



**HAL**  
open science

# Segmentation and Quantitative Analysis of Epithelial Tissues

Benoit Aigouy, Benjamin Prud'homme

► **To cite this version:**

Benoit Aigouy, Benjamin Prud'homme. Segmentation and Quantitative Analysis of Epithelial Tissues. *Drosophila*, 2540, Springer US, pp.387-399, 2022, *Methods in Molecular Biology*, 10.1007/978-1-0716-2541-5\_20 . hal-03844061

**HAL Id: hal-03844061**

**<https://hal.science/hal-03844061v1>**

Submitted on 8 Nov 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Segmentation and quantitative analysis of epithelial tissues

**Benoit Aigouy<sup>1</sup> and Benjamin Prud'homme<sup>1</sup>**

1: Aix Marseille Univ, CNRS, IBDM, Marseille, France

**e-mail:** benoit.aigouy@univ-amu.fr

## Summary

Epithelial tissues regulate exchanges with the environment. They are very dynamic and can acquire virtually any shape. At the cellular level, they are composed of cells tightly connected by junctions. Most often epithelia are amenable to live imaging; however the vast number of cells composing an epithelium makes large scale studies tedious. Here we present Tissue Analyzer (TA), an open source tool that can be used to segment epithelia and monitor cell and tissue dynamics.

**Key Words :** Epithelia; Quantitative biology; Tissue Analyzer; Cells; Segmentation; Tracking; 2D; 3D; Deep learning; Open source

## 1. Introduction

Epithelia cover body structures, organs and cavities. They act as barriers and regulate the passage of molecules between the internal milieu and the environment. *Drosophila* epithelia often consist of a single layer of cells tightly connected by junctions. These monolayers are very dynamic and can acquire almost any shape. Their plasticity is driven by a series of cellular events, including cell death, proliferation, shape changes and neighbour exchanges. Fully understanding epithelial development therefore requires the quantitative description of cellular events as well as collective behaviours. Such a quantitative analysis is facilitated, nowadays, by the advent of deep learning segmentation that largely automates cell segmentation. The latter can then, in combination with

various computer science algorithms, be used to convert the pixel information of images into biological knowledge. Quantitative biology is not only accurate and unbiased but it also allows for the development and in depth validation of theoretical modelling of biological processes [1-6].

Here we introduce Tissue Analyzer (TA), an open source Python software, capable of segmenting and analysing single-layered epithelia. TA conveniently integrates, in a single package, all the tools necessary to study epithelial development, from cell segmentation using deep learning up to quantitative analysis of cell properties such as area and shape, as well as neighbour relationships. In this chapter we describe software installation (**Subheading 2.1**), image pre-processing (**Subheading 3.2**) and introduce epithelial segmentation (**Subheading 3.3**). Later we discuss cell tracking (**Subheading 3.5**). We finally give a brief overview of the data collected by TA (**Subheading 3.7**) and demonstrate some of the plotting capabilities of the software (**Subheading 3.8**).

## 2. Software

### 2.1. Software installation

1. The software can be installed on any recent computer capable of running Conda and Python 3.7 (or superior) (see **Notes 1 and 2**). Ideally, the computer should be equipped with a CUDA® compatible graphic card (see **Note 4**).
2. Open the system command line (see **Note 3**).
3. Then type “**conda create -n TA python==3.7**”, without the quotes, to create and activate a Conda environment for Tissue Analyzer.
4. Type “**conda deactivate**” to leave the default Conda environment.
5. Enter the TA environment by typing “**conda activate TA**”.
6. Finally install Tissue Analyzer by running the following command “**pip install --upgrade epyseg**”.

### 2.2. Run Tissue Analyzer

1. Open a terminal (see **Note 3**).

2. Type “**conda activate TA**” in the terminal to activate the environment.
3. Type “**python -m epyseg.ta**” to open the TA graphical user interface.

## 2.2. Input data

4. TA supports single and multi-channel images.
5. TA supports pixel intensities up to 32 bits per channel.
6. TA reads TIF, CZI, LIF, LSM, JPEG and PNG images (see **Notes 5 and 6**).
7. TA can create two-dimensional (2D) projections by extracting the surface of epithelial cells in three dimensional (3D) stacks using deep learning [7].
8. TA can segment two dimensional (2D) images of epithelial cells.

## 3. Methods

TA can be used to segment still images of immunostained epithelia or image sequences of a living sample expressing a fluorescently tagged protein.

### 3.1. Image acquisition guidelines

Ideally, the images to be segmented and analysed should have a good signal to noise ratio. However, nowadays, good results can be obtained with images exhibiting low signal to noise (see **Notes 7 and 8**).

1. TA is designed to detect cell outlines in single layered epithelia -- cells need to be labelled with a membrane marker (e.g. E-cadherin in *Drosophila* and ZO-1 in vertebrates).
2. If pixel intensity measurements are important for the study, make sure to avoid both bleaching and pixel saturation.
3. Post-acquisition, ensure epithelial integrity of live samples by verifying that the animal continues to develop normally.

### 3.2. Image pre-processing

Classically epithelia are acquired as 3D stacks; however, tissue segmentation is often faster, more reliable and best suited for publication when performed on 2D images. Therefore, creating a 2D

projection from a 3D epithelial is usually a good enough approximation of the epithelial tissue. This is especially true since most fly epithelia are monolayers lying in a plane and this avoids the hassle of having to correct segmentation errors in 3D which is long and difficult.

Such a 2D image can easily be created from a 3D stack using the maximum projection algorithm; however this method accumulates the noise of the entire stack, generating a 2D image that is very noisy. To avoid noise accumulation and to select only the relevant planes carrying the epithelial signal within a wide and noisy 3D epithelial stack, several algorithms or deep learning methods have been developed [8-16,7,17]. Only the deep learning based method for surface extraction [7] is currently implemented in TA (see Note 9).

However, there are rare cases, i.e. when the epithelial monolayer is not flat (e.g. the tissue is curved or bent) where the 2D projection can lead to severe underestimation of several cell metrics (e.g. cell area, perimeter ...) and need be corrected. Importantly, fixing 2D cellular metrics to accommodate for non-flat tissues is easy to achieve, it requires a 2D segmentation mask along with a height map (depth localization) of the cells in the tissue [18,12]. In parallel to extracting 2D cell cortices from 3D epithelial stacks, TA can also map cell heights. Combining 2D cell segmentation with a height map allows to obtain reliable 3D cell metrics [12]. Below is a description of the procedure to perform surface extraction in TA using the CARE model to generate height maps from 3D epithelial images. Please note, however, that if the epithelium you study is perfectly flat or if you have already obtained a 2D projection using a 3rd party software (see Note 9), you may move directly to **subheading 3.3**.

1. Select the “Pre-process” tab (see Fig. 1A)
2. Drag and drop single time points of epithelia 3D stacks over the TA interface (Fig. 1B).
3. Select the channel containing the cell outlines (see Fig. 1C and Note 10).
4. Press “Surface projection” (see Notes 11 and 12)

### 3.3. Automated segmentation

1. TA relies on EPySeg [19], a deep learning software, to segment epithelial cells. We will now

guide you through the segmentation procedure.

2. Click on the “Segmentation” tab (see **Fig. 1E**).
3. If you skipped the pre-processing step (see **Subheading 3.1**), drag and drop 2D projections of epithelial cells over the TA interface, otherwise move to **step 4**.
4. Select the channel that needs be segmented (see **Fig. 1C**).
5. Press the “Deep learning/EPySeg segmentation” button and wait until the segmentation mask is overlaid.

### 3.4. Correction of segmentation masks

Even though deep learning segmentation is globally excellent (compare **Fig. 2A and B**), few errors may remain. TA therefore allows to edit segmentation masks (**Fig. 2**).

1. Select an image from the list and scroll over it to detect missing or erroneous cell junctions (see **steps 2 and 3 and Notes 13 and 14**).
2. Adding missing cell contacts: Left click on the centroid of all the cells surrounding missing junctions (compare **Fig. 2C and D**) and press “Ctrl/Cmd+M” to run the watershed algorithm [20] locally and have the missing junctions appear (see **Fig. 2E**).
3. Removing false cell contacts: Right click on any pixel of the junction (compare **Fig. 2I and J**) and press “Enter” or “Shift+Enter”. All the pixels contained between the two vertices of the targeted junction will disappear (see **Fig. 2K**).
4. Once editing the segmentation mask of an image is done, press the “Save” button (see **Fig. 1F**) to store the changes.
5. Finalize segmentation and compute cell metrics
  - a) Repeat steps 2 to 5 for all the images in the list until there are no segmentation errors left.
  - b) Select the “Analysis” tab (see **Fig. 1G**) and press the “Measure cell properties” button.

### 3.5. Cell tracking

Identifying the same cells in time and space is essential to understand morphogenesis. Cell tracking is achieved in TA by matching cells in consecutive frames and assigning them a unique

identity (represented by a unique colour) throughout the movie (**see Note 15**).

1. Please complete “Epithelial cell segmentation” before tracking cells.
2. Select the “Tracking” tab (**see Fig. 1H**).
3. Press “Track cells”.

### 3.6. Correction of cell tracks

The tracking algorithm in TA is usually robust, especially when the segmentation is good. It can also accommodate for large time gaps between consecutive frames. However, tracking may, in rare cases, contain two types of errors: (i) cell swapping errors where two adjacent cells exchange their identities (**compare Fig. 3D and E**) and (ii) interrupted tracks where the identity of a cell is changed from one frame to the next (**Compare Fig. 3E and F**). TA now ships with a tool to rapidly spot tracking and segmentation errors (**see Fig. 3**). This tool measures the similarities of the cellular environments in consecutive frames (**Fig. 3G, H**). A low score (i.e. dark colour coded cell) using this metric indicates that the cell environment is rapidly changing; this is often (**see Note 16**) due to a segmentation or a tracking error and calls for a careful user inspection. In contrast, a high score (bright colour coded cell) indicates environmental preservation and does not require user attention.

1. Visually identifying tracking or segmentation errors:
  - a) Select the “Advanced segmentation and tracking correction” tab (**see Fig. 1I**).
  - b) Click on the “Fix masks and tracks dynamically” button and wait until a window with three rows appears (**see Fig. 3**).
  - c) Visually identify a dark cell (**see Fig. 3G, H**) and zoom (“Ctrl/Cmd +”) on it and move the mouse to it.
  - d) Using the synchronization of mouse pointer search for associated segmentation errors (**see Fig. 3A-C**) and fix them as described in subheading 3.4. If there is no segmentation error, check the tracks (**Fig. 3D-F**) for errors (**swapping error or lost track**) and fix them (**see steps 2 and 3**).
2. Correction of a cell swapping error:

Please correct all the segmentation errors and repeat the tracking procedure (**see subheading 3.5**) before attempting to correct for tracking errors. Indeed, as most tracking errors are linked to segmentation errors, fixing the latter is often sufficient to fix the former. However, if tracking errors persist, please do as follows.

- a) Move the mouse cursor over the image where the swapping is first detected (**see arrow in Fig. 3E**).
  - b) Left click on the centroid of the two swapped cells.
  - c) Press ‘Enter’ while keeping the mouse pointer on this image to fix the error for the current and following frames.
3. Correction of an interrupted track error:
- a) Place the mouse cursor on the last image where the cell is still tracked properly (**i.e. shows the right colour/identity in Fig. 3E**).
  - b) Left click the cell that is about to lose its identity/colour (**o**).
  - c) Place the mouse cursor on the following image (**Fig. 3F**) and left click on the cell with the wrong identity/colour (**+**).
  - d) Press ‘Enter’ while keeping the mouse on this image, this will fix the tracking error for the current and following frames.

### 3.7. Data analysis and quantifications

TA provides the user with a thorough description of cells including area, perimeter, elongation, planar distribution of proteins and cell neighbourhood (number and identity of neighbours) [1]. TA additionally provides information on junction length, orientation and intensity. Most cell metrics can also be corrected for tissue bending as if they would have been obtained directly from a 3D segmentation when a height map is provided (**see also subheading 3.2**).

Any data extracted by TA is stored in an internal SQLite database (**see Notes 17 and 18**). This data can also be exported to facilitate plotting using 3<sup>rd</sup> party software as follows:

1. Click on the “Export (CSV files)” tab (**see Fig. 1J**).



2. To export cell data, press the “Export cell data” button.
3. To export junction data, press the “Export bond data” button.

### 3.8. Data presentation

Because visual presentation of quantitative data is often key to human understanding of tissue dynamics, TA offers the possibility to overlay/blend virtually any data over the original epithelial images.

1. Select the “Preview” tab (**Fig 1K**).
2. Select the data to be visualized from the ‘Image/Data’ drop-down list
3. Enable ‘Colour coding’
4. Select the look up table to be used from the “LUT” drop-down list
5. Enable “Overlay/Blend” and select the channel of interest for the background image.
6. Set the “foreground opacity” value (0=totally transparent; 1=totally opaque).
7. The view can be exported as a TIF for use with 3<sup>rd</sup> party tools or for publication by pressing the ‘Single image’ or the ‘Stack’ button.

### 3.9. Going Further with Tissue Analyzer

Here we have covered basic TA functionalities from surface extraction to segmentation and data presentation. However, due to space limitation and for sake of simplicity, we have left aside several useful TA functionalities. We leave it to you try by yourself more advanced functionalities that haven't been documented here. To assist you in that process, we offer an online tutorial ([https://github.com/baigouy/tissue\\_analyzer](https://github.com/baigouy/tissue_analyzer)). Alternatively, place the mouse cursor over the TA component of interest and wait for a few seconds and until a small tooltip text appears.

## 4. Notes

1. If Python is not installed on your system or if the installed version is < 3.7 (**see Note 2**). Please download and install Miniconda from <https://docs.conda.io/en/latest/miniconda.html>
2. To determine the version of Python running on a computer

- a) Open a terminal (**see Note 3**)
  - b) Type in the terminal “**python -V**” without the quotes.
3. Open a terminal:
- a) On a Mac: open the command line terminal by pressing together the Command and Space keys, then type in the new window “**terminal**” without the quotes.
  - b) On Windows: press the “Windows” key and the “R” letter, then type “cmd” without the quotes.
  - c) On Linux (Ubuntu): press Ctrl+Alt+T
4. TA currently relies on deep learning for segmentation and image pre-processing. Even though a graphic card is not mandatory to use trained deep learning models; being equipped with a CUDA® compatible graphic card will significantly speed up the analysis. For example, we recommend to use, whenever possible, Nvidia® graphics cards with, at least, Turing or Ampere microarchitectures.
5. The JPEG compression is deeply modifying pixel data and should not be used for quantifications. Instead we recommend using TIF images with non-lossy or no compression.
6. When a file format cannot be read by TA; open the image using a third party tool such as ImageJ or FIJI [21,22], then export it as a TIF file. TIF files can be read flawlessly by TA.
7. Deep learning segmentation models can be trained using images having a low signal to noise ratio; such models can then achieve good segmentation even for tissues showing weak fluorescence.
8. Deep learning models can be used to reconstruct/denoise images prior to segmentation [7]. Note however, that if intensity measurements are key for your analysis, denoising algorithms should be used with a lot of care, as they heavily modify pixel intensities and are shown to statistically guarantee a good quality and a biologically relevant denoising only when the images used are of the same tissue and acquired in the exact same conditions as the ones used for training the model. This will almost inevitably not be the case for your tissue unless you

- do train the model on you own data, a step that is often difficult and time consuming.
9. In our hands surface extraction is best achieved using deep learning.
  10. TA output might be channel dependent, the software will warn the user when channel selection is required.
  11. Most often surface extraction works well but it is rarely error free; very importantly, any error in the height map will impair the 3D correction of 2D measurements such as area and perimeter. It is therefore important to check for the quality of the height map. To help you visually assess height map quality, TA generates a file called “height\_map\_quality\_test.tif” for each surface extraction it performs. If there are discontinuities/holes in the cortical signal of the region of interest, then the height map is not of good quality and the 3D corrections applied to the 2D cell metrics will be locally erroneous. In such case, and if the epithelium studied is flat, it is probably a good idea to rely on raw 2D measurements rather than using the 3D corrected ones, or, alternatively, to correct the height map. Finally, when using a 3<sup>rd</sup> party tool to create the 2D surface projections from a 3D epithelial stack, we recommend you set the projection tool in a way that only one frame is used for building the final 2D projection, this allows to see height map errors better. Indeed, height map errors are hard to visualize when several frames are summed or averaged to create a 2D projection and most tools use several frames to compensate for their low quality height map extraction.
  12. At this step, TA must have extracted the ratio: voxel depth/voxel width. Given the importance of this ratio for the 3D correction of 2D cell metrics, it is important to verify its correctness. This can be done in the “Properties” tab of TA (see **Fig. 1D**). If there is an error, it is possible to edit the value stored in the table and to save those changes.
  13. To enter full screen mode and see cells better, press “F12”.
  14. Some segmentation errors may be hard to see. It is often useful to switch back and forth between the original image and the image overlaid with its segmentation mask to detect errors. This can be achieved by pressing the “M” key repeatedly, to show or hide the segmentation

mask.

15. When tracked, each cell is given a unique pseudo-random colour defining its identity. A cell keeps this identity as long as it is successfully tracked. It is attributed a new identity when lost.
16. Please note that a low score is not necessarily due to an error in segmentation or tracking, it can instead indicate that a cell underwent a rapid, yet natural, environmental change (e.g. such change can result from a division or a cell delamination).
17. For each image, Tissue Analyzer stores the segmentation mask and various output files in a folder. To open this folder double click on the file of interest in the list (see **Fig. 1B**).
18. Those databases can be accessed directly by users having experience with manipulating SQL databases. Since SQL databases are heavily used by TA, it is important that all TA generated tables remain unaltered. Experienced users can nevertheless create extra tables or duplicate existing ones and modify them at will. We recommend closing TA before accessing databases using an external software to avoid database locking problems and similarly we recommend closing external database browsers before launching TA.

## **Acknowledgement**

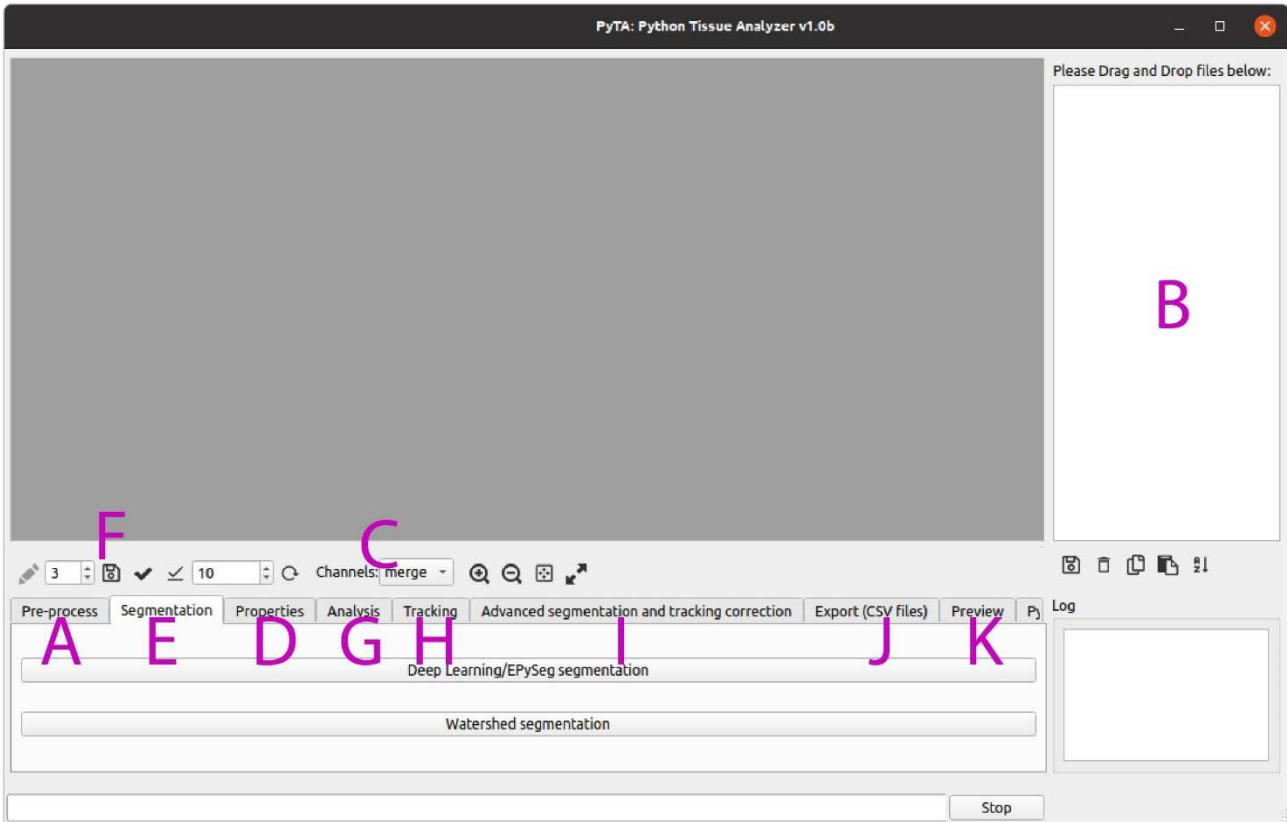
This work was supported by CNRS.

## **References**

1. Aigouy B, Farhadifar R, Staple DB, Sagner A, Röper J-C, Jülicher F, Eaton S (2010) Cell Flow Reorients the Axis of Planar Polarity in the Wing Epithelium of *Drosophila*. *Cell* 142 (5):773–786. doi:10.1016/j.cell.2010.07.042
2. Bosveld F, Bonnet I, Guirao B, Tlili S, Wang Z, Petitalot A, Marchand R, Bardet P-L, Marcq P, Graner F, Bellaïche Y (2012) Mechanical Control of Morphogenesis by Fat/Dachsous/Four-Jointed Planar Cell Polarity Pathway. *Science* 336 (6082):724–727. doi:10.1126/science.1221071
3. Etournay R, Popović M, Merkel M, Nandi A, Blasse C, Aigouy B, Brandl H, Myers G, Salbreux G, Jülicher F, Eaton S (2015) Interplay of cell dynamics and epithelial tension during morphogenesis of the *Drosophila* pupal wing. *eLife* 4:e07090. doi:10.7554/eLife.07090
4. Saadaoui M, Rocancourt D, Roussel J, Corson F, Gros J (2020) A tensile ring drives tissue flows to shape the gastrulating amniote embryo. *Science* 367 (6476):453–458. doi:10.1126/science.aaw1965
5. Umetsu D, Aigouy B, Aliee M, Sui L, Eaton S, Jülicher F, Dahmann C (2014) Local Increases

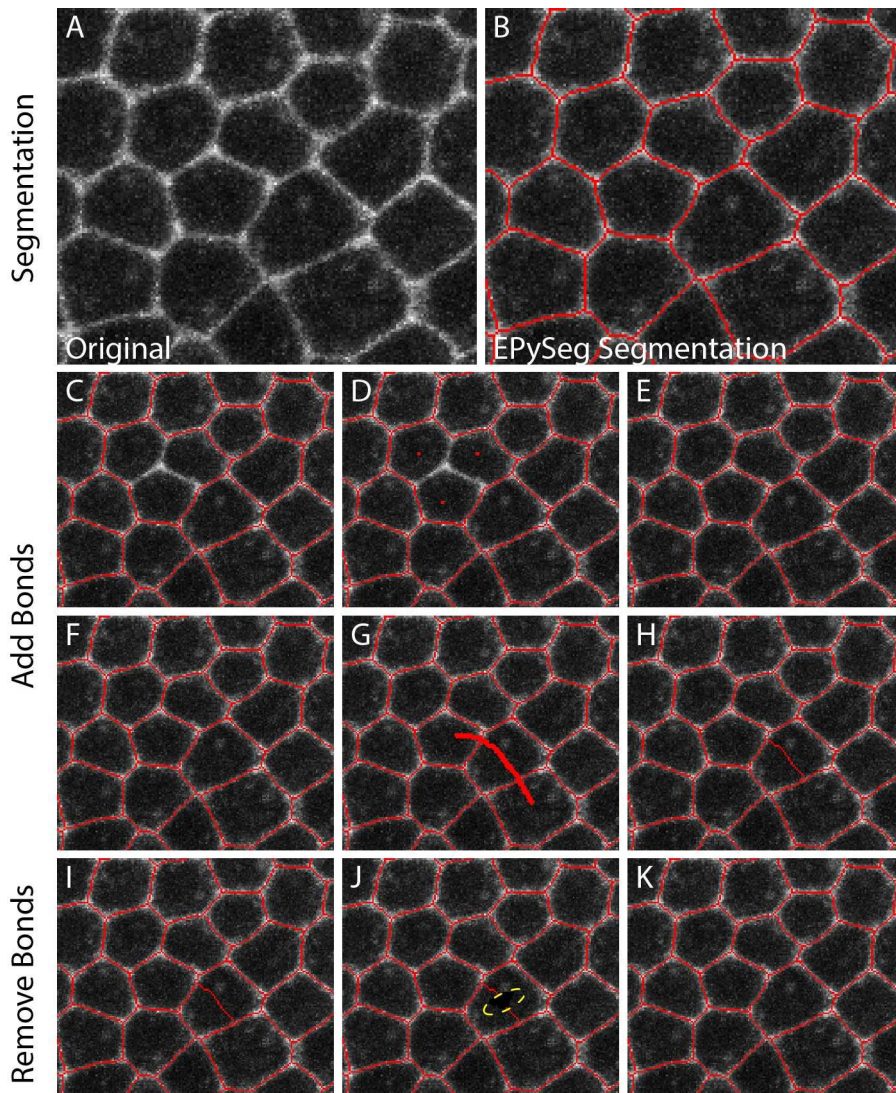
- in Mechanical Tension Shape Compartment Boundaries by Biasing Cell Intercalations. *Current Biology* 24 (15):1798–1805. doi:10.1016/j.cub.2014.06.052
6. Valon L, Davidović A, Levillayer F, Villars A, Chouly M, Cerqueira-Campos F, Levayer R (2021) Robustness of epithelial sealing is an emerging property of local ERK feedback driven by cell elimination. *Developmental Cell* 56 (12):1700–1711.e1708. doi:10.1016/j.devcel.2021.05.006
7. Weigert M, Schmidt U, Boothe T, Müller A, Dibrov A, Jain A, Wilhelm B, Schmidt D, Broaddus C, Culley S, Rocha-Martins M, Segovia-Miranda F, Norden C, Henriques R, Zerial M, Solimena M, Rink J, Tomancak P, Royer L, Jug F, Myers EW (2018) Content-aware image restoration: pushing the limits of fluorescence microscopy. *Nature Methods* 15 (12):1090–1097. doi:10.1038/s41592-018-0216-7
8. Basu S, Rexhepaj E, Spassky N, Genovesio A, Paulsen RR, Shihavuddin A FastSME: Faster and Smoother Manifold Extraction from 3D Stack. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018-06 2018. pp 2362–23628. doi:10.1109/CVPRW.2018.00305
9. Blasse C, Saalfeld S, Etournay R, Sagner A, Eaton S, Myers EW (2017) PreMosa: extracting 2D surfaces from 3D microscopy mosaics. *Bioinformatics* 33 (16):2563–2569. doi:10.1093/bioinformatics/btx195
10. Erguvan Ö, Louveaux M, Hamant O, Verger S (2019) ImageJ SurfCut: a user-friendly pipeline for high-throughput extraction of cell contours from 3D image stacks. *BMC Biology* 17 (1):38. doi:10.1186/s12915-019-0657-1
11. Forster B, Ville DVD, Berent J, Sage D, Unser M (2004) Complex wavelets for extended depth-of-field: A new method for the fusion of multichannel microscopy images. *Microscopy Research and Technique* 65 (1-2):33–42. doi:10.1002/jemt.20092
12. Herbert S, Valon L, Mancini L, Dray N, Caldarelli P, Gros J, Esposito E, Shorte SL, Bally-Cuif L, Aulner N, Levayer R, Tinevez J-Y (2021) LocalZProjector and DeProj: a toolbox for local 2D projection and accurate morphometrics of large 3D microscopy images. *BMC Biology* 19 (1):136. doi:10.1186/s12915-021-01037-w
13. Li K, Wu X, Chen DZ, Sonka M (2006) Optimal Surface Segmentation in Volumetric Images—A Graph-Theoretic Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (1):119–134. doi:10.1109/TPAMI.2006.19
14. Lombardot B (2017) Minimum Cost Z surface Projection. [https://imagej.net/Minimum\\_Cost\\_Z\\_surface\\_Projection](https://imagej.net/Minimum_Cost_Z_surface_Projection).
15. Shihavuddin A, Basu S, Rexhepaj E, Delestro F, Menezes N, Sigoillot SM, Del Nery E, Selimi F, Spassky N, Genovesio A (2017) Smooth 2D manifold extraction from 3D image stack. *Nature Communications* 8 (1):15554. doi:10.1038/ncomms15554
16. Umorin M (2006) Stack Focuser. <https://imagej.nih.gov/ij/plugins/stack-focuser.html>.
17. Wu X, Chen DZ Optimal Net Surface Problems with Applications. In: Widmayer P, Eidenbenz S, Triguero F, Morales R, Conejo R, Hennessy M (eds), 2002 2002. *Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, pp 1029–1042. doi:10.1007/3-540-45465-9\_88
18. Barbier de Reuille P, Routier-Kierzkowska A-L, Kierzkowski D, Bassel GW, Schüpbach T, Tauriello G, Bajpai N, Strauss S, Weber A, Kiss A, Burian A, Hoffhuis H, Sapala A, Lipowczan M, Heimlicher MB, Robinson S, Bayer EM, Basler K, Koumoutsakos P, Roeder AH, Aegerter-Wilmsen T, Nakayama N, Tsiantis M, Hay A, Kwiatkowska D, Xenarios I, Kuhlemeier C, Smith RS (2015) MorphoGraphX: A platform for quantifying morphogenesis in 4D. *eLife* 4:e05864. doi:10.7554/eLife.05864
19. Aigouy B, Cortes C, Liu S, Prud'Homme B (2020) EPySeg: a coding-free solution for

- automated segmentation of epithelia using deep learning. *Development* 147 (24). doi:10.1242/dev.194589
20. Vincent L, Soille P (1991) Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13 (6):583–598. doi:10.1109/34.87344
21. Schindelin J, Arganda-Carreras I, Frise E, Kaynig V, Longair M, Pietzsch T, Preibisch S, Rueden C, Saalfeld S, Schmid B, Tinevez J-Y, White DJ, Hartenstein V, Eliceiri K, Tomancak P, Cardona A (2012) Fiji: an open-source platform for biological-image analysis. *Nature Methods* 9 (7):676–682. doi:10.1038/nmeth.2019
22. Schneider CA, Rasband WS, Eliceiri KW (2012) NIH Image to ImageJ: 25 years of image analysis. *Nature Methods* 9 (7):671–675. doi:10.1038/nmeth.2089



**Figure 1. The Tissue Analyzer user interface**

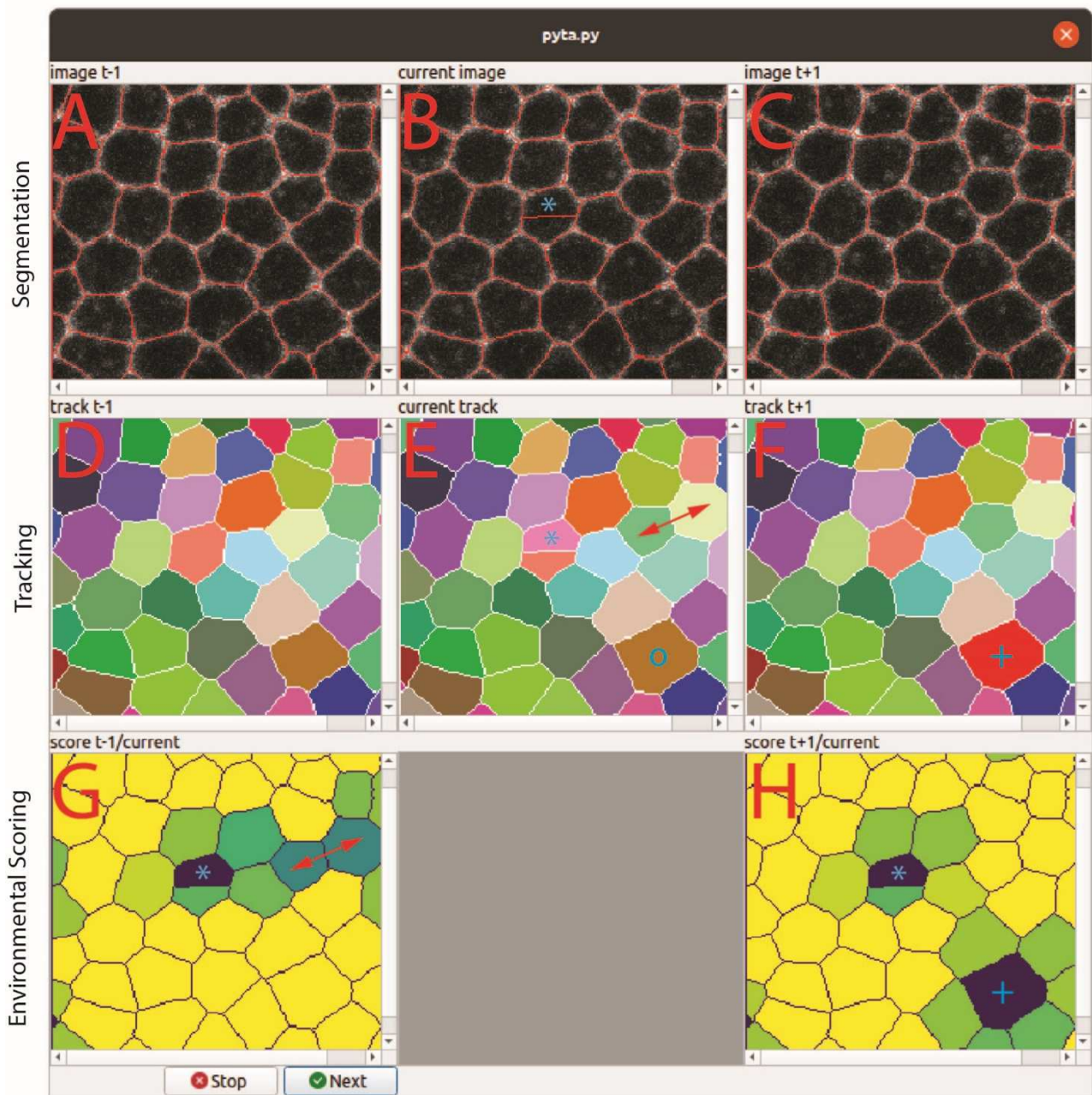
(A) “Pre-processing” tab, (B) Image list where images can be dropped, (C) Channel selection tool, (D) “Properties” tab, (E) “Segmentation” tab, (F) “Save” button, (G) “Analysis” tab, (H) “Tracking” tab, (I) “Advanced segmentation and tracking correction” tab, (J) “Export (CSV files)” tab and (K) “Preview” tab.



**Figure 2. Epithelial segmentation and mask editing**

A) Original image. B) The red overlay is the result of deep learning (EPySeg) segmentation of the image shown in (A). C) Image with three missing junctions. D) Same as (C) where cell centroids have been drawn. E) Locally running the watershed algorithm (“Ctrl/CMD+M”) using seeds drawn in (D) restores the missing contacts. F) Segmented image. G) Same as in (F) except that a new junction bisecting a cell has been drawn manually (note the new junction is intersecting two existing junctions). H) The software removes extra mask pixels when “Enter” is pressed. I) Segmented image. J) Same image as in (I) except that a small piece of a cell contact mask has been removed using the mouse right click (dashed outline). K) The entire junction is removed after pressing “Enter”.





**Figure 3. Tool to rapidly detect tracking and segmentation errors**

(A-C) Three consecutive time points of the same epithelial tissue overlaid with their segmentation masks in red. (D-F) Cell tracks corresponding to (A-C). (G, H) Colour code of the environmental score obtained when comparing the current track with the previous (G) and next tracks (H). Any dark coloured cell in (G, H) has undergone important environmental changes between consecutive frames and needs to be scrutinized for potential errors. In contrast, bright cells in (G, H) have kept a similar environment between consecutive frames and can be ignored safely. For example, the asterisk in (G)

indicates an over-segmented cell. This corresponds to a newly appeared cell (asterisk in E) that does not exist in the previous (D) or next time frame (F). This over-segmentation is visible in the segmentation mask (asterisk in B), it can be fixed as described in **subheading 3.4 step 3**. The arrow in (E) indicates two cells tracks that have been swapped (compare with D). Note that this cell swap generates a (darkening) strong environmental change (arrow in G). Cell swapping can be fixed as described in **subheading 3.6 step 2**. Finally, the dark colour coded cell (labelled with a + in H) shows a strong environmental change caused by a tracking failure. Compare the identity given to the cell in (E) and (F) (o and + symbols in E and F). This error can be fixed as described in **subheading 3.6 step 3**.