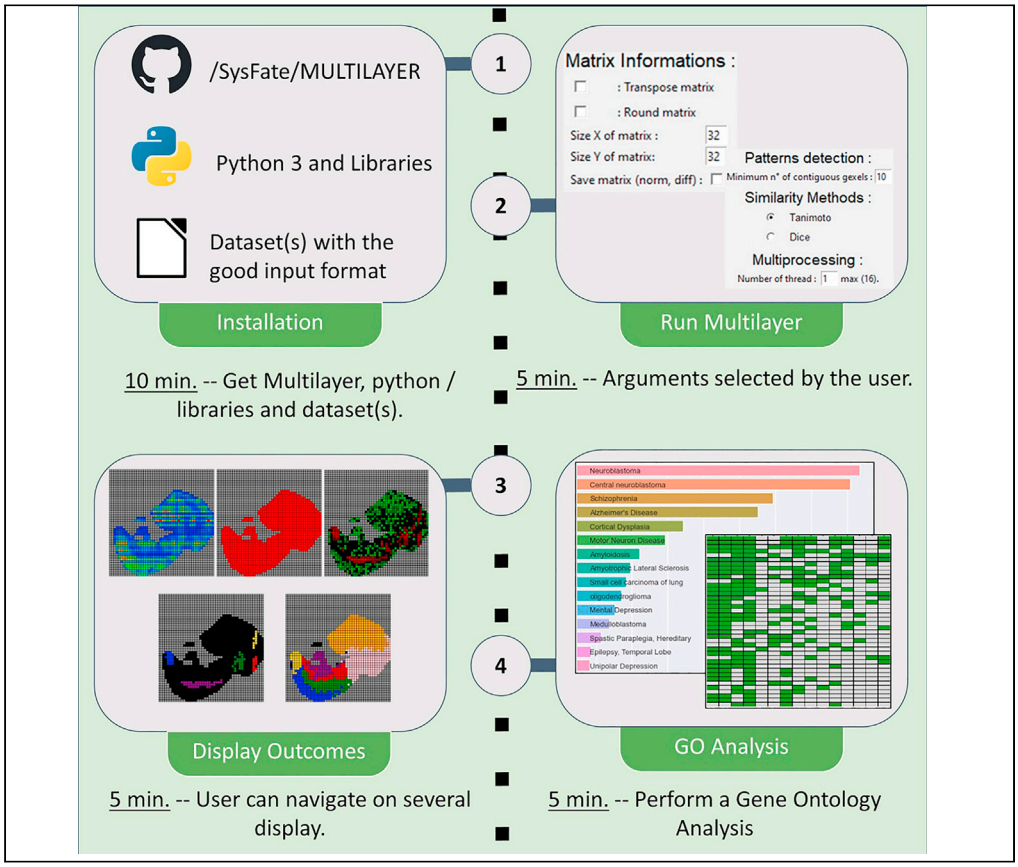
STAR Protocols

CellPress
OPEN ACCESS

Protocol

Protocol for using MULTILAYER to reveal molecular tissue substructures from digitized spatial transcriptomes



Spatially resolved transcriptomics (SrT) allows us to explore organ/tissue architecture from the angle of the gene programs involved in their molecular complexity. Here, we describe the use of MULTILAYER to reveal molecular tissue substructures from the analysis of localized transcriptomes (defined as gexels). MULTILAYER can process low- and high-resolution SrT data but also perform a comparative analysis within multiple SrT readouts.

Julien Moehlin, Aysis Koshy, François Stüder, Marco Antonio Mendoza-Parra

mmendoza@genoscope.cns.fr

Highlights

MULTILAYER reveals spatial transcriptomes (ST) into biologically relevant substructures

MULTILAYER provides a user-friendly environment for ST processing

MULTILAYER has been used on a variety of data, including high-resolution ST

A "Visium data convertor" and a "high-resolution data compressor" are also available

Moehlin et al., STAR Protocols *2*, 100823
December 17, 2021
https://doi.org/10.1016/j.xpro.2021.100823

STAR Protocols

CellPress
OPEN ACCESS

Protocol

Protocol for using MULTILAYER to reveal molecular tissue substructures from digitized spatial transcriptomes

Julien Moehlin,[1,2,3] Aysis Koshy,[1,2] François Stüder,[1] and Marco Antonio Mendoza-Parra[1,4,*]

[1]Génomique Métabolique, Genoscope, Institut François Jacob, CEA, CNRS, University of Evry, University Paris-Saclay, 91057 Évry, France

[2]These authors contributed equally

[3]Technical contact

[4]Lead contact

*Correspondence: mmendoza@genoscope.cns.fr
https://doi.org/10.1016/j.xpro.2021.100823

SUMMARY

Spatially resolved transcriptomics (SrT) allow researchers to explore organ/tissue architecture from the angle of the gene programs involved in their molecular complexity. Here, we describe the use of MULTILAYER to reveal molecular tissue substructures from the analysis of localized transcriptomes (defined as gexels). MULTILAYER can process low- and high-resolution SrT data but also perform comparative analyses within multiple SrT readouts.

For complete details on the use and execution of this protocol, please refer to Moehlin et al., 2021.

BEFORE YOU BEGIN

Timing: < 30 min

This section includes the minimal hardware requirements, the installation procedures, as well as the format of the files to be processed by MULTILAYER.

Hardware

Local-Memory: a minimum of 8GB required

Downloading MULTILAYER toolkit

1. The Multilayer toolkit can be downloaded from https://github.com/SysFate/MULTILAYER (Figure 1A). This is done by clicking in the green 'Code' tab and downloading the zip folder. It contains the data set used as examples in the paper, the gene ontology databases, the tutorial in a pdf format, the MULTILAYER tool, MULTILAYER compressor tool, "Visium Converter" tool and a ReadMe file.
2. After downloading the zip folder, extract all files.

Anaconda Python Platform

MULTILAYER is functional on all operating systems (Windows, Linux and Mac OSX) with Python 3. Python version 3.8 is recommended. A simplified strategy to run Python on any operating systems is to use Anaconda.

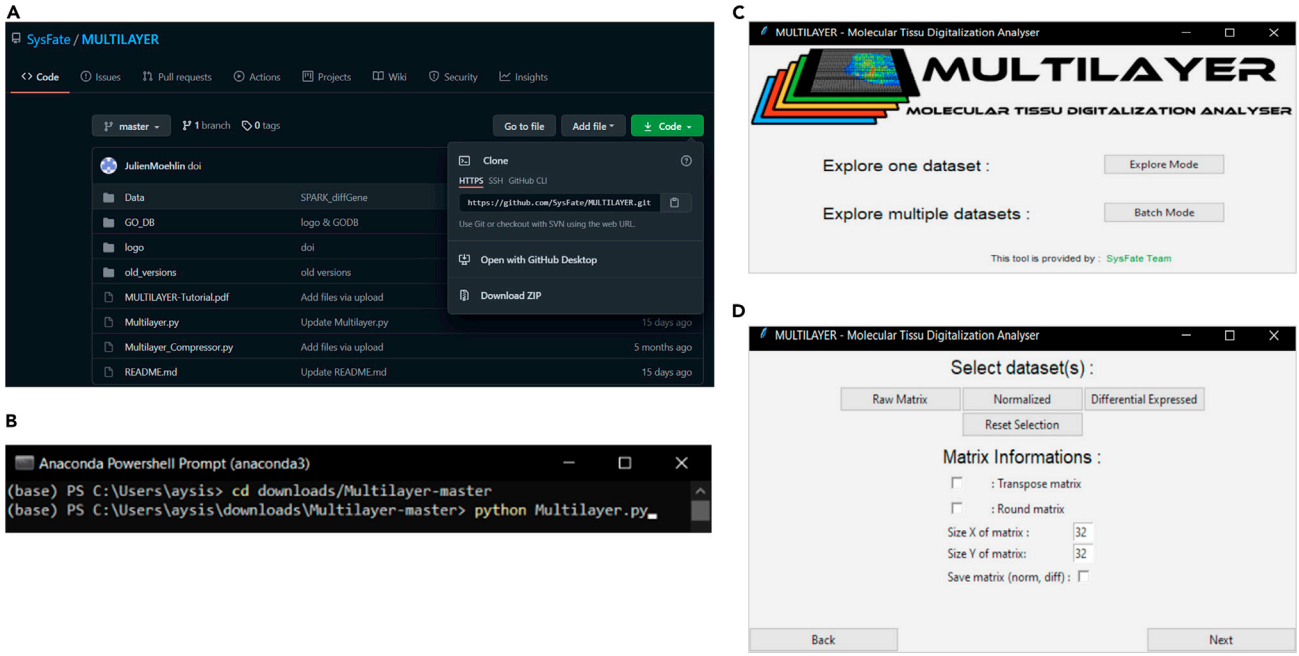


Figure 1. Download and Launching MULTILAYER

(A) The MULTILAYER toolkit is available at https://github.com/SysFate/MULTILAYER.

(B) The PowerShell prompt console with the written function to launch MULTILAYER.

(C) The MULTILAYER application home page providing either an "explore mode" for the analysis of a single SrT dataset, or the "batch mode" dedicated for processing multiple files.

(D) The MULTILAYER interface allowing to upload either raw, normalized or differentially-expressed SrT matrix data. In addition, users can transpose and/or round off the grid coordinates, as well as select the grid size if required.

3. Anaconda can be downloaded from https://www.anaconda.com/products/individual according to individual computer specifications.
4. Once Anaconda is installed, open up the Anaconda Navigator to launch the PowerShell Prompt console.
5. Before using MULTILAYER for the first time, it is necessary to install the following dependencies as following:

```
> pip install numpy

> pip install matplotlib

> pip install pandas

> pip install scipy

> pip install scikit-learn

> pip install seaborn

> pip install networkx

> pip install python-louvain

> pip install pillow
```

Launching MULTILAYER

6. To launch MULTILAYER, you have to first access the files by tracing the correct path from the PowerShell Prompt Console (Figure 1B). For e.g., If the Multilayer.py is located in the Multilayer-master folder, which is located in the downloads folder, type:

```
> cd downloads\Multilayer-master
```

7. To launch the Multilayer tool, type

```
> python3 Multilayer.py
```

or

```
> python Multilayer.py
```

(This will depend on which version of python has been installed.)

Data collection

Spatially resolved transcriptomics (SrT) promotes the transition of histological studies towards a data science context, notably by the generation of digitized maps of tissue architecture. MULTILAYER takes advantage of such a digitized view to reveal molecularly-defined spatial signatures, with the use of agglomerative clustering over contiguous gene expression elements, herein defined as gexels (in analogy to pixels retrieved on digital images). Considering the recent heterogeneity of platforms available for generating SrT data (e.g., use of DNA arrays (Rodriques et al., 2019; Ståhl et al., 2016) or microfluidic channels (Liu et al., 2020)), users might refer to the corresponding platforms for the preprocessing steps (spatial barcodes demultiplexing, read counts alignment, gene/transcript read counts association, etc). In all cases, the spatial transcriptome output requires to be in the format of a matrix composed of columns associated to spatial coordinates and rows to gene identifiers.

8. Depending on the SrT platform in use, such a spatial matrix is generated within the corresponding analytical pipeline or it can be obtained as following:
   a. In cases in which the structure of the matrix is transposed (i.e., columns are associated to gene identifiers and rows to spatial coordinates) or where the provided coordinates are not as round numbers; MULTILAYER provides options to handle these types of situations. Within the "Select dataset(s)" panel, user can opt to transpose and/or round off the entry matrix (Figure 1D). At this stage MULTILAYER is not able to recognize the format of the dataset in use.
   b. In cases in which the data is available as a three column dataframe (spatial coordinates, gene identifiers and read counts), the ad-hoc module called "MULTILAYER compressor", allows to convert it into the required matrix. In addition, this module is able to decrease the resolution of the available data by agglomerating contiguous gexels defined by a user-provided compression factor (refer to section "processing high-resolution SrT data").
   c. In cases in which the data is issued from the commercial solution "Visium", the ad-hoc module, called "VisiumConverter", allows to generate the required matrix. For this, users need to download the 10×Genomics proprietary tool spaceranger from here: https://support.10xgenomics.com/spatial-gene-expression/software/downloads/latest and proceed with the following Visium generated files:
      i. a matrix in h5 format.
      ii. the spatial imaging data containing the tissue_positions_list.csv file.

iii. the feature barcodes matrix files containing the corresponding features.tsv.gz file.

On the PowerShell Prompt Console, users can covert h5 matrix files generated by Visium to csv format as following:

```
> spaceranger mat2csv cellmatrix_HDF5.h5 out_file_matrix.csv
```

Where cellmatrix_HDF5.h5 is the h5 matrix files to convert to the csv format. For a detailed description of these steps, users can follow the spaceranger dedicated tutorial here: https://support.10xgenomics.com/spatial-gene-expression/software/pipelines/latest/output/matrices

The obtained csv matrix can be processed by our ad-hoc module, called "VisiumConverter" (available at https://github.com/SysFate/MULTILAYER) module as follows:

```
> python visiumConverter.py -m out_file_matrix.csv -p spatial/tissue_positions_list.csv -g
raw_feature_bc_matrix/features.tsv.gz -o matrix_multilayer.tsv –compressor
```

Where -m defines the matrix to be converted; -p: the tissue positions information, -g: the features and -o: the matrix compatible with MULTILAYER. Additionally, the "–compressor" option generates a three-column format (spatial coordinates / Gene ID / read counts) file, which can be processed by "MULTILAYER compressor" in case users need to decrease gexels density prior to MULTILAYER processing. Since new Visium DNA arrays present an interstitial printed spot, the matrix generated with our "VisiumConverter" tool tends to stretch the maps on the y-axis, without a real impact on the spatial information, as illustrated on Figure 2.

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Whole_mouse_embryo raw matrix (DBiT-Seq)	Liu et al., 2020	https://github.com/SysFate/MULTILAYER/tree/master/Data/Whole_mouse_embryo
High_resolution_brain raw matrices (Slide-seq)	Rodriques et al., 2019	https://github.com/SysFate/MULTILAYER/tree/master/Data/High_resolution_brain
Prostate_cancer raw matrices (ST)	Berglund et al., 2018	https://github.com/SysFate/MULTILAYER/tree/master/Data/Prostate_cancer
Software and algorithms		
MULTILAYER	Moehlin et al., 2021	https://github.com/SysFate/MULTILAYER
MULTILAYER compressor	Moehlin et al., 2021	https://github.com/SysFate/MULTILAYER
Visium Converter	This article	https://github.com/SysFate/MULTILAYER

STEP-BY-STEP METHOD DETAILS

Herein we describe Step-by-step methods for analyzing spatial read counts, from the loading of raw data, to its normalization, differential gene expression detection, gene co-expression pattern mapping and finally revealing the spatial communities corresponding to biologically relevant tissue substructures. To illustrate these various steps, we use as an example, the analysis of the Whole Mouse Embryo data generated by Liu and colleagues (Liu et al., 2020).

Open data on MULTILAYER

Timing: 10 min

1. Use Explore mode for this data. This is used for the analysis of one sample. (Figure 1C).

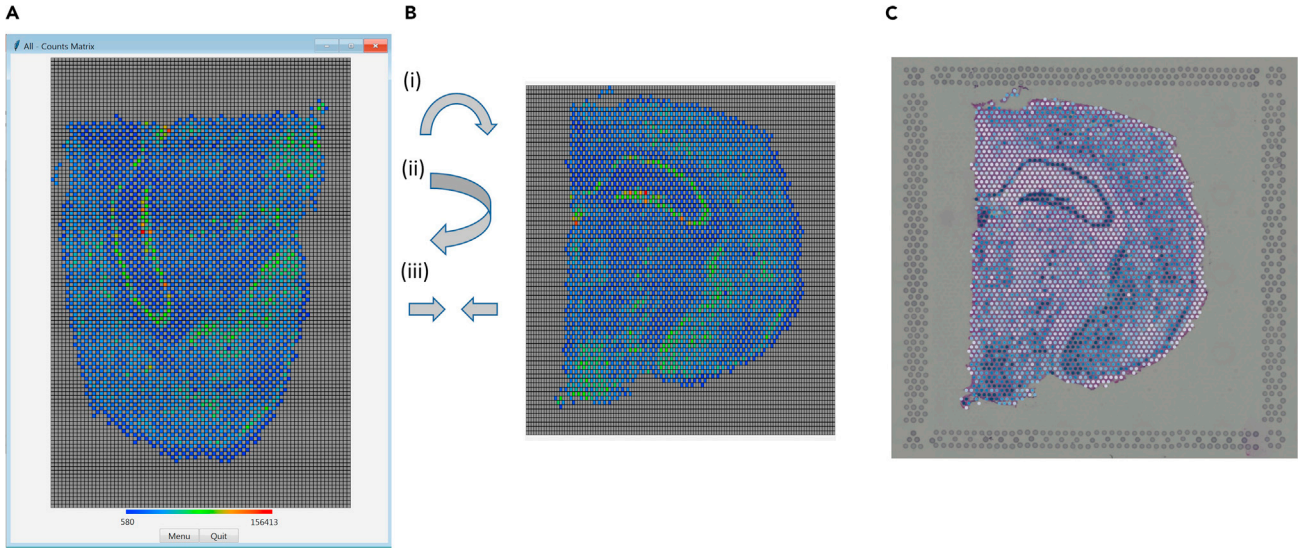


Figure 2. MULTILAYER display for datasets generated by the commercial platform Visium

A dataset corresponding to the analysis of a mouse brain section (Coronal) has been downloaded from 10×Genomics https://www.10xgenomics.com/resources/datasets/mouse-brain-section-coronal-1-standard. As indicated on "Data Collection" section, we have converted the corresponding cell matrix HDF5 (filtered) to csv with spaceranger; then we have used our "visiumConverter" module to generate a MULTILAYER compatible matrix.
(A) Raw count matrix corresponding to the coronal mouse brain section, displayed within the MULTILAYER platform. Notice that the image seems stretched on the y-axis. This is due to the fact that Visium DNA arrays present interstitial printed spots.
(B and C) By applying an image rotation of 90°; a flipping conversion and a stretch of the image on the new x-axis we can obtain a view identical to that provided by the commercial platform (displayed in C).

2. Provide your dataset by clicking Raw matrix and select the appropriate data; in this case Folder 'Multilayer-master', Folder 'Data', Folder 'Whole_mouse_embryo', File 'GSM4096262_0725cL.tsv'. (Figure 1D).

   Note: A .tsv file (table separated values) should be provided as raw matrix. The correct format is a matrix with gene names as rows, coordinates as columns (coordinates should be in the format 'XxY') (see data collection). It is necessary to enter a raw matrix. Optionally, if you provide your own normalized matrix and / or differential expressed matrix MULTILAYER will use them. If you provide only raw matrix, you can save those generated by MULTILAYER by checking the option for 'Save Matrix (norm, diff)'.

3. Select option to transpose matrix.

   Note: Transposing the dataset is useful when you have a matrix with gene names as columns and coordinates as rows (see data collection). When it is not performed (but required), the following error comes up – AttributeError: 'NoneType' object has no attribute 'columns'.

4. Optional setting to round off the matrix; default is unchecked.

   Note: MULTILAYER requires to round out spatial coordinates for the analysis. For this, the tool needs an integer as X & Y (coordinates) which will get rounded off. For eg: a given coordinate '3.36×13.94' will be converted to '3×14'.

5. Indicate maximum matrix size or leave as default option 32 × 32.
6. Click Next.
7. Indicate threshold for up and down regulated differential gene expression; default range is 1 to −1 (in log2).

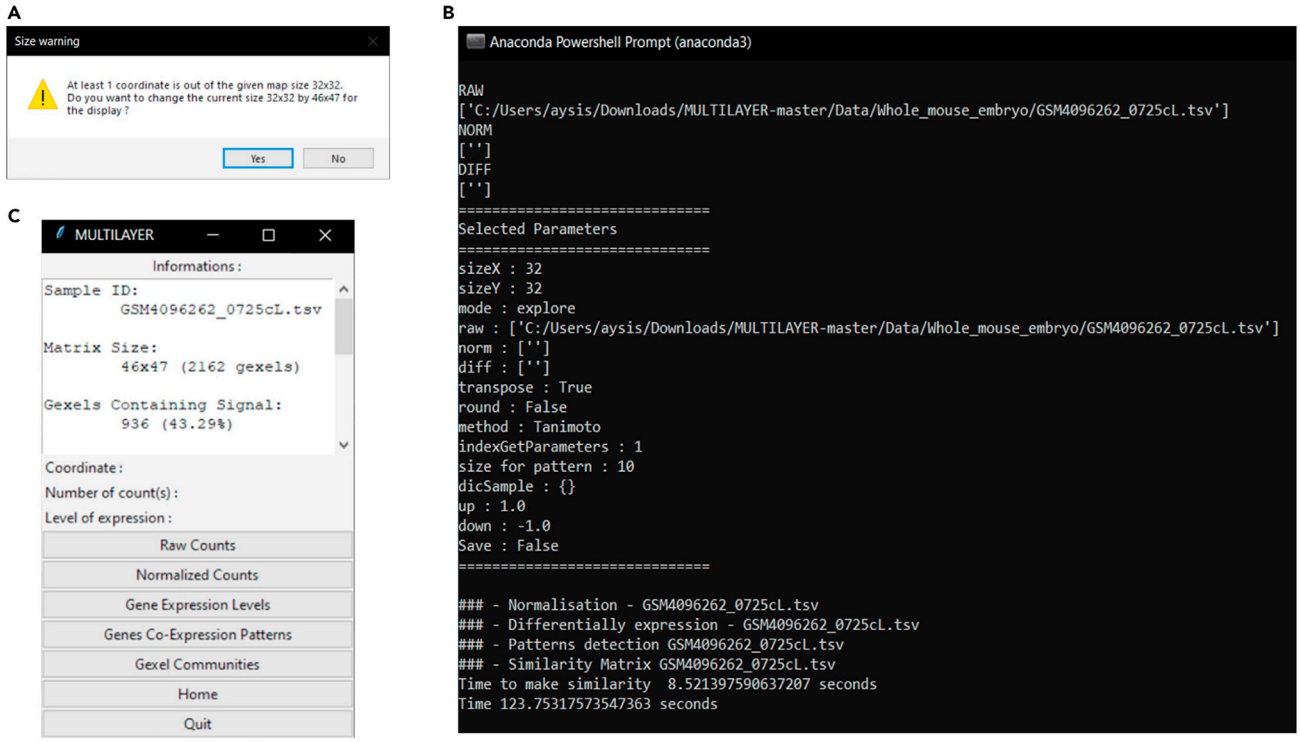


Figure 3. Processing whole_mouse_embryo data on MULTILAYER

(A) When the size matrix defined by the user doesn't comply with the grid size retrieved within the data, a size warning error is displayed. User can decide whether or not to accept the proposed grid size modification. This option does not impact the computing steps, but only the matrix display.

(B) As MULTILAYER processes the data, the Anaconda prompt interface displays a listing of the different steps taking place.

(C) Once it is finished, the processed data is available via the MULTILAYER interface.

8. Indicate minimum number of contiguous Gexels; default is 10.

   Note: In Pattern detection, the tool uses agglomerative clustering on gexels (https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html) to determine a contiguous pattern. It considers a pattern if the number of gexels is equal to or higher than this value. Default value is 10. This value is to be adapted to the size of the tissue, and the resolution of the data (number of total gexels within the matrix). An empirical evaluation of the minimal number of contiguous gexels is required on the basis of the pattern's detection performance (as illustrated in Figure 10).

9. Indicate similarity methods- Tanimoto or dice; default is Tanimoto.

   Note: Similarity methods calculate the similarity between patterns.

- Tanimoto: the similarity is calculated as A Union B / A Intersection B.

- Dice- the similarity is calculated as 2*(A Union B) / A Intersection B.

10. Indicate multiprocessing i.e., number of threads; default is 1.
11. Click next.
12. If the indicated matrix size is inaccurate and/or data is detected in points outside the matrix size suggested, a dialog box will appear indicating a size warning and ask if you want to open the file in a suggested size; here for e.g., 46 × 47. Click yes (go with MULTILAYER's recommendation) (Figure 3A).

13. The MULTILAYER interface allows at this stage to access to the various types of processed data (Raw counts, Normalized counts, Gene expression levels, Gene co-expression patterns, gexel communities) (Figures 3B and 3C).

Visualize raw and normalized data on MULTILAYER

Timing: <10 min

14. On the MULTILAYER interface, select the type of data you want to view (Raw counts, or Normalized counts).

Note: In contrast to Raw counts, the Normalized matrix provides a view in which all read counts were corrected by using a quantile normalization strategy (as described in (Moehlin et al., 2021)). Briefly, global read counts normalization is essential to correct technical bias affecting local read count levels across the tissue section of interest. Hence, all downstream steps (differential gene expression, gene co-expression patterns detection, spatial communities' detection) are issued from the normalized data. In cases in which the user might prefer to use their own normalization strategy, they can upload the normalized and corresponding raw matrix at the beginning of the analysis. In such a situation, MULTILAYER will not apply the quantile normalization correction, but instead it will use the provided normalized data for all downstream steps.

15. In both cases, the Menu button can be used to search for specific genes.
16. When a gene is selected, users can indicate the minimum or maximum read count threshold to enhance the display; as well as to convert it to log2 (Figure 4).

Visualize differential gene expression data on MULTILAYER

Timing: <10 min

17. On the MULTILAYER interface, select "Gene Expression levels" (Figure 4A).
18. By clicking the "Menu" button, a panel displaying a list of the upregulated genes ranked by the number of associated gexels is displayed (Figure 5A).
19. Similarly, the list of ranked downregulated genes is available by clicking on "Show repressed genes".
20. To visualize the gene expression location, users can either enter the gene name in the search panel, or double-click on the ranked list.
21. To enhance the display, users can modify the heatmap intervals (within the "threshold panels"), and also modify the differential expression threshold (threshold expression panels).

Visualize gene co-expression patterns

Timing: <10 min

22. On the MULTILAYER interface, select "Gene co-expression patterns" (Figure 4A).
23. A dialog box will appear requesting the minimum number of gexels to consider as part of a gene expression pattern (10 gexels by default) (Figure 5B).

Note: The minimum number of contiguous gexels for defining a gene expression pattern depends on the size of the tissue under study, its complexity as well as the resolution of the SrT data.

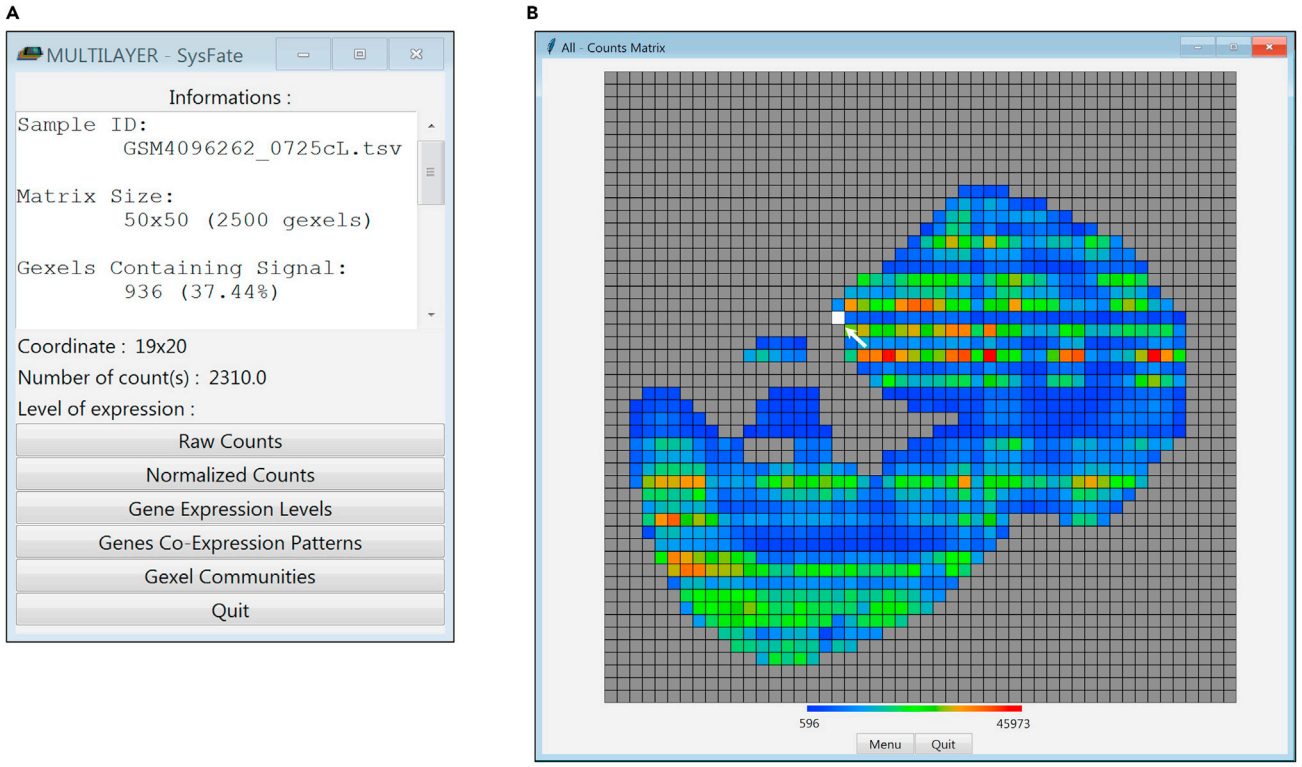


Figure 4. Visualizing a Whole Mouse Embryo data with MULTILAYER

(A) MULTILAYER interface providing general information, including the dataset sampel ID, matrix size, and the fraction of gexels presenting read counts.

(B) Raw count matrix displayed when selecting "raw counts" on the MULTILAYER interface. Each read count matrix (raw, Normalized, etc.), is accompanied by a menu bar dedicated to query for genes of interest. Read count levels per gexel are displayed as a heatmap. Hovering over a gexel on the image (indicated by white square and the pointer arrow) gives information of that particular area on the MULTILAYER interface (coordinate, number of counts or level of expression when visualizing the gene expression level matrix). The heatmap refers to the total read counts retrieved per gexel.

24. After clicking on "Run patterns", a matrix will be displayed. Like in the previous cases, the Menu button displays a panel in which gene patterns are ranked by the number of occupied gexels (Figure 5B).
25. To visualize the gene pattern location, users can either enter the gene name in the search panel, or double-click on the ranked list.
26. In some cases, multiple patterns (represented by different colors) can be displayed for a given gene (Figure 5B).

Note: Multiple patterns per gene potentially correspond to biologically relevant events, notably if they are separated by several gexels (excluding a technical discontinuity).

27. Select a pattern by clicking on one of their associated colored gexels.
28. Select the minimal gene co-expression similarity (in percent).
29. By clicking on "similarity", MULTILAYER will compute spatial gene co-expression signatures for the selected pattern. Gene co-expression is displayed by a spread of gexels from the selected pattern (corresponding to other gexels associated to co-expressed genes), as well as a heatmap corresponding to the gene co-expression similarity; i.e., from red meaning 100% of co-expression similarity (defining the location of the initial gene query), till dark blue corresponding to the least co-expression similarity level. (Figure 6A). A comprehensive list of the co-expressed genes

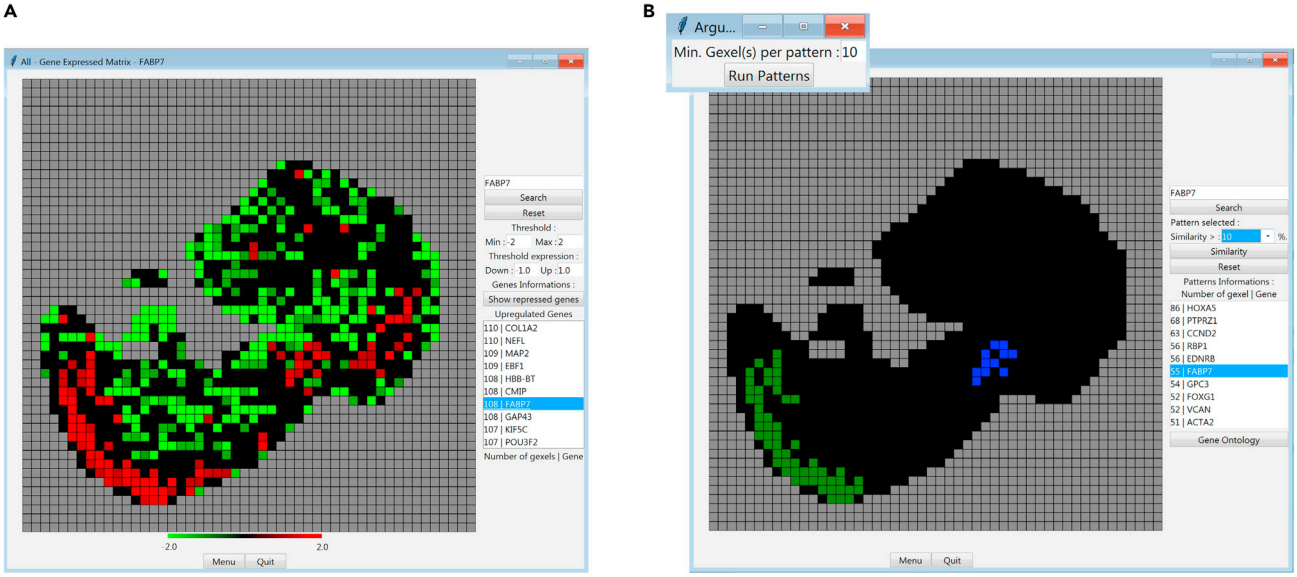


Figure 5. Revealing the differential gene expression within digitized tissue sections

(A) By selecting "Gene Expression levels" on the MULTILAYER interface, users can access a matrix view in which all precomputed differentially expressed genes are ranked on the basis of the number of their associated gexels. A double-click on the gene FABP7, retrieved on the ranked list, allows to visualize the over-expression (in red) or repressed (in green) signature of this gene over the whole tissue. The number at the left side of the Gene name (in this case "108") indicates the number of gexels within the whole tissue presenting a differential expression for that corresponding gene. The heatmap refers to the differential expression levels (log2) retrieved per gexel.

(B) when selecting "Gene co-expression patterns" on the MULTILAYER interface, a dialog box appears, allowing to select the minimal number of gexels per gene expression pattern. Like above, users can access a list of ranked genes, this time on the basis of the number of gexels per pattern. A double-click on the gene FABP7 reveals this time a cleaner view relative to (A), in which two distinct patterns (green and blue) complying with the minimal number of gexels per pattern criteria (at least 10) is displayed. For downstream analyses, users have to select a pattern of interest by clicking on one of their associated colored gexels.

and their corresponding similarity with the query gene is displayed within the Powershell prompt console (Figure 6B).

30. Click on "Gene Ontology" to access a dialog panel to select a GO terms database.
31. Select the adequate GO terms, then press "Run" to access to the Gene Ontology outcome, available either as a Barplot format or a gene vs GO terms heatmap (Figure 6C and 6D).

Note: In case users might require to interrogate a particular GO terms database; users can add the corresponding information to the folder "GO_DB" folder provided together with the MULTILAYER tool. For this purpose, users might adjust their GO terms database of interest to the format associated to the already available collections.

Visualize spatial communities defining tissue substructures

Timing: <10 min

While with the "Gene co-expression patterns" function we can query for a gene of interest and visualize the list of other genes that are spatially co-expressed, the "Gexel Communities" function allows to screen for all gene co-expression patterns over the whole tissue, and to then classify them within spatial communities. This is performed as follows:

32. When selecting "Gexel Communities" on the MULTILAYER interface a dialog box allows users to select the gene co-expression similarity threshold (in percentage), and also the possibility to perform 15 iterative runs and include the gene co-expression similarity levels as a "weight"

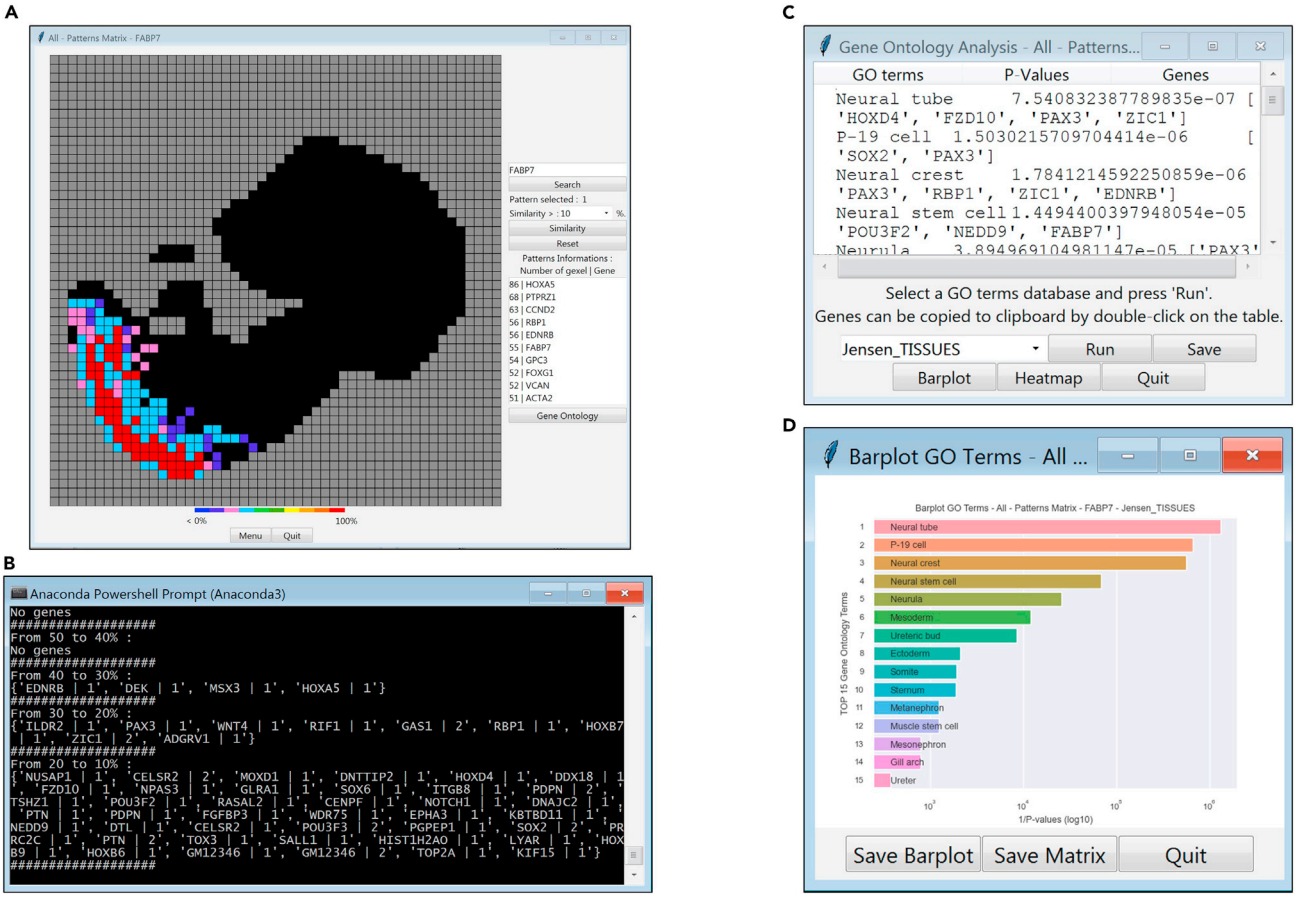


Figure 6. A functional analysis of co-expressed genes revealed by MULTILAYER

(A) Gene co-expression analysis for the spatial pattern associated to the gene FABP7. After selecting one of the FABP7 patterns shown in 5b (by clicking on one of the colored gexels associated to the pattern of interest), a gene co-expression similarity (minimal threshold: 10%) is computed. The illustrated matrix, displays gexels occupancy by spatial co-expressed genes with FABP7 (heatmap: similarity level with the query gene).

(B) A comprehensive list of the identified co-expressed genes is displayed within the PowerShell prompt.

(C) By clicking on the "Gene Ontology" button in (A), a new dialog panel allows to select for a pertinent GO terms database (here Jensen TISSUES). By pressing the Run button, MULTILAYER assesses the enrichment for the pertinent GO terms and displays their ranking on the basis of their confidence.

(D) Enriched GO terms associated to the co-expressed genes associated to FABP7 displayed as a Barplot.

parameter within the gene regulatory networks used for their spatial communities partitioning (Figure 7A).

Note: Louvain algorithm is performed over multiple iterations (default value: 15 times) and the most frequent outcomes are retained. This can be seen in the PowerShell prompt console.

33. By clicking the "Run Communities" button, MULTILAYER displays a matrix where the spatial communities are highlighted (color-coded gexels) (Figure 7A).

Note: When all communities are displayed, it's possible for communities to overlap. If there are overlaps between communities, the tool will color the gexel as part of the community with the highest similarity.

34. By choosing one community, the over-expressed genes colocalized in that region are listed (Figure 7B).

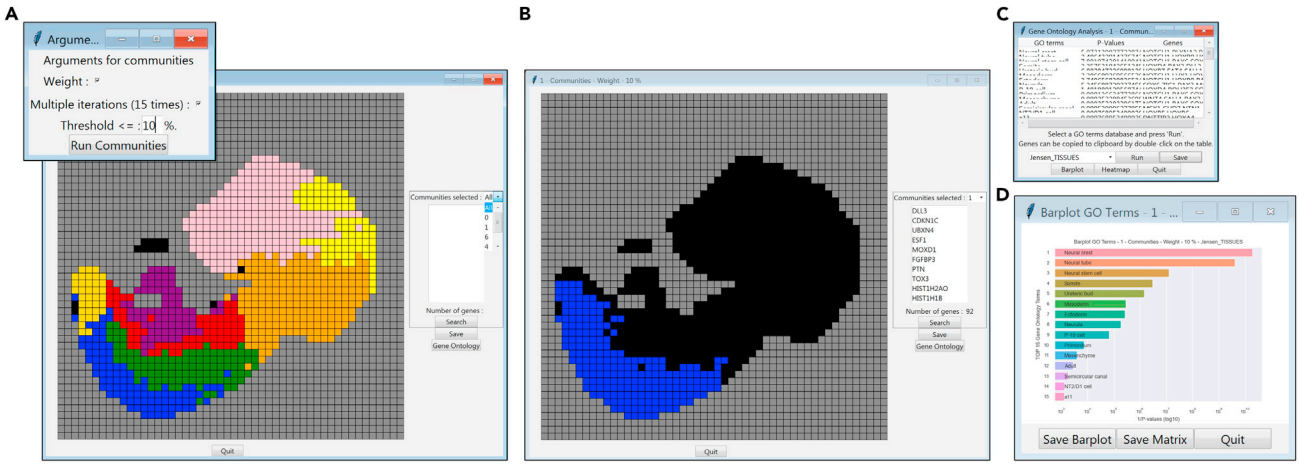


Figure 7. Revealing spatial tissue substructures by spatial communities partitioning

(A) Dialog panel defining the minimal gene co-expression similarity to detect gene co-expression Communities. By running this process, a matrix view in which all inferred gene co-expression communities within the tissue section is displayed. Note the presence of a scrolling bar (right side) by which users can access each of the communities.

(B) When selecting community "1", the list of the co-expressed genes and their corresponding spatial location is displayed.

(C) By clicking on Gene Ontology button in (B), a dialog panel allows to select for a pertinent GO terms database (here "Jensen_TISSUES").

(D) Barplot display of the enriched GO terms associated to the community "1" displayed in (B).

35. Similar to the gene expression analysis described on step 4, by clicking on the "Gene Ontology" button users can access the Gene ontology analysis (Figure 7C and 7D).

Note: A video illustrating the aforementioned steps is available here: https://www.youtube.com/watch?v=zByIdsUyJPg

Processing high-resolution SrT data

Step-by-step method of analyzing high resolution data. In this particular case, datasets issued from the technology Slide-seq (Rodriques et al., 2019) is used (Hippocampus Brain section). High resolution data are, in general, stored as a three-column format, instead of a gene/coordinate matrix. Prior to MULTILAYER processing, high-resolution files need to be condensed into larger gexels, thus reducing the number of gexels to display, but simultaneously gain on read counts per gexel.

Compress data with MULTILAYER compressor

Timing: 10 min

36. In order to compress the file, it is necessary to open the MULTILAYER compressor tool and also indicate the correct directory path to the data.
37. In order to access the compressor tool, type in the PowerShell prompt console:

```
> python3 Multilayer_compressor.py -i input.tsv -o output.tsv -cx 60 -cy 60
```

or

```
> python Multilayer_compressor.py -i input.tsv -o output.tsv -cx 60 -cy 60
```

For e.g.,

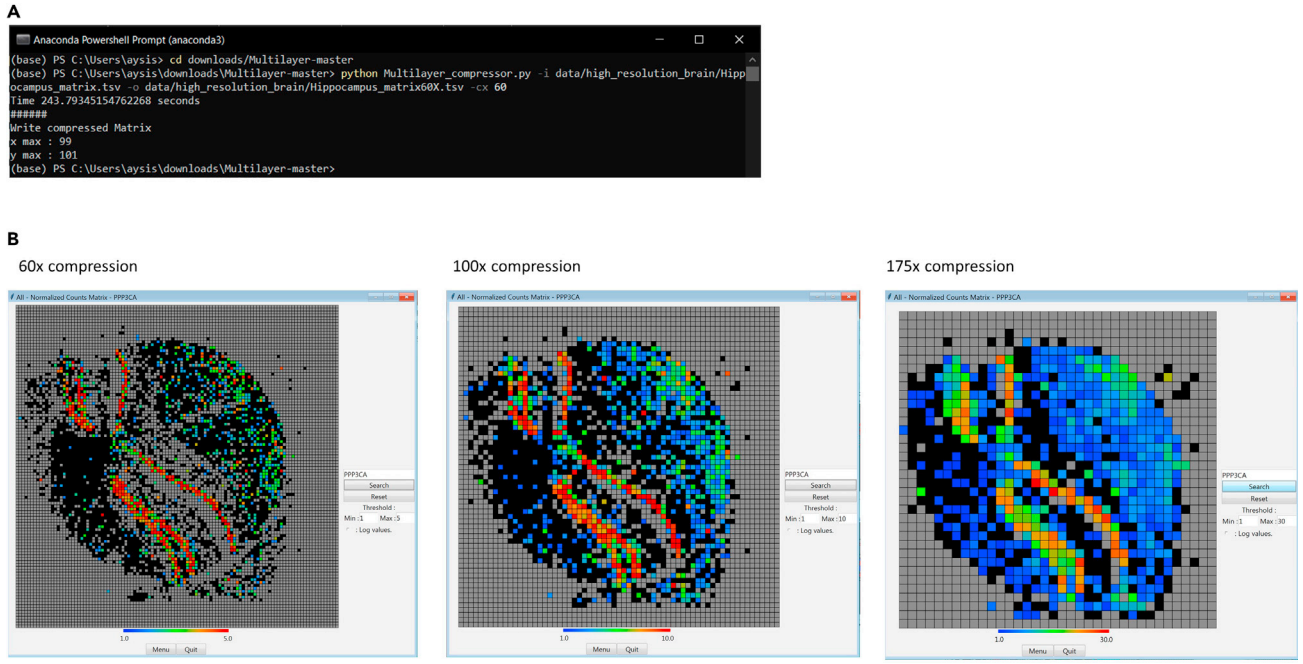


Figure 8. Processing high-resolution hippocampus SrT with MULTILAYER

(A) The PowerShell prompt console displaying the function to use the MULTILAYER compressor tool. The "-cx" parameter defined to "60", indicates a compression factor of 60 folds; i.e., the aggregation of proximal gexels such that the size of the matrix is reduced by 60 folds.

(B) High-resolution hippocampus spatial transcriptomics data compressed by a factor of 60×; 100× and 175× respectively and analyzed by MULTILAYER. Display of the spatial expression of the gene PPP3CA for these three compression factors demonstrates a conservation of its spatial signature. The heatmap refers to the total read counts retrieved per gexel.

```
> python Multilayer_compressor.py -i data/high_resolution_brain/Hippocampus_matrix.tsv
-o data/high_resolution_brain/Hippocampus_matrix60X.tsv -cx 60
```

(Figure 8A)

Note: The format of the input data has to be in 3 columns, where the first column correspond to barcode coordinates (XxY) with header 'bc'; the second column to genes with header 'gene'; and the third column to the gene counts with header 'count'.

The Multilayer_Compressor.py has several arguments: The function -i indicates the input matrix, -o indicates the output matrix, -cx and -cy indicates the compression factor (if -cx and -cy are identical only -cx needs to be defined). By typing

```
> python Multilayer_compressor.py -h
```

into the PowerShell console, you get access to the help section of the compressor.

38. The data will be stored in your output directory and can now be opened using the MULTILAYER tool.

Visualize compressed data on MULTILAYER

Timing: minutes

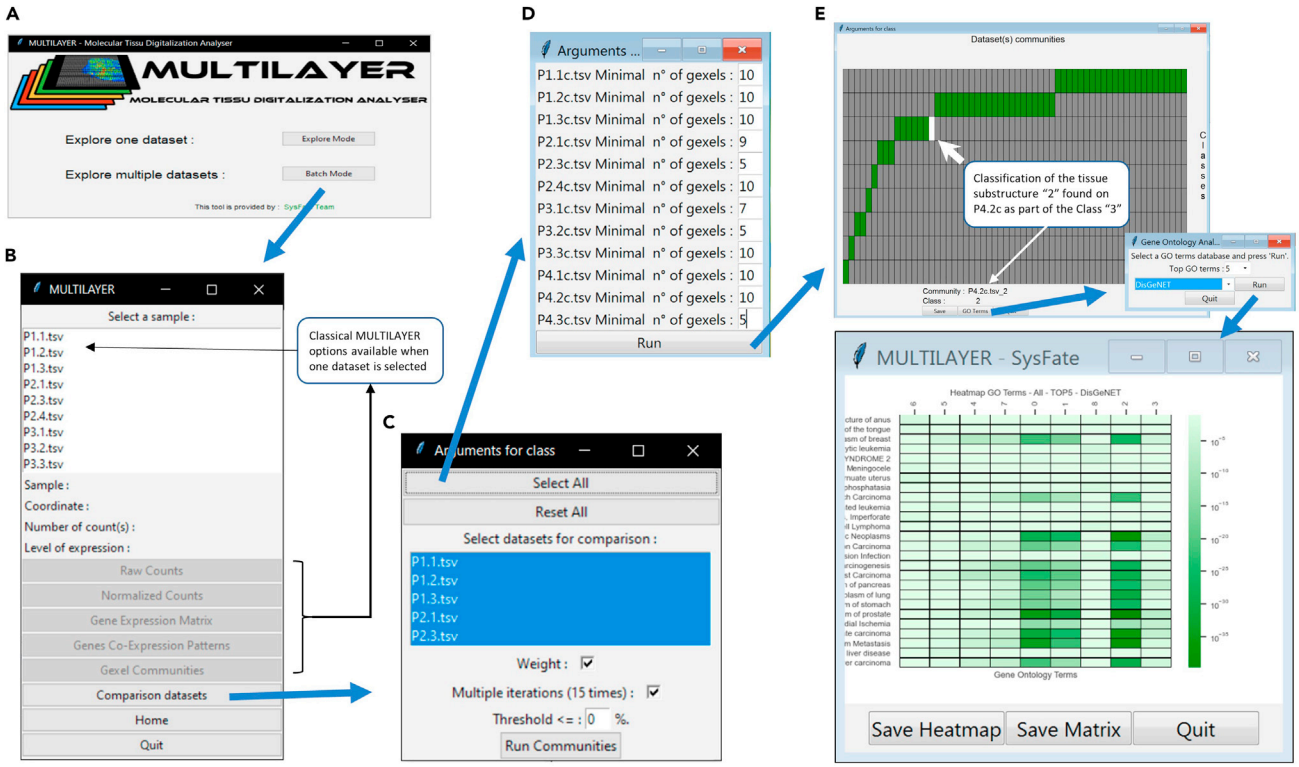


Figure 9. Use of the "batch-mode" available on MULTILAYER for processing several datasets at once

(A) MULTILAYER main interface allowing the selection of a folder containing all datasets to process.

(B) Once all datasets are processed (in this case a set of twelve prostate sections), MULTILAYER provides a similar interface than that used for the "explore mode", giving access to all different analytical panels when one dataset is selected. In addition, a "Comparison datasets" button is displayed.

(C) This "Comparison datasets" module, allows us to select the samples of interest for processing. Furthermore, it allows us to define the parameters for the spatial community's detection.

(D) For selected datasets, users can also define the minimal number of gexels for patterns detection.

(E) Panel illustrating the classification of all spatial communities detected per analyzed datasets. The interactive panel allows us to visualize the spatial tissue substructure ID and its corresponding class. This information can also be downloaded as a table for further analyses (e.g., graphs display). Furthermore, this panel allows us to access the gene ontology analysis as described for the "explore" mode.

39. After being compressed, datasets can be visualized on MULTILAYER tool as described above in the example of Whole Mouse Embryo (Figure 8B).

Note: A video illustrating the aforementioned steps is available here: https://www.youtube.com/watch?v=ww4x2aP6ENA

Processing multiple SrT data at once

Step-by-step method of using the batch-mode option. To illustrate this module, we have used twelve prostate cancer datasets generated by Berglund et al. (2018).

Open data on MULTILAYER

Timing: 10 min

40. Use Batch mode for this data (Figure 9A).

Note: The Batch mode is useful for analyzing multiple datasets from the same sample.

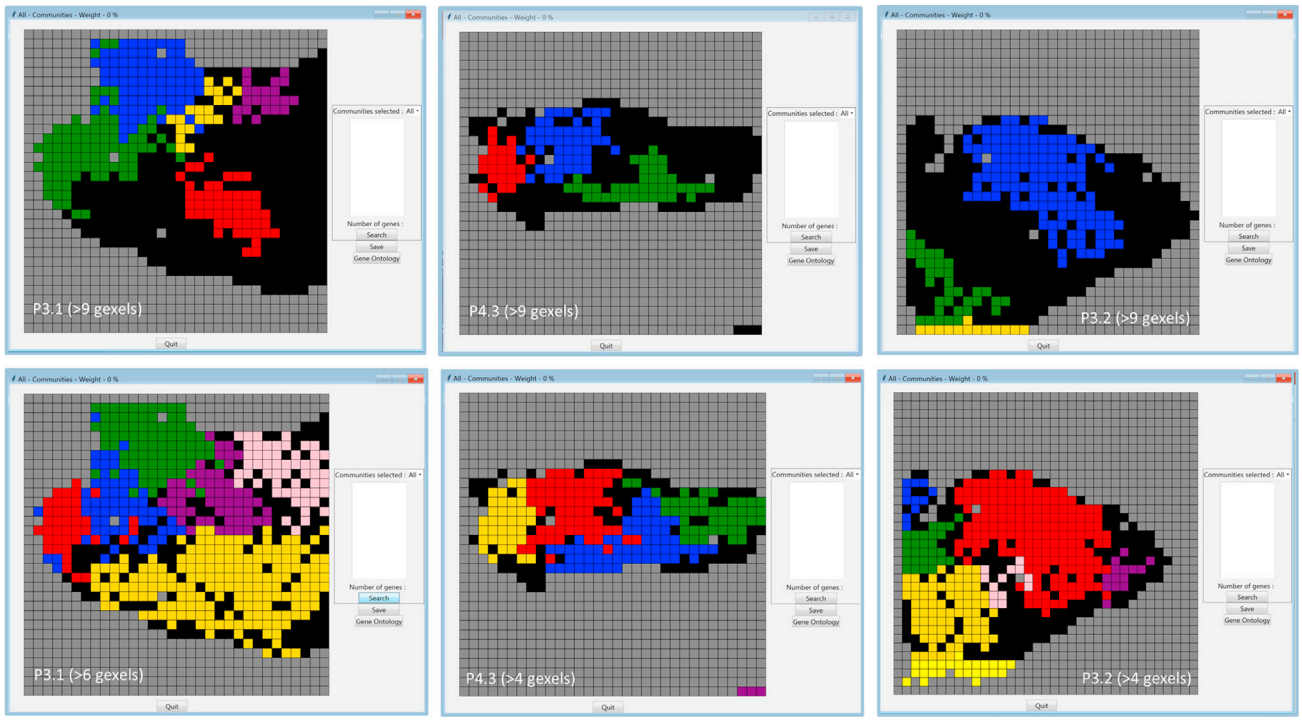


Figure 10. Spatial communities revealed over prostate sections as computed with different minimal number of gexels for pattern detection

Three of the twelve prostate sections analyzed in Figure 9, are displayed (P3.1, P4.3 and P3.2). When using a minimal number of gexels of 10 (i.e., >9) for pattern detection, several gexels within the tissues are not associated to any of the inferred spatial communities (black-colored gexels; Top panels). By decreasing the minimal number of gexels criteria, it is possible to maximize the number of gexels associated to any of the inferred spatial communities (Bottom panels).

41. Provide your dataset by clicking Raw matrix and select the appropriate folder where all datasets to process are available (in this case the "Prostate Cancer" datasets). All .tsv files in the selected folder will be considered as the input matrix.
42. All values are kept as default values.
43. Click next.

Note: Since multiple files are being processed, this step takes ~6 min.

44. You are now able to visualize digitally, the raw counts, normalized counts etc. for individual samples as in the example for Whole Mouse Embryo or a comparison dataset for several samples at once on the basis of their associated spatial communities (Figure 9B).

Compare multiple spatial communities on MULTILAYER

Timing: minutes

45. On the Multilayer interface, select "comparison datasets" and select all the files to be compared ("Select All" button) or make your own selection (Figure 9C).
46. Default values for community's detection are kept constant for all datasets (weight is checked, multiple iterations (15 times) is checked, threshold is 0%). Furthermore, users can adjust the minimal number of gexels for patterns detection (Figure 9D). This option is of major interest in case users have previously identified the minimal number of gexels (Figure 10).

Note: Users can empirically evaluate the minimal number of contiguous gexels for the analysis, by evaluating the fraction of gexels that are associated to at least one of the inferred spatial communities. As illustrated in Figure 10, by decreasing the minimal number of gexels from 10 to 5, the spatial communities retrieved within the displayed prostate tissue sections managed to incorporate most of the gexels of the tissue within the inferred patterns.

47. Run communities and save the .tsv file.

Note: When saving the files for the comparison analysis, the tool saves 2 files (ex. save.tsv, save_filter.tsv).

48. An Interactive heatmap comes up with all the dataset communities. The x-axis represents all the dataset communities, and the y-axis represents all the classes. A class is a group of 1 or several communities. (Figure 9E).
49. For gene ontology analysis, click GO terms.
50. Select the number of GO terms and database. In this example, top GO terms =5, database= DisGeNET.
51. This gives a heatmap of the GO terms that can be saved as a plot or matrix (Figure 9E).

EXPECTED OUTCOMES

MULTILAYER provides an intuitive platform for processing SrT data issued from a variety of technological platforms (e.g., DNA-array based methodologies (Rodriques et al., 2019; Ståhl et al., 2016), microfluidic channels (Liu et al., 2020)). Furthermore, thanks to additional modules, MULTILAYER can also analyze high-resolution data (MULTILAYER compressor module), as well as those issued from the commercial platform "Visium" (Visium converter module).

As outcome, MULTILAYER processes SrT maps as a digital image, in which users can visualize gene expression signatures, reveal gexel patterns resulting from agglomerative clustering, and even infer biologically relevant tissue substructures.

Considering the future needs for processing multiple datasets on a comparative manner, MULTILAYER also provides a "batch-mode", allowing to process multiple samples at once, as well as compare them, notably by classifying the assessed tissue substructures per dataset into groups that might reveal common biological functions.

LIMITATIONS

The current version of MULTILAYER reveals spatial gene co-expression signatures, but it is not able to infer their related co-regulatory relationships. Furthermore, it is not yet adapted for integrating single-cell transcriptome data. MULTILAYER is currently in progress to be updated with these two aspects. From a technical angle, MULTILAYER is limited with its visual functionalities for datasets represented within matrices > 250 gexels per side. In such a situation, the use of the MULTILAYER compressor module is strongly recommended.

TROUBLESHOOTING

The most common problem when running MULTILAYER is the format of the input dataset.

Problem 1

MULTILAYER fails to process the input dataset because the matrix has spatial coordinates as rows and genes as columns.

Potential solution

Use the "Transpose" option at the time of the data upload.

Problem 2
MULTILAYER fails to process the input dataset because the matrix has decimal values within the gene counts.

Potential solution
Use the "Round" option at the time of the data upload.

Problem 3
MULTILAYER fails to process the input dataset despite taking care of transposing the matrix or rounding off the read counts.

Potential solution
The provided data might miss values for the read counts. Make sure to fill empty cells or replace "NAs" by a value ("0" by default).

Problem 4
MULTILAYER fails to process the input dataset, notably because it has utilized all the available RAM of the computer.

Potential solution
High-resolution data might require large amounts of RAM for processing. Either use the "MULTILAYER Compressor" module to reduce the matrix size, or increase the RAM of your system.

Problem 5
Multilayer fails to process when the gene identifiers are not unique within the provided dataset.

Potential solution
To avoid this issue, we recommend using unique identifiers, for instance corresponding to transcripts (e.g., EnsemblID). Since the gene ontology analyses requires access to the gene symbol ID, an optimal solution is to concatenate both gene symbol and transcript ID into a single identifier separated by a "_" (e.g., 'geneSymbol_ensemblID'). MULTILAYER is able to separate such a gene ID structure to recognize the first part of the identifier as the corresponding gene symbol to be used for the gene ontology processing when required.

Problem 6
MULTILAYER processing does not provide gene patterns and/or spatial communities.

Potential solution
The processed data does not have sufficient differential expression data, or the minimal number of contiguous gexels is not satisfied. To evaluate such potential reasons, decrease the minimal number of contiguous gexels and/or decrease the differential gene expression threshold.

Problem 7
The predicted spatial communities do not cover the whole surface of the tissue (i.e., a significant part of the tissue displays black gexels, not associated with any of the predicted communities).

Potential solution
The minimal number of contiguous gexels for patterns detection is too high. Evaluate multiple lower thresholds to identify empirically the least number of contiguous gexels that allows to predict spatial communities within the whole (maximum) surface of the tissue.

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Marco Antonio Mendoza-Parra (mmendoza@genoscope.cns.fr).

Materials availability

This study did not generate new unique reagents.

Data and code availability

The datasets and code used during this study are available at https://github.com/SysFate/MULTILAYER

ACKNOWLEDGMENTS

This work was supported by the institutional bodies CEA, CNRS, and Université d'Evry-Val d'Essonne. J.M. was supported by Genopole Thematic Incentive Actions funding (ATIGE-2017); A.K. by the "Fondation pour la Recherche Medicale" (FRM; funding ALZ°201912009904); and F.S. by the funding 2OI9-L22 from the Institut National du Cancer (INCa).

AUTHOR CONTRIBUTIONS

Conceptualization, J.M. and M.A.M.-P.; protocol elaboration, A.K., J.M., and M.A.M.-P.; software development, J.M.; "Visium converter" development, F.S.; scientific evaluation, M.A.M.-P.; writing, review, and editing, J.M., A.K., and M.A.M.-P.; funding acquisition, M.A.M.-P.

DECLARATION OF INTERESTS

The authors declare no competing interests.

REFERENCES

Berglund, E., Maaskola, J., Schultz, N., Friedrich, S., Marklund, M., Bergenstråhle, J., Tarish, F., Tanoglidi, A., Vickovic, S., Larsson, L., et al. (2018). Spatial maps of prostate cancer transcriptomes reveal an unexplored landscape of heterogeneity. Nat. Commun. *9*, 2419.

Liu, Y., Yang, M., Deng, Y., Su, G., Enninful, A., Guo, C.C., Tebaldi, T., Zhang, D., Kim, D., Bai, Z., et al. (2020). High-spatial-resolution multi-omics sequencing via deterministic barcoding in tissue. Cell *183*, 1665–1681.e18.

Moehlin, J., Mollet, B., Colombo, B.M., and Mendoza-Parra, M.A. (2021). Inferring biologically relevant molecular tissue substructures by agglomerative clustering of digitized spatial transcriptomes with multilayer. Cell Systems *21*, 694–705. https://doi.org/10.1016/j.cels.2021.04.008.

Rodriques, S.G., Stickels, R.R., Goeva, A., Martin, C.A., Murray, E., Vanderburg, C.R., Welch, J., Chen, L.M., Chen, F., and Macosko, E.Z. (2019). Slide-seq: a scalable technology for measuring genome-wide expression at high spatial resolution. Science *363*, 1463.

Ståhl, P.L., Salmén, F., Vickovic, S., Lundmark, A., Navarro, J.F., Magnusson, J., Giacomello, S., Asp, M., Westholm, J.O., Huss, M., et al. (2016). Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. Science *353*, 78–82.