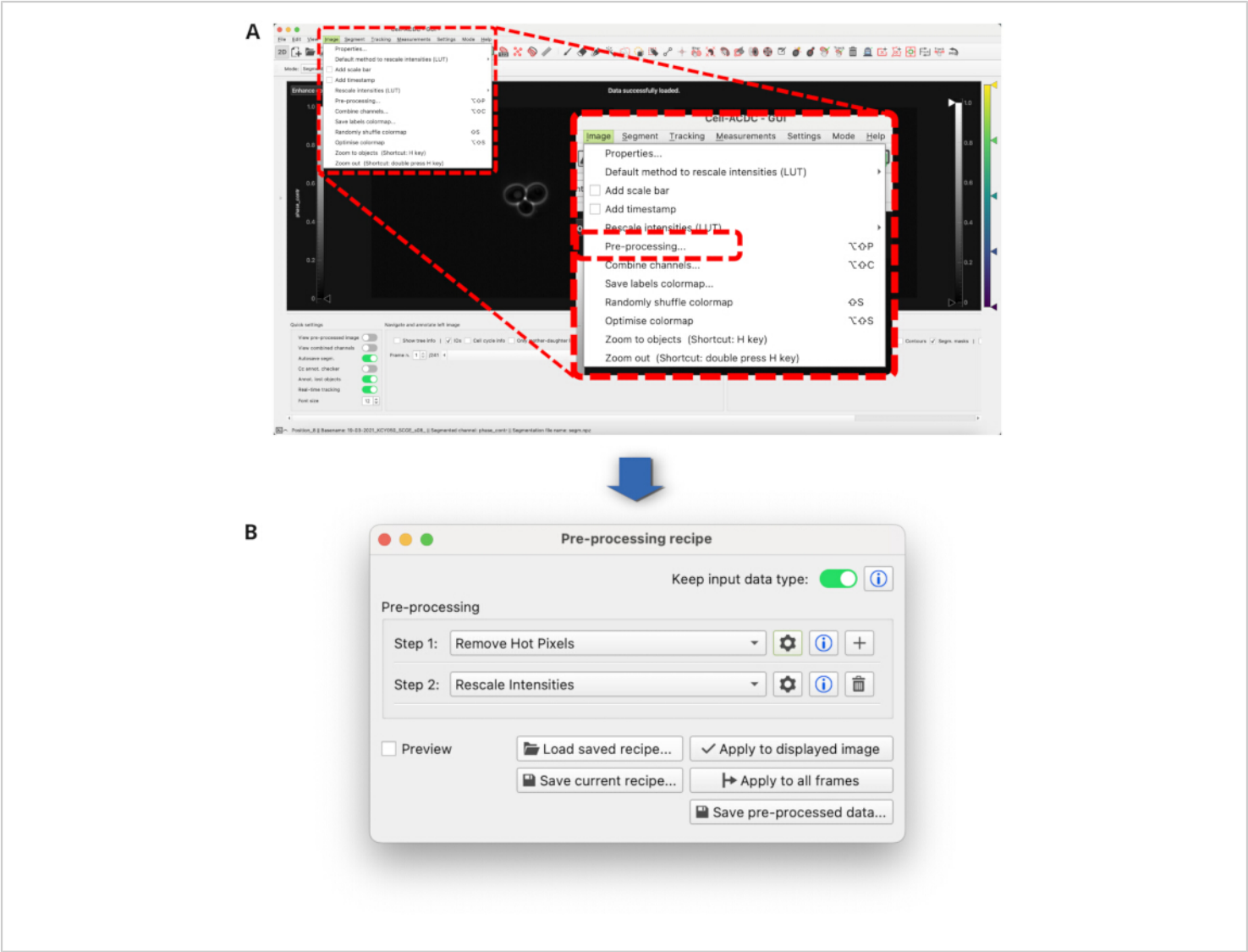


Figure 7: Visualize pre-processing in the main GUI. (A) Open pre-processing dialogue. **(B)** Pre-processing dialogue with parameters used in Protocol initialized. [Please click here to view a larger version of this figure.](#)

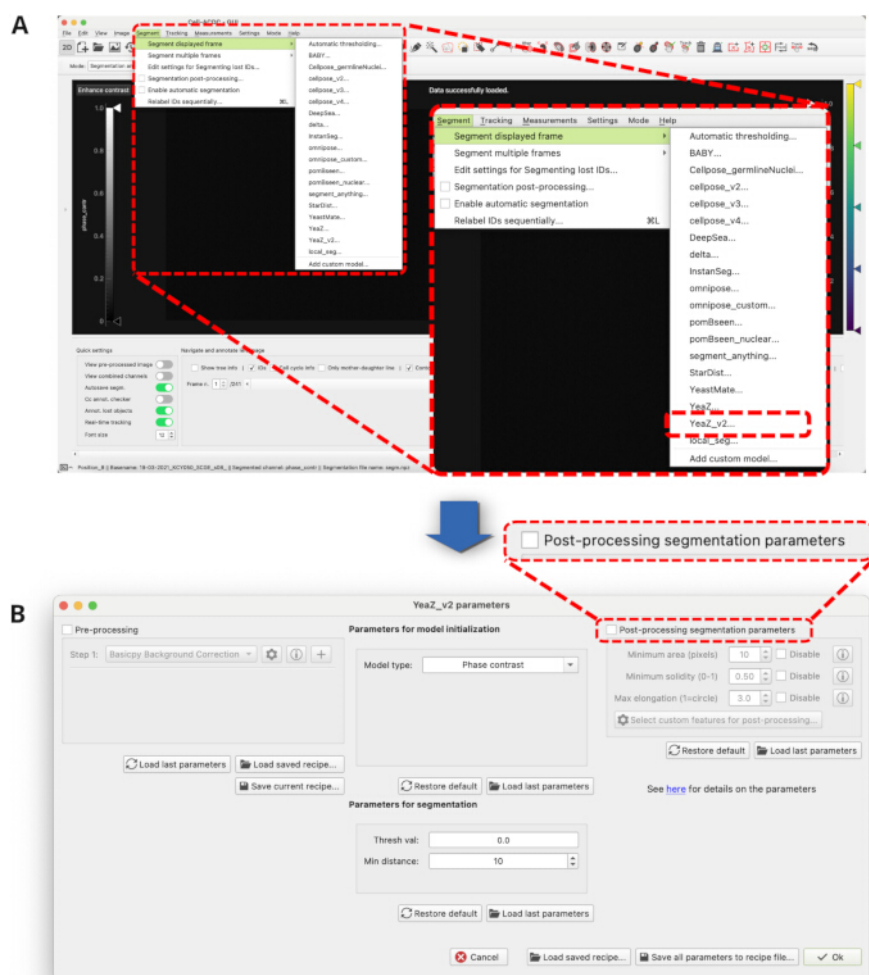


Figure 8: Visualize segmentation output in the main GUI. (A) Select a segmentation model. (B) Set up segmentation parameters. [Please click here to view a larger version of this figure.](#)


```

Experiment folder
├── Position_1
│   └── Images
│       ├── basename_s01_metadataXML.txt
│       ├── basename_s01_metadata.csv
│       ├── basename_s01_ChannelName1.tif
│       ├── basename_s01_ChannelName2.tif
│       └── ...
├── Position_2
│   └── Images
│       ├── basename_s01_metadataXML.txt
│       └── ...
└── Position_n
    └── Images
        ├── basename_s0n_metadataXML.txt
        └── ...
    
```

Figure 9: Folder structure required by Cell-ACDC. To work with Cell-ACDC, the data must be arranged in a specific folder structure. While Cell-ACDC provides a module to automatically generate this structure, it is important to understand what this structure should look like. First, all files must be inside a folder called **Images**. Next, they all need to start with the same name, the so-called "basename_". The minimum set of required files is a single-channel TIFF file (2D, 3D z-stack, or 3D+time) and a CSV file ending with "_metadata.csv". This file must be a table with two columns, the first column called **Description** and the second column called **values**, and it should contain at least entries for **SizeT**, and **SizeZ** for the number of frames and z-slices, respectively. If an image file does not have z-slices, **SizeZ** must be set to 1. The same is true for no time-lapse images, where **SizeT** must be 1. Being a CSV file, an entry is a single line of **Description,value**, separated by a comma, e.g., **SizeZ,1**. For multiple channels, one TIFF file per channel must be generated. The folder **Images** must then be placed inside a folder called "Position_1". Multiple positions are allowed, and they need to be named with a consecutive number. When loading data into any of the Cell-ACDC modules, the user can either select a specific Position folder or the entire experiment folder. [Please click here to view a larger version of this figure.](#)

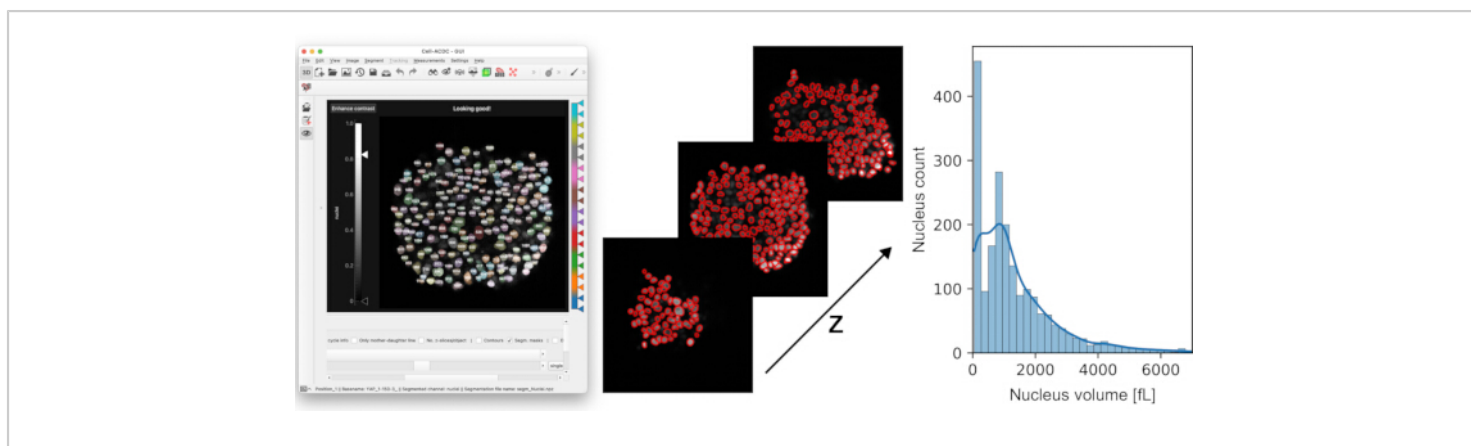


Figure 10: 3D quantification of tumor organoids. Screenshot of a representative tumor organoid (data from⁹) loaded into the third module GUI of Cell-ACDC (left), example z-slices with red contours highlighting the segmentation masks (center), and histogram of the cell volume distribution computed from the 3D segmentation masks (right). [Please click here to view a larger version of this figure.](#)

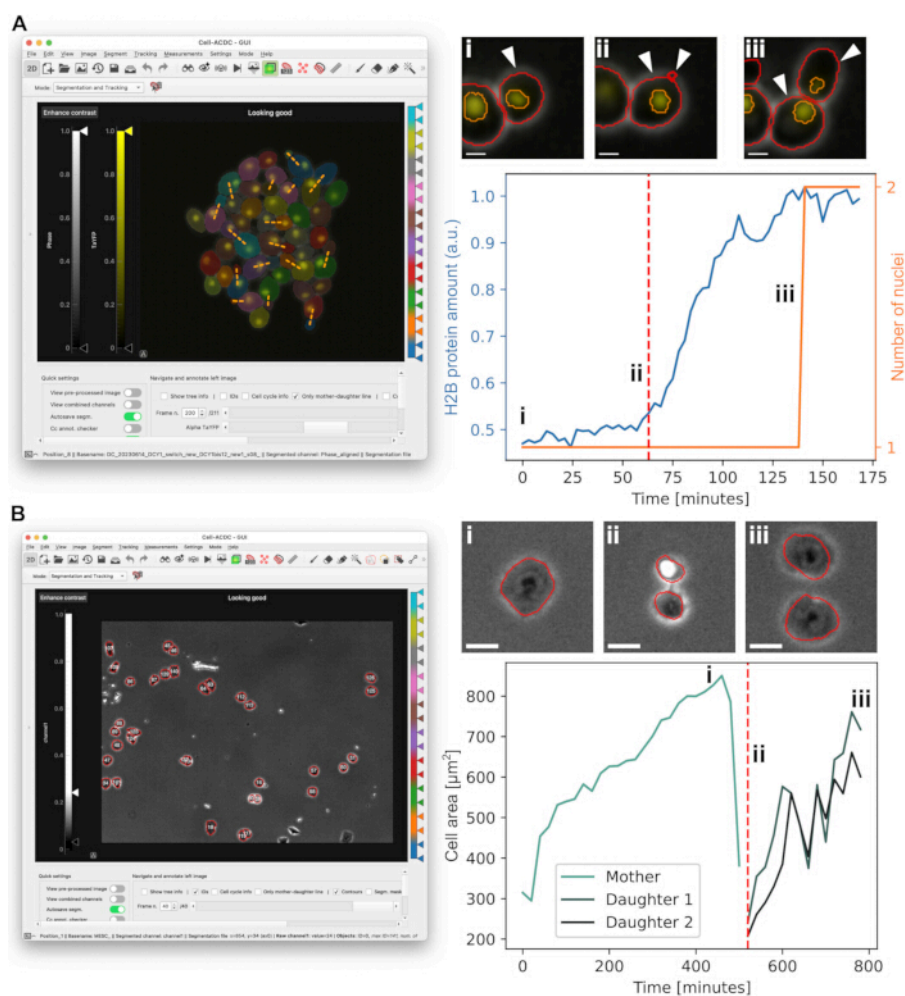


Figure 11: Quantification of time-lapse microscopy data. (A) Screenshot of time-lapse microscopy data of budding yeast cells loaded into the third module GUI of Cell-ACDC (left), and histone H2B protein amount quantification over time in a representative cell cycle (right). The zoomed images show the example cell and its bud (white arrows) at the start of the cell cycle (i), at bud emergence (ii), and at nuclear division (iii). Data is taken from Chatzitheodoridou et al.³⁵ **(B)** Screenshot of time-lapse microscopy data of mouse embryonic stem cells loaded into the third module GUI of Cell-ACDC (left) and cell area (μm^2) plotted as a function of time of a representative cell undergoing cell division. The zoomed images show the example cell and its daughters at maximum cell area before division (i), division into two daughter cells (ii), and the last analyzed frame after division (iii). Data is taken from Zargari et al.^{22,33}. [Please click here to view a larger version of this figure.](#)

Discussion

The analysis of multi-dimensional microscopy data often requires the use of cutting-edge AI-driven models. However,

these tools are developed quickly and have a significant entry barrier to adoption. Additionally, biologists often need visualization of the result for effective error correction. Here,

it is demonstrated how scientists can use Cell-ACDC to address these challenges.

A critical step of the presented protocol is selecting the optimal segmentation model. Cell-ACDC includes a GUI enabling visual selection (**Figure 1A**, "Visualise and correct" module). Tools for preparing data and batch processing of multiple datasets are also provided (**Figure 1A**, Create data structure, Data pre-processing, Segment and Track modules, and Utilities). Users are encouraged to report issues and provide feedback on the **image.sc** forum (using the tag #cell-acdc) or by opening an issue on the Cell-ACDC GitHub page.

While Cell-ACDC has already been used on relatively large data, such as with spheroids in images with thousands of nuclei⁹, or time-lapse data of budding yeast with hundreds of cells per frame, the program must load the entire single-channel data into RAM to function correctly. Approximately three times the image file size in available memory is recommended to ensure stable performance. At present, datasets exceeding available system memory cannot be processed, but support for out-of-core (lazy) loading is planned through the integration of OME-Zarr³⁷.

Currently, one of the main limitations of Cell-ACDC is the partial support for 3D+time datasets, as most of the tools have been developed for either 3D z-stack or 2D+time data. Hence, future versions of Cell-ACDC will extend its capabilities with full support for 3D+time data. Additionally, we are working on extending the cell pedigree annotation framework for "symmetrically" dividing cells (e.g., mammalian cells), which are those cells where the mother cell divides into two daughter cells (opposed to budding yeast, where the mother cell, once per cell cycle, gives rise to a single daughter cell through budding). Finally, as of now, Cell-ACDC is limited to data

whose single-channel data fits entirely into RAM. Therefore, we plan to implement lazy loading as mentioned above.

Since its release, Cell-ACDC has been constantly improved, for example, by adding new segmentation and tracking models, new annotation frameworks (e.g., for mammalian cells, currently under beta-testing), and various performance improvements. Thanks to these developments, scientists could employ Cell-ACDC to accelerate research and enable scientific discovery^{6,9,16,35,38,39,40,41,42,43,44,45,46,47,48,49}.

What makes this software framework unique compared to existing methods^{14,27,50,51} is its ability to leverage and complement existing models, therefore benefiting from the developments of the community. Continued development of Cell-ACDC is actively maintained to establish it as a reference framework for analyzing multidimensional microscopy data.

Disclosures

The authors declare that they have no conflict of interest.

Acknowledgments

We thank Mario Vitacolonna and Rudolf Rüdiger for sharing the spheroids data in Fig. 10, and colleagues at the Institute of Functional Epigenetics, Pascal Falter-Braun and Carsten Marr, for valuable discussions. Special thanks to the several users who provided invaluable feedback, tested new features, and patiently reported issues. This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through project 416098229, the Helmholtz Association, and the joint research school Munich School for Data Science.

References

1. Cuny, A. P., Schlottmann, F. P., Ewald, J. C., Pelet, S., Schmoller, K. M. Live cell microscopy: from image to insight. *Biophys Rev.* **3**, 021302 (2022).
2. Hugelier, S., Colosi, P. L., Lakadamyali, M. Quantitative single-molecule localization microscopy. *Annu Rev Biophys.* **52**, 139-160 (2023).
3. Tavakoli, M. R. et al. Light-microscopy-based connectomic reconstruction of mammalian brain tissue. *Nature.* **642**, 398-410 (2025).
4. Seal, S. et al. Cell painting: a decade of discovery and innovation in cellular imaging. *Nat Methods.* **22**, 254-268 (2025).
5. Robertson, M. J., Meyerowitz, J. G., Skiniotis, G. Drug discovery in the era of cryo-electron microscopy. *Trends Biochem Sci.* **47**, 124-135 (2022).
6. Al-Refaie, N. et al. Fasting shapes chromatin architecture through an mTOR/RNA Pol I axis. *Nat Cell Biol.* **26**, 1903-1917 (2024).
7. Ueda, H. R. et al. Tissue clearing and its applications in neuroscience. *Nat Rev Neurosci.* **21**, 61-79 (2020).
8. Ko, J., Hyung, S., Cheong, S., Chung, Y., Li Jeon, N. Revealing the clinical potential of high-resolution organoids. *Adv Drug Deliv Rev.* **207**, 115202 (2024).
9. Vitacolonna, M. et al. A multiparametric analysis including single-cell and subcellular feature assessment reveals differential behavior of spheroid cultures on distinct ultra-low attachment plate types. *Front Bioeng Biotechnol.* **12**, 1422235 (2024).
10. Qian, N., Weinstein, J. A. Spatial transcriptomic imaging of an intact organism using volumetric DNA microscopy. *Nat Biotechnol.* (2025).
11. Wallace, C. T., St. Croix, C. M., Watkins, S. C. Data management and archiving in a large microscopy-and-imaging, multi-user facility: microscopy and data management. *Mol Reprod Dev.* **82**, 630-634 (2015).
12. Schindelin, J. et al. Fiji: an open-source platform for biological-image analysis. *Nat Methods.* **9**, 676-682 (2012).
13. Bankhead, P. et al. QuPath: open source software for digital pathology image analysis. *Sci Rep.* **7**, 16878 (2017).
14. Sofroniew, N. et al. napari: a multi-dimensional image viewer for Python. *Zenodo.* (2025).
15. Padovani, F., Mairhörmann, B., Falter-Braun, P., Lengefeld, J., Schmoller, K. M. Segmentation, tracking and cell cycle analysis of live-cell imaging data with Cell-ACDC. *BMC Biol.* **20**, 174 (2022).
16. Piñeiro López, C., Rodrigues Neves, A. R., Čavka, I., Gros, O. J., Köhler, S. Segmentation of *C. elegans* germline nuclei. *microPubl Biol.* (2023).
17. Ohira, M., Rhind, N. pomBseen: an automated pipeline for analysis of fission yeast images. *PLoS One.* **18**, e0291391 (2023).
18. Goldsborough, T. et al. InstanSeg: an embedding-based instance segmentation algorithm optimized for accurate, efficient and portable cell segmentation. *arXiv.* (2024).
19. Pietsch, J. M. et al. Determining growth rates from bright-field images of budding cells through identifying overlaps. *Elife.* **12**, e79812 (2023).
20. Bunk, D. et al. YeastMate: neural network-assisted segmentation of mating and budding events in *Saccharomyces cerevisiae*. *Bioinformatics.* **38**, 2667-2669 (2022).

21. Cutler, K. J. et al. Omnipose: a high-precision morphology-independent solution for bacterial cell segmentation. *Nat Methods*. **19**, 1438-1448 (2022).
22. Zargari, A. et al. DeepSea is an efficient deep-learning model for single-cell segmentation and tracking in time-lapse microscopy. *Cell Rep Methods*. **3**, 100500 (2023).
23. Lugagne, J. B., Lin, H., Dunlop, M. J. DeLTA: automated cell segmentation, tracking, and lineage reconstruction using deep learning. *PLoS Comput Biol*. **16**, e1007673 (2020).
24. Kirillov, A. et al. Segment anything. *arXiv*. (2023).
25. Weigert, M., Schmidt, U. Nuclei instance segmentation and classification in histopathology images with StarDist. *IEEE Int Symp Biomed Imaging Challenges (ISBIC)*. (2022).
26. Dietler, N. et al. A convolutional neural network segments yeast microscopy images with high accuracy. *Nat Commun*. **11**, 5723 (2020).
27. Stringer, C., Pachitariu, M. Cellpose3: one-click image restoration for improved cellular segmentation. *Nat Methods*. **22**, 592-599 (2025).
28. Pachitariu, M., Stringer, C. Cellpose 2.0: how to train your own model. *Nat Methods*. **19**, 1634-1641 (2022).
29. Gallusser, B., Weigert, M. Trackastra: transformer-based cell tracking for live-cell microscopy. *arXiv*. (2024).
30. Allan, D. B., Caswell, T., Keim, N. C., van der Wel, C. M., Verweij, R. W. soft-matter/trackpy: v0.6.4. *Zenodo*. (2024).
31. Ulicna, K., Vallardi, G., Charras, G., Lowe, A. R. Automated deep lineage tree analysis using a Bayesian single cell tracking approach. *Front Comput Sci*. **3**, 734559 (2021).
32. Doersch, C. et al. TAPIR: tracking any point with per-frame initialization and temporal refinement. *arXiv*. (2023).
33. Zargari, A. et al. DeepSea is an efficient deep-learning model for single-cell segmentation and tracking in time-lapse microscopy. *Cell Rep Methods*. **3**, 100500 (2023).
34. O'Connor, O. M., Alnahhas, R. N., Lugagne, J. B., Dunlop, M. J. DeLTA 2.0: a deep learning pipeline for quantifying single-cell spatial and temporal dynamics. *PLoS Comput Biol*. **18**, e1009797 (2022).
35. Chatzitheodoridou, D., Bureik, D., Padovani, F., Nadimpalli, K. V., Schmoller, K. M. Decoupled transcript and protein concentrations ensure histone homeostasis in different nutrients. *EMBO J*. **43**, 5141-5168 (2024).
36. Schmoller, K. M., Turner, J. J., Kõivomägi, M., Skotheim, J. M. Dilution of the cell cycle inhibitor Whi5 controls budding-yeast cell size. *Nature*. **526**, 268-272 (2015).
37. Moore, J. et al. OME-Zarr: a cloud-optimized bioimaging file format with international community support. *Histochem Cell Biol*. **160**, 223-251 (2023).
38. Seshadri, A., Badrinarayanan, A. Exonuclease action of replicative polymerase gamma drives damage-induced mitochondrial DNA clearance. *EMBO Rep*. **26**, 1385-1405 (2025).
39. Dengler, L. et al. When mitochondria fall apart: unbalanced mitochondrial segregation triggers loss of mtDNA in the absence of mitochondrial fusion. *bioRxiv*. (2025).
40. Xiao, J., Turner, J. J., Kõivomägi, M., Skotheim, J. M. Whi5 hypo- and hyper-phosphorylation dynamics control cell-cycle entry and progression. *Curr Biol*. **34**, 2434-2447.e5 (2024).

41. Roussou, R. et al. Real-time assessment of mitochondrial DNA heteroplasmy dynamics at the single-cell level. *EMBO J.* **43**, 5340-5359 (2024).
42. Padovani, F. et al. SpotMAX: a generalist framework for multi-dimensional automatic spot detection and quantification. *bioRxiv.* (2024).
43. Lanz, M. C. et al. Genome dilution by cell growth drives starvation-like proteome remodeling in mammalian and yeast cells. *Nat Struct Mol Biol.* **31**, 1859-1871 (2024).
44. Kukhtevich, I. et al. The origin of septin ring size control in budding yeast. *bioRxiv.* (2024).
45. Chadha, Y., Kukhtevich, I. V., Padovani, F., Schneider, R., Schmoller, K. M. Single-cell imaging reveals a key role of Bck2 in budding yeast cell size adaptation to nutrient challenges. *bioRxiv.* (2024).
46. Seel, A. et al. Regulation with cell size ensures mitochondrial DNA homeostasis during cell growth. *Nat Struct Mol Biol.* **30**, 1549-1560 (2023).
47. Schuh, L. et al. Altered expression response upon repeated gene repression in single yeast cells. *PLoS Comput Biol.* **18**, e1010640 (2022).
48. Kukhtevich, I. V. et al. Quantitative RNA imaging in single live cells reveals age-dependent asymmetric inheritance. *Cell Rep.* **41** (7), 111656 (2022).
49. Freitag, M. et al. Single-molecule experiments reveal the elbow as an essential folding guide in SMC coiled-coil arms. *Biophys J.* **121**, 4702-4713 (2022).
50. Sugawara, K., Çevrim, Ç., Averof, M. Tracking cell lineages in 3D by incremental deep learning. *Elife.* **11**, e69380 (2022).
51. Ershov, D. et al. TrackMate 7: integrating state-of-the-art segmentation algorithms into tracking pipelines. *Nat Methods.* **19**, 829-832 (2022).