Software for Multidimensional Data Analysis and Classification

SMIAL, a versatile toolkit developed for the analysis and classification of multidimensional data. The following document provides users with step-by-step instructions for the analysis and classification of multidimensional data. For up-to-date documents and code, please visit:

https://github.com/EwaGoldys/SMIAL.git

Table of Contents

1 Download and Installation

To install and launch the SMIAL software, follow the steps below:

1. **Download SMIAL**
   - Visit the official GitHub repository and download the SMIAL folder from: https://github.com/EwaGoldys/SMIAL.git
2. **Install SMIAL**
   - Run the *MyAppInstaller_web* file and follow the on-screen prompts.
   - Select a destination folder for installation.
   - Note if Windows displays a warning about an unknown publisher, click "More info" and then "Run anyway". This warning will not appear in future launches.
3. **Open SMIAL**
   - Once installed, SMIAL software is available in the selection application folder (see **Figure 1-1**).
   - Double-click the SMIAL icon to activate the software.

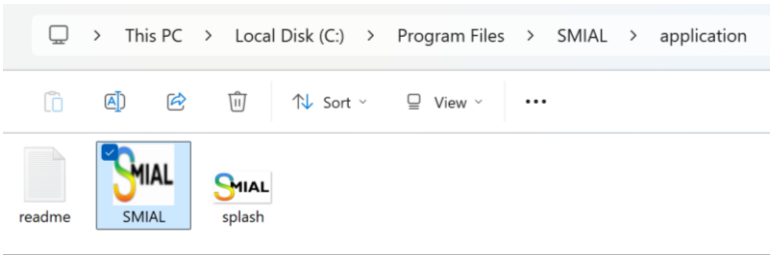

Figure 1-1. SMIAL installation folder and software icon

2 Overview

SMIAL features a user-friendly graphical user interface, organised into six main panels (see **Figure 2-1 Box 1**):

1. **Load Images**
2. **Segmentation and Masks**

3. **Pre-processing**
4. **Feature Generation**
5. **Data Cleaning and Statistics**
6. **Feature Analysis**

This manual provides detailed descriptions of each panel in SMIAL, including required input formats and step-by-step instructions for using the software. All generated files, along with the parameters and settings used are automatically saved in a dedicated "SMIAL" folder, serving as the master directory. This folder is created in the same directory where the user uploads their data into SMIAL. To support reproducibility, SMIAL also generates log text files that record all parameters and settings applied for each panel.

Recommended workflow diagrams for using SMIAL with various types of input files are available in Section 9.

Tooltips are available throughout the interface. Hover over buttons or text elements for quick guidance on specific functions.

A direct link to this manual is also accessible from within the software (see **Figure 2-1** ***Box 2****).*

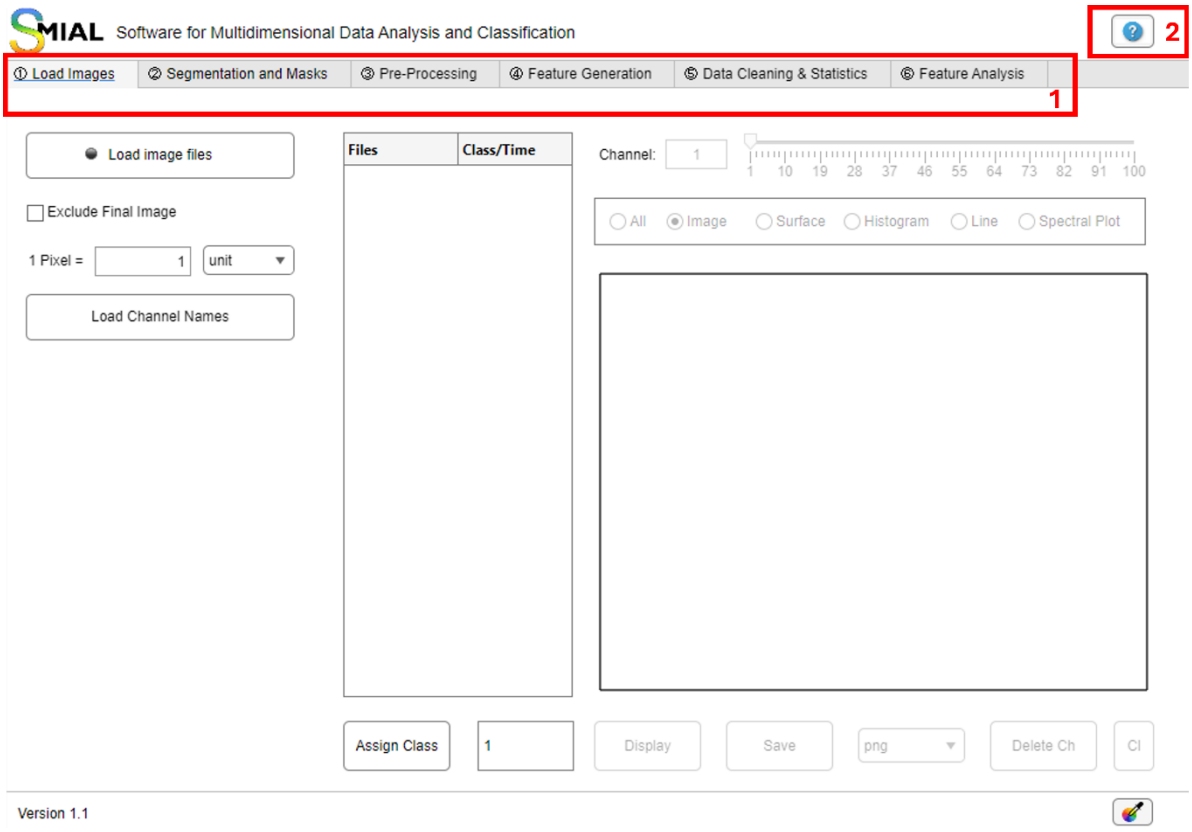

Figure 2-1. Modular structure of SMIAL

2.1 Built in indicators to monitor errors

SMIAL includes built-in indicators to help users monitor the software's status during option:

- Progress bar appears during computations.

- Status lamp providing real-time feedback:
  - **Orange**: Processing in progress
  - **Green**: Processing complete
  - **Red**: Error occurred

If an error occurs, an error message will be displayed alongside the red lamp to help identify the issue.

⚠ *If the software behaves unexpectedly, such as window resizing or unresponsive clicks, it is recommended to close and restart the GUI.*

Computational errors will be saved in an error logging file.

The following subsections describes each panel in detail, including their functions and input requirements.

3 Load Images Panel

The first panel in SMIAL is designed for uploading and managing 2D imaging data. It provides a user-friendly interface for importing, organising and previewing image files. The corresponding workflow of the Load Images panel is provided in Figure 9-1.

SMIAL only accepts 2D images, however if a user has a 3D data input, image slices may be imported individually into SMIAL for analysis.

3.1 Loading Image Files

- Users can upload image files in various formats, including .tif, .png, .jpg, and .mat. To upload, navigate to **Box 1** (**Figure 3-2**) and select the desired files from your folder. If your images are in another format, please convert them to one of the supported formats using an image processing tool (e.g. FIJI, ImageJ or Python). Users can post questions on the Github forum if they have further questions.

- **Exclude final image:** If this checkbox is ticked, it is assumed that the final image of the image-stack is from a different modality (e.g. brightfield).

- **Spatial Scale:** Input spatial resolution of your imaging data. By default, **1** pixel = **1** unit.
- **Temporal Scale:** Select the time interval for your data (i.e. ms, s, h)
- **Load channel names (optional):** An excel sheet may be uploaded with reference to some details of the channels, such as the excitation/emission ranges for every channel. The format of this excel sheet is a single column file with the number of rows equal to the number of imaging channels (**Figure 3-1**).

1	**ChannelNames**
2	ex325_em414
3	ex339_em414
4	ex343_em414
5	ex356_em414
6	ex381_em414

Figure 3-1. Example excel sheet of channel names.

For first time SMIAL use, users will need to initially select the file directory where their images are located from their file explorer. However, SMIAL will take this new selected directory as the automatic new default directory.

SMIAL only allows one file format at a time to be uploaded. In cases where there are multiple file formats in a single folder directory, SMIAL will only upload images with the majority file format.

SMIAL currently supports analysis of 2D spatial imaging data only. However, users working with 3D spatial datasets (e.g., ZYX stacks) can still visualise and analyse their data by importing individual slices into SMIAL as separate 2D images. This enables layer-by-layer analysis, which may be sufficient for many pre-processing and feature extraction workflows involving z-projection or plane-specific evaluation."

3.2 Viewing Metadata and File List

- **Box 2** (**Figure 3-2**) displays metadata for the selected image, such as x and y pixel dimensions, number of channels, and spatial and temporal resolution. Note that the user must first define the resolution parameters and click *Display* to view Metadata.
- **Box 3** shows a list of all uploaded image files.

3.3 Folder Structure and Class Label Assignment

SMIAL supports a **hierarchical folder structure**:

- If all image files are in a **single folder**, SMIAL treats this as the **master folder**. In this case, users must **manually assign a class label** to each image. The field **'Assign Class'** in **Box 3 Figure 3-2** is used to assign a label for supervised classification. Enter a class number or a name for non-time-varying data (e.g. *melanoma* or *fibroblast* cells) or a numerical value as a reference to the timepoint for dynamic image sequences. This field can also be used to overwrite any class labels inferred from the folder structure.
- Alternatively, users can organise images into **subfolders**, where each subfolder represents a different class. When these subfolders are uploaded together from a master folder, SMIAL will **automatically assign class labels** based on the subfolder names (displayed in **Box 3**).

3.4 Preview Image Channels

- Use **Box 4** (**Figure 3-2**) to visualise different channels of the loaded images such as the red, green, blue markers or autofluorescence channels, by selecting a specific channel.
    - **All:** Displays selected image for all channels in a separate pop-up window. To save individual images, please click save on the new pop-up window.
    - **Image:** Displays the selected channel image after applying intensity normalisation to the selected image and channel.
    - **Surface:** Displays a 3D surface plot of the selected image.
    - **Histogram:** Allows the user to select a region of interest by scrolling and shifting. A histogram of the pixel intensity distribution of that region is generated.
    - **Line:** Enables the user to draw a line of interest on the image. A line graph is displayed showing the mean intensity along that line.

- **Spectral plot:** Displays the selected channel and allows the user to select a region of interest by scrolling and shifting. A line graph is generated showing the mean intensity across all channels for the selected region.
- **Display**: Shows the selected display option.
- **Save**: Saves the figure on the SMIAL display panel as a .png, .jpg, .tif or .fig (Matlab) file.

- Use **Box 5 'Delete Ch'** icon to remove unnecessary channels. Click '**Cl'** to undo any previous channel removal. Deleted channels will be listed in an excel sheet (DeletedChannel.xls), only after *Feature Generation* (see **Section 6**).

3.5 Adjust Plot Settings

- Use **Box 6** 'Adjusts colours for figures' icon to modify the displayed output colour settings. This tool is available in all subsequent panels.

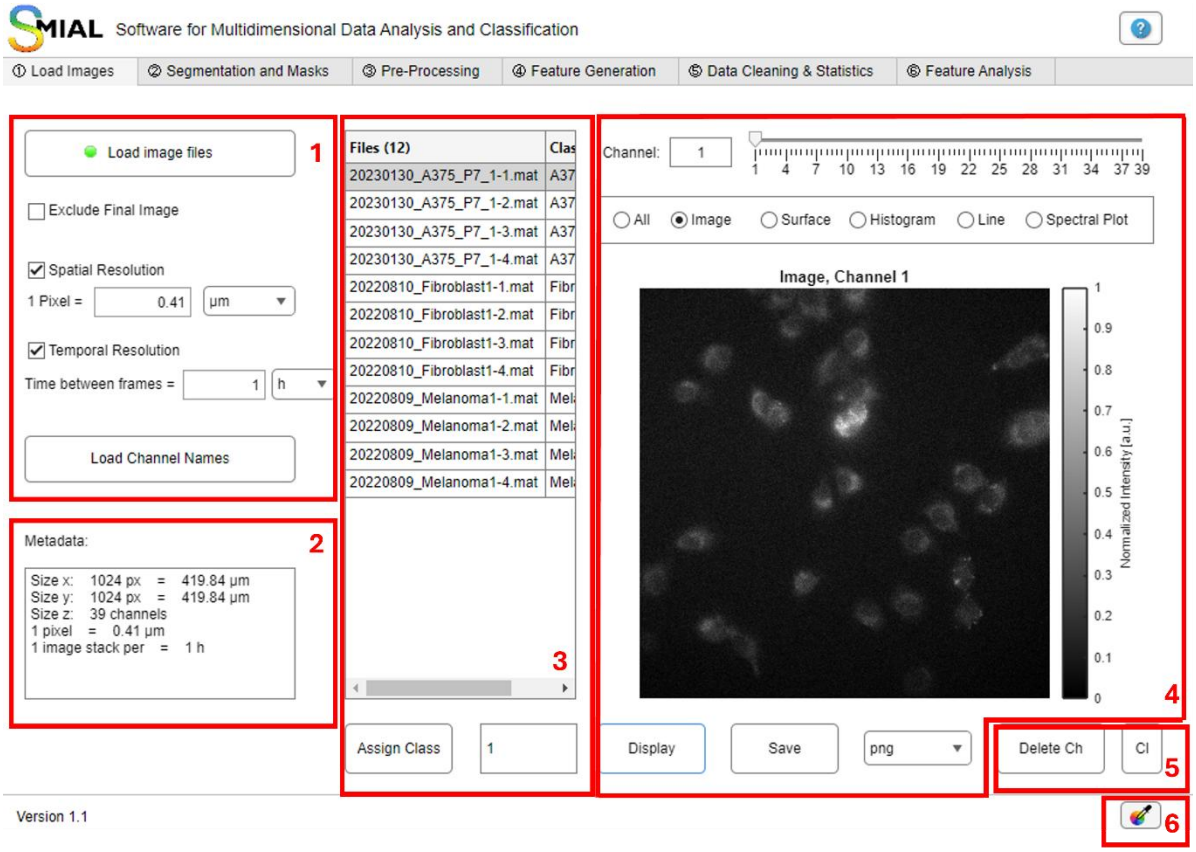

Figure 3-2: Load Images Panel.

4 Segmentation and Masks Panel

The second panel in SMIAL provides tools for segmenting images to identify objects of interest or for uploading pre-generated image masks. Users can also upload optional masks for background exclusion and ground truth comparison. The corresponding workflow of the Segmentation and Masks panel is provided in **Figure 9-2**Figure 9-1.

- **Mask Subpanel**: Upload masks that define the foreground objects (e.g. cells).

- **Background Mask Subpanel (Optional)**: Upload masks that define the background pixels of the image. These masks can be used in the subsequent *Pre-Processing* panel if available. The filename must use the suffix '_back'.

- **Ground truth Mask Subpanel (Optional)**: Upload *ground truth* masks to evaluate the accuracy of previously obtained segmentation results. When provided, SMIAL computes key performance metrics: **F1 score**, **Precision**, **Recall** and **Intersection over Union (IoU)** to compare the results against the ground truth. This is useful for validating segmentation outputs (e.g. CellPose or ilastik outputs) against a known accurate reference (e.g. hand-drawn annotations). Performance metrics can be saved as a ***MaskComparison.xls*** file (**Figure 4-1**).

  Note: Ground-truth masks must have the **same filename and extension** as the corresponding foreground masks. It is recommended to test this comparison on a **small sample set** first to ensure segmentation quality.

⚠ *Ground truth masks must have the **same filename and extension** as the corresponding foreground masks.*

💡 *It is recommended to test this comparison on a **small sample set** first to ensure segmentation quality prior to batch processing.*

	A	B	C	D	E
1	Filename	F1 Score	IoU	Precision	Recall
2	Average	0.609894273	0.441686612	0.669565505	0.579030668
3	20220809_Melanoma1-1.png	0.559744751	0.388642743	0.716716985	0.459177659
4	20220809_Melanoma1-2.png	0.598191372	0.426728271	0.496019108	0.753374962
5	20220809_Melanoma1-3.png	0.554789323	0.383881279	0.692798618	0.462630737
6	20220809_Melanoma1-4.png	0.575832885	0.404329575	0.536054443	0.621988119
7	20220810_Fibroblast1-1.png	0.592612336	0.421072566	0.688504882	0.520165508
8	20220810_Fibroblast1-2.png	0.600343896	0.428922429	0.810758395	0.476641963

Figure 4-1. Example exported Mask Comparison metrics.

4.1 Mask Subpanel Options

The mask subpanel has two options: (1) Segment Image; (2) Load Mask.

Segment Image Subpanel

Users can segment images directly within SMIAL using a simple threshold-based method.

Steps:

1. Select a file from the **Image Files Box** (**Figure 4-3 Box 1**) and choose the desired **channel** using the slider (**Box 2**).
2. Adjust the following parameters (**Box 1**):
   - **Contrast**: Enhances image contrast using the Contrast Limited Adaptive Histogram Equalization (CLAHE) method. Input a scale value (0–1); higher values increase contrast.
   - **Binary Threshold**: Converts the image to a binary mask. Pixels above the threshold (0–1) become white (1), others become black (0).

- **Fill Holes**: Fills holes in binary images. Choose **connectivity** (4 or 8):
    - 4-connectivity: horizontal and vertical neighbours.
    - 8-connectivity (default): all surrounding pixels.
- **Remove Pixel Components**: Removes connected components smaller than a user-defined pixel area.
- **Erode**: Applies morphological erosion using a **disk** or **diamond** structuring element with a user-defined radius.
- **Smooth**: Applies a **median filter** of user-defined order to smooth the mask.

3. Click **'Display'** (**Figure 4-3 Box 2**) to view the generated mask.
4. Use **'Overlay Mask'** to visualise the mask outline (green) over the original image.
5. Save the binary mask using the **'Save'** icon. Files are saved in a subfolder labelled **'Masks'** in the master directory.
6. Use '**Batch process**' to segment all available image files with the set parameters.
7. A ***SegmentationInfo.txt*** file (**Figure 4-2**) with all parameters used is automatically saved in the folder.

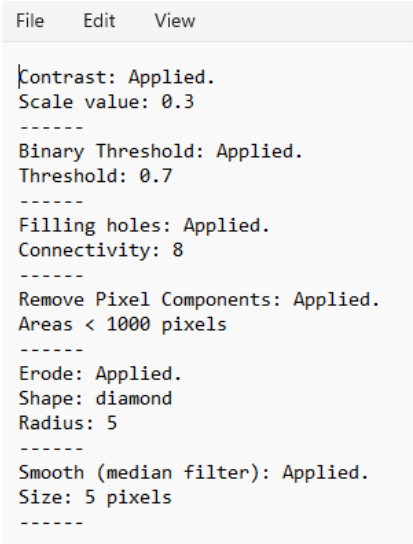

Figure 4-2. Example exported Segmentation Information text file.

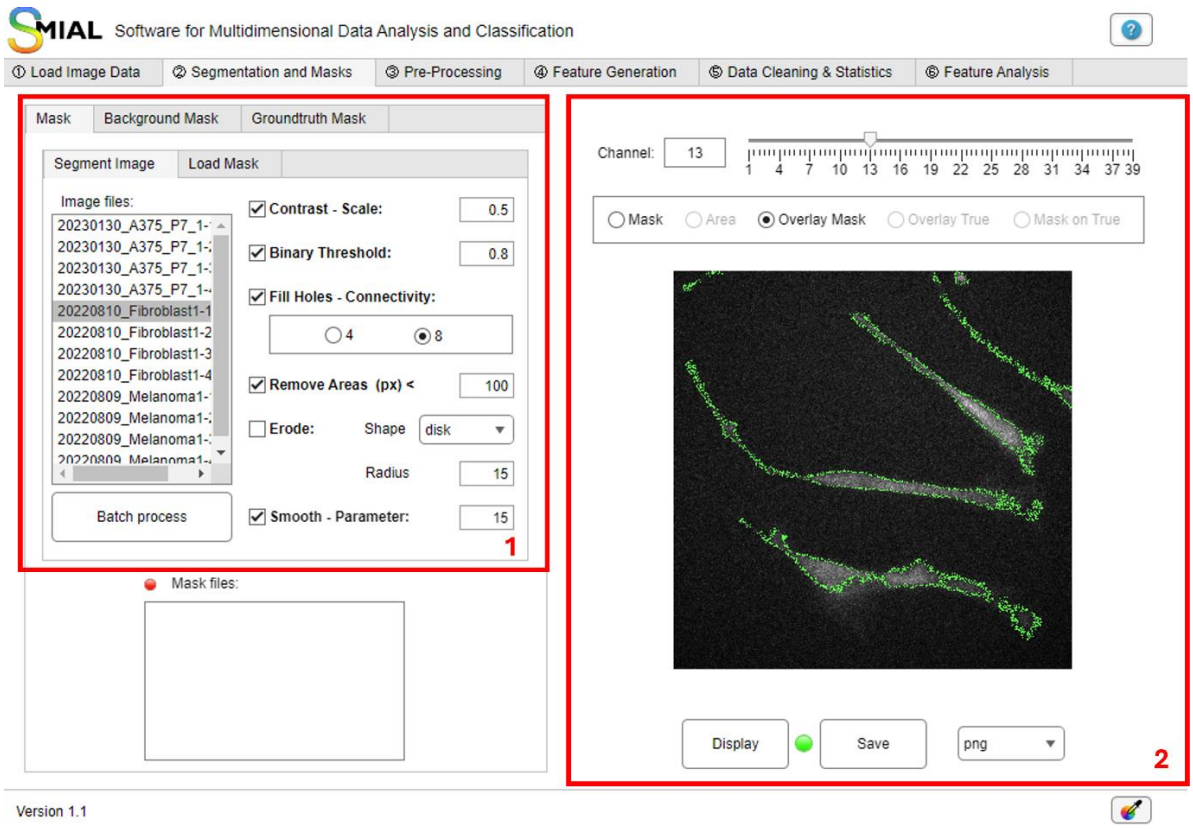

Figure 4-3: Segmentation and Masks panel overview. When using the Load Image Panel, users are required to either segment regions of interest from their image files, or upload a mask. Background and grouthtruth masks are optional.

Load Mask subpanel

Users can upload pre-generated binary masks for analysis.

Steps:

- **Upload Masks**: Upload mask files in .png, .jpg, or .tiff formats. Filenames must **exactly match** the corresponding image filenames from the first **Load Images** panel.
    - A **red indicator** appears if a mask is missing; hover to view missing filenames.
    - A **green indicator** confirms all masks are correctly matched.
- **Area Threshold**: Remove small objects in the mask below a user-defined pixel threshold. Useful for cleaning up noisy masks generated from external software.
- **Transform Masks (Optional)**:
    - **Remove Border Objects**: Deletes objects touching the image border. Define the number of pixels from the edge to apply this filter (Threshold).
    - **Split Objects**: Separates touching objects (e.g. clustered cells).
- **Display option to check masks:**
    - **Mask**: View mask file selected from files table. Masked objects will appear red and background appears black.
    - **Area**: Show included masked objects on file and channel selected, that have an area greater than the inputted Area Threshold.

- **Overlay Mask** (foreground): Show masked objects (red) from Mask, overlayed on the selected image (select file and channel)
- **Overlay True**: Show masked objects (red) from True, overlayed on the selected image (select file and channel). Note that masks also need to be loaded in the Mask subpanel to generate.
- **Mask on True** (ground-truth): Overlay Mask and True mask.

5 Pre-processing Panel

*Note that the pre-processing panel is optional, and does not need to be used for downstream analysis.

The **Pre-Processing** panel allows users to apply a range of 2D image pre-processing techniques, including background subtraction, smoothing, and image flattening. It consists of four subpanels (**Figure 5-3**, **Box 1**):

- **Image**: Apply pre-processing parameters directly to the original images.
- **Background**: Subtract a background image from the main image.
- **Calibration**: Apply a calibration factor to correct image intensities.
- **Quality:** Evaluate and compare the quality of the pre-processed images to the original images.

To trial different pre-processing methods, users can click **'Preview'** (**Figure 5-3**, **Box 2**) to apply and preview the selected method on a specific image file and channel (adjustable via the channel slider). Once the desired methods and parameters are selected, click **'Compute Pre-processing'** (**Figure 5-3**, **Box 3**) to generate the processed image files for all uploaded image files, which can be viewed and assessed in the adjacent figure panel (**Figure 5-3**, **Box 4**). These will be saved in a new folder **'Preprocessing'** in the master directory with **'_preprocessed'** appended to the original filename. The corresponding workflow of the Pre-processing panel is provided in **Figure 9-3**.

- A **SmoothingInformation.txt** file (**Figure 5-1**) detailing the selected smoothing and pre-processing parameters will also be saved for reproducibility.

⚠ *To ensure compatibility with later steps, pre-processed files must retain the "_preprocessed" suffix and remain in the original image directory. Comparisons between original and pre-processed images are only available after processing.*

```
Spike Removal (all files & channels):
Window size: 0.3
Threshold: 15
------
Image:
- Smoothing Method: Moving Average
- Parameter: 5, 5, 5, 5, 5, 5, 5, 5, 5,
------
Background:
Not applied.
------
Calibration:
Not applied.
------
Run time:
00:04:39 h:m:s
------
```

Figure 5-1. Example exported Smoothing Information text file.

5.1 Image Subpanel

Spike removal: Removes spikey data artefacts defined by sharp intensity values relative to neighbouring pixel intensities, by replacing them with the median of neighbouring pixels. Recommended for images with saturated pixels.

- Controlled by **Window Size** and **Threshold**.
- Applied to **all image channels, background, and calibration images**.

Smooth Image:

Reduces noise and enhances signal clarity. Users can choose from several smoothing methods:

- Moving average (default): Applies a moving average filter of size *N* (*N* = smoothing parameter).
- Median filter: Replaces each pixel with the median value of an *N x N* matrix (*N* = smoothing parameter).
- Wavelet bior4.4: Denoises the image using an empirical Bayesian method (no parameters required).
- Wavelet CDF9/7: Applies Cohen-Daubechies-Feauveau wavelet with a threshold based on the smoothing parameter. This method requires square images.
- FORPDN: First order spectral roughness penalty denoising performed in wavelet space (see [1] for details).
- HyRes: Hyperspectral image restoration using sparse and low-rank modelling (no parameters required; see [2] for details).
- Network: Uses a pretrained denoising convolutional neural network, DnCNN, to estimate a denoised image (no parameters required).

⚠ *For Wavelet CDF9/7 smoothing, images MUST be square sized.*

Smoothing parameter:

- Required for methods with adjustable parameters.
- Set a **uniform parameter** across all channels or assign **individual values** per channel (select 'Channel specific parameters' and use slider for channel selection).
- Input via field 'Smoothing parameter', or upload from:
    - Excel file: a single-column file with one row per channel containing the smoothing parameter to apply (see **Figure 5-2** for example below).
    - .mat file (SmoothingP_date) generated by SMIAL

💡 *High smoothing values may remove important features (e.g. cells), while low values may leave noise.*

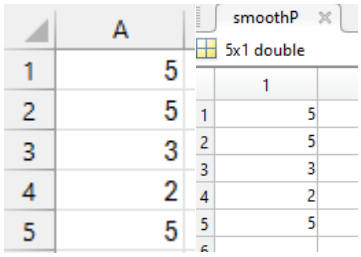

Figure 5-2. Example file containing preprocessing smoothing parameters. The format is a single column with each row specifying the smoothing parameter for a corresponding channel. Left: Excel input file. Right: SmoothP_date.mat file generated by SMIAL.

Other filters

Enhances image quality by subtracting filtered versions that estimate background intensity.

Options:

- **Top-hat Filter**: Also known as morphological opening is erosion followed by dilation.
- **Bottom-hat Filter**: Also known as morphological closing is dilation followed by erosion.
- **Gaussian Convolution**: 2D Gaussian filtering with user-defined **Sigma**.
- **Structuring Element**: Choose **disk** or **diamond** shape and define **radius**.

Small structuring elements may remove important features; large ones may retain curvature artifacts.

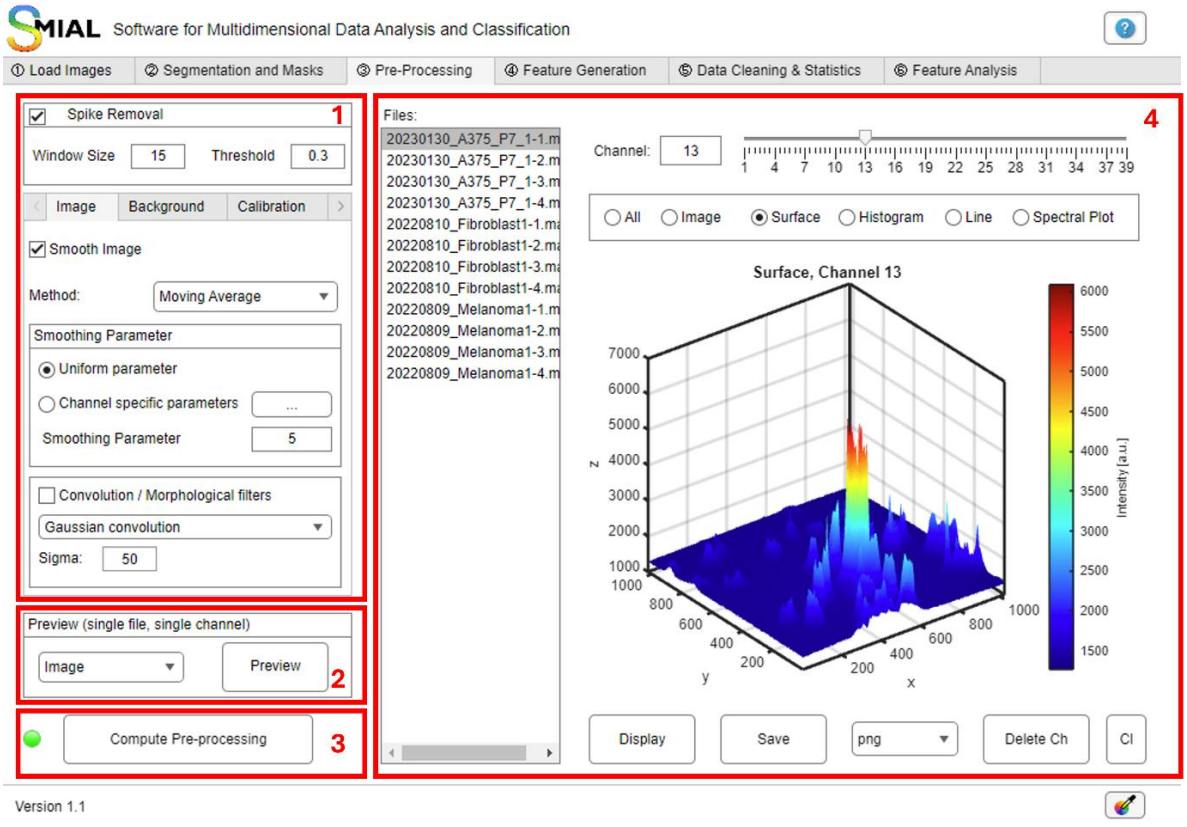

Figure 5-3: Pre-processing panel layout.

5.2 Background Subpanel (optional)

Upload a **background image** taken without a sample, e.g., water for autofluorescence measurements or dark image. This background image will be subtracted from all uploaded images to correct for system-related signal (Equation 1, where *y* is the original image or smoothed image if smoothing was applied, and y_{bkg} is the background image).

- **Background Shifting**: If background masks are provided (see **Section 4**), the **median pixel value** within the user-defined background regions is set to zero to further flatten the image.
- Smoothing of the background image is strongly recommended before subtraction.

> ⚠ *Background image must match the format and number of channels as the original image files uploaded in the Load Images panel.*

$$y_{bkgsub} = y - y_{bkg} \quad (1)$$

5.3 Calibration Subpanel (optional)

This subpanel allows users to upload calibration data to correct image intensities, remove curvature effects, and align measurements with a calibrated standard. Calibration is typically performed using a reference fluid (e.g., NADH, FMN) that produces measurable signals across all excitation/emission wavelength combinations [3]. Equation 2 is applied for calibrating images for *p* pixels *and* c channels, where *y* is the original image (or pre-processed image if previously performed), y_{bkg} is the background image, y_{cal} represents the calibration image and $f(c)$ is the calibration factor for each channel.

$$y_{flat}(p,c) = \frac{f(c)\times\left(y(p,c)-y_{bkg}(p,c)\right)}{y_{cal}(p,c)-y_{bkg}(p,c)} \quad (2)$$

Calibration Inputs

- **Upload Calibration Image:** Upload an image acquired using your imaging system that contains the reference fluid. A correction equation is applied to each imaging channel to standardise intensity values.
- **Load Calibration Factor (Optional):** Upload an Excel file containing calibration factors measured on a reference device (e.g. spectrofluorometer). These factors scale each imaging channel based on the signal intensity of the reference fluid at corresponding excitation/emission wavelengths.

  Expected Excel Format (see Figure 5-4):

  - Column 1: Channel number
  - Column 2: Reference intensity values

	A	B
1	Channel	Fluoromax
2	1	0.07832
3	2	0.07832
4	3	0.048329
5	4	0.128382
6	5	0.088632
7	6	0.319655
8	7	0.258362
9		

Figure 5-4. Expected excel format for inputting calibration factors. The first column lists the channels and the second column contains the corresponding reference intensity values.

5.4 Quality Subpanel (optional)

The **Quality Subpanel** enables users to assess and compare the **pre-processed image** with the **original image** using a range of quantitative metrics. To begin, select the desired metrics and click the **'Quality'** button to perform analysis.

Additionally, the **'Display Raw & Processed'** option allows side-by-side visual comparison of the original and pre-processed images.

Available metrics:

- SNR (signal-to-noise ratio): requires a background mask (from the Background Mask subpanel).

$$SNR = 10\log_{10}\left(\frac{preprocessed_{\max value} - background_{median}}{background_{std}}\right)$$

- MSE (mean squared error): measures the average squared difference between the pre-processed and original image. Lower values indicate higher similarity. In the equation below, n equals the number of pixels.

$$MSE = \frac{1}{n}(Preprocessed - original)^2$$

- PSNR (peak signal-to-noise ratio): evaluates the peak error between the pre-processed and original image. Higher values indicate better quality.

$$PSNR = 10\log_{10}\left(\frac{peakvalue^2_{preprocessed}}{MSE}\right)$$

- RMSE (root mean square error): the square root of MSE, providing error in the same units as the image intensity.

- SSIM (structural similarity index): measures structural similarity between the pre-processed and original image. Values range from 0 (poor) to 1 (excellent).

- MS-SSIM (multi-scale SSIM): an extension of SSIM that evaluates similarity across multiple image scales. Also ranges from 0 to 1, with higher values indicating better quality. Default uses five scales.

The resulting metrics are displayed in the GUI and automatically saved as an excel sheet (‘*Filename_date_preprocessingQuality.xls’*, **Figure 5-5**, each row is a separate channel) in the “*Preprocessing*” folder under the “*Quality*” subfolder.

	A	B	C	D	E
1	SNR Original (O)	SNR Smoothed (S)	SNR S/O [%]	PSNR	SSIM
2	19.82373707	19.32790393	98.34841464	-26.23930391	0.034861445
3	15.58164699	15.61193992	100.1944142	-23.81477561	0.040749691
4	15.61172202	15.44884625	98.95670844	-24.42186802	0.046362446
5	19.62427833	19.59923926	99.8724077	-30.47456653	0.061947151
6	14.59713497	14.6332938	100.2477118	-27.34104206	0.030564812
7	22.88938158	21.22335663	92.72140692	-28.39170038	0.028222101

Figure 5-5. Example exported excel spreadsheet comparing original and pre-processed image quality metrics.

6 Feature Generation Panel

The **Feature Generation** panel allows users to either upload their own numeric feature table or generate imaging features directly from their data using SMIAL. If features are being generated, these are calculated from either the pre-processed image (if the Pre-Processing panel was used) or from the original images uploaded (from the Load Images panel). Additionally, the user must have the associated masks (from the Segmentation and Masks panel). The corresponding workflow of the Feature Generation panel is provided in **Figure 9-4**.

6.1 Load Feature Table

When using this option, select a path to load the feature table. Please follow the following format that enables compatibility with downstream computations. The first four columns must be:

1. **Idx, Area > "threshold value"**
   - Provides an option to the user to remove certain objects based on a known area threshold in pixels (e.g. a small debris that was misclassified as a cellular object) The threshold value is user-defined and filters foreground objects by area.
   - Set this value to **1** for each row if you don’t wish to apply an area-based thresholding to the observations.
2. **Class**
   - Specifies the class label or group for each row/observation (e.g., *melanoma*, *fibroblasts* for non-varying imaging data and 1 or 2 for time-varying data . Both text and numerical values allowed.
3. **File**
   - A filename for each observation.
   - Required for downstream visualisation to correctly link features to their corresponding files. If visualisation is not required, set all entries to **1**.
4. **ID**
   - Identifier for individual objects within an image (e.g. ID_1, ID_2)
   - If not available, use ID_**1** for all entries.

All subsequent columns (5th to last) should contain **quantitative** numeric feature values for each observation, with each column header representing a different feature. Any text features that can be converted into numeric values should be specified outside of SMIAL.

⚠ *If multiple feature tables are uploaded simultaneously in SMIAL, the same number of feature columns must be used and their names must match, otherwise this will produce the following error:*
"Selected feature table files have different feature names. Please check feature names for all files to ensure they match."

Users may also **Define image folder directory to** link the model with the folder where the original images are stored, for downstream visualisation analysis.

6.2 Generate Features

SMIAL can compute **62 features**, grouped into four categories:

- **Intensity**
- **Morphology**
- **Texture**
- **Time**

See **Section 10.1** for a complete list and description of all features.

To generate features:

1. Upload images (Load Images Panel) and corresponding mask files (Segmentation and Masks Panel).
2. Select the desired features from the intensity, morphology, texture subpanels. (See Time-based features in the next section)
3. Ensure the tick boxes for **'Image files'** and **'Mask files'** are auto-populated (**Box 3, Figure 6-1**).
4. Click **'Compute Features' in Box 2**, **Figure 6-1**.

Output:

- A general feature table: *featureTable.xlsx*
- Class-specific tables: featureTable_<categoryName>.xlsx
- The list of features selected to calculate: *featureList.xls*
- A list of any channels deleted by the user that was not included in the feature generation: *DeletedChannels.xls*
- Files are saved in a folder labelled '*Feat*' in the master directory.
- Existing feature tables will be **overwritten** unless renamed.

Time Features

SMIAL supports a range of time-based features that quantify how a value changes across time points. These features are key for tracking changes in spatial features (i.e. morphology, intensity and texture) especially relative to a baseline or previous timepoint.

Each time point is treated as a 'Class', which can be defined in the Class/Time column within the Load Images panel for each image file. It can accept both numeric/text description. For example,

- To compare treated cells at different times, class 1 could represent treatment data at 24 hours, and class 2 at 48 hours.

- To compare treated and untreated cells, Class 1 could be untreated data at 24 hours and Class 2 treated at 48 hours.

To compute Time-based features, the Time toggle in **Box 1 Figure 6-1**, needs to be enabled, otherwise general intensity, morphology and texture features are computed.

Time-based features can be computed in two ways:

1. Relative to the baseline (e.g. t = 0)
2. Relative to the previous time point (e.g. between $t = TimeX$ and $t = TimeX - 1$)

The follow metrics are available where the $TimeReference$ is either the baseline or a previous time point):

- **Difference**:

$$Difference = Feature_{TimeX} - Feature_{TimeReference}$$

- **Fold change:**

$$Fold\ change = \frac{Feature_{TimeX}}{Feature_{TimeReference}}$$

- **Percentage change:**

$$Percentage\ change = \frac{Feature_{TimeX} - Feature_{TimeReference}}{Feature_{TimeReference}} \times 100\%$$

- **Percentage difference:**

$$Percentage\ difference = \frac{\left|Feature_{TimeX} - Feature_{TimeReference}\right|}{\left(\frac{Feature_{TimeX} + Feature_{TimeReference}}{2}\right)} \times 100\%$$

- **Rate of change:**

$$Rate\ of\ change = \frac{Feature_{TimeX} - Feature_{TimeReference}}{TimeX - TimeReference}$$

Time-based features are saved in a separate Excel file.

⚠ *To analyse time-based features, re-upload the feature table via the* ***Load Feature Table*** *subpanel.*

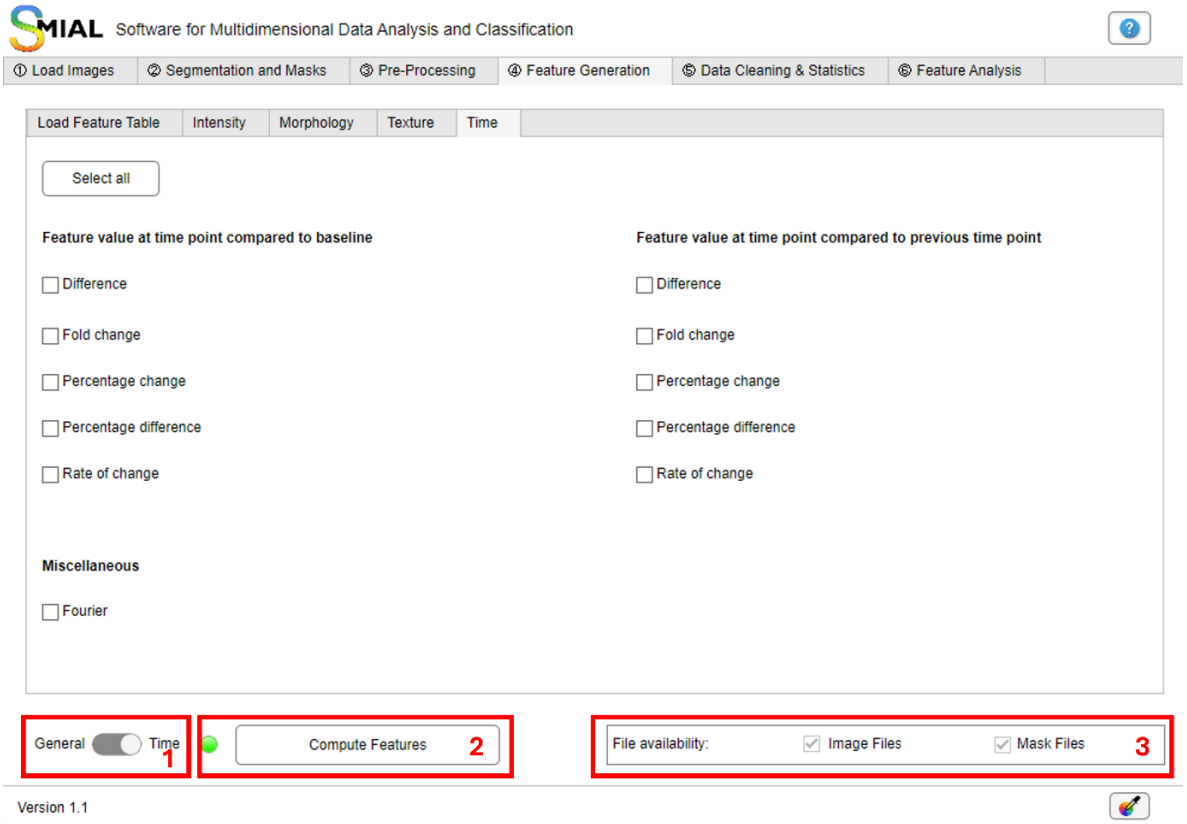

Figure 6-1. Time features available in SMIAL. Box1: Toggle for time features to be computed. Otherwise general intensity, morphology and texture features will be computed if not enabled. Box2: Compute features button to generate feature table of selected features for all image files. Box3: Automatic check if image files and mask files are available to compute features. This will automatically be 'ticked' if image files and masks were uploaded in the Load Images and Segmentation & Masks panels respectively. Features cannot be computed if either are not present.

6.3 Plot Features

The Plot feature tool appears after the features have been generated. It enables the visualisation of feature distributions per imaging channel across class/time groups using the following plot types.

Select the features to be visualised from **Box 1** (**Figure 6-2**) and the plot type **Box 2**. Use the **Display** button to visualise feature distributions (**Box 3**). Click **Save** to export plots to a folder named **'Feat'** in the original data directory as specified in the Load Images Panel or Feature Generation Panel. Available plot types include:

- **Bar graphs** and **line plots**: Shows mean feature values across Class/Time labels.
- **Boxplots**: Displays feature distributions across Class/Time labels.
- **Scatter plot matrix**: Visualises pairwise correlations between features using scatter plots. Requires two or more features selected.
- **Heatmap (mask)**: Displays a colour-coded matrix of feature values onto the selected image in the Files subpanel.
- **Heatmap (overlay)**: Overlays the colour-coded matrix of feature values onto the selected image in the Files subpanel and selected channel using the slider.

6.3.1 Initial Feature Selection

Delete Features: Removes selected features from feature table. Users can manually manage features for further analysis (**Figure 6-2, Box 1**):

- Press the **Backspace** or **Delete** key (hold **Ctrl** to select multiple). Deleted features will be excluded from further computations.
- **Reset Features**: Restores the original feature table, including any deleted features and resets all calculated statistics in the **Data Cleaning & Statistics** panel.
- **Add to Manual Selection**: Adds selected features to the **Manual Feature Selection** list in the **Feature Analysis** panel. Hold **Ctrl** to select multiple features.
- **Delete Manual Selection**: Deletes all features from the **Manual Feature Selection** list in the **Feature Analysis** panel.

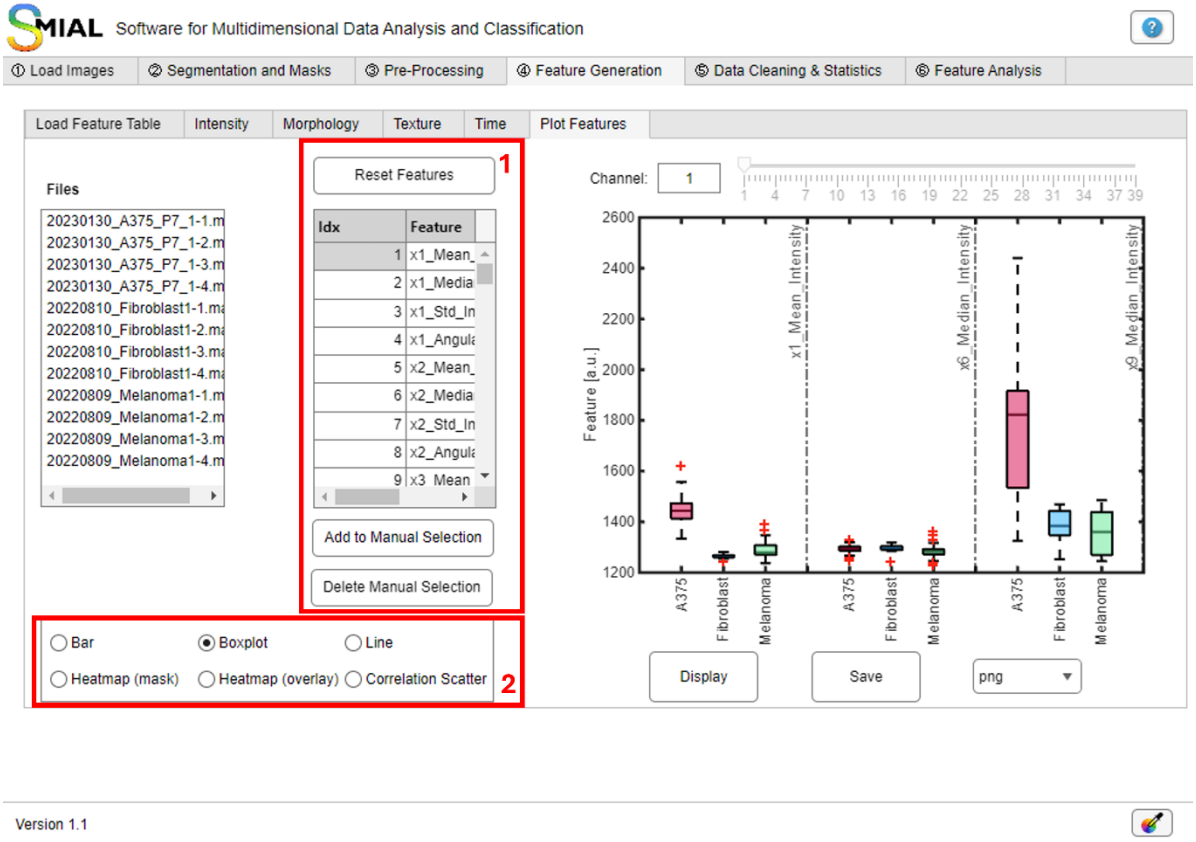

Figure 6-2. Plot Features Subpanel.

7 Data Cleaning & Statistics

The Data Cleaning & Statistics panel enables users to explore, clean, and statistically analyse their feature data to improve the quality and interpretability of downstream analysis. The corresponding workflow of the Data Cleaning and Statistics panel is provided in **Figure 9-5**.

This panel is divided into three main sections:

- Statistical Study (**Figure 7-1**, **Box 1**)
- Compute Statistic (**Figure 7-1**, **Box 2**)
- Display Options (**Figure 7-1**, **Box 3**)

Images and results generated in this panel are saved in a subfolder labelled '**Statistics'**.

⚠ *The **Data Cleaning & Statistics** panel is **optional**, however it is highly recommended to remove values such as not a number (NaN), infinity (Inf) and rows with empty values, otherwise errors may occur in the subsequent **Feature Analysis** panel.*

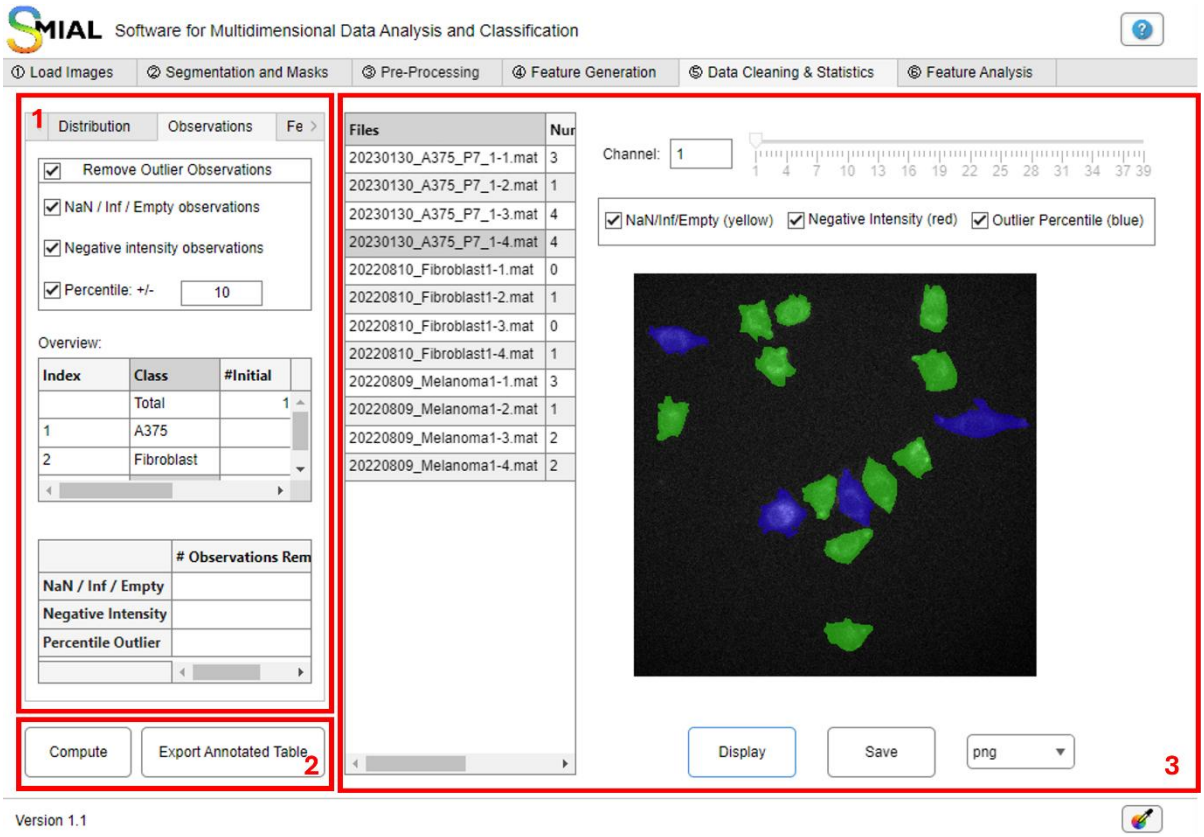

Figure 7-1: Data Cleaning and Statistics panel.

7.1 Statistical Study

This section provides tools for assessing the structure of the feature table, identifying outliers, and evaluating feature significance and correlation. An overview file is saved to the Statistics subfolder within the master directory file path upon each computation. Removed observations are summarised in the file overview, indicating the number of outliers and their type. Additionally, an annotated feature table identifying flagged outliers and excluded features can be exported to the *Statistics* folder using the '**Export Annotated Table**' button.

7.1.1 Distribution

This subpanel enables the users to assess the results of the null hypotheses that the feature data comes from a normal distribution. Users can assess the distribution of feature data using the following statistical tests:

- **One-sample Kolmogorov–Smirnov test**: Recommended for datasets with more than 50 observations
- **Shapiro–Wilk test:** Recommended for datasets with 50 or fewer observations. Additionally, **normal probability plots** can be generated to visually assess the normality of each feature.

7.1.2 Observations

This subpanel allows the users to clean their data by removing outliers and visualise the results:

- **Remove NaN, Inf, and Empty**:
  Removes observations with undefined or non-numeric values. Use only when these values are limited to a small subset of data.
- **Remove Negative Intensity Values**:
  Excludes observations with negative intensity values.
- **Remove Values Within a Percentile Threshold**:
  Filters out observations outside a user-defined percentile range.

Display Options:
Visualise excluded or valid observations in the image viewer (image and mask files required):

- **Yellow**: Objects with NaN, infinity, or empty values
- **Red**: Objects with negative intensity
- **Blue**: Objects with pixel intensities outside percentile range
- **Green**: Remaining valid objects

7.1.3 Features

- **Remove NaN/Inf/Empty features:** Removes values with undefined or non-numeric values (**Figure 7-2**, **Box 1**). Note that for downstream Feature Analysis all NaN, Inf and empty values must be removed otherwise errors will occur in subsequent panels.
- **Significance** (**Figure 7-2**, **Box 2**):

This subpanel allows users to apply statistical tests to filter out features. Users can choose from three methods: parametric (t-test for 2 class/time labels, ANOVA for three or more) and non-parametric (Wilcoxon signed rank test for 2 class/time labels, Kruskal-Wallis for three or more) and automatic (checks for normal or non-normal distribution for parametric or non-parametric study respectively).

Option to remove features with **p-values** above a user-defined threshold, p.

- **Remove Entropy = 0** (**Figure 7-2**, **Box 3**):
  Removes features with **zero Shannon entropy across all classes**, indicating no variability across classes/ Time labels.

  $$Entropy = -\sum p * \log_2 p$$

  where *p* is the normalised histogram count.
- **Remove Standard Deviation = 0** (**Figure 7-2, Box 4**):
  Removes features with **zero standard deviation** across all classes/Time labels.
- **Correlation** (**Figure 7-2, Box 5**):
  Computes pairwise correlation between features: Features with correlation coefficients above a user-defined threshold **r** is removed.
  - **Pearson's** for normally distributed data
  - **Spearman's Rho** for non-normally distributed data

Figure 7-2. Feature subpanel in Data Cleaning & Statistics.

7.1.4 Feature Summary

Feature summary will appear after features have been filtered. This subpanel provides an overview of all features removed during the statistical tests, including the number of features removed per test type.

7.2 Compute

Click **Compute** to apply all selected statistical operations to the feature table. The updated feature table is automatically passed to the **Feature Analysis** panel (see **Section 8**). Click **Export Annotated Table** to export an excel file detailing measured statistical information for all features (**Figure 7-3**).

A *StatsInformation.txt* file containing all statistical analysis performed (e.g. removal of Nan/Inf/Empty observations, outlier removal, removal of insignificant features) is automatically generated and saved in the *Statistics* subfolder. It also contains information on the number of features removed after data cleaning (**Figure 7-4**).

	A	B	C	D	E	F	G	H	I
1	Outlier	Idx, Area > 5	Class	File	ID	x1_Mean_Intensity	x1_Median_Intensity	x1_Std_Intensity	x1_Angular_Second_Moment
2						Zero Entropy	Zero Entropy	Zero Entropy	Highly Correlated
3	Included	1	A375	20230130_A375_P7_1-1.mat	ID_1	1450.3948	1430.6667	91.0872	0.14895
4	Included	2	A375	20230130_A375_P7_1-1.mat	ID_2	1462.593	1452	94.2073	0.2353
5	Included	3	A375	20230130_A375_P7_1-1.mat	ID_3	1472.8626	1459.3333	103.3679	0.15
6	Included	4	A375	20230130_A375_P7_1-1.mat	ID_4	1480.3121	1469.3333	93.9246	0.13859
7	Included	5	A375	20230130_A375_P7_1-1.mat	ID_5	1393.2695	1389.3333	60.2763	0.46154
8	Included	6	A375	20230130_A375_P7_1-1.mat	ID_6	1551.8836	1544.3333	131.0166	0.21271
9	Included	7	A375	20230130_A375_P7_1-1.mat	ID_7	1445.7277	1430	87.8022	0.12323

Figure 7-3. Example of the exported statistics annotated excel table.

```
Stats Analysis, 2024-12-10_15-49
------
Data distribution: Automatic = not normal
Computation method: One-Sample Kolmogorov-Smirnov Test
------
Observations:
- NaN / Inf / Empty: 0 observations removed
- Negative intensity: 0 observations removed
- Lower and upper percentile: 34 observations removed
  -> Percentile: ±1
------
Number of observations:
Class1 -> initial observations: 468, post-removal: 460
Class2 -> initial observations: 844, post-removal: 828
Class3 -> initial observations: 574, post-removal: 564
------
Features:
- Insignificant removed (p = 1e-05): 15, 6
  (Computation method: Kruskal-Wallis test)
- Zero entropy removed: 1, 2, 3, 4, 7, 9, 11, 12, 16, 17, 18, 19
- Zero standard deviation removed: -
- Correlation removed (r = 0.6): 5, 13, 14
  (Computation method: Spearman correlation)
------
Number of features:
-> initial number of features: 19
-> post-removal number of features: 2
------
```

Figure 7-4. Example of the StatsInformation.txt file.

8 Feature Analysis Panel

The **Feature Analysis** panel allows users to either upload a pre-trained classification model from SMIAL or build a new SMIAL model directly from the latest feature table data. This panel provides tools to perform model training, validation, and application, with options for normalisation, feature selection, and class balancing. Some optional visualisation options require availability of image files. The corresponding workflows of the Feature Analysis panel is provided in **Figure 9-6** and **Figure 9-7**. Please note that a feature table must be either generated or imported as detailed in **Section 6**, to compute a classification model.

8.1 Load Pre-built Model

Users can upload a previously trained classification model and associated data for evaluation in the **Load Pre-built Model** *subpanel* (**Figure 8-1**). The button **'Apply Model'** applies a previously uploaded classification model to the current dataset.

- **Load Model**: allows users to upload a **.mat** file containing a classification model (Mdl.mat) previously generated using SMIAL. This file also stores relevant analysis parameters and settings, which are essential for accurately regenerating and reproducing the model and its associated outputs.
- **Load Data Model**: enables users to upload a **.mat** file containing the feature table and corresponding class labels (Data.mat) that was used with the pre-trained model.
- **Define image folder directory**: link the model with the folder where the original images are stored, for downstream visualisation analysis.

Usage scenarios:

- Regenerating results: uploading both the model and its associated data allows users to recreate plots and performance metrics from a previous analysis.
- Applying to new data: users may also upload a pre-trained model alongside a new feature table (either generated or uploaded via the **Feature Generation** panel, see **Section 6**), to apply the model to fresh data and evaluate its performance.

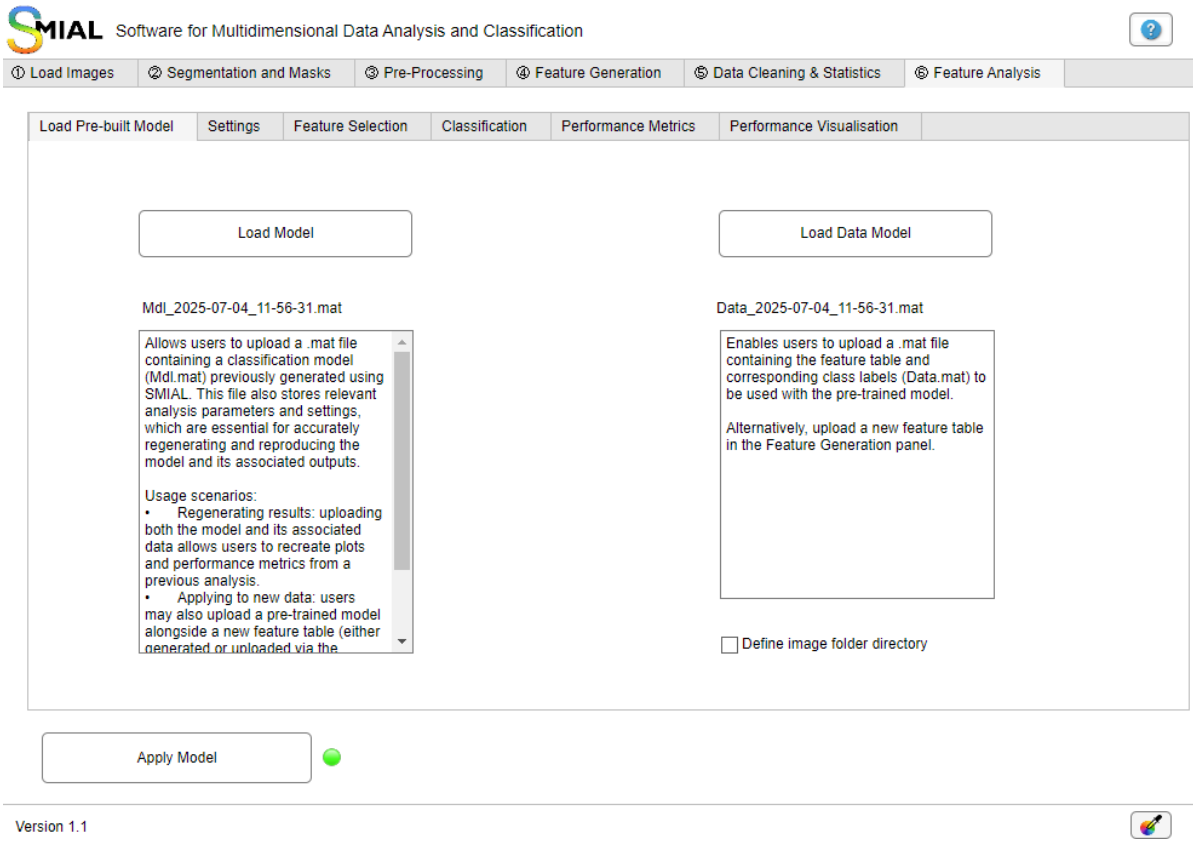

Figure 8-1. Load Pre-built Model Subpanel.

8.2 Compute Model

Users can build a new classification model from a feature table using the **'Compute Model'** button. The model is generated based on the parameters specified under **Settings**, **Feature Selection**, and **Classification** subpanels.

8.2.1 Settings

Validation Methods

Allows the user to select a model validation strategy using (**Figure 8-2, Box 1**):

- **Hold-out**: Splits data into training and test sets based on *X* percentage of the Training and Test datasets.
- **Cross-validation** (default): Splits data into *k* folds where *k* is user defined. *k* must be less than the minimum number of observations per class.
- **Custom folds** (optional): Allows uploading an Excel spreadsheet defining custom fold assignments. The file must contain one column with as many rows as observations. Each

entry should be an integer indicating the assigned fold (e.g. for 10-fold cross-validation, use integers 1–10).

- **Leave-one-out**: Uses each data point once as a test sample. Ideal for small datasets.

Normalisation

Allows the user to choose from five normalization methods (**Figure 8-2, Box 2**):

- **Z-score** (default)

$$I_{Zscore} = \frac{I - I_{mean}}{I_{std}}$$

- **Min-Max**

$$I_{min-max} = \frac{I - I_{min}}{I_{max} - I_{min}}$$

- **Robust Scaling**

$$I_{robustscaling} = \frac{I - I_{median}}{I_{Interquartile\ range}}$$

- **Unit Vector**

$$I_{unitvector} = \frac{I}{\sqrt{\sum I^2}}$$

Z-score normalization is recommended when using PCA, to standardise the data and ensure that all variables have equal contribution to the analysis.

Other

These options provide additional control over how data is handled during model training and evaluation (**Figure 8-2, Box 3**):

- **Class Labels Provided**:
  Indicates whether the uploaded feature table/s includes class labels, which are necessary for supervised classification and performance evaluation.
- **Correct Unbalanced Sample Size**:
  Addresses class imbalance by applying either **Upsampling** (increasing samples from the minority class) or **Downsampling** methods (random downsampling of majority class), helping to prevent biased model training.

Upsampling Options

Upsampling addresses class imbalance by synthetically increasing the number of samples in underrepresented classes. When working with imbalanced data, users can use the following upsampling methods.

- **SMOTE** (Synthetic Minority Over-sampling Technique): Generates synthetic samples between nearest neighbours.

(GitHub: https://github.com/dkbsl/matlab_smote/releases/tag/1.0)

- **Borderline SMOTE**: Focuses on samples near the decision boundary.
- **Safe-level SMOTE**: Assigns a "safe level" to each instance and generates samples near the safest ones.
- **ADASYN** (Adaptive Synthetic Sampling): Adds more variance to synthetic samples by introducing randomness.

> ⚠ *If all neighbouring points are minor class samples, only SMOTE is available.*

Downsampling Options

Downsampling addresses class imbalance by randomly removing samples from the overrepresented class to match the number of samples in the minority dataset.

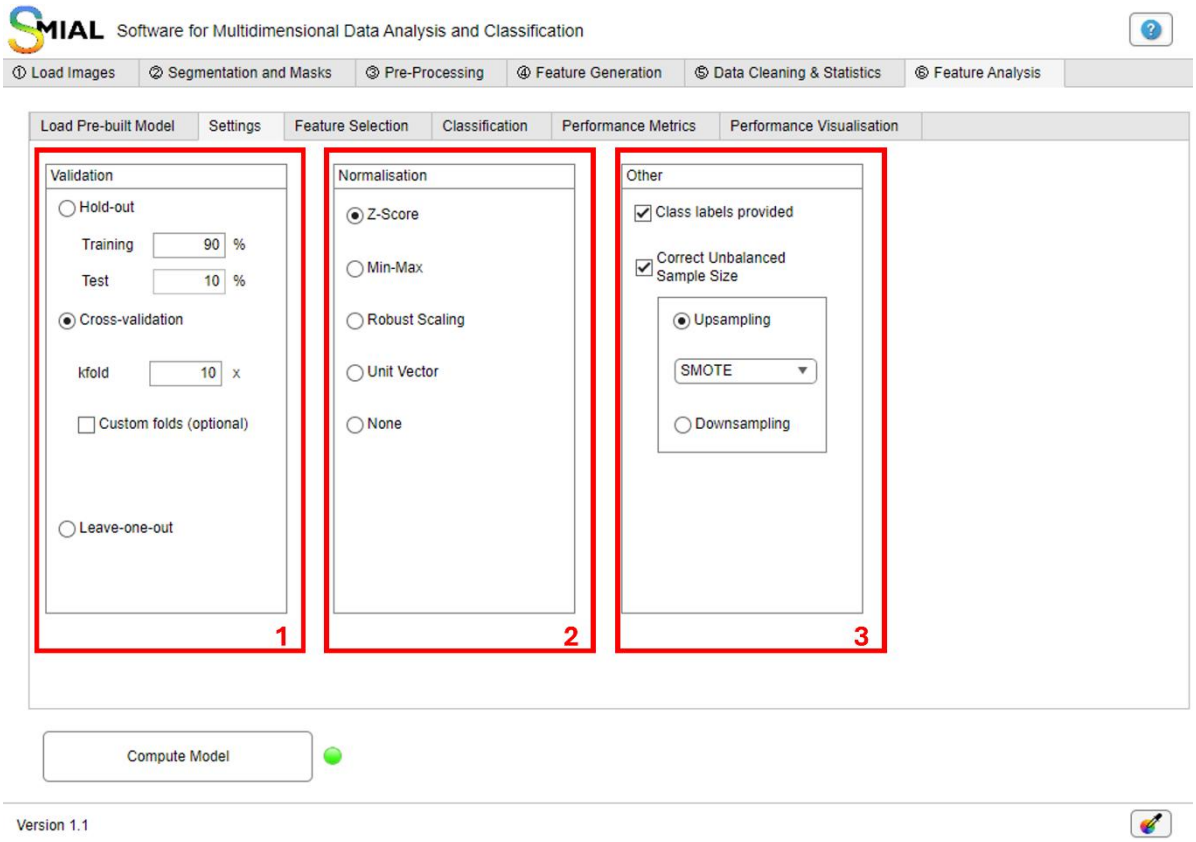

Figure 8-2. Feature Analysis Settings subpanel.

8.2.2 Feature selection

Once a feature table is loaded or updated, users can perform feature selection to improve model performance. When *k*-fold cross-validation is applied, the feature selection method is applied to each fold. An excel file containing the selected features from each fold is automatically exported and saved in a subfolder named **'Classification'** is created in the master directory.

8.2.2.1 Number of features

Users can define how many features to include in the model using one of the following options (**Figure 8-3 Box 1**):

- **User Input**: Manually specify the number of features.
- **Square Root of Observations** (default): The number of features equal to the square root of the number of observations in the minority class.
- **All**: Uses all available features. Caution is advised when the number of features exceeds the number of observations, as this may lead to overfitting.

8.2.2.2 Feature selection methods

Users may skip feature selection by choosing **None**, or select from the following methods (**Figure 8-3 Box 2**):

- **Chi-square**: Ranks features using chi-square statistics by looking at how strongly a feature is related to a class. Higher chi-square scores considered more relevant for classification.
- **Gini Index**: Ranks features based on how well they separate two classes by measuring impurity (measurement of how mixed the classes are within a group). A lower impurity indicates the group contains mostly one class, suggesting the feature is more informative. This is a non-parametric method.
- **Forward Feature Selection**: Iteratively adds features to the classification model one at a time. At each iteration, the feature that leads to the greatest model performance is added. The user can choose the evaluation metric used to guide selection: AUC, F1 score, correctly labelled %, sensitivity, specificity (see Section 8.3 for definitions).

  ⚠ Requires user-defined **number of folds (k-fold)**.

  💡 Computationally intensive: best used after initial feature reduction using the Data Cleaning and Statistics panel.
- **Laplacian**: Ranks features based on variance and similarity in a nearest-neighbour graph. It selects features that do not change drastically between classes.
- **MRMR (Maximum Relevance Minimum Redundancy)**: Selects features that are highly relevant yet minimally redundant to each other. Maximum relevance selected features have a strong statistical relationship, whilst minimum redundancy avoids selecting features providing repetitive information.
- **Particle Swarm**: Optimises feature selection based on Fisher distance (between-class vs. within-class variance). A higher Fisher distance indicates better class separation, making the selected features more unique.
- **Pearson, ANOVA**: Combines two selection methods through ANOVA (analysis of variance) to assess statistical significance of each feature, and Pearson correlation to filter out highly correlated features (eliminating redundant features). Requires user-defined correlation threshold (**r**).

- **ReliefF**: Ranks features based on their ability to distinguish between classes using k-nearest neighbours, where *k* refers to the number of nearest data points considered for each instance. Features that show different values for neighbours from different classes, and similar values for neighbours within the same class are ranked higher and deemed more informative for classification.
- **Significance:** Selects features with the highest statistical significance. Depending on the data, different significant tests can be applied: parametric (normal distribution) and non-parametric (non-normal distribution). Note 'Automatic' checks for normal or non-normal data distribution for parametric or non-parametric study respectively.
- **Bhattacharyya**: Statistical measure to evaluate how effectively a feature can distinguish between two classes. It quantifies the similarity between the probability distributions of the two groups (assuming normal distribution). A higher Bhattacharyya distance suggests greater separation between the classes. Particularly suited for binary classification tasks, where only two distinct classes are involved.
- **Entropy**: Measures relative entropy (Kullback-Leibler Divergence) of each feature between classes. A greater Kullback-Leibler Divergence value suggests better distinction between the classes. This assumes the data follows a normal distribution and is only applicable to binary classification problems.
- **ROC**: Assesses how well a feature can distinguish between two classes by plotting the true positive rate against the false positive rate at various threshold settings. The area under the curve (AUC) is measured where a greater AUC value indicates better class separation. This method is non-parametric and is for binary classification only.
- **t-test**: Statistical approach comparing the means of each feature across two classes to determine if they are statistically significant. The absolute value of the t-test is used as the ranking criterion, with a greater absolute t-value, indicating the feature has a stronger discriminative power. This method assumes each class is normally distributed and is only for binary classification problems.
- **Wilcoxon**: Also known as the Mann-Whitney U Test, is a non-parametric test for feature ranking. It assesses whether the distribution of the feature values differ significantly between two classes. This method does not assume that the data follows a normal distribution and is only for binary classification problems.
- **Manual**: Users can directly specify which features (columns) to include in their analysis by inputting their indices, e.g. 1, 28, 37. This approach gives full control to the user, enabling selection based on domain knowledge or prior experience. Note that the number of inputted features must match the number in **User Input** for number of selected features.

*Indices can be referenced from the **Plot tab** in the **Generate Features** panel. Separate multiple indices with commas.*

- **Multi-select** (default): Ranks features based on their scores across all applied feature selection methods. Binary selection methods are only used for binary classification tasks. Manually added features are not considered.

8.2.2.3 Feature extraction

Feature extraction transforms the original feature space into a new subspace, reducing dimensionality whilst preserving the data's variance. Users may skip this step by selecting '**None**' or selecting one of the available methods (**Figure 8-3 Box 3**).

Supervised Methods:

1. **LDA (Linear Discriminant Analysis)**: projects data onto a lower-dimensional space by maximising class separation.
2. **PCA + LDA** (default): combines principal component analysis to decorrelate data, followed by LDA to enhance class discrimination.

Unsupervised Methods:

1. **PCA (Principal Component Analysis):** dimensionality reduction method that transforms the data into principal components. Requires a cut-off value (default = 0.95), representing the percentage of variance to retain.
2. **Sparse Filtering:** an unsupervised method that learns features by making them sparse and varied to highlight important patterns in the data.

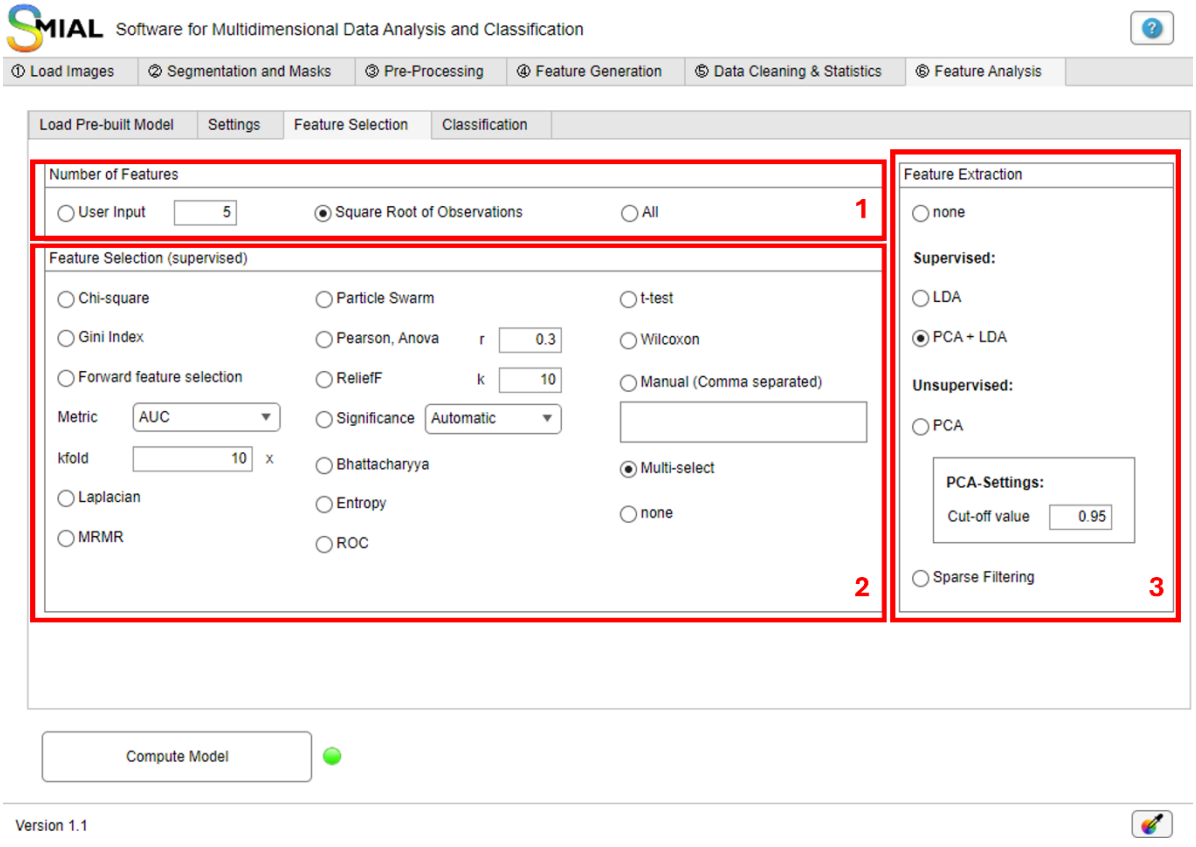

Figure 8-3. Feature Analysis Feature Selection subpanel.

8.2.3 Classification

The **Classification** subpanel (**Figure 8-10**) allows users to select and apply a classification algorithm to separate data into defined classes. Classification methods are grouped into two categories:

Supervised Classification

Supervised classification methods require class labels for each image file, as outlined in **Section 3**. Note that two or more distinct class labels are required for supervised classification. SMIAL offers nine classifiers to choose from (**Figure 8-10 Box 1**).

- **SMV (linear)** (default): a linear support vector machine (SVM) finds the best straight-line (hyperplane) that separates data into classes with a maximum margin.
- **SVM (polynomial)**: a polynomial SVM is similar to linear SVM, however uses polynomial functions to separate data that is not linearly separable.
- **kNN**: k-Nearest Neighbours classifies data points based on the majority class of their closest neighbours.
- **LDA**: linear discriminant analysis projects the data onto a lower-dimensional space to maximise class separation.
- **Logistic Regression**: models the probability of a class using a logistic function, working better for binary classification problems.
- **Naïve Bayes**: a probabilistic classifier based on Baye's Theorem with the assumption that features are independent to classify data.
- **QDA:** quadratic discriminant analysis (QDA) is like LDA, however allows each class to have its own covariance matrix, making it have more flexible boundaries to capture class-specific variability.
- **Random Forest:** an ensemble method that builds multiple decision tress and combines their outputs for more accurate classification.
- **Multi:** runs all available supervised classifiers in a batch mode and automatically selects the best performing model based on a chosen performance metric.

When cross-validation is applied, the resulting model represents the median performance across *k*-folds, based on a user selected performance metric: AUC, F1 score, correctly labelled %, sensitivity, specificity (see **Section 8.3** for definitions). The table below provides suggestions on which performance metric could be used in various scenarios.

Table 1. Use cases and limitations of the available classification metrics.

Metric	Usage Scenario	Limitations
AUC	Evaluating overall model discrimination across all thresholds.	Can be misleading when there is severe class imbalance.

F1 Score	Handling class imbalance; prioritising cost-sensitive false negatives/false positives.	Ignores true negatives; may not reflect full performance.
Accuracy	When classes are balanced; general performance check	Biased towards the majority class; poor with class imbalance.
Sensitivity	When missing positives is critical (e.g. disease detection)	May reduce precision; ignores false positives.
Specificity	When false positives need to be minimised.	May miss true positives; ignores false negatives.

Unsupervised Classification

These methods do not require class labels. Users must define the **expected number of clusters** or enable **automatic cluster detection**.

Available methods (**Figure 8-10 Box 2**):

- **k-Means**: clustering algorithm that partitions the data into *k* groups to minimise the distance between data points and the centre of their assigned cluster.
- **Fuzzy C-Means**: an extension of k-Means, however allows data points to belong to multiple clusters.

⚠ If no labels are available, this must be specified in the **Settings** tab.

⚠ If labels are provided, the number of clusters must not exceed the number of defined groups.

Hyperparameter Tuning

SMIAL supports hyperparameter tuning for all supervised classifiers except Quadratic Discriminant Analysis (QDA) and the Multi classifier. When enabled (**Figure 8-10 Box 3**), hyperparameter optimisation is applied to within each fold during k-fold cross-validation, ensuring the most suitable parameters are selected for each training subset. This process uses Matlab's built-in optimisation framework, using Bayesian optimisation to minimise a specified loss function (typically the cross-validation loss) over multiple iterations (30 by default). Users can customise the cross-validation data partition indices. While the hyperparameter tuning models automatically handle data partitioning, the user must specify the number of folds (k) if not set. The tuned hyperparameters can be accessed via the myMdl.mat output file.

If hyperparameters tuning is not selected, the classifiers will use Matlab's default hyperparameter settings (see https://au.mathworks.com/help/stats/classreg.learning.paramoptim.hyperparameteroptimizationoptions.html)

⚠ Hyperparameter tuning can enhance classification performance. However, for smaller datasets, caution is advised as hyperparameter tuning may increase model complexity and lead to overfitting.

For larger datasets, hyperparameter tuning can significantly increase computational time.

Output files

After computing the model by clicking *'Compute model'* (**Figure 8-10 Box 4**), a subfolder labelled **'Classification'** is created. This folder contains all essential files for model reuse, reproducibility and documentation including:

- **Mdl_Date.mat**: the trained classification model (can be reloaded via the **Load Pre-built Model** subtab). This file (**Figure 8-4**) includes two models:
  - Training/Validation Model: built using a split of the dataset into training and test sets, used for evaluation model performance (see Section 8.3).
  - Final model: trained on the entire dataset and intended for application to new, unseen data.

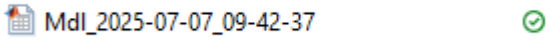

Figure 8-4. Example Matlab Model file.

- **Data_Date.mat**: the data used for training the classifier (can be reloaded via the **Load Data** tab).

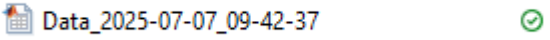

Figure 8-5. Example Matlab Data file.

- **Selected_features.xlsx**: two separate files displaying the list of features used in the validation classification model (Figure 8-6) and the final classification model (Figure 8-7).

	A	B	C	D	E	F	G	
1	Index (fold 1)	Feature Name (fold 1)	Index (fold 2)	Feature Name (fold 2)	Index (fold 3)	Feature Name (fold 3)	Index (fold 4)	Feature N
2	1225	x43_Skewness	5753	x199_75th_Percentile	5724	x198_75th_Percentile	5223	x181_Std_
3	4558	x158_Maximum_Intensity	5782	x200_75th_Percentile	5666	x196_75th_Percentile	5233	x181_Vari
4	1254	x44_Skewness	5666	x196_75th_Percentile	5753	x199_75th_Percentile	5252	x182_Std_
5	1283	x45_Skewness	5637	x195_75th_Percentile	5782	x200_75th_Percentile	5262	x182_Vari
6	1196	x42_Skewness	5695	x197_75th_Percentile	5637	x195_75th_Percentile	5753	x199_75th

Figure 8-6. Selected features in the validation model

	A	B
1	Index	Feature Name
2	66	x17_Median_Intensity
3	31	x8_Std_Intensity
4	153	Area_unit
5	35	x9_Std_Intensity
6	28	x7_Angular_Second_Mo

Figure 8-7. Selected features in the final model

- **Model_information.txt**: A comprehensive log of all parameters and settings using during model training including:

- Validation method
- Normalisation method
- Feature selection and extraction methods
- Classification mode and algorithm
- Hyperparameters on/off

```
myMdl, 2025-07-07_09-42-37
# Input features for classification: 6515
(# Manually deleted features: 0)
------
Validation method: 5-fold cross-validation
------
Normalisation method: Z-Score
------
Number of selected features: 5
Feature selection method: Gini index
Selected features:
5753    x199_75th_Percentile;
5782    x200_75th_Percentile;
5666    x196_75th_Percentile;
5637    x195_75th_Percentile;
5695    x197_75th_Percentile
------
Feature extraction: pca
→ Number of selected PCs: 2
(PCA cut-off: 0.95)
Classification mode: Supervised
Classifier: Polynomial SVM (selected based on performance metric: AUC, hyperparameter tuning)
------
```

Figure 8-8. Example exported model information text file.

- **PredictedClass.xls**: excel file showing the predicted class per observation for training and test data.

	A	B	C	D
1	Filename	ID	Partition	Predicted Class
2	20230130_A375_P7_1-1.mat	ID_1	training	A375
3	20230130_A375_P7_1-1.mat	ID_2	training	A375
4	20230130_A375_P7_1-1.mat	ID_3	training	A375
5	20230130_A375_P7_1-1.mat	ID_4	training	A375
6	20230130_A375_P7_1-1.mat	ID_5	training	Melanoma
7	20230130_A375_P7_1-1.mat	ID_6	test	A375
8	20230130_A375_P7_1-1.mat	ID_7	training	A375
9	20230130_A375_P7_1-1.mat	ID_8	training	A375
10	20230130_A375_P7_1-1.mat	ID_9	test	A375
11	20230130_A375_P7_1-1.mat	ID_10	training	A375

Figure 8-9. Predicted class for each observation excel file.

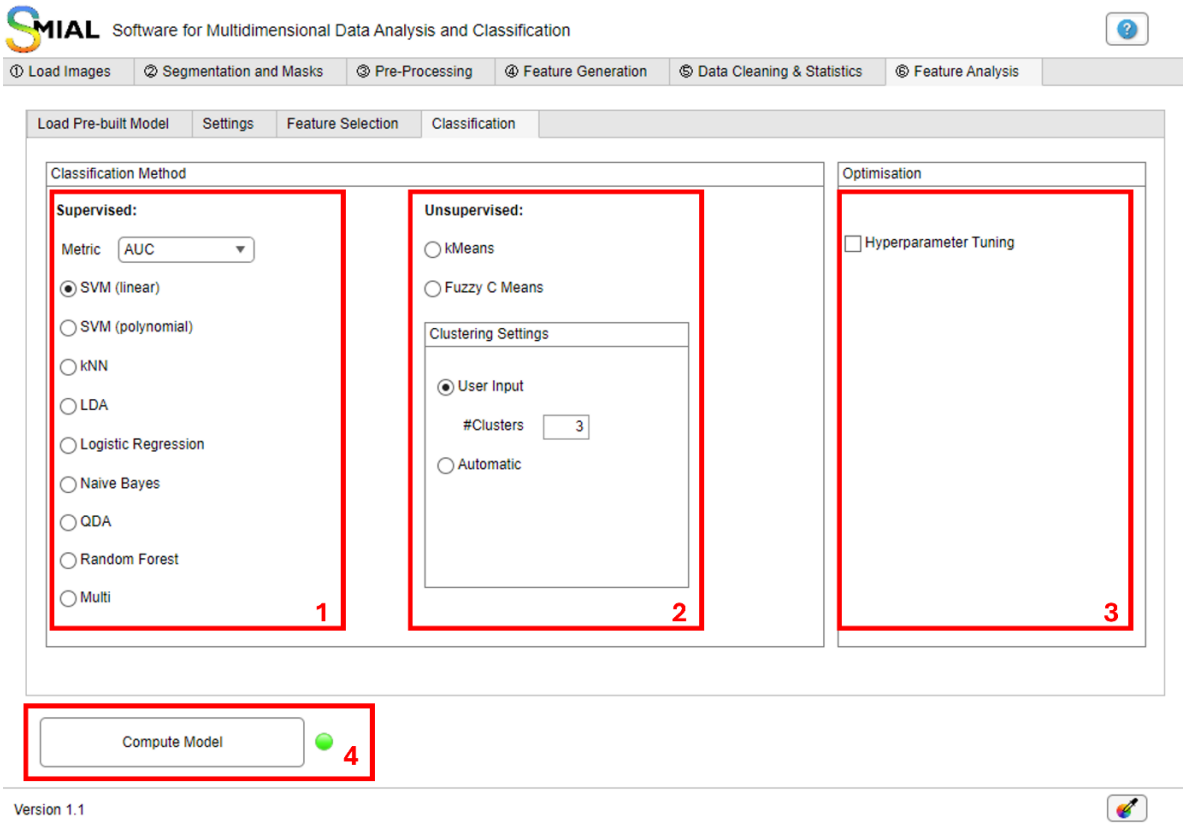

Figure 8-10. Classification subpanel.

8.3 Performance Metrics

The **Performance Metrics** subpanel becomes available after a classification task is completed. It allows users to evaluate model performance using up to **12 metrics** that can be measured for the test, training or total (combined training and test) dataset:

Available Metrics

- **AUC**: area under the receiver operating characteristics curve.
- **Confusion Matrix**: displays true/false positives and negatives:
  - **TP: True Positives**
  - **TN: True Negatives**
  - **FP: False Positives**
  - **FN: False Negatives**
- **Balanced Accuracy**: ranges from 0 to 1, indicating poor to perfect performance respectively.

$$Balanced\ accuracy = \frac{sensitivity + specificity}{2}$$

- **Correctly Labelled** (%):

$$Accuracy = \frac{\#\ correctly\ identified\ samples}{\#\ total\ samples} * 100\%$$

- **Diagnostic Odds Ratio**:

$$DOR = \frac{sensitivity * specificity}{(1 - sensitivity)(1 - specificity)}$$

- **F1 Score**:

$$F1 = 2 \times \frac{precision \times recall}{precision + recall}$$

- **False Positive Rate (FPR):**

$$FPR = \frac{FP}{FP + TN}$$

- **Negative Predicted Value (NPV):**

$$NPV = \frac{TN}{TN + FN}$$

- **Positive Predicted Value (PPV):**

$$PPV = \frac{TP}{TP + FP}$$

- **Sensitivity**:

$$Sensitivity = \frac{TP}{TP + FN}$$

- **Specificity**:

$$Specificity = \frac{TN}{TN + FP}$$

Compute Performance

- Once metrics are selected (**Figure 8-12 Box 1**), click **Compute Performance** to generate a performance table (**Figure 8-12 Box 2**). This table is exported and saved as '*Validation_Date.xls*'.

Row	Class1_training	Class2_training	Class3_training	Class1_test	Class2_test	Class3_test	Cl
AUC	0.8377±0.055	0.8088±0.0116	0.8321±0.0237	0.7261±0.1079	0.686±0.0503	0.7116±0.065	
Balanced Accuracy	0.6063±0.0462	0.5823±0.0599	0.6187±0.0367	0.6092±0.0715	0.5233±0.0924	0.5736±0.072	
F1 Score	0.3404±0.1362	0.285±0.1803	0.7922±0.0151	0.3571±0.2074	0.1818±0.2208	0.7551±0.0397	
Diagnostics Odds Ratios	Inf±NaN	13.7394±11.2804	Inf±NaN	32.6471±NaN	1.6786±2.7466	3.5238±NaN	
False Positive Rate	0.0044±0.0061	0.0295±0.0256	0.7501±0.079	0.0089±0.0194	0.0784±0.0562	0.7778±0.1334	
True Positive Rate	0.217±0.095	0.1942±0.1254	0.9875±0.0204	0.2273±0.144	0.125±0.2167	0.925±0.0523	
Negative Predicted Value	0.8663±0.0153	0.7941±0.0232	0.9467±0.0869	0.8672±0.0195	0.7705±0.0432	0.6667±0.2528	
Positive Predicted Value	0.9267±0.1011	0.6827±0.181	0.6617±0.021	0.8333±0.25	0.3333±0.2887	0.6379±0.0481	
Correctly Labelled [%]	21.6993±9.4957	19.4154±12.5408	98.75±2.0373	22.7273±14.4049	12.5±21.6654	92.5±5.2291	
Precision	0.9267±0.1011	0.6827±0.181	0.6617±0.021	0.8333±0.25	0.3333±0.2887	0.6379±0.0481	
Sensitivity	0.217±0.095	0.1942±0.1254	0.9875±0.0204	0.2273±0.144	0.125±0.2167	0.925±0.0523	
Specificity	0.9956±0.0061	0.9705±0.0256	0.2499±0.079	0.9911±0.0194	0.9216±0.0562	0.2222±0.1334	

Figure 8-11. Example exported Validation excel sheet

- If **Confusion Matrix** is selected, it will appear in the adjacent figure panel (**Figure 8-12 Box 3**) and are automatically saved in the selected file format (PNG, JPG, TIFF, FIG).

It is highly recommended for the user to look at a variety of performance metrics to fully evaluate the model.

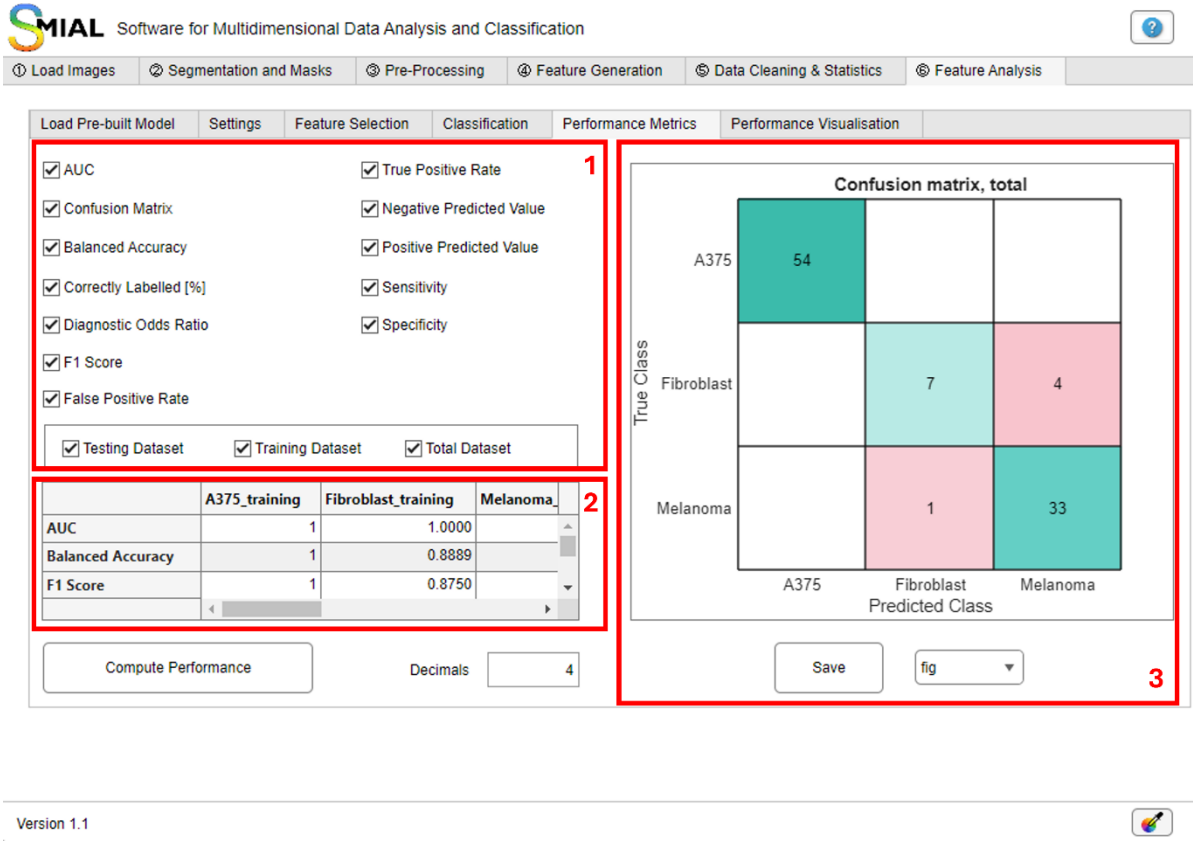

Figure 8-12. Feature Analysis Performance Metrics subpanel.

8.4 Performance Visualisation

The Performance Visualisation subpanel allows users to explore and save visual representations of classification results. Users can view training, test, combined, and final datasets (**Figure 8-13 Box 1**), and identify misclassified data points in 2D or 3D cluster plots (**Figure 8-13 Box 2**). These dataset options are defined as follows:

- **Test**: Data used for evaluating model performance during validation.
    - Scatter plots display results from the test data of the most representative (median) model produced during training.
    - ROC and Precision-Recall (PR) curves aggregate test data across all validation folds if available.
- **Training**: Input data used to train the most representative (median) model during the validation phase.
- **Combined**: Includes both training and test data from the most representative (median) model developed during the validation process.

- **Final**: The complete dataset used to train the final model, intended for deployment on new datasets.

Figures can be saved in PNG, JPG, TIFF, or FIG formats.

8.4.1 Plot Classification

This subpanel allows users to select the type of plot they want to generate (**Figure 8-13 Box 3**) using the options below, where figures are displayed in the adjacent display panel (**Figure 8-13 Box 4).**

8.4.1.1 Classification

ROC Curve: Displays the Receiver Operating Characteristic (ROC) curve for all classes. Options include:

- Average ROC (micro and macro averaging): computes the average ROC curve using either micro averaging (aggregates the contributions from all classes to compute the average metrics) or macro averaging (metrics are computed individually for each class then the average it taken).
- Confidence Intervals: represents the uncertainty of the ROC curve.
- Diagonal Line: reference line from (0,0) to (1,1) that represents the performance of a random classifier.

Precision-Recall Curve: Shows the trade-off between precision and recall, with optional confidence intervals.

8.4.1.2 PCA Visualisations

- **Scatter Plot:** Displays data points in PCA-reduced space.
- **Biplot**: Combines PCA scores and loadings to show feature contributions.
- **Scree Plot**: Shows the variance explained by each principal component.

8.4.1.3 Final Transform

- **Scatter Plot**: Displays final transformed data with each class in a different colour.
    - Optionally, plot Standard Deviational Ellipses for each class and highlight misclassified data points.
- **IoU (Intersection over Union):** Measures overlap between ellipses:
    - **All Classes**: Computes IoU for all class pairs.
    - **Selected Classes**: Input class indices (e.g. 2,4) to compute IoU between specific classes.
    - Results are shown in the **'IoU (%)**' field after clicking '**Display'**.

8.4.1.4 t-SNE

Visualises high-dimensional data in 2D using t-distributed Stochastic Neighbour Embedding. Optionally, mislabelled data points can be highlighted for easier identification.

8.4.1.5 Feature contribution

- **Swarmchart:** Scatter plot of the SHAP summary for each class. Each point represents a single observation, where its SHAP magnitude indicates the influence of the feature on the model's prediction.
- **Bar plot:** SHAP values quantifying the contribution of each selected feature.

Figure 8-13. Feature Analysis Performance Visualisation subpanel.

8.4.2 Classification Overlay

This tool overlays classification results directly onto the original image files. It requires image files and masks to be available.

- **Select Image and Channel**: Choose an image and channel from the file table.
- **Display**: Shows the image with objects colour-coded by class (e.g., red = group1, blue = group2).
  Requires both **image** and **mask** files.
- **Save**: Saves the current view (including zoom or rotation) in the selected format.

Note this feature is not available for time-based features and will generate an error.

9 Workflow diagrams

The following subsections provide general workflow diagrams for using SMIAL with various types of input files. These diagrams outline the recommended sequence of steps and the corresponding panels within the SMIAL interface.

Each workflow is designed to guide users through specific panels in SMIAL:

- Workflow 1: Load images panel
- Workflow 2: Segmentation and masks panel
- Workflow 3: Pre-processing panel
- Workflow 4: Feature generation panel
- Workflow 5: Data cleaning and statistics panel
- Workflow 6: Feature analysis panel - creating a new classification model
- Workflow 7: Feature analysis panel – using pre-generated classification models

9.1 Workflow 1: Load Images

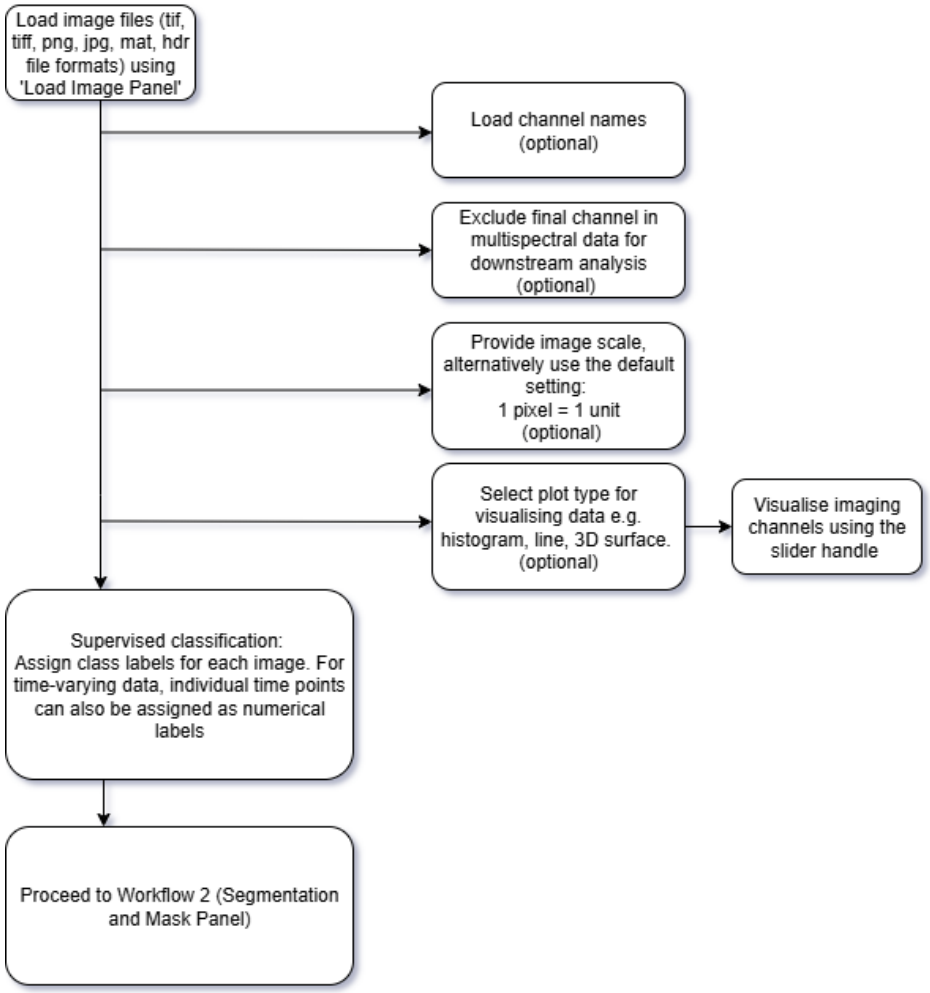

Figure 9-1. Workflow 1: Load and visualise imaging data using the 'Load Images' panel.

9.2 Workflow 2: Perform segmentation and/or evaluate segmentation performance

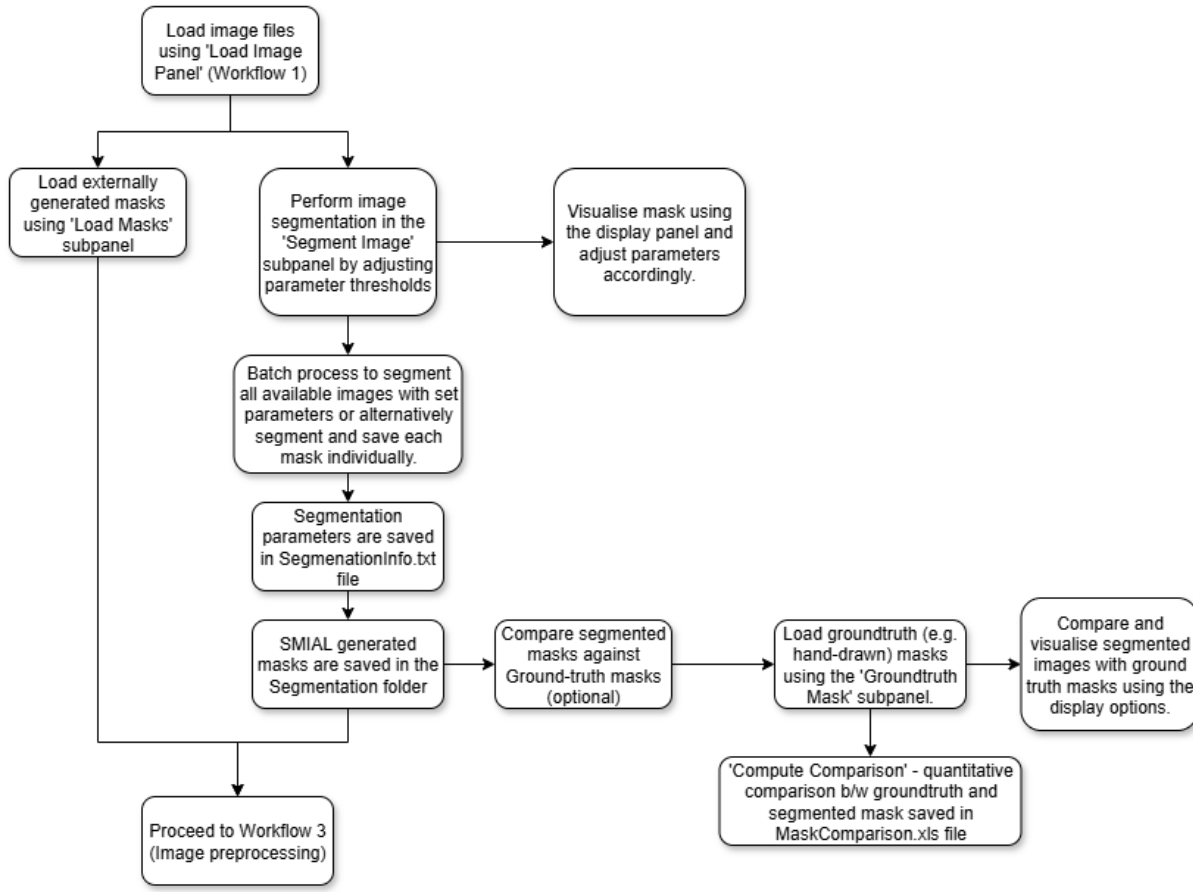

Figure 9-2. Workflow 2: Perform image segmentation and compare with ground truth masks using the 'Segmentation and masks' panel.

9.3 Workflow 3: Perform pre-processing

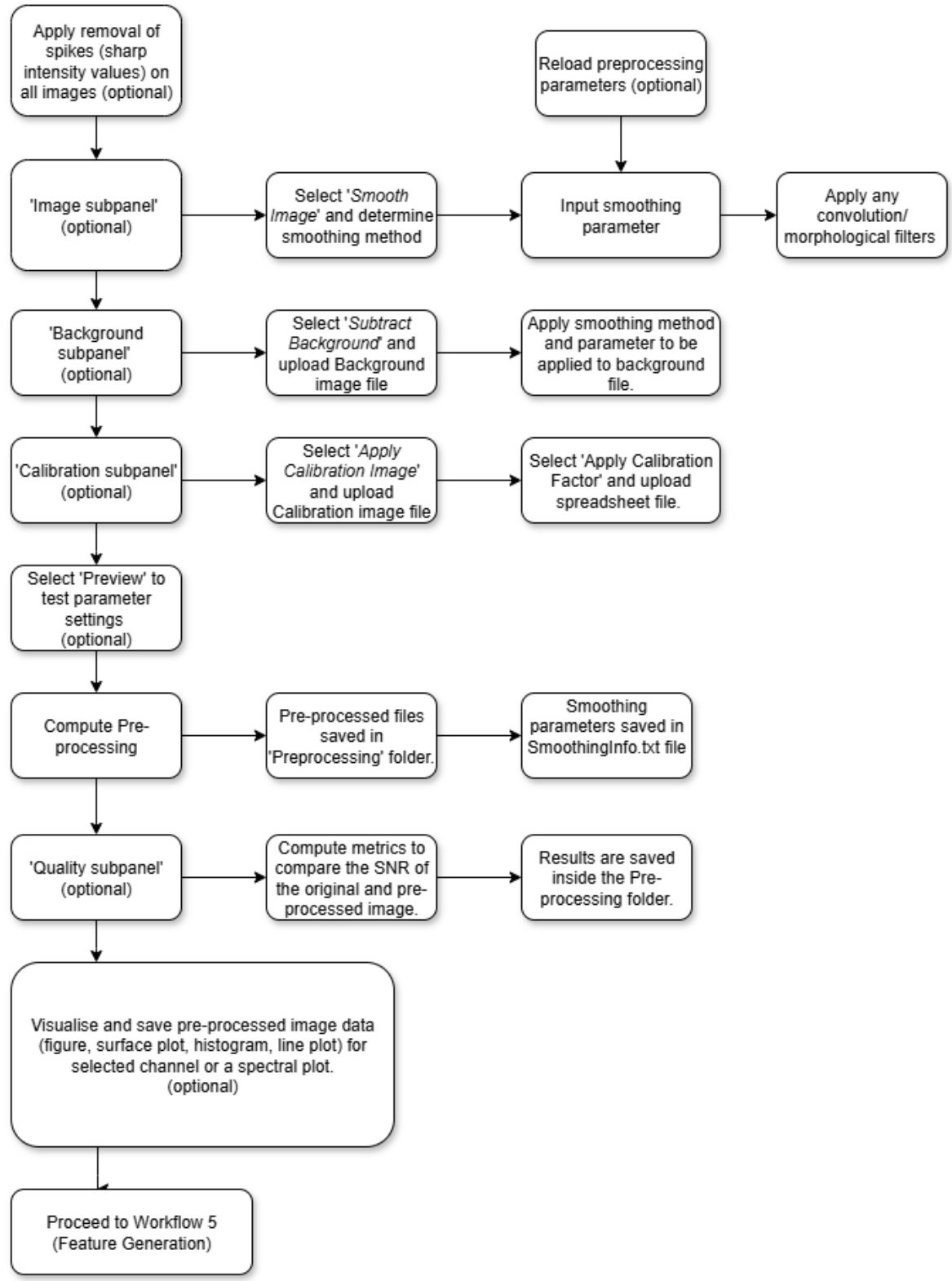

Figure 9-3. Workflow 3: Perform image pre-processing using the 'Pre-processing' panel.

9.4 Workflow 4: Generate features

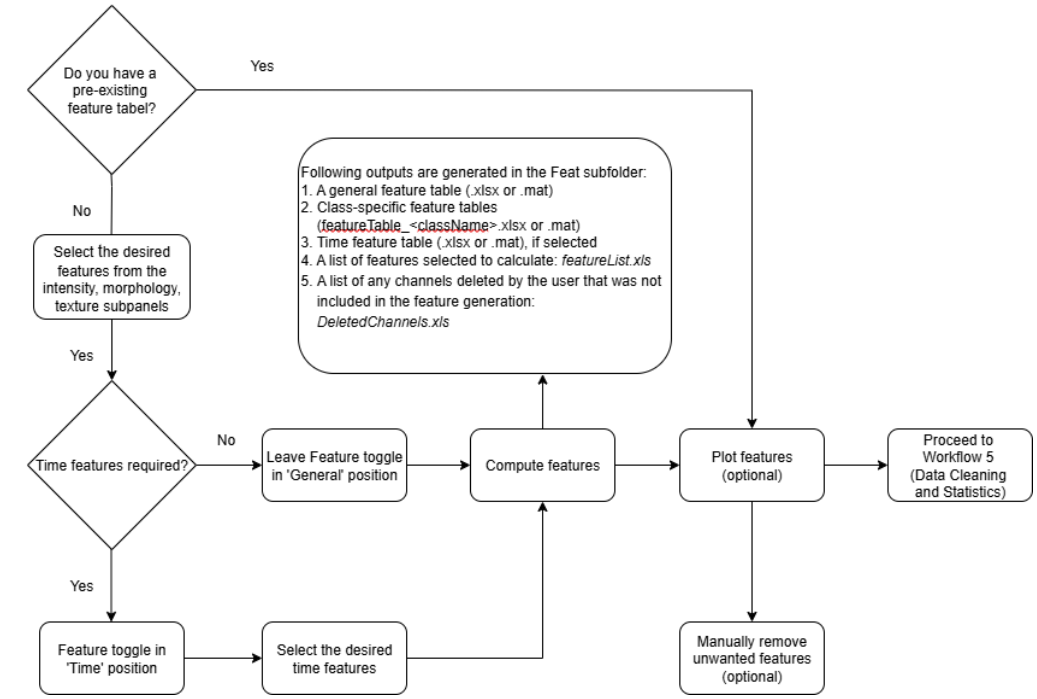

Figure 9-4. Workflow 4: Compute features using the 'Feature Generation' panel.

9.5 Workflow 5: Perform data cleaning

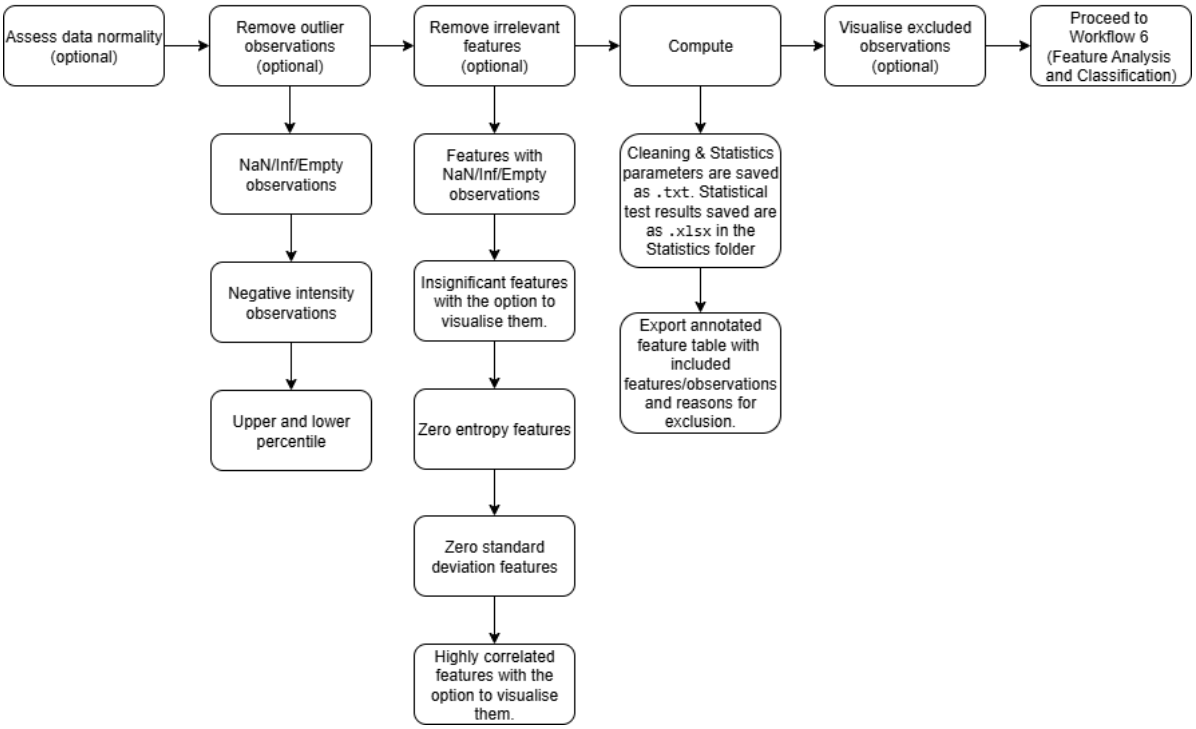

Figure 9-5. Workflow 5: Assess and clean data using the 'Data cleaning and statistics' panel.

9.6 Workflow 6: Creating a new classification model

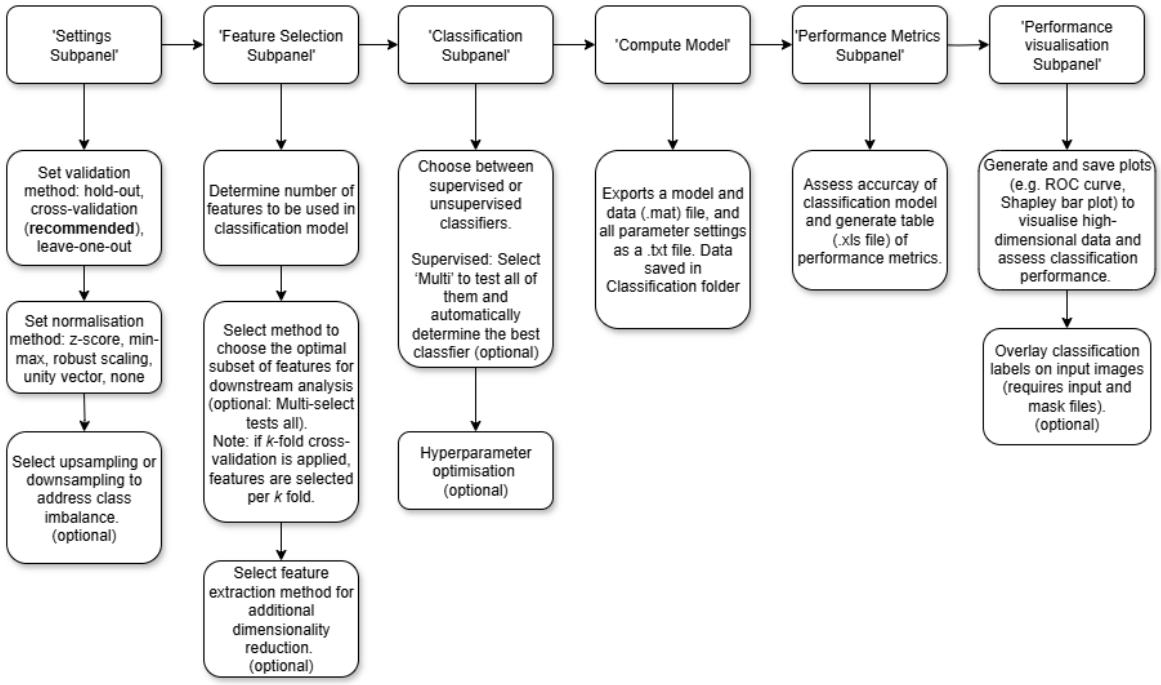

Figure 9-6. Workflow 6: Perform classification using the 'Feature Analysis' panel to generate a new classification model.

9.7 Workflow 7: Using pre-generated classification models

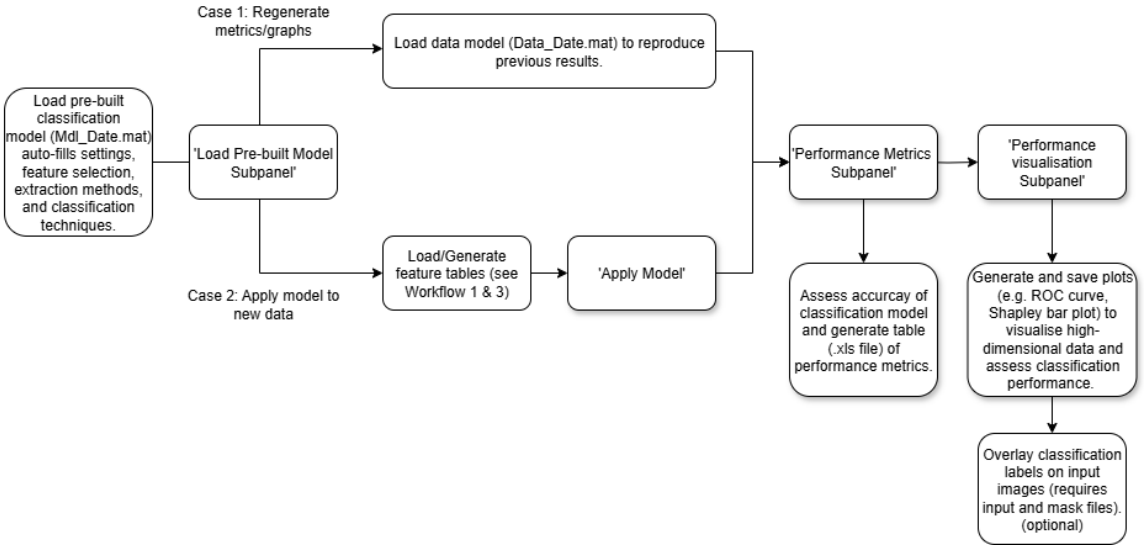

Figure 9-7. Workflow 7: Perform classification using the Feature Analysis panel to work with pre-generated classification models.

10 Appendix

10.1 List of available features

The table below lists and defines the features that can be generated in SMIAL. Each object refers to one observation (e.g. cell). See [4] for equations used to generate texture features.

Table 2. List and definition of features available in SMIAL. Adjusted from [4]

Morphological features	**Definition (unit = pixels)**
Area	Total area per object.
Perimeter	Total perimeter per object.
Major axis length	Length of the major axis of the ellipse that has the same normalised second central moments as the object.
Minor axis length	Length of the minor axis of the ellipse that has the same normalised second central moments as the object.
Eccentricity	Ratio of the distance between the foci of the ellipse (that has the same second-moments as the object) and its major axis length. 0=circle, 1=line.
Orientation	Angle between the x-axis and the major axis of the ellipse (that has the same second-moments as the object).
Convex area	Area of the convex image, that is the image that shows the convex hull (smallest convex polygon that can be made in the object per object).
Circularity/form factor	Roundness of an object (also known as form factor). $Form\ factor = 4\pi * area/perimeter^2$
Filled area	Area of the filled image (bounding box per object).
Euler Number	Number of objects per image minus number of holes in objects.
Equivalent diameter	Diameter of a circle with the same area as the object. $Equiv\ dia = 4 * area/\pi$
Solidity	Proportion of pixels in the convex hull that are also in the object. Measured as $\frac{area}{convex\ area}$
Extent	Ratio of pixels in the object to pixels in the total bounding box. Measured as area/area of bounding box.
Max feret diameter	Maximum distance between two parallel lines tangent on either side of the object.
Max feret angle	Angle of the maximum feret diameter and horizontal axis.
Min feret diameter	Minimum distance between two parallel lines tangent on either side of the object.
Min feret angle	Angle of the minimum feret diameter and horizontal axis.
Interconnectivity	Calculated as mean area/mean perimeter.
Fragmentation	Degree of fragmentation measured as $\frac{area*perimeter}{\#\ objects\ per\ image}$
Intensity features	**Definition (units = A.U. Intensity values may be scaled when using the pre-processing panel in SMIAL)**
Mean intensity	Mean pixel intensity within each object.
Median intensity	Median pixel intensity within each object.
Std intensity	Standard deviation of pixel intensities within each object.
Minimum intensity	Minimum pixel intensity within each object
Maximum intensity	Maximum pixel intensity within each object

Mode	Most frequent pixel intensity within each object
Skewness	Sample skewness for elements, that is the measure of asymmetry around the mean value.
Kurtosis	Sample kurtosis for elements, that is a measures the "tailedness" of a distribution for each object.
Variance	Variance of pixel intensities within each object
Channel ratios	For each object, ratio of mean intensity between all pairs of imaging channels (e.g. red/green)
Channel products	For each object, product of mean intensity between all pairs of imaging channels (e.g. red × green)
Top 10% of pixel values	Mean intensity of the top 10% brightest pixels per object.
75^{th} percentile pixel values	Average of pixel intensities within the 75^{th} percentile per object.
25^{h} percentile pixel values	Average of pixel intensities within the 25^{th} percentile per object.
Top 50% of pixel values	Mean intensity of the top 50% brightest pixels per object.
Top x% of pixel values. Specify x.	Mean of the top X% highest pixel intensities per object, with X set by the user . Choose non-zero value.
Top x% channel ratio. Specify x1/x2	Ratio of mean pixel intensities between two imaging channels, computed per object using , the top X% brightest pixels in each channel. User defines percentage for each channel. Choose non-zero values.
Total intensity	Sum of all pixel intensities within each object
Texture features	**Definitions (Dimensionless)**
Angular second moment (energy)	Measure of object homogeneity. 1=uniform object, <1 higher intensity variation.
Contrast	Measure of local variation in an image. 0=uniform object, higher values indicating a higher degree of local variation.
Correlation	Measure of linear dependency of intensity values in an object. Larger areas of similar intensity have a higher correlation. Values of 1 or -1 for perfectly positively or negatively correlated objects respectively.
Variance	Measure of the variation of object intensity values. For an image with uniform intensity the variance = 0.
Inverse difference moment (homogeneity)	Represents object contrast. Low values for inhomogeneous object and a relatively higher value for homogeneous object.
Sum average	Average of the normalised gray-scale object in the spatial domain.
Sum variance	Variance of the normalised gray-scale object in the spatial domain.
Sum entropy	Measure of randomness within an object.
Entropy	Indication of the complexity within an object with a more complex image having a higher entropy value.
Difference variance	Object variation from normalised co-occurrence matrix.
Difference entropy	Indication of the amount of randomness in an object.
Information measure of correlation I	Measure of the total amount of information contained within an object of pixels derived from the recurring spatial relationship between specific intensity values.
Information measure of correlation II	Measures association between intensity values similar to correlation I but on a different scale.

Maximal correlation coefficient	Measure of dependence between x and y. Equals 0 when x and y are independent.
Time-based features	**Definition** Time-based features can be computed in two ways: 1. Relative to the baseline (e.g. t = 0) 2. Relative to the previous time point (e.g. between $t = TimeX$ and $t = TimeX - 1$) Units are the same as the original feature. The follow metrics are available where the $TimeReference$ is either the baseline or a previous time point):
Difference	$$Difference = Feature_{TimeX} - Feature_{TimeReference}$$
Fold change	$$Fold\ change = \frac{Feature_{TimeX}}{Feature_{TimeReference}}$$
Percentage change	$$Percentage\ change = \frac{Feature_{TimeX} - Feature_{TimeReference}}{Feature_{TimeReference}} \times 100\%$$
Percentage difference	$$Percentage\ difference = \frac{\left\lvert Feature_{TimeX} - Feature_{TimeReference} \right\rvert}{\left(\frac{Feature_{TimeX} + Feature_{TimeReference}}{2}\right)} \times 100\%$$
Rate of change	$$Rate\ of\ change = \frac{Feature_{TimeX} - Feature_{TimeReference}}{TimeX - TimeReference}$$

11 References

[1] B. Rasti, J. R. Sveinsson, M. O. Ulfarsson, and J. A. Benediktsson, "Hyperspectral image denoising using first order spectral roughness penalty in wavelet domain," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol. 7, no. 6, pp. 2458-2467, 2013.

[2] B. Rasti, M. Ulfarsson, and P. Ghamisi, "Automatic Hyperspectral Image Restoration Using Sparse and Low-Rank Modeling," *IEEE Geoscience and Remote Sensing Letters,* vol. 14, pp. 2335-2339, 11/06 2017, doi: 10.1109/LGRS.2017.2764059.

[3] S. B. Mahbub, M. Plöschner, M. E. Gosnell, A. G. Anwer, and E. M. Goldys, "Statistically strong label-free quantitative identification of native fluorophores in a biological sample," *Scientific Reports,* vol. 7, no. 1, p. 15792, 2017/11/17 2017, doi: 10.1038/s41598-017-15952-y.

[4] S. Handley, A. G. Anwer, A. Knab, A. Bhargava, and E. M. Goldys, "AutoMitoNetwork: Software for analyzing mitochondrial networks in autofluorescence images to enable

label-free cell classification. LID - 10.1002/cyto.a.24889 [doi]," (in eng), *Cytometry. Part A : the journal of the International Society for Analytical Cytology,* no. 1552-4930 (Electronic), 2024, doi: 10.1002/cyto.a.24889.