Traject3d User Manual

Contents

1 Before starting

1.1 Folder Structure

The computer you will be running the KNIME analysis from should have either access to, or a local, copy of the directory containing your phase images, and output from CellProfiler. The Experiment Key and directory structure template (SampleData) can be found in the Traject3d GitHub repository.

For reference, this directory should have the following structure:

1. DatasetName
   - Experiment___1
     - Data
     - Experiment Key
     - Phase
     - PhaseGrayCystOutlines
   - Experiment___n
   - Output

Experimental/biological replicate folders are named "Experiment___n", where n corresponds to the replicate number (please note double underscore).

Prior to analysis using this KNIME pipeline, the above folders should contain:

- *Data:* Your data as a CSV file named “PhaseGrayCysts”, in which each column is either metadata or a measured morphological feature, and each row corresponds to a single object, from a single image. This file is generated by CellProfiler and automatically added to the Data subdirectory.
- *Experiment Key:* A Microsoft Excel spreadsheet containing relevant metadata including, sample and experiment names, and well IDs. In the example spreadsheet provided ensure the following columns are completed in the “Experiment Key” sheet (tab); “Metadata_ExperimentName”, “Metadata_Plate”, “Metadata_Well”, “Condition”. The remaining columns in this sheet will be auto-populated using this information. The other sheets in the Excel workbook will also be automatically filled in. The contents of “Metadata_ExperimentName” should be unique to each experimental/biological replicate, and “Condition” naming (e.g. cell line and/or manipulation) should be consistent (in terms of case, spelling, symbols or spaces used) between, each experimental/biological and technical replicate.
- *Phase:* Phase images, in RGB (3-channel) TIFF format. Filenames should be in the following format: ExperimentName_Plate_Well_Site_Date_Time (please see the sample data provided for an example). Image and filename formatting is typically set at the point of export from the imaging system utilised.
- *PhaseGrayCystOutlines:* Binary (white on black background) images of the object outlines, as PNGs. There should be one, equivalently named, outline image for each image in the “Phase” folder. These images are generated by CellProfiler and automatically added to the PhaseGrayCystOutlines subdirectory.

The default output directory for KNIME analysis results will be the “Output” subdirectory of this folder.

1.2 KNIME installation

We have used KNIME v4.0.2, which can be downloaded from here. Newer versions of the KNIME software have not been tested by us with the Traject3d pipeline.

1.2.1 Notes for KNIME workflow

The KNIME software base installation doesn't come with all possible functionality. As such, additional functionality needs to be activated by the user through installation of extensions to the base software. The KNIME extensions used by the Traject3d pipeline are listed below. These can be installed from within the KNIME software, by following File → Install KNIME Extensions... in the toolbar at the top of the user interface.

Required KNIME version and extensions:

Name	Version
KNIME Analytics Platform	4.0.2.v201909300912
KNIME Interactive R Statistics Integration	4.0.1.v201908131226
KNIME Core	4.0.2.v201909300912
KNIME Quick Forms	4.0.2.v201909242005
KNIME Math Expression (JEP)	4.0.2.v201909242005
KNIME Python Integration	4.0.0.v201906241606
KNIME Image Processing	1.8.0.201911140609
KNIME SVG Support	4.0.1.v201908131226
KNIME Virtual Nodes	4.0.0.v201905311239
KNIME Distance Matrix	4.0.2.v201909260824
KNIME Data Generation	4.0.0.v201905311239
KNIME File Handling Nodes	4.0.1.v201908131226
Vernalis KNIME Nodes	1.24.2.v201911141223
KNIME Excel Support	4.0.1.v201908131226
KNIME HCS Tools	4.0.0.v201906200802

1.3 R Integration

The Traject3d KNIME workflow uses an R integration for some steps of the analysis. The workflow was built using R version 4.2.0, which is available here: Windows, macOS.

Windows: First install Rtools by following the instructions in sections 2b-c of this guide. Then install Rserve v1.8-10 by opening the R application and running the following line of code:

```
install.packages("Rserve", repos = "https://cran.r-project.org", type="win.binary")
```

macOS: Instructions for setting up the R integration can be found here.

All users: To finish setting up the integration, in the KNIME Analytics Platform user interface, go to File → Preferences. From the list on the left, under "KNIME", select "R". In "Path to R Home" enter the path indicating the location where R is installed on your computer. This path can be found from within R by typing R.home(). Set the "Rserve receiving buffer size limit" to 0.

The R scripts within the KNIME workflow should automatically download and install any missing R packages that are required for the analysis. A dialogue box may come up asking you to select a CRAN mirror - if so, select whichever location is local to you. If, for any reason, the automatic download and installation is unsuccessful, any missing packages can be installed manually from within the R application (externally to KNIME).

With the exception of cytofkit and Rserve (manual installation is required, as described above), all other packages can be installed manually within R by running the command:

```
install.packages(PACKAGE_NAME)
```

Manual installation of cytofkit2 requires running the following code within R:

```r
if(!require(devtools)) install.packages('devtools')
devtools::install_github("JinmiaoChenLab/cytofkit2", dependencies=TRUE)
```

Traject3d utilises the following packages:

Package	Version
ClassDiscovery	3.4.0
circlize	0.4.14
cytofkit2	0.99.80
DescTools	0.99.44
factoextra	1.0.7
FactoMineR	2.4
fields	13.3
ggdendro	0.1.23
ggimage	0.3.1
ggnewscale	0.4.7
ggplot2	3.3.6
ggseqlogo	0.1
imputeTS	3.2
MASS	7.3.51.4
pheatmap	1.0.12
plyr	1.8.7
png	0.1-7
RColorBrewer	1.1-3
reshape2	1.4.4
Rserve	1.8-10
Rtsne	0.16
vcd	1.4-9
viridis	0.6.2

1.4 Python Integration

The KNIME workflow requires Python in order to run the GeoSketch algorithm for subsampling data.

First, install Anaconda, which can be found here. Preconfigured Python environments are provided on the Traject3d Github repository. Download the provided environments, and import them into Anaconda. This can be easily done using the Anaconda Navigator application (which will have been installed in the earlier step), instructions for which can be found here under "Importing an environment". Then follow the instructions under "Option 1" here, and select the provided environments within your KNIME Python preferences.

Package	Version
GeoSketch	1.0

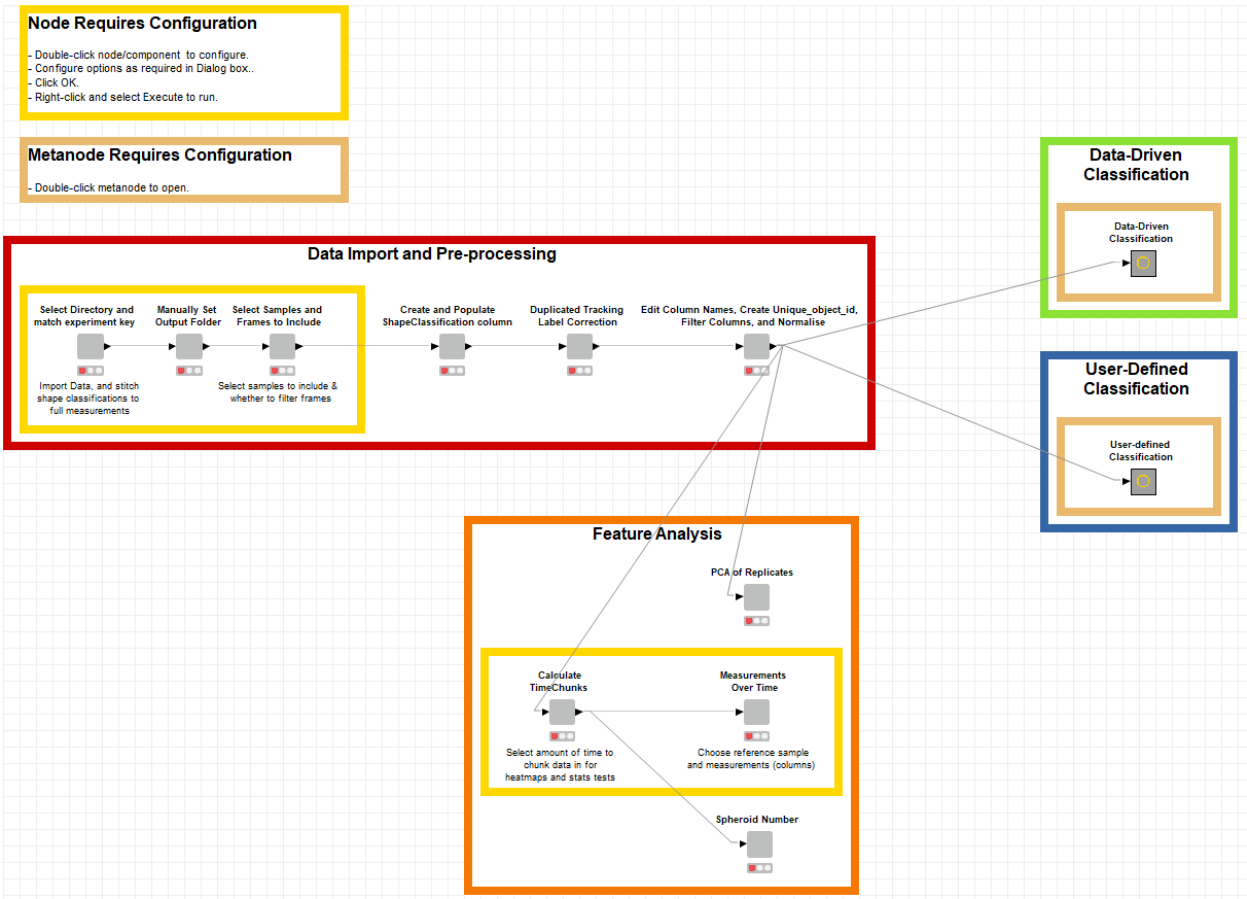
2 KNIME

2.1 Using KNIME

KNIME Analytics Platform is free, open-source software that can be used to design modular data science workflows using a graphical user interface, without any need for coding. It also allows integration with other tools such as R and Python in a single workflow. Introductory instructions for using KNIME can be found here.

Our workflow for Traject3d (Traject3d_v1.knwf) can be found on the Traject3d Github repository. In brief, to load the workflow into the KNIME Analytics Platform: in the main toolbar at the top of the KNIME user interface select File → Import KNIME Workflow and browse to select the workflow file from wherever in the local file directory it was downloaded to. Click "Finish" to import the selection. The Traject3d_v1 workflow should then open in the Local Workspace, viewed in the KNIME Explorer pane on the left of the user interface. Double-click on the workflow (Traject3d_v1) within this pane to open it in the Workflow Editor.

2.2 Workflow Structure



The KNIME workflow is comprised of four parts, each addressed in one section of this manual:

Data Import and Pre-processing: Essential for all subsequent parts of the workflow. This part allows import of data and merging with experimental keys, filtering to exclude any time intervals or treatment groups if required, parsing of user-defined classifications, correction of duplicated tracking labels, and data normalisation.

Feature Analysis: Generates heatmaps of size, shape and movement features over time with statistical comparison between control and treatment groups. Can also generate counts of objects analysed and perform PCA of experimental/biological and technical replicates.

User-Defined Classification: Generates heatmaps of user-defined states over time with statistical comparison between control and treatment groups. Can also calculate means of all measured features and generate representative outlines for each state. Phase images are also overlaid with outlines colour coded by user-defined classification. Dimensionality reduction using t-SNE can be performed for data visualisation.

Data-Driven Classification: Dataset may be subsampled to identify data-driven states, from which trajectories

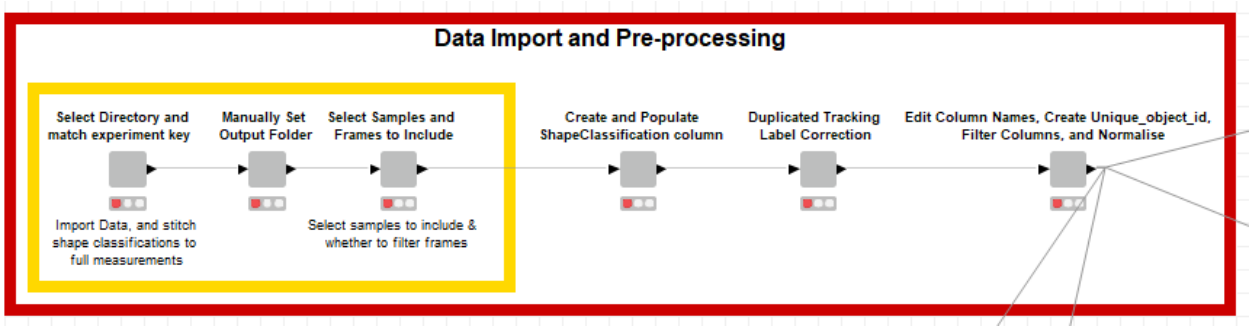
are subsequently identified. Both are quantified as heatmaps. Trajectory motifs and transition plots are created to summarise trajectories. Representative object outlines are identified to represent each data-driven state classification, and representative tracked objects are identified to represent each trajectory. Phase images overlaid with outlines colour coded by state classification are also generated.

A KNIME workflow is comprised of three types of building block: nodes, metanodes, and components. The most basic of these are nodes, each of which represents a function being performed on the data. For simplicity, we have encapsulated nodes into components. These act as a wrapper, breaking the workflow into logical steps in the analysis, and condensing any user input required by the respective section into a dialog box. For example, each grey element within the "Data Import and Pre-processing" (red) section in the image above is a component. We have then used metanodes as an additional wrapper, enabling sections of the workflow (components) to be collapsed and hidden until required by the user. For example, two metanodes in the workflow are "User-Defined Classification" and "Data-Driven Classification", shown in blue and green, respectively, in the image above. We have indicated in each section of this user manual and, where necessary, on the pipeline itself (yellow boxes) which nodes/components require configuration by the user, usually through the selection of options in dialog boxes.

Tips for those new to KNIME:

- Double-click on a node/metanode/component. If the element is a node/component, a dialog box will open for configuration. Configure as appropriate, and click OK to apply the changes. If no configuration is required, the dialog box will be empty. Conversely, if the element is a metanode, double-clicking will cause it to open revealing the nodes/components within.
- After configuration, right-click on a node/component and select "Execute" to run it.
- To open a component, right-click on it and select Component → Open. This will reveal the contents (nodes) of the component within a new tab in the KNIME user interface.
- Nodes that fail to run will be indicated by a red X to aid troubleshooting.
- Error messages are printed in the Console panel in the bottom right of the KNIME user interface to aid troubleshooting.
- The panel on the right hand side of the KNIME user interface provides additional information about a selected node/metanode/component.
- After a node/component has run successfully, right-click and select the bottom-most option (usually called "Port 1" for components) to view the processed data at this stage. This can aid visual inspection for quality control and troubleshooting if there are any problems with the analysis.

3 Data import and pre-processing



3.1 Select Directory and match experiment key

Configuration required.

This component reads in the CSV files output by CellProfiler and combines them with their respective experiment keys.

Double-click to configure the node and select Browse to set the root directory containing all your experimental/biological replicate folders in the format described in Section 1.1 Folder Structure. There is also the option to search for and replace a String value in user-specified columns. Right-click and select "Execute" to run the component.

3.2 Manually Set Output Folder

Configuration required.

This component can override the default output directory.

Double-click to configure the component. If you wish to override the default output directory ("Output" subdirectory of your root directory), select "Yes", and "Browse" choose an alternative directory for output. Otherwise, select "No". Right-click and select "Execute" to run the component.

3.3 Select Samples and Frames to Include

Configuration required.

This component applies sample, frame, and lifetime filtering of objects in the dataset.

Double-click to configure the component.

To filter samples: Samples to be retained for analysis should be in the green "Include" box, and others in the red "Exclude" box.

To filter by frame number (time point): In order to include all frames, set "Filter by timepoints" to "No". Otherwise, set it to "Yes", and set the "Start Frame Number" and "Stop Frame Number" values to the lower- and upper-bound frames of interest, respectively.

To filter by lifetime: Set "Filter objects by lifetime" to "Yes", and set the maximum and minimum lifetime values, where lifetime is the number of frames an object was tracked for by CellProfiler.

Right-click and select "Execute" to run the component.

3.4 Create and Populate ShapeClassification column

No configuration required. Just right-click and select "Execute".

If user-defined shape classifications are present, this component parses the classification columns from CellProfiler into a new "ShapeClassification" column. The new column is populated with the classification names as defined in CellProfiler.

3.5 Duplicated Tracking Label Correction

No configuration required. Just right-click and select "Execute".

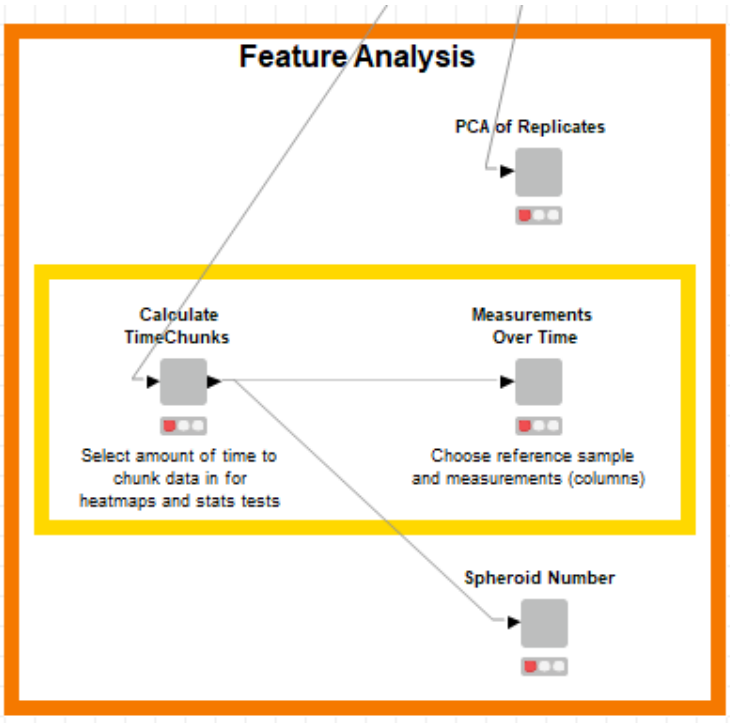
In CellProfiler, when a tracked object splits into two, or more, new objects, the software assigns the tracking label of the parent to the daughter objects. This creates conflicts downstream in the KNIME workflow, whereby multiple objects have the same tracking ID. This component ensures that the tracked objects in each Experiment, Plate, Well, and Site combination, have unique tracking labels. Where duplicate tracking labels are present, the object that is largest at the initiation of the duplication retains this label, and the smaller object(s) is assigned a new tracking label. For more information on this please refer to the manuscript.

3.6 Edit Column Names, Create Unique_object_id, Filter Columns, and Normalise

No configuration required. Just right-click and select "Execute".

This component removes unnecessary prefixes from column names. The AspectRatio is then calculated. Unique IDs are created for each tracked object. Finally, the filename of the originating image is extracted, and measurement columns are Z-Score normalised.

4 Feature Analysis



4.1 PCA of Replicates

No configuration required. Just right-click and select "Execute".

This component performs Principal Component Analysis (PCA) on the dataset replicates, and outputs results for two PCAs: one in which each point is a well ("PCAofReplicates_perWell.pdf"), and another in which each point is an experiment ("PCAofReplicates_perExperiment.pdf"). A plot of PCA Loadings is also output for each analysis: "PCAofReplicates_perWell_Loadings.pdf", "PCAofReplicates_perExperiment_Loadings.pdf".

4.2 Calculate TimeChunks

Configuration required.

Visualisation of the change in a variable over time can be complex to display when many timepoints are present. To simplify this, data can be presented in grouped segments of time, rather than at individual timepoints. This component calculates time interval chunks for the input data table.

Double-click to open dialog box. Configure the component to indicate how many timepoints (frames) to include in each timechunk. If chunking is not required, set this value to 1. Right-click and select "Execute" to run the component.

4.3 Measurements Over Time

Configuration required.

This component generates a heatmap showing change in measurements of area, shape, and movement over time (for a set of user-specified measurements). For each sample, the average value of the measurement is calculated at each of the previously defined time chunk intervals. For the purposes of presentation, resulting values are Z-score normalised per measurement. T-tests are performed to compare samples at each time interval, to the specified reference sample. A Bonferroni adjustment is applied to adjust for multiple testing. Two files are generated in the main Output directory:"MeasurementsOverTime_CONTROLSAMPLEControl.pdf" and "MeasurementsOverTime_CONTROLSAMPLEControl.csv"

Double-click to open dialog box. Select Reference Sample from the drop-down menu to set a reference sample for statistical comparison. Measurements to be plotted in the heatmap should be included in the green "Include" box - all others should remain in the red "Exclude" box. Descriptions of each of the measurements generated by CellProfiler can be found in the CellProfiler documentation. Selecting multiple measurements will generate one heatmap with all values on the same colour scale. Selecting one measurement at a time will result in maximum and minimum values set for each heatmap independently.

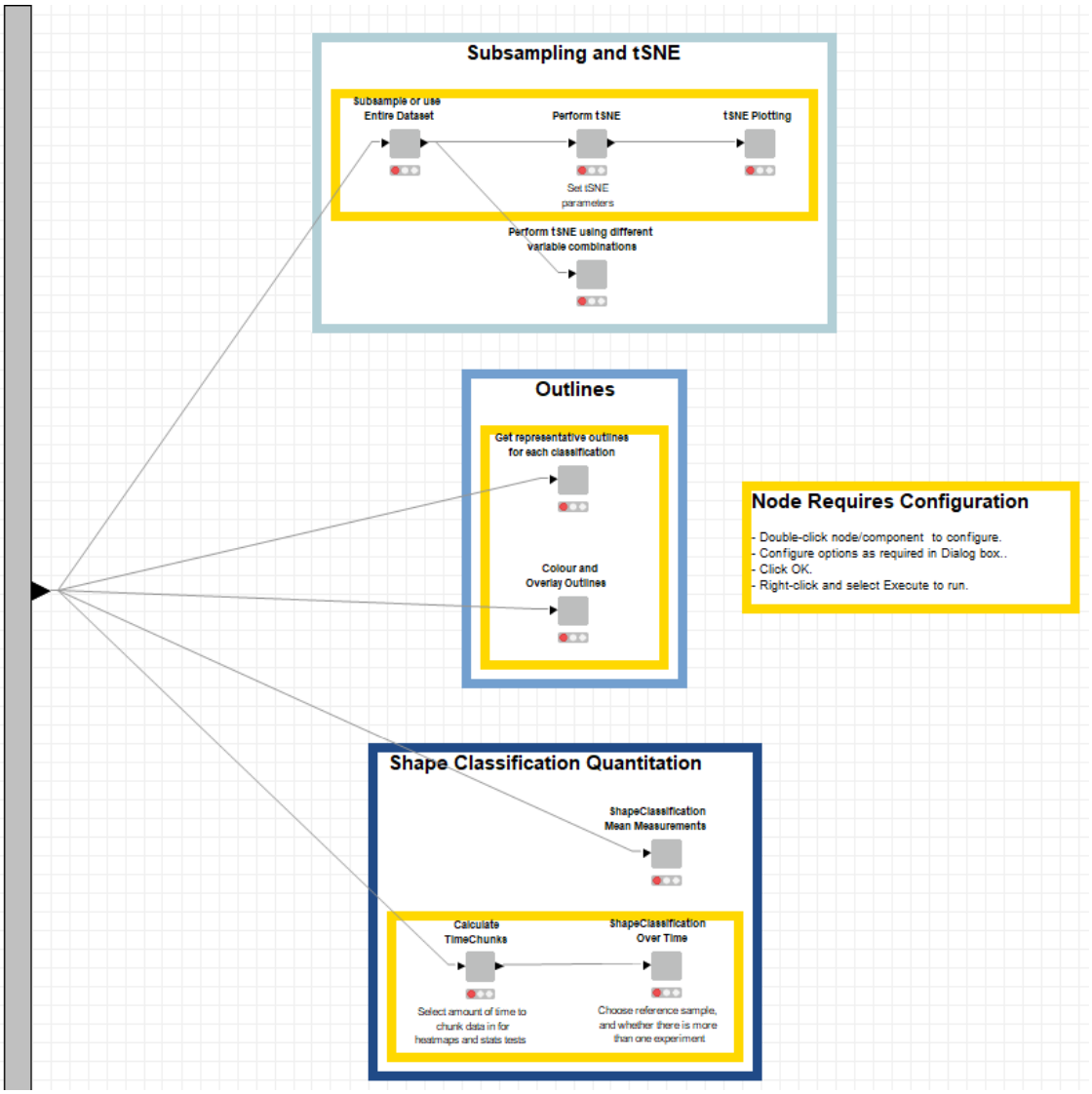
Finally, indicate how the rows (samples) in the heatmap should be ordered: alphabetically or in user-specified ordering. If a user-specified order is to be applied, use the text box to define this order (from top row of the heatmap, to bottom). Sample names should be written as they appear in the Experiment Key, separated by only commas. The easiest way to ensure this is done correctly is by using keyboard shortcuts (control/command + c, control/command + v) to copy and paste sample names directly from the Experiment Key. **Tip**: Be aware of any leading or trailing spaces in sample names in the Experiment Key, and if present, ensure not to delete them when configuring this text box.

4.4 Spheroid Number

No configuration required. Just right-click and select "Execute".

This component outputs the initial counts of spheroids in each treatment/sample per experiment ("InitialSpheroidNumbersPerExperiment.csv"), and the relative change from this initial count over time ("SpheroidNumberOverTime.csv" and "SpheroidNumberOverTime_LinePlot.pdf").

5 User-defined Classification



5.1 Subsample or use Entire Dataset

Configuration required.

This component first ensures equal sample size. Then, if the data is to be subsampled, this is done to user specifications. Otherwise all the remaining data is retained.

Double-click to open dialog box. Configure to indicate whether data is to be subsampled. If so, select which subsampling method to use (Random vs GeoSketch [Geometric Sketching]), and input the number of objects to subsample. To view the total number of objects and inform on how many objects to subsample, right-click on the previous component in the workflow and select "Port 1"; the total number of objects is the "Rows" shown at the top of the tab in the resulting window, and the subsampling choice should be lower than this value. After configuration, right-click the component and select "Execute" to run.

5.2 Perform tSNE

Configuration required.

If you will also be running the data-driven analysis, skip the following tSNE sections and instead run the equivalent components in section 6. This will ensure matching tSNE plots of your user-defined classes and the identified data-driven states.

Performs tSNE using Area, Zernike features, Displacement, and Distance Travelled.

Double-click to open dialog box. Configure the component to set values for perplexity, theta, the number of iterations to perform, and to indicate whether PCA should be performed first. If unsure of what values to set for these parameters, run the component called "Perform tSNE using different variable combinations" in section 5.4, and use the resulting plots to inform variable selection. Right-click and select "Execute" to run the component.

Optional Customisation: We use Area, Zernike features, Displacement and DistanceTraveled for analysis (see manuscript Results section for rationale). If you would like to change the set of features used for analysis, right-click on this component and select Component > Open. Customise the selection of features (green box) within the "Column Filter" node highlighted in yellow.

5.3 tSNE Plotting

Configuration required.

After ensuring equal sample size, and subsampling as required by the user, this component produces tSNE plots of the input data table. Plots are produced in which points are: coloured by sample, both in individual plots and all together; coloured by their value for each measurement; for each sample, coloured by point density; and coloured by user-defined classification. Plots are saved to Output > tSNE, within a folder whose name summarises the tSNE parameters as set by the user.

Double-click to open dialog box. Configure component to indicate whether to subsample prior to plotting - this value should be lower than the number of objects used for tSNE analysis. This can be used to adjust the number of points if the plot is too dense. Additionally, select a colour scheme, and point size to be used for plotting. Right-click and select "Execute" to run the component.

5.4 Perform tSNE using different variable combinations

No configuration required. Right-click and select "Execute" to run the component.

tSNE is performed (on Area, Zernike features, Displacement, and DistanceTravelled) using combinations of different values for all parameters. Plots for these are coloured by user-defined classification and saved within Output > tSNE. These are intended to aid in selecting the appropriate values to use for best possible visualisation.

perplexity: 25, 50, 100

theta: 0.25, 0.5

iterations: 5000

PCA: TRUE

Optional Customisation: We use Area, Zernike features, Displacement and DistanceTraveled for analysis (see manuscript Results section for rationale). If you would like to change the set of features used for analysis, right-click on this component and select Component > Open. Customise the selection of features (green box) within the "Column Filter" node highlighted in yellow, making sure to include the "ShapeClassification" column.

5.5 Get representative outlines for each classification

Configuration required.

This component selects and outputs a representative outline for each user-defined classification. This is selected from outlines of x objects nearest (in terms of Euclidian distance) to the mean measurements of the group. Outlines are saved in the ShapeClassification > RepresentativeOutlines subdirectory. Plots and the selected representative object outline are subsequently saved in ShapeClassification > RepresentativeOutlines > Plots subdirectory.

Double-click to open dialog box. Configure to indicate the number of outlines from which to select a representative per classification group. Right-click and select "Execute" to run the component.

5.6 Colour and Overlay Outlines

Configuration required.

This component overlays object outlines, coloured by their user-defined classification, onto the original phase images. The resulting images are saved in Output > ShapeClassification > OutlineOnPhase.

Double-click to open dialog box. Configure this component to select an experiment, plate, and site for which to overlay outlines. Next indicate whether this is to be performed for all wells, or a user-defined subset. If the latter, populate the associated text input with the wells of interest, separated by commas but no spaces. Finally indicate whether to process frames for the whole duration of the experiment, or for a user-defined subset of timepoints/frames. If a subset is to be used, fill in the appropriate text input to indicate the frames of interest, again separated by commas but no spaces.

If you are not using "round", "spread" and "spindle" classes you will need to adjust this component to assign colours to your specific classifications. Within the component (right-click, select Component > Open) configure the "Color Manager" node, highlighted by a yellow box, to set the colours to be used for each classification; note the colours that are used for each shape classification in the tSNE plots etc, if you want these to match.

Right-click and select "Execute" to run the component.

5.7 ShapeClassification Mean Measurements

No configuration required. Right-click and select "Execute" to run the component.

This component generates a CSV file containing the mean values of measurements for each user-defined class. The resulting file is named "ShapeClassificationMeanMeasurements.csv" and is located in Output > ShapeClassification.

5.8 Calculate TimeChunks

Configuration required.

Visualisation of the change in a variable over time can be complex to display when many timepoints are present. To simplify this, data can be presented in grouped segments of time, rather than at individual timepoints. This component calculates time interval chunks for the input data table.

Double-click to open dialog box. Configure the component to indicate how many timepoints (frames) to include in each timechunk. If chunking is not required, set this value to 1. Right-click and select "Execute" to run the component.

5.9 ShapeClassification Over Time

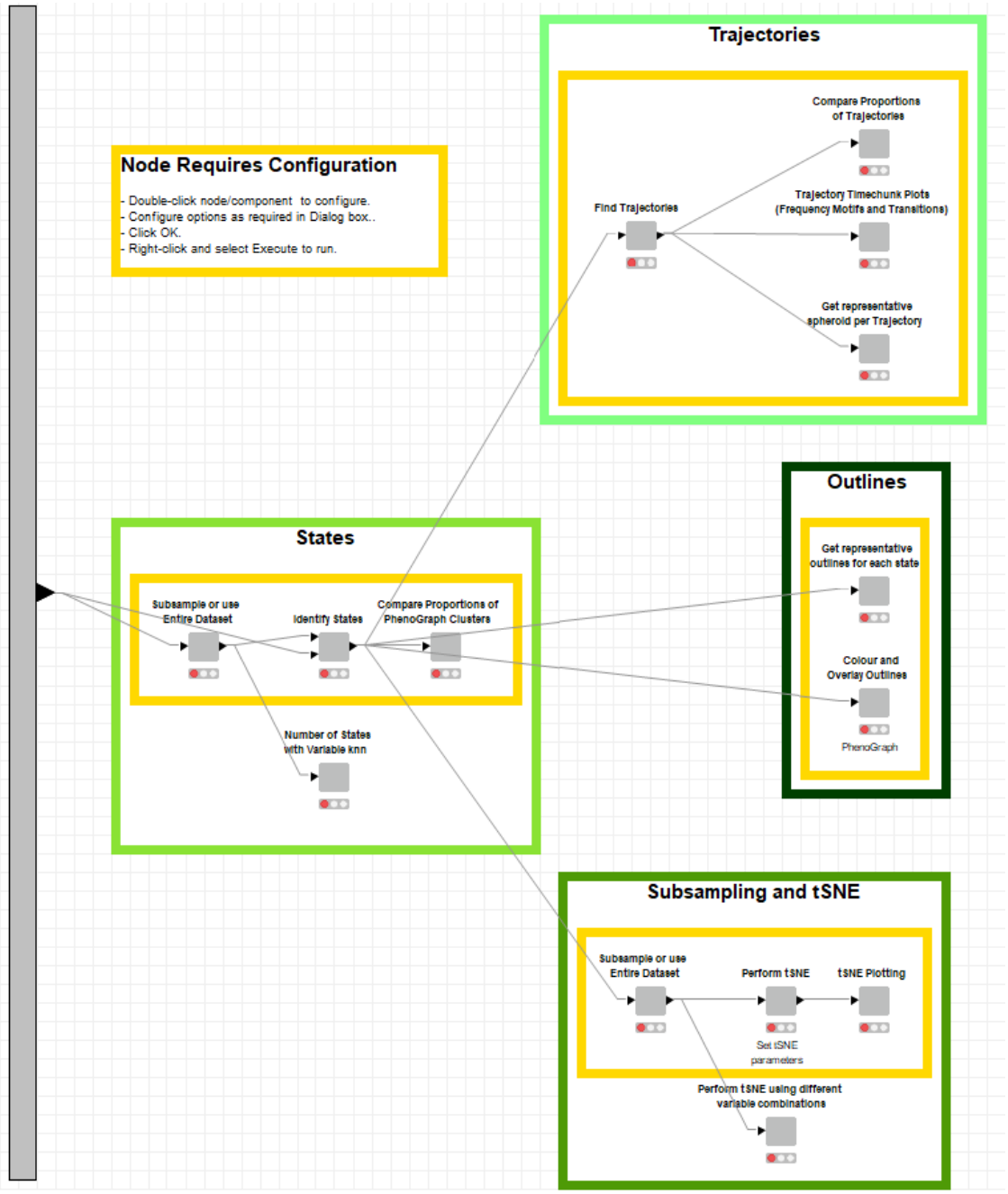
Configuration required.

This component compares, between a reference and non-reference samples, the proportions of each user-defined classification at every time chunk interval. A heatmap, and CSV file of results is output to Output > ShapeClassification, named: ShapeClassification_ShapeChangeOverTimeHeatmap_StatsTest_REFERENCE

SAMPLE NAME, where StatsTest represents: Cochran-Mantel-Haenszel test (CMH), Chi-Squared test (ChiSq).

Double-click to open dialog box. Configure the component to set a reference sample for statistical comparison. Select the statistical test to be used. Generally, Cochran-Mantel-Haenszel is used when comparing across multiple experimental/biological replicates, whereas Chi-squared is used when there is only one replicate. Finally, indicate how rows (samples) in the heatmap should be ordered: alphabetically, dendrogram clustered, or by user-specified ordering. If dendrogram clustering is used, the dendrogram is output as a separate file (ShapeClassification_ShapeChangeOverTimeHeatmap_Dendrogram_REFERENCE SAMPLE NAME). If a user-specified order is to be used, use the text box to define this order (from top row of the heatmap, to bottom). Sample names should be written as they appear in the Experiment Key, separated by only commas. The easiest way to ensure this is done correctly is by using keyboard shortcuts (control/command + c, control/command + v) to copy and paste sample names directly from the Experiment Key. **Tip**: Be aware of any leading or trailing spaces in sample names in the Experiment Key, and if present, ensure not to delete them when configuring this text box. Right-click and select "Execute" to run the component.

6 Data-Driven Classification



6.1 Subsample or use Entire Dataset

Configuration required.

This component first ensures equal sample size. Then, if the data is to be subsampled, this is done to user specifications. Otherwise, all the remaining data is retained.

Double-click to open dialog box. Configure to indicate whether data is to be subsampled. If so, select which subsampling method to use (Random or GeoSketch [Geometric Sketching]), and input the required number of points to be subsampled. To view the total number of objects and inform on how many objects to subsample, right-click on the previous component in the workflow and select "Port 1"; the total number of objects is the

"Rows" shown at the top of the tab in the resulting window, and the subsampling choice should be lower than this value. After configuration, right-click the component and select "Execute" to run.

6.2 Number of States with Varying knn

No configuration required. Right-click and select "Execute" to run the component.

The PhenoGraph clustering algorithm is performed on the Area, Zernike features, Displacement, and Distance Travelled columns of the input data table. This is done using a varying knn value of 20-100. The number of identified states across knn iterations is output as a line plot (PDF) named "VaryingKNN_LinePlot" within in Output > PhenoGraph. This is intended to aid in selecting a knn value for the analysis.

Optional Customisation: We use Area, Zernike features, Displacement and DistanceTraveled for analysis (see manuscript Results section for rationale). If you would like to change the set of features used for analysis, right-click on this component and select Component > Open. Customise the selection of features (green box) within the "Column Splitter" node highlighted in yellow.

6.3 Identify States

Configuration required.

The PhenoGraph clustering algorithm is performed on the Area, Zernike features, Displacement, and Distance Travelled columns of the input data table. The data previously excluded when ensuring equal sample size and during subsampling is then retrofitted to the identified states, using Euclidian distance. Mean measurements (i.e. Area, AspectRatio) of the identified states are calculated, and output as a heatmap. The heatmap is saved as "PhenoGraph_MeanExpressionHeatmap.pdf" within Output > PhenoGraph .

Double-click to open dialog box. Configure the component to set a value for the PhenoGraph k-nearest neighbours parameter. If in doubt, use the plot generated by the component called "Number of States with Varying knn" (Section 6.2) and select a value (x-axis) near the elbow point of the line plot. Right-click the component and select "Execute" to run.

Optional Customisation: We use Area, Zernike features, Displacement and DistanceTraveled for analysis (see manuscript Results section for rationale). If you would like to change the set of features used for analysis, right-click on this component and select Component > Open. Customise the selection of features (green box) within the "Column Splitter" and "Similarity Search" nodes highlighted in yellow.

6.4 Compare Proportions of PhenoGraph Clusters

Configuration required.

This component compares the proportions of objects that belong to each PhenoGraph (behaviour state/shape) classification. This is done pairwise between reference and non-reference samples. Output files include a heatmap of the fold changes from control and statistical comparisons (PDF), and a CSV file containing the heatmap plot data, are saved within Output > PhenoGraph as "PhenoGraph_TotalProportionHeatmap_StatsTest_REFERENCE SAMPLE NAME". StatsTest represents: Cochran-Mantel-Haenszel test (CMH), Chi-Squared test (ChiSq).

Double-click to open dialog box. Configure the component to set a reference sample, and to select which statistical test to perform. Generally, Cochran-Mantel-Haenszel is used when comparing across multiple experimental/biological replicates, whereas Chi-squared is used when there is only one replicate. Next, indicate whether rows (states) and columns (samples) should be ordered alphabetically, clustered (dendrogram), or by a user-specified ordering. If a user-specified ordering is required, input the desired order into the appropriate text input, separated by commas without spaces. Sample names should be written as they appear in the Experiment Key.; the easiest way to do this is by using keyboard shortcuts to copy and paste directly from the Experiment Key. If clustered ordering is selected, the associated dendrogram will be output as a separate PDF file in the location stated above. Right-click the component and select "Execute" to run.

6.5 Get representative outlines for each state

Configuration required.

Ensure this component is run prior to attempting section 6.10 "Trajectory Timechunk Plots (Frequency Motifs and Transitions)".

This component selects and outputs a representative outline for each identified state. This is selected from outlines of x objects nearest (in terms of Euclidian distance) to the mean measurements of the group. Outlines are saved in the PhenoGraph > RepresentativeOutlines subdirectory. Plots and the selected representative object outline are subsequently saved in PhenoGraph > RepresentativeOutlines > Plots subdirectory.

Double-click to open dialog box. Configure to indicate the number of outlines from which to select a representative per classification group. Right-click the component and select "Execute" to run.

Optional Customisation: We use Area, Zernike features, Displacement and DistanceTraveled for analysis (see manuscript Results section for rationale). If you would like to change the set of features used for to compare between the group mean and each object, right-click on this component and select Component > Open. Customise the selection of features (green box) within the "Similarity Search" node highlighted in yellow, making sure to include the PhenoGraph column.

6.6 Colour and Overlay Outlines

Configuration required.

This component overlays object outlines, coloured by their state classification, onto the original phase images. The resulting images are saved in Output > PhenoGraph > OutlineOnPhase.

Double-click to open dialog box. Configure this component to select an experiment, plate, and site for which to overlay outlines. Next indicate whether this is to be performed for all wells, or a user-defined subset. If the latter, populate the associated text input with the wells of interest, separated by commas but no spaces. Finally indicate whether to process frames for the whole duration of the experiment, or for a user-defined subset of timepoints. If a subset is to be used, fill in the appropriate text input to indicate the frames of interest, again separated by commas but no spaces. Right-click and select "Execute" to run the component.

6.7 Find Trajectories

Configuration required.

This component strings the state classifications into a time sequence for each tracked object. After a series of filtering steps to remove objects that were only tracked for a short period of time, the PhenoGraph algorithm is used to identify recurring "trajectories" of state change over time. A heatmap of trajectories, as well as the heatmap data are saved in Output > PhenoGraph > Trajectory, as "TrajectoryHeatmap_knn.pdf", and "TrajectoryHeatmap_Data_knn.csv"

A CSV file containing the number/percentages of total values plotted in the heatmap, imputed values, and retrofitted PhenoGraph classifications are also saved ("TrajectoryHeatmap_DataSourceCounts.csv").

Line plots are output per measurement, of mean values for each identified trajectory: Output > PhenoGraph > Trajectory > MeasurementAverageOverTime

Line plots are output per trajectory, of normalised mean Area, Displacement, and DistanceTravelled: Output > PhenoGraph > Trajectory > AreaMovementPlots

Double-click to open dialog box. Configure the component to set a value for the PhenoGraph k-nearest neighbours parameter. A value of 40 is a good starting point. Generally, increasing this value will cause identified trajectory classes to merge, whereas decreasing the value will cause them to split; use visual inspection of "TrajectoryHeatmap_knn.pdf" to inform choice and adjust as necessary. Right-click the component and select "Execute" to run.

6.8 Find Trajectories in One Image Sequence

Configuration required.

This component finds an image sequence in which objects of user-specified trajectories are nearest to each other. The outlines (coloured depending on Trajectory ID) of the objects in the specified trajectories are overlayed onto the phase image. The resulting images are saved in Output > PhenoGraph > Trajectory > OverlayedOutlines.

Double-click to open dialog box. Populate the text box to indicate the trajectories of interest: separated by commas only (no spaces). Additionally input how many image sequences/movies are to be output. Right-click the component and select "Execute" to run.

Note on the algorithm: First, the component determines which Experiment, Plate, Well, and Site combinations (image sequences) contain objects from each of the trajectories defined by the user. Next, Euclidian distance is calculated pairwise between the trajectories of interest; specifically, for each spheroid in Trajectory X, the nearest spheroid belonging to Trajectory Y is identified. For each pairwise comparison, the shortest distance between the pairs of spheroids is retained. Finally, for each image sequence, the pairwise distances are summed and the sequence with the shortest total pairwise distance (smallest sum) is output.

6.9 Compare Proportions of Trajectories

Configuration required.

This component compares the proportions of objects that belong to each Trajectory classification. This is done pairwise between reference and non-reference samples. Output files include a heatmap of the fold changes from control and statistical comparisons (PDF), and a CSV file containing the heatmap plot data. These are saved within Output > PhenoGraph > Trajectory as "Trajectory_TotalProportionHeatmap_StatsTest_REFERENCE SAMPLE NAME". StatsTest represents: Cochran-Mantel-Haenszel test (CMH), Chi-Squared test (ChiSq).

Double-click to open dialog box. Configure the component to set a reference sample, and to select which statistical test to perform. Generally, Cochran-Mantel-Haenszel is used when comparing across multiple experimental/biological replicates, whereas Chi-squared is used when there is only one replicate. Next, indicate whether rows (Trajectories) and columns (samples) should be ordered alphabetically, clustered (dendrogram), or by a user-specified ordering. If a user-specified ordering is required, input the desired order into the text input, separated by commas without spaces. Sample names should be written as they appear in the Experiment Key; the easiest way to do this is by using keyboard shortcuts (control/command + c, control/command + v) to copy and paste directly from the Experiment Key. If clustered row and/or column ordering is selected, the associated dendrograms will be output as separate PDF files in the location stated above. Right-click the component and select "Execute" to run.

6.10 Trajectory Timechunk Plots (Frequency Motifs and Transitions)

Configuration required.

Ensure you have run section 6.5 "Get representative outlines for each state" prior to attempting to run this component.

Ensure the "Get representative outlines for each state" component has been completed. This component generates summary plots for each identified Trajectory. Three types of plots are generated for each Trajectory: 1) Frequency motif of states at the defined time chunk interval. 2) Chord diagram of transitions between states, segmented by the defined time chunk interval. 3) Transitions plotted as line segments onto a PCA plot of states (one plot per time chunk). The files are saved in Output > PhenoGraph > Trajectory, in subdirectories named "Frequency Motifs", and "Transition Plots".

Double-click to open dialog box. Configuring this component is required to indicate the time chunk interval. A transition frequency (as a percentage) filter is required for both types of transition plot - transitions that occur at a lower frequency (within a time chunk interval) will not appear in the plots. Finally, a minimum

line thickness is required for generating the PCA transition plots. Right-click the component and select "Execute" to run.

6.11 Get representative spheroid per Trajectory

Configuration required.

This component selects and outputs phase images of x "representative" spheroids for each trajectory. Images are saved as square crops around the selected spheroid, but can also be output as uncropped files. The resulting images are saved in Output > PhenoGraph > Trajectory > RepresentativeSpheroids.

Double-click to open dialog box. Configure to indicate how many representative spheroids should be output per trajectory, and whether to save uncropped phase images (in addition to the cropped images). Right-click the component and select "Execute" to run.

Note on the algorithm: First a consensus sequence (matching the sequence appearing in the state frequency motif) is determined for each trajectory - this is comprised of the most frequent state at each point in time. Next, the sequence of state classifications for each spheroid in a trajectory is determined, which is then compared to the consensus sequence for the trajectory group using a string comparison algorithm. Only mismatches at equivalent timepoints are assessed, while insertion/deletion type alterations are not taken into consideration. This results in a similarity score. Images of x spheroids deemed most similar, in terms of this score, to the consensus sequence are retrieved. The spheroid outlines are overlayed onto the phase image, cropped, and output.

6.12 Subsample or use Entire Dataset

Configuration required.

This component first ensures equal sample size. Then, if the data is to be subsampled, this is done to user specifications. Otherwise all the remaining data is retained.

Double-click to open dialog box. Configure to indicate whether data is to be subsampled. If so, select which subsampling method to use (Random vs GeoSketch [Geometric Sketching]), and input the number of objects to subsample. To view the total number of objects and inform on how many objects to subsample, right-click on the previous component in the workflow and select "Port 1"; the total number of objects is the "Rows" shown at the top of the tab in the resulting window, and the subsampling choice should be lower than this value. After configuration, right-click the component and select "Execute" to run.

6.13 Perform tSNE

Configuration required.

Performs tSNE using Area, Zernike features, Displacement, and Distance Travelled.

Double-click to open dialog box. Configure the component to set values for perplexity, theta, the number of iterations to perform, and to indicate whether PCA should be performed first. If unsure of what values to set for these parameters, run the component called "Perform tSNE using different variable combinations" in section 6.15, and use the resulting plots to inform variable selection. Right-click and select "Execute" to run the component.

Optional Customisation: We use Area, Zernike features, Displacement and DistanceTraveled for analysis (see manuscript Results section for rationale). If you would like to change the set of features used for analysis, right-click on this component and select Component > Open. Customise the selection of features (green box) within the "Column Filter" node highlighted in yellow.

6.14 tSNE Plotting

After ensuring equal sample size, and subsampling as required by the user, this component produces tSNE plots of the input data table. Plots are produced in which points are: coloured by sample, both in individual plots

and all together; coloured by their value for each measurement; for each sample, coloured by point density (one plot per sample); and coloured by PhenoGraph classification (data-driven state). Plots are saved to Output > tSNE, within a folder whose name summarises the tSNE parameters as set by the user.

Double-click to open dialog box. Configure component to indicate whether subsampling should occur - this value should be lower than the number of objects used for tSNE analysis. Additionally, select a colour scheme, and point size to be used for plotting. Right-click and select "Execute" to run the component.

6.15 Perform tSNE using different variable combinations

No configuration required.

tSNE is performed (on Area, Zernike features, Displacement, and DistanceTravelled) using combinations of different values for all parameters. Plots for these are coloured by PhenoGraph classification and saved within Output > tSNE. These are intended to aid in selecting the appropriate values to use for best possible visualisation.

perplexity: 25, 50, 100

theta: 0.25, 0.5

iterations: 5000

PCA: TRUE

Optional Customisation: We use Area, Zernike features, Displacement and DistanceTraveled for analysis (see manuscript Results section for rationale). If you would like to change the set of features used for analysis, right-click on this component and select Component > Open. Customise the selection of features (green box) within the "Column Filter" node highlighted in yellow, making sure to include the PhenoGraph column.

7 Troubleshooting Tips

Problem: Cannot install KNIME extensions or Conda environments as described in Section 1.2 "KNIME installation" and Section 1.4 "Python Integration".

Solutions: This is likely to be an internet connection issue. If you are setting the software up using an institute/university network connection, which is otherwise stable and functioning as expected, the problem may have to do with the institute network settings (i.e. proxy server). Please see if your IT department can help troubleshoot this.

Problem: Data won't load into "Data import and pre-processing metanode"Select Directory and match experiment key" component.

Solutions: if folders or data are not in the format descripted in Section 1.1 "Folder Structure" this component will have trouble locating the necessary files. Please check the following:

- That you are configuring this node to direct to the Directory Folder, not an "Experiment___n" folder.
- That the Directory Folder has no spaces in the name.
- Each "Experiment___n" folder has double underscore before the number and no additional spaces/characters.
- That "Phase" folder contains all images in RGB (3-channel) TIFF format with filenames in following format; ExperimentName_Plate 1_Well_Site_Date_Time.
- That "PhaseGrayCystOutlines" folder contains binary images of object outlines as PNGs. Should be 1 image, with equivalent file name, per Phase image in "Phase" folder.
- That "Data" folder contains CSV file named 'PhaseGrayCysts' in correct format with all relevant fields populated.
- That all fields in 'Experiment Key' and the appropriate 'Plate' tabs are filled in or auto-populated correctly.

- That value in "Metadata_ExperimentName" is different across experimental/biological replicates in each Experiment Key.
- That "Conditions" are consistent in Experiment Keys for both experimental/biological and technical replicates.
- Data described in Experiment Key corresponds to data analysed by CellProfiler and included in "PhaseGrayCysts" CSV file. Any data not included in analysis should be removed from the Experiment Key, and subsequently excluded using the component "Select Samples and Frames to Include" (Section 3.3).
- All data analysed by CellProfiler and included in "PhaseGrayCysts" CSV file is described in Experiment Key.

Problem: in "Select Samples and Frames to include" component there are more Sample names than there should be.

Solution: Check that samples names are exactly the same for both biological and technical replicates in "Condition" field in Experimental Keys.

Problem: R Snippet node not running in various components.

Solution: As R Snippets are used to perform various processes in the workflow, this can occur for a variety of reasons. The KNIME console should print any errors from R to help inform on what the problem is. If the error indicates an issue with Rserve, or a missing package (i.e. "there is no package called 'x' ", or "could not find function 'x' "), in the first instance ensure correct version of R and all relevant packages installed correctly as detailed in Section 1.3 "R Integration". Manually install any packages that are missing.

Problem: Error encountered when running statistical analysis using Cochran-Mantel_Haenszel test.

Solution: This test requires two or more experimental/biological replicates for *every* sample. If you have only one experimental/biological replicate, use the Chi-squared test instead. If you are certain you have multiple replicates, check that:

- Sample names are exactly the same for both experimental/biological and technical replicates in "Condition" field in Experimental Keys.
- Each experimental/biological replicate has a unique value in the "Metadata_ExperimentName" column of its respective Experiment Key.

Problem: Overlay outlines components not working.

Solution: Check that:

- Phase images are in 3-channel (RGB) TIFF format, and saved within a folder called "Phase" (in the respective experimental replicate folder).
- Outline images show white outlines on black background, are PNG format and contained within a folder called "PhaseGrayCystOutlines".
- The names of the phase and corresponding outlines images (described above) match exactly.
- User-defined classification component (section 5.6): ensure that you have set outline colours for your user-defined classes (as described in section 5.6).