

1 EpiGraph: an open-source platform to quantify 2 epithelial organization.

3 4 **Supplementary Information.**

5 6 **SUPPLEMENTARY MATERIAL AND METHODS**

7 **EpiGraph Source code.**

8 The project's code is accessible through Github
9 <https://github.com/ComplexOrganizationOfLivingMatter/Epigraph>. It is open
10 source and available under GPLv3 license.

11 **Pipeline.**

12 The image processing pipeline entails cell recognition, valid cell identification,
13 graph of cell-cell contact creation, polygon distribution analysis and, finally, Epi-
14 Hexagons, Epi-Random and Epi-Voronoi5 calculation. These stages are properly
15 described in the subsequent paragraphs. Moreover, a full set of tutorials
16 explaining how to install and use EpiGraph are available at EpiGraph's wiki
17 (<https://imagej.net/EpiGraph>). In addition, **Movie S1** summarizes a general
18 example with default options.

19 20 - Recognition of existing cells

21 EpiGraph uses segmented images from natural tissues or simulations as an
22 input. These images must be built in binary format, where one colour should be
23 presented as the background, forming the body of cells, and the other one the
24 cells outline. Each cell is assigned a label using MorphoLibJ (Legland et al.,
25 2016). This identifier allows us to have a record of every cell and its location on
26 the image, which is necessary for the following steps.

27 - Identification of valid cells

28 Once we have properly labelled the image, we proceeded to fully analyse it. By
29 default, an invalid region is created, which is initially defined by the boundaries of
30 the image itself. We select all the cells that fall outside of this border and set them

1 as non-valid cells (note that non-valid cells are different from invalid regions, i.e.
2 the latter are not cells). In this way, we avoid incorrect characterization of polygon
3 distribution due to the lack of real neighbours in the margins of the image.
4 However, it is also possible to create a personalized invalid region, by selecting
5 as many cells as you want.

6 In addition, we define another two kinds of valid cells: the 3-distance valid cells
7 and the 4-distance valid cells. The former represents the cells that do not have a
8 non-valid cell within a distance of three cells connexions. The later kind is formed
9 by the cells having all valid cells in a maximum length of four cells connexions. 7-
10 motifs and 10-motifs require 3-distance valid cells and 17-motifs and 29-motifs 4-
11 distance valid cells.

12 - Neighbourhood creation

13 An epithelium can also be considered as a tessellation because there is no
14 space between each cell. Thus, inspecting the number of sides of the cells, we
15 can measure the number of neighbours that will surround it. We, therefore, extend
16 a mask from each pixel of the cell with a given shape and size, both selected by
17 the user. Then, we capture all the cells distinct from the actual cell that fall into
18 this mask and add them as neighbours of the concrete cell. Afterwards, we create
19 a neighbourhood network, modelling each cell as a node and connecting two cells
20 with an edge if they are neighbours.

21 - Polygon distribution analysis

22 The polygon distribution is defined by the number of sides of every valid cell.
23 We specify three different areas: the global zone formed by all valid cells, the
24 region in which 3-distance valid cells are contained and the territory in which 4-
25 distance valid cells are placed. For 7-motifs and 10-motifs, we use a path of 4-
26 nodes length (3-distance valid cells as orbits nodes, and any type of valid cell as
27 its branching nodes), and for 17-motifs and 29-motifs the area is defined by the
28 cells contained within the path of 5-nodes length (4-distance valid cells as orbits
29 nodes, and valid cells as its branching nodes). Finally, we calculate the polygon
30 distribution regarding the cells involved in the graphlet calculation.

1 - Epi-Hexagons, Epi-Random and Epi-Voronoi5 calculation

2 We have adapted Graphlet degree Distribution agreement Distance (GDD) from
3 Yaveroğlu et al. (Yaveroğlu et al., 2014) to fully integrate it with FIJI java
4 environment. We use GDD to compare two images and describe how similar they
5 are. In particular, minimum distance value (0) means two images are equal, and
6 maximum distance value (1) specifies that they are very different in terms of
7 graphlets. We compute all the graphlets in which the valid cells participate. Then,
8 we use only the graphlets in which 3-distance or 4-distance valid cells are
9 included (depending on the set used). We defined three references to compare
10 with the real images. The first one is the Graphlet degree Distribution agreement
11 Distance Random Voronoi (Epi-Random). A random Voronoi (RV) emerges from
12 500 seeds randomly placed. Then, we computed the GDD between each one of
13 the 20 RV images and the input image and used the mean of these 20 GDDs as
14 final distance. We also calculated the Graphlet degree Distribution agreement
15 Distance Hexagons (Epi-Hexagons), which measures the difference between a
16 given image and a regular tessellation of hexagons, in a similar way to the Epi-
17 Random. Finally, we computed the Graphlet degree Distribution agreement
18 Distance Voronoi 5 (Epi-Voronoi5) by comparing a given image with 20 diagram
19 5s from the CVT path and calculating its mean.

20 - Statistical analysis

21 Right before adding all the GDD values to the table, we performed a statistical
22 analysis (see **Material and methods, statistical analysis**) to calculate the
23 closest diagram on the CVTn. This test provides a confidence score determining
24 if the source image is part of the CVTn path of reference or, on the contrary, is
25 an outlier.

26 **Graphical user interface.**

27 ImageJ's FIJI distribution (Schindelin et al., 2015, 2012) provides a framework,
28 which offers a range of functions and a suitable predefined application
29 programming interface (API). Furthermore, it allows users with no programming
30 expertise to execute and exploit all their functionalities. EpiGraph is prepared for
31 heavy processes. In these cases, some operations such as “calculate graphlets”,
32 run in the background, giving the user the possibility to move around the

1 application smoothly. In addition, a progress bar displays the evolution of the task
2 in case it may take longer than expected. FIJI works on the three main Operative
3 Systems (Linux, Mac and Windows) without requiring Java installation since it is
4 usually embedded in the program. FIJI describes different ways it can be
5 operated by external software. One of them is constructing a plugin, which is used
6 by EpiGraph as basic structure to start building. Using Java Swing GUI libraries,
7 we have designed the three windows of the EpiGraph plugin.

8 - Main Window

9 The program starts after clicking on *Plugins>EpiGraph*. The first window that
10 appears is the Main Window. It contains a table of processed images (therefore,
11 initially it is empty). Each table row corresponds to a single processing of image
12 and comprises the following columns: Colour, label, Epi-Hexagons, Epi-Random,
13 Epi-Voronoi5, percentage of hexagons, radius of shape, type of shape, kind of
14 graphlet calculation, closest diagram, confidence score and a checkbox. *Colour*
15 column lets you pick the colour of the point from the palette to be visualized later;
16 *Label* shows the name of each image; *Epi-Hexagons*, *Epi-Random* and *Epi-*
17 *Voronoi5* represent the GDD of the image against hexagons, random Voronoi
18 diagrams and Voronoi 5 diagrams from CVTn, respectively; *Percentage of*
19 *hexagons* contains the proportion of hexagons for all valid cells involved in
20 graphlet calculation; *Radius* specifies the shape size used to calculate
21 neighbours; *Shape* characterizes the type of form used to calculate
22 neighbourhood; *Kind of calculation* displays which set of graphlets was used;
23 *Closest diagram* represents which number Voronoi diagram from the CVTn is
24 closest to the image; *Confidence score* is the certainty of the image being an inlier
25 of the point cloud of that diagram. Finally, *Select* column lets you choose, by
26 means of a check box, whether you visualise the calculated data in a 3D viewer
27 or not, and in the same way, removes the selected rows if you click on the button
28 *remove rows*. Additionally, you do not need to calculate the graphlet properties
29 every single time. Instead, it is possible to import your own dataset from a
30 properly formatted Excel file, using the *import table* button. Likewise, once you
31 have already analysed several images, you can export your information into an
32 Excel file, using the *export table* button. The exported table includes the polygon
33 distribution of the cells involved in the graphlets calculation. Lastly, you have the

1 option of representing all table rows, by ticking the “true” check box. For that, you
2 should click on the *visualize* button to launch the mentioned 3D viewer.

3 - Image processing Window

4 This window will be automatically triggered when a supported file format is
5 correctly selected, after clicking the *open* main window button. This window is
6 composed of a button panel with different processing options, a canvas with the
7 loaded image embedded in it and a polygon distribution legend. The only enabled
8 button when the window is first opened, will be the mode in which you label the
9 image. You can choose to label this image using either 8-connectivity or 4-
10 connectivity and then click on the *label image* button to execute it. After these
11 other buttons become enabled. These buttons are classified in 3 panels:

- 12 • *Region of interest*. The main objective of this panel is selecting regions of
13 interest so that only valid regions are processed and invalid ones are
14 discarded. The *create ROI* and *select cells* buttons let you select various
15 rectangular (default) regions and pick individual cells using the FIJI’s ROI
16 manager tool. You can repeatedly combine both options to establish a
17 valid region in which to operate. Furthermore, you can change the ROI
18 selection shape, from rectangular to another shape, using FIJI’s control
19 panel. On the other hand, there is the option to choose an invalid region
20 by clicking over it, after clicking the *add invalid regions* button. This action
21 only lets you store a single set of invalid ROIs at a time, so if you wanted
22 to add an invalid region after saving previous invalid regions, you must
23 delete the former to include a new one.
- 24 • *Neighbourhood*. This panel allows you to visualize valid and non-valid
25 cells, invalid regions and polygon distribution values from the current
26 image. To calculate the neighbourhood you should select, using the
27 number selector (located close to *radius*), a size of shape in pixels and a
28 specific geometrical shape using the tool described above (located next to
29 *shape*). This selection depends on the width of the cell outline. If the border
30 cells have a width of 1 pixel, it is enough using a size of 3 pixels to explore
31 the cells vicinity. By default, we expand 3 pixels each cell, using the
32 circular shape, to look for their neighbours. For wider border cells, you can
33 choose higher size of mask with either circular or squared shape. Once

1 the parameters have been selected, the *test neighbours* button can be
2 pressed to calculate the neighbourhood for the valid cells belonging to the
3 ROI. If the ROI is the whole image, a column next to the legend of polygons
4 shapes, headed by “Graphlets”, would be filled with the polygon
5 distributions of the valid cells. Otherwise, if the ROI is a subsection of the
6 image, a new column appears next to “Graphlets” headed by “ROIs” that
7 will be filled by the polygon distribution of the valid cells belonging to this
8 ROI. Note that in this case, the “Graphlets” column would be filled with the
9 polygon distribution of the all valid cells participating in the ROI graphlets
10 calculation.

11 Finally, when the columns are filled, an overlay is displayed on the canvas,
12 representing each cell colour coded by polygon number. These colours
13 are the same that the ones at the legend located at the left of the window.
14 In addition, the *toggle overlay* button lets you choose whether to visualize
15 the overlay created by *test neighbours*. The invalid regions and the cells
16 outlines are labelled in black in the overlay, dark grey marks non-valid
17 cells, and the rest of colours (represented in the polygon distribution
18 legend) are reserved for valid cells. It is important to highlight that the
19 bright colours of the legend displayed over the canvas represent the
20 number of sides of the valid cells into the ROI. The same pale colours
21 represent the polygon distribution of valid cells located out of the ROI, yet
22 which still contribute to the graphlet calculation of cells within the ROI. A
23 4-distance valid cell is a cell that do not have a no valid cell within a
24 distance of four cells connexions. Moreover, a 3-distance valid cells have
25 no valid cells in at least three branched cells.

- 26 • *Graphlets*. This panel is designed with the aim of saving graphlet data
27 internally and externally. There is a text box where you can add the data
28 label (image name by default), located after *image label* text. You can also
29 select a colour label for your data, using *pick a colour* button. Finally, you
30 can choose the appropriate method to calculate graphlets data. The
31 options are: 26 graphlets representing 29 cellular motifs (29-motifs), 17
32 graphlets that are contained in 17 cellular motifs (17-motifs), 9 graphlets
33 on behalf of 10 cellular motifs (10-motifs) and 7 graphlets typifying 7
34 cellular motifs (7-motifs). 7-motifs and 10-motifs are formed by graphlets

1 of maximum 4 nodes, while 17-motifs and 29-motifs make use of graphlets
2 of maximum 5 nodes. 7-motifs and 10-motifs are most useful when images
3 have a small number of cells, since both require fewer cells than 29-motifs
4 and 17-motifs. Once the method has been selected, you can then click the
5 *calculate graphlets data* button to acquire data for all calculated graphlets.
6 When this process is complete, data are automatically added to the main
7 window table. Furthermore, by clicking on *Export Graphlet data* you can
8 export a ZIP file containing: a JPG image representing the neighbourhood,
9 another JPG image capturing the label of the cells, a CSV file storing all
10 calculated graphlet data and a .sif that represent the neighbourhood
11 network.

12 The window has a progress bar that estimates the process state. You can
13 modify the image zoom by pressing control and rotating the mouse wheel at same
14 time.

15 - Visualizing Window

16 We use a 3d viewer to display our calculated results stored in the main window's
17 table. When a row's checkbox is ticked, it will be plotted. To develop this window,
18 we have used an open source library named Jzy3D (<http://www.jzy3d.org/>) that
19 is able to generate different graphical representations. In particular, we make use
20 of the 3d scatter plot class.

21 This window is displayed after clicking on the *visualize* button (located in the
22 main window), even if your table is still empty. This window is composed of the
23 scatter plot figure located on the left and a set of components to modify the
24 appearance of this figure on the right. The plotted figure is delimited by a 3D box
25 with 3 axes: Percentage of hexagons, Epi-Hexagons and Epi-Random by default.
26 These 3 axes can be replaced by any of the following configurations ($X - Y - Z$
27 axes) using the drop-down list with the label *Axes of figure* (upper right corner of
28 the window):

- 29 1- Epi-Hexagons, Epi-Random, Percentage of hexagons (default)
- 30 2- Epi-Hexagons, Epi-Random, Epi-Voronoi5
- 31 3- Epi-Hexagons, Epi-Voronoi5, Percentage of hexagons
- 32 4- Epi-Random, Epi-Voronoi5, Percentage of hexagons

1 The Percentage of hexagons axis encapsulate values between 0 – 100, and the
2 others between 0 – 1. These limits can be modified for zooming in on the
3 individual axes. We used three *rangeSliders* to select the range for each axis to
4 be represented due to the limitations of the Jzy3D library. This library only
5 provides you the possibility of zooming the Z axis turning the mouse wheel. You
6 can visualize the three *rangeSliders* (one per axis) just below the *Axes of figure*
7 drop-down list.

8 By default, the scatter plot displays the CVTn path, shown as individual dots:
9 The darkest dot represents the average of 20 Voronoi diagrams 1 in CVTn and
10 the lightest one is the average of 20 Voronoi diagrams 700 in CVTn. You have
11 the option to disable the visualization of these references clicking on the
12 checkbox ‘Show reference’ (just below *rangeSliders*), to only display calculated
13 data. One can also adjust the size of the dots by modifying the position of the
14 slider bar, located just below the previously mentioned checkbox. To select the
15 different modes in which graphlet data of the CVTn reference can be calculated,
16 a drop-down list can be deployed by clicking on the label *Motifs of CVTn*
17 *reference*. This list allows you to represent the CVTn path depending on the
18 method with which the graphlets were calculated: 29-motifs, 17-motifs (by
19 default), 10-motifs and 7-motifs. *In this way, you can export the visualized CVTn*
20 *path by clicking on the button with label “Export actual CVT reference” as an excel*
21 *file with the GDDs to represent yourself your acquired data elsewhere (as in*
22 *“Export table” of the Main Window).*

23 The dots shown in the figure in the Main Window can be modified by changing
24 the colour box in the Main Window’s table. In addition, a .png figure screenshot
25 can be saved, by pressing the *Export view* button. For more detail, different
26 angles and modes of visualization are available. By clicking the figure and moving
27 the mouse, one can change the viewing angle, while double clicking on the figure
28 will automatically perform a 360 degrees’ rotation.

29 **Functionalities.**

30 Along the program execution pipeline, there is the opportunity to develop a set
31 of functionalities:

1 - Label image

2 Once you have selected an image to process, this image is binarized and the
3 background is analysed to detect the cells and their outlines. If the number of
4 white pixels is higher than black pixels, white pixels will be considered as cell's
5 body, and vice versa. After that, each cell's body region of the image is assigned
6 with a unique label. Thanks to the extensible architecture of FIJI, in which you
7 can install plugins to add additional functionalities, this process of labelling
8 regions is made using MorphoLibJ functionalities. Specifically, we have used the
9 connected component labelling, which transforms a binary image into a labelled
10 one by assigning a specific number to each connected component.

11 - Create ROI

12 Though it is possible to analyse the entire image, you can also process a smaller
13 region of interest (ROI). Through FIJI's Roi Manager we can manage the ROIs,
14 saving and performing operations on one at a time. We have selected two default
15 operations within EpiGraph: rectangle (or any available shape) and multipoint
16 selection. The former creates a rectangle selection (by default) and defines all
17 the cells that fall inside it as valid cells. The latter enables the multipoint function
18 with which it is possible to select individual cells. However, it is also possible to
19 change these default forms of selection, by going straight to FIJI's main window
20 and picking any from the toolbar.

21 - Select invalid regions

22 As your image may contain artefacts in the form of false valid cells, we have
23 made it possible to mark these zones as invalid regions. All valid cells
24 surrounding the mentioned invalid region will be considered as "non-valid cells".
25 It is possible to select several areas of the image using the multipoint tool, to
26 convert them to invalid sections. To validate this action, the selected invalid
27 region should have the same colour as the cells' background.

28 - Testing neighbours

29 As mentioned the previous section, EpiGraph is able to calculate the polygon
30 distribution using 2 essential parameters: a given pixel radius and an element
31 shape. The element shape expands to a given radius of the cells of interest,
32 looking for cells neighbouring. To ensure that the neighbourhood is correctly
33 captured, we allow the users to verify if the image has the right parameters by

1 themselves. The polygon distribution of the image will appear with numbers at
2 the left side of the window and each cell will also be painted with a colour
3 representing its number of sides. Additionally, non-valid cells are coloured dark
4 grey (almost black). Depending on the method chose to calculate graphlets (29-
5 motifs or 17-motifs considering 4-distance valid cells; 10-motifs or 7-motifs
6 considering 3-distance valid cells), the zone coloured with bright colours will be
7 established only by a group of central cells. In the same way, when we select a
8 ROI, the cells affecting the cells within the ROI will be represented in pale colours.
9 Furthermore, due to the variant number of 3-distance and 4-distance valid cells
10 that are going to be filling the ROI, you may encounter when a series of
11 particularities:

- 12 ○ Some cells of your defined ROI could not be final valid cells (3-distance
13 or 4-distance). If the ROI is not surrounded by a minimum of 4 cells in all
14 directions, the number of 4-distance valid cells is going to be lower than
15 the number of 3-distance valid cells. Thus, if you select 17-motifs or 29-
16 motifs, some cells will not be 4-distance valid cells and will not be used
17 in the final graphlet computation as principal nodes.
- 18 ○ An empty ROI. If you select a region with no 3-distance or 4-distance
19 valid cell, you will obtain an empty neighbourhood.
- 20 ○ The pale coloured cells affecting the 3-distance or 4-distance valid cells
21 will differ, when switching from 17-motifs to 7-motifs (for example).
22 Therefore, a different number of cells will be affecting the bright coloured
23 cells.

24 Finally, the calculations for neighbourhood will be exploited by *calculate*
25 *graphlets* module, if parameters are not modified since this step.

26 - Calculate graphlets

27 The main function of EpiGraph is the graphlet comparison. It begins by checking
28 if there are any selected cells or ROIs. Then, in case any configuration has
29 changed, we re-compute the neighbourhood, otherwise we take the information
30 from a previous computation. From this neighbourhood and valid cells, it
31 calculates the graphlets for the involved network of neighbours. As mentioned in
32 previous sections, it would be incoherent not to filter the graphlets, so we refine
33 it by adding only the nodes at a fixed distance from the border nodes to the final

1 graphlets. We first calculate the total set of graphlets for valid cells and then select
2 a filter that depends on the chosen type of Graphlets to be implicated: 29-motifs,
3 17-motifs, 10-motifs and 7-motifs. The involved orbit nodes for graphlet
4 calculations will be referred as 3-distance and 4-distance valid cells. When we
5 have the final graphlets, we calculate the three distances (Epi-Hexagons, Epi-
6 Random and Epi-Voronoi5). Depending on the number of graphlets selected, a
7 variable number of orbits are used in the comparison. Finally, the results will be
8 added automatically to the table on Main Window.

9 - Statistical analysis

10 Once all GDD data from an input image have been calculated, a statistical
11 analysis is carried out to check if the GDD values of a certain image matches with
12 the CVTn scale or if the image is out of the CVTn path in terms of organization.
13 First, the closest CVTn diagram to the image is computed, estimating the
14 Euclidean distance by considering 3 dimensions: Epi-Hexagons, Epi-Random
15 and Epi-Voronoi5. Thereafter, it is checked whether the image could belong to
16 the closest diagram CVTn point cloud. At this point, the method generates a
17 confidence score for the comparison to the CVTn (see **Material and methods,**
18 **statistical analysis**). The closest diagram and the confidence score are
19 computed and added to the main window table immediately upon calculating the
20 GDD values or importing from an existing excel document. Alongside these two
21 parameters, when exporting to an excel, the Euclidean distance to the closest
22 diagram is also presented as a column.

23 - Visualization

24 Visualizing results properly is a major feature and an important challenge to
25 interpret results. Thus, we have embedded Jzy3d chart in a Java *JDialog* where
26 you can visualize the calculated results' three coordinates (any combination of
27 Epi-Hexagons, Epi-Random, Epi-Voronoi5 and percentage of hexagons) as a
28 point in a scatter plot. Once the points are represented you can compare them
29 with our CVTn reference. At this point, you may want to change the illustrated
30 CVTn selecting the number of graphlets used on the computations, although it is
31 advisable to compare images with the same configuration (7-motifs, 10-motifs,
32 17-motifs or 29-motifs). Additionally, it is possible to increase or decrease the size

1 of the dots and zoom each axis manually. Finally, you can export the actual view
2 of the chart to an image file.

3 **Dependencies.**

4 FIJI is designed to add functionalities via several routes, one of which is Plugin.
5 In order to have simple control of dependencies and project settings, it uses
6 Maven. Maven is a tool that is broadly used for supervising and building Java-
7 based projects. It integrates several tools, such as Javadoc, to make all the
8 programming steps easy. Dependencies are downloaded and updated when
9 available. Furthermore, it helps you create the package with all your code and
10 zips it in a .jar file that will be the Plugin format of FIJI. All this is achieved through
11 its project object model (POM), which you can shape to your project adding
12 mailing lists, issue tracking and more.

13 **Supported file formats.**

14 EpiGraph's input is an image. This image should be properly segmented and
15 grey-scale, however, there are several options within EpiGraph to configure it
16 depending on the type of the image. We allow images with different sizes of
17 borders, since it is possible to increase the radius of the mask, changing the way
18 in which the neighbourhood is built. However, it is mandatory that images are
19 binary images (8 and 32 bits RGB images can be loaded, but they might not be
20 correctly computed), where one colour is presented as the background, forming
21 the body of cells, and the other the cells outline. Regarding the image file
22 extensions, we entrust FIJI with the image opening and the supported files, so
23 individual image file extensions allowed by FIJI would be supported in EpiGraph.
24 We cannot admit any sequence of images and the program is sensitive to single
25 images with high resolution. Accordingly, we warn the user when the images
26 overpass the 3000px either along the height or width, to prevent possible
27 execution problems due to the high computational complexity.

28 **Quick step-by-step EpiGraph's usage.**

29 - Installation

30 *Through FIJI/ImageJ update site:*

31 The usual way to install a FIJI plugin is through his on-site updater. It's usually
32 located on "Help > Update FIJI". Once it is open, you click on "Manage updates

1 *sites*", look for EpiGraph and tick the checkbox next to it. Finally, "*Apply changes*"
2 and you should have successfully installed this plugin. With this option, you can
3 automatically get the latest version of EpiGraph.

4 *Manually:*

5 On the other hand, you may just want to download the .jar file from
6 <http://bit.ly/EpiGraph-1-0-2> (or even generate your own .jar from source code)
7 and install it manually. To do this, you can click either *Plugins>Install* or
8 *Plugins>Install Plugin*. Then you select the provided .jar and you should be able
9 to run EpiGraph.

10 - Simple example of a complete analysis

11 Calculate Epi-Hexagons, Epi-Random, Epi-Voronoi5 and percentage of
12 hexagons from a given image and visualize it in the 3D visualizer.

- 13 1. Select *open*.
 - 14 a. Select a supported image.
 - 15 b. If the image is supported, a window with the image will open.
- 16 2. Pick connectivity:
 - 17 a. Select the connectivity of your image. Usually 8-connectivity.
 - 18 b. Press *label*.
- 19 3. Default configuration:
 - 20 a. Radius of 3 pixels.
 - 21 b. Circle shape.
- 22 4. Calculate graphlets and statistics:
 - 23 a. Write a name for your image.
 - 24 b. Pick a colour for your image.
 - 25 c. Select *29-motifs* in the combo box.
 - 26 d. Press *calculate graphlets*. This will calculate Epi-Hexagons, Epi-
27 Random and Epi-Voronoi5.
 - 28 e. When GDDs have been calculated, a statistical analysis is carried
29 out providing a confidence score and the closest CVTn diagram to
30 the image. Once it is finished, if the name box is not empty, all data
31 will be automatically added to the main window table.
- 32
33 5. Export graphlets data:

- 1 a. Press the *Export graphlets data* button.
- 2 b. EpiGraph create a zip folder containing several files; an image with
- 3 all the cells labelled with their corresponding identifier, another
- 4 image representing the polygon distribution and, finally, a '.csv' file
- 5 with the graphlets for all the valid cells.
- 6 6. Visualize your results:
 - 7 a. Return to the main window.
 - 8 b. Click on *visualize*.
- 9 7. Export view:
 - 10 a. Staying in the visualizing window, you have the option of exporting
 - 11 the actual view into a '.png' file at every stage.
 - 12 b. Press *export view*.

13 Congratulations! You have done the complete pipeline. Now, check where your
14 image is in regard to the CVTn scale. You should analyse whether or not your
15 image is near the CVTn (reference), which can be done using its closest diagram
16 and confidence score from the table.

17 - Create region of interest

18 If you wish to test a sub-region of your images, you can create a region of
19 interest (ROI). We have already opened an image and labelled it within EpiGraph.

- 20 1. In the image processing window:
 - 21 a. Press *Create ROI*.
 - 22 b. The default ROI is square shaped; however, you may want to test
 - 23 another shape. You can now return to the FIJI application and
 - 24 select any of the existing ones on the toolbar.
 - 25 c. Click *done*. Your ROI is managed and stored in FIJI Roi Manager.

26 You have created your ROI; however, you cannot see how it affects the image
27 yet.

- 28 2. Press *Testing neighbours*:
 - 29 a. An image representing each cell by its number of neighbours
 - 30 will appear as an overlay in your image. Remember that the valid
 - 31 cells in the ROI, which appear with intense colour, are the 3-
 - 32 distance or 4-distance valid cells (depending on your graphlet
 - 33 selection), and the pale colours represent the rest of valid cells that

1 affect the computation of graphlets. The result will be represented
2 in the polygon distribution legend.

3

4 Once this procedure has been accomplished, you can continue the execution in
5 the same way as before: graphlet calculation and statistics, export of results and
6 visualization.

7 - Combine ROIs

8 As regions of interest are managed by the ROI manager you can add several
9 ROIs to the image. You have already opened the image and properly labelled it.

10 1. Press *Create Roi*:

11 a. Select a left-hand region with the rectangular ROI shape. You
12 should not select any of the cells on the right-hand side yet.

13 b. *Done*.

14 2. Select individual cells:

15 a. Click *Select cells*.

16 b. Pick only cells on the right-hand side of the image.

17 3. Test your setup:

18 a. You should be able to see the cells that fall into both ROIs.

19 b. Press *test neighbours*.

20 4. Oh! You realize you have made a mistake:

21 a. You do not want the first ROI.

22 b. Go to the ROI manager.

23 c. Select the first ROI and click Delete.

24 5. Test the new selection:

25 a. You will now represent only the existing ROI on the right.

26 b. Click *test neighbours* again.

27 If you create some ROIs, the selected cells will be all the cells that fall into any
28 of the ROIs. The logical operation would be an OR, so ROIs could be
29 disconnected.

30 - Selection of the number of motifs

31 There are 4 possible configurations. In this example we will create an analysis
32 with 29-motifs using 26 graphlets.

- 1 1. Open an image representing cells with 4-connectivity and borders with 1-
2 pixel width.
- 3 2. Label it:
 - 4 a. Click *label*. A warning notice advises that there are very few cells in
5 the image. Your full image is tagged with the same label because
6 all the cells are connected.
 - 7 b. Select now 4-connectivity.
 - 8 c. Click *label*. That works as expected! Each cell has a different tag.
- 9 3. Test neighbours by default.
- 10 4. 29-motifs:
 - 11 a. This configuration has the maximum number of graphlets present
12 in our study.
 - 13 b. Change the combo box to *29-motifs*.
- 14 5. Calculate graphlets:
 - 15 a. Add a proper name to your image.
 - 16 b. Press calculate graphlets to add the full results to the table.
- 17 6. Visualizing:
 - 18 a. Change the colour of your row to a pink-ish colour. This will let you
19 differentiate your data from the reference, which is black, grey and
20 almost white.
 - 21 b. Click on *Visualize*.
 - 22 c. As a default, your reference is *17-motifs*. So, change this option in
23 the combo box to *29-motifs*.
 - 24 d. Modify the axes of figure to 'Epi-Hexagons, Epi-Random, Epi-
25 Voronoi5'.
 - 26 e. Change the range sliders at your right side to get an adjusted
27 visualization per axis.

28 You will see that the reference has changed regarding the default 17-motifs
29 reference. Your image is probably aligned to the CVTn reference. This is because
30 your image is within the CVTn path, otherwise the image organization would be
31 different to our model of reference. You can verify this with the confidence score,
32 which also provides the closest diagram to the input image.

1 - Increase the radius of neighbourhood.

2 You have a segmented image with a border wide of 3 pixels, and you want to
3 capture the real number of neighbouring of each cell. You have already opened
4 the image and properly labelled it.

5 1. Test if neighbours are correct:

6 a. Since you do not know which pixel radius you should select, test
7 various shapes and radii in order to see if the cells with 6
8 neighbours are actual 6-sided cells.

9 b. Change parameters and click on *test neighbours* to make sure the
10 neighbourhood is correct.

11 Once your polygon distribution is correct enough, you can follow the next step
12 of the analysis protocol.

13

14 - Import/export table

15 You may want to save your results and to do this you can export your results to
16 an excel file, simply by pressing the button export in the main window.

17 On the other hand, if you would like to continue your session right where you
18 left off, you could import your exported excel file to the table and carry on with
19 your analysis. You can add as many .xls files to the table as you want and we
20 allow duplicates. If you want to remove them, just tick them at the *Select all*
21 column and click on *Delete rows*.

22 Another option is to calculate the mean values of a particular set of images
23 outside EpiGraph. To do this, import the excel with the mean values. EpiGraph
24 will automatically calculate all the values of the closest diagram and its associated
25 confidence score, enabling you to improve your analysis on that dataset.

26

27 Troubleshooting

28 If the visualization is not working properly, you may need to update FIJI after
29 the installation.

30 If you find something is not working with any functionality of the application, you
31 can send an e-mail to pvicente1@us.es or pgomez-ibis@us.es. Reporting bugs
32 can also be done through Github

1 <https://github.com/ComplexOrganizationOfLivingMatter/Epigraph/issues>. Known
2 issues are held at the same page.

3

4 **License Information.**

5 To encourage the sharing of resources, EpiGraph is published under an
6 open-source (GPLv3) license, which can be downloaded from

7 <https://github.com/ComplexOrganizationOfLivingMatter/Epigraph/blob/master/LI>
8 CENSE.

9

10

1 **SUPPLEMENTARY FIGURES**

2

3 **Figure S1. Graphlets and orbits configuration.** Illustration of graphlets
4 networks used in (Pržulj, 2007). Each graphlet configuration is labelled with G_n ,
5 in which, 'n' is the graphlets number (from G_0 to G_{29}). These labels match with
6 the cellular motifs in (**Fig. 1**). Each vertex represents a cell and each edge the
7 connection between two cells. Therefore, the graphs represent the connectivity
8 network for each matched cellular motif. The four graphlets labelled in red (G_{20} ,
9 G_{22} , G_{25} and G_{27}) are the ones discarded in our work. The digits (from 0 to 72) at
10 some mauve vertices indicate the orbits number counted for the GDD calculation.

11

1 **Figure S2. Pipeline for the Graphlet degree Distribution agreement Distance**
2 **(GDD) calculation.** Scheme representing the protocol sequence to calculate the
3 GDD between two segmented images: Eye and Diagram 1. First, a network of
4 cell-to-cell contacts is computed defining the centroids of valid cells as nodes and
5 its connections with neighbouring cells as edges. The nodes are represented with
6 the same colour code as in **Fig. 2**. Second, the graphlets are extracted from the
7 network. This enables the calculation of an index of the distribution of every
8 graphlet. Finally, a comparison between the two Graphlet degree Distributions is
9 performed to obtain the GDD value.
10

1 **Figure S3. CVTn scale capturing tissue organization.** Representation
2 showing how the CVTn iteratively evolves (**Material and Methods**) from a
3 disordered and heterogeneous tessellation (V1) to a more homogeneous and
4 uniform Voronoi (V700). Each diagram is presented in circles surrounded with
5 greyscale tones. The series of dots between the circles represent the middle
6 diagrams amid V4 and V700. CVTn, based on the original CVT path, has been
7 shown to be a good descriptor for identifying a tissue in homeostatic state. The
8 graphic displays the resulting relationship between the natural samples and their
9 most akin Voronoi diagram (signed by a double arrow) in terms of organization
10 computed by EpiGraph. In addition, it shows which biological samples were
11 contained within the scale (IN) and which were not (OUT). The colour code of the
12 actual samples is to the same as that in **Fig. 2D**.

13

14

1 **Figure S4. Comparisons between epithelia and CVTn path.** 5-dimensional
2 scatter plot representing every possible comparison between Epi-Hexagons, Epi-
3 Random, Epi-Voronoi5, percentage of hexagons and shape index. The X and Y
4 axes for each chart correspond to its column and row names respectively. The
5 CVTn path is shown starting at diagram 1 until diagram 100 (from 1 to 20, from
6 30 to 100 by steps of 10), and they are represented in greyscale beginning in
7 black and reducing its darkness with the increase of the diagrams. A set of natural
8 epithelia are display: cNT (16 samples, light blue), dWL (15 samples, green),
9 dWP (16 samples, red), Eye (3 samples, orange) and dMWP (3 samples, violet).
10 The mean value is represented as a circle and their individual values as smaller
11 circumferences. In the shape index column, values up to 3.81 represent solid
12 tissues, painted in orange; shape index values greater than 3.81 indicate fluid
13 tissues, in green. The histograms represent the dots density along each column
14 of comparisons. These graphs complement **Fig. 2**.
15
16

1 **Figure S5. Comparisons of the different simulations regarding the CVTn**
2 **and natural tissues.** All the possible combinations from Epi-Hexagons, Epi-
3 Random, Epi-Voronoi5, percentage of hexagons and shape index in 2-
4 dimensional graphics. The CVTn path is shown starting at the diagram 1 until the
5 diagram 100 (from 1 to 20, from 30 to 100 by steps of 10). The average values
6 of 20 replications per diagram, are represented as greyscale dots beginning in
7 black and reducing its darkness with the increase of the diagrams. Images from
8 biological samples are marked as dots representing their mean: cNT (16
9 samples, light blue), dWL (15 samples, green), dWP (16 samples, red), Eye (3
10 samples, orange) and dMWP (3 samples, violet); Simulations are represented
11 with their mean (circle) and their individual values (circumference): Proliferative
12 Control (20 replicates, carnation pink), Case III (17 replicates, hot pink) and Case
13 IV (15 replicates, purple); Non-proliferative control (20 replicates, blue bell),
14 Squared (20 replicates, azure blue) and Elongated simulations (20 replicates,
15 Cornflower Blue). In the shape index column, values up to 3.81 represent solid
16 tissues, painted in orange; shape index values greater than 3.81 stand for fluid
17 tissues, in green. The histograms represent the dots density along each column
18 of comparisons. These graphs complement **Fig. 3.**
19

1 **Figure S6. Shape index values for the analysed images.** Plot with shape index
2 and Epi-Random as axes. The orange region represents the shape index values
3 that define a tissue as 'solid', and the green zone as 'fluid'. Circles are the
4 average value obtained from the individual samples from the natural images: Eye,
5 cNT, dWL, dWP and dMWP; CVTn (from diagram 1 to 100), Proliferative Control,
6 Case III and Case IV; Non-proliferative control, Squared and Elongated
7 simulations. Triangles stand for the average value from solid/fluid tissues: WT,
8 (blue) and *Mbs-RNAi* (orange).
9

1 LEGENDS OF SUPPLEMENTARY FILES

2 **Movie S1. A general example of the usage of EpiGraph.** The movie shows all
3 the functionalities of EpiGraph and how to use them, in general terms.

4
5 **Table S1. GDD values between pairs of images from images of biological**
6 **samples and Voronoi Diagrams.** Graphlet degree Distribution agreement
7 Distance mean between each row and column is shown. Data is divided by the
8 used cellular motifs (17-motifs and 29-motifs). The used samples are Voronoi 1,
9 4 and 5: 20 replicates; dWP: 16 replicates; dWL: 15 replicates; Eye: 3 replicates.

10

11 **Table S2. GDDs and percentage of hexagons of CVTn reference using all**
12 **the cellular motifs sets.** Data are distributed depending on the cellular motifs
13 used (7-motifs, 10-motifs, 17-motifs, 29-motifs). Mean and standard deviation of
14 percentage of hexagons, Epi-Hexagons, Epi-Random and Epi-Voronoi5 are
15 shown, along with their associated diagram. 20 replicates of each diagram are
16 represented.

17

18 **Table S3. Outlier detection results of natural images, simulations and**
19 **rigid/soft tissues.** For the mean of each set of images, the closest diagram and
20 confidence score are calculated (see Material and methods, statistical analysis).
21 In green, the confidence score above 0.95 corresponding to inliers. In red are
22 marked the confidence scores below 0.95, which corresponds to outliers.

23

24 **Table S4. Measurements of fluidity from simulations and actual epithelia**
25 **using the shape index.** The shape index of all the tessellations used along the
26 manuscript have been calculated in terms of the average of its median and mean
27 values for each sample. The green colour is referred to a tissue with a shape
28 index defined as fluid, and the orange colour is identified as solid.

29

30 REFERENCES

31 Legland, D., Arganda-Carreras, I., Andrey, P., 2016. MorphoLibJ: integrated
32 library and plugins for mathematical morphology with ImageJ. *Bioinformatics* 32,
33 btw413. doi:10.1093/bioinformatics/btw413

34 Pržulj, N., 2007. Biological network comparison using graphlet degree

1 distribution. *Bioinformatics* 23, 177–183. doi:10.1093/bioinformatics/btl301
2 Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch,
3 T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B., Tinevez, J.-Y., White, D.J.,
4 Hartenstein, V., Eliceiri, K., Tomancak, P., Cardona, A., 2012. Fiji: an open-
5 source platform for biological-image analysis. *Nat. Methods* 9, 676–682.
6 doi:10.1038/nmeth.2019
7 Schindelin, J., Rueden, C.T., Hiner, M.C., Eliceiri, K.W., 2015. The ImageJ
8 ecosystem: An open platform for biomedical image analysis. *Mol. Reprod. Dev.*
9 82, 518–529. doi:10.1002/mrd.22489
10 Yaveroğlu, Ö.N., Malod-Dognin, N., Davis, D., Levnajic, Z., Janjic, V.,
11 Karapandza, R., Stojmirovic, A., Pržulj, N., 2014. Revealing the Hidden
12 Language of Complex Networks. *Sci. Rep.* 4, 1–9. doi:10.1038/srep04547
13