# Package ‘synapsis’

December 5, 2025

**Type** Package

**Title** An R package to automate the analysis of double-strand break repair during meiosis

**Version** 1.17.0

**Description** Synapsis is a Bioconductor software package for automated (unbiased and reproducible) analysis of meiotic immunofluorescence datasets. The primary functions of the software can i) identify cells in meiotic prophase that are labelled by a synaptonemal complex axis or central element protein, ii) isolate individual synaptonemal complexes and measure their physical length, iii) quantify foci and co-localise them with synaptonemal complexes, iv) measure interference between synaptonemal complex-associated foci. The software has applications that extend to multiple species and to the analysis of other proteins that label meiotic prophase chromosomes. The software converts meiotic immunofluorescence images into R data frames that are compatible with machine learning methods. Given a set of microscopy images of meiotic spread slides, synapsis crops images around individual single cells, counts colocalising foci on strands on a per cell basis, and measures the distance between foci on any given strand.

**biocViews** Software, SingleCell

**Depends** R (>= 4.1)

**Imports** EBImage, stats, utils, graphics

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.1.1

**VignetteBuilder** knitr

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0), ggplot2, tidyverse, BiocStyle

**Config/testthat/edition** 3

**git_url** https://git.bioconductor.org/packages/synapsis

**git_branch** devel

**git_last_commit** cc1375e

**git_last_commit_date** 2025-10-29

**Repository** Bioconductor 3.23

**Date/Publication** 2025-12-04

**Author** Lucy McNeill [aut, cre, cph] (ORCID:
<https://orcid.org/0000-0003-1752-4882>),
Wayne Crismani [rev, ctb] (ORCID:
<https://orcid.org/0000-0003-0143-8293>)

**Maintainer** Lucy McNeill <luc.mcneill@gmail.com>

# Contents

annotate_foci_counting . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2
annotate_foci_counting_adjusted . . . . . . . . . . . . . . . . . . . . . . . . 3
append_data_frame . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
auto_crop_fast . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6
count_foci . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
crop_single_object_fast . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10
get_blobs . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12
get_C1 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
get_coincident_foci . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
get_foci_per_cell . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
get_overlap_mask . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
get_pachytene . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18
keep_cells . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20
make_foci_mask . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21
make_strand_mask . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21
remove_XY . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22

**Index** **24**

---

`annotate_foci_counting`

## *annotate_foci_counting*

### Description

Contains all plotting routines for count foci annotation

### Usage

```
annotate_foci_counting(
  img_file,
  cell_count,
  img_orig,
  img_orig_foci,
  artificial_amp_factor,
  strands,
  coincident_foci,
```

```
  foci_label,
  alone_foci,
  percent_px,
  foci_per_cell
)
```

### Arguments

| | |
|---|---|
| `img_file` | cell's file name |
| `cell_count` | unique cell counter |
| `img_orig` | original strand crop |
| `img_orig_foci` | cropped foci channel |
| `artificial_amp_factor` | amplification factor |
| `strands` | black white mask of strand channel |
| `coincident_foci` | mask of overlap between strand and foci channel |
| `foci_label` | black and white mask of foci channel |
| `alone_foci` | estimated number of foci that are NOT on a strand. |
| `percent_px` | percentage of foci mask that coincides with strand channel small number indicates potentially problematic image. |
| `foci_per_cell` | number of foci counted per cell |

### Value

displays key steps from raw image to coincident foci count

---

## `annotate_foci_counting_adjusted`
*annotate_foci_counting_adjusted*

---

### Description

Contains all plotting routines for count foci annotation

### Usage

```
annotate_foci_counting_adjusted(
  img_file,
  cell_count,
  img_orig,
  img_orig_foci,
  artificial_amp_factor,
  strands,
  coincident_foci,
```

```
  foci_label,
  alone_foci,
  percent_px,
  foci_per_cell
)
```

## Arguments

| | |
|---|---|
| `img_file` | cell's file name |
| `cell_count` | unique cell counter |
| `img_orig` | original strand crop |
| `img_orig_foci` | cropped foci channel |
| `artificial_amp_factor` | amplification factor |
| `strands` | black white mask of strand channel |
| `coincident_foci` | mask of overlap between strand and foci channel |
| `foci_label` | black and white mask of foci channel |
| `alone_foci` | estimated number of foci that are NOT on a strand. |
| `percent_px` | percentage of foci mask that coincides with strand channel small number indicates potentially problematic image. |
| `foci_per_cell` | number of foci counted per cell |

## Value

displays key steps from raw image to coincident foci count

---

# `append_data_frame` *append_data_frame*

## Description

applies new row to data frame

## Usage

```
append_data_frame(
  WT_str,
  KO_str,
  WT_out,
  KO_out,
  img_file,
  foci_areas,
  df_cells,
  cell_count,
```

```
  stage,
  foci_per_cell,
  image_mat,
  percent_px,
  alone_foci,
  discrepant_category,
  C1
)
```

## Arguments

| | |
|---|---|
| `WT_str` | string in filename corresponding to wildtype genotype. Defaults to ++. |
| `KO_str` | string in filename corresponding to knockout genotype. Defaults to –. |
| `WT_out` | string in output csv in genotype column, for knockout. Defaults to +/+. |
| `KO_out` | string in output csv in genotype column, for knockout. Defaults to -/-. |
| `img_file` | cell's file name |
| `foci_areas` | pixel area of each foci |
| `df_cells` | current data frame |
| `cell_count` | unique cell counter |
| `stage` | meiosis stage of interest. Currently count_foci determines this with thresholding/ object properties in the synaptonemal complex channel by previosly calling the get_pachytene function. Note that if using this option, the count_foci function requires that the input directory contains a folder called "pachytene" with the crops in it. |
| `foci_per_cell` | foci count for cell |
| `image_mat` | matrix with all pixel values above zero |
| `percent_px` | percentage of foci mask that coincides with strand channel small number indicates potentially problematic image. |
| `alone_foci` | estimated number of foci that are NOT on a strand. |
| `discrepant_category` | estimated number of foci that are NOT on a strand. |
| `C1` | criteria |

## Value

data frame with new row

---

`auto_crop_fast` *auto_crop_fast*

---

### Description

crop an image around each viable cell candidate.

### Usage

```
auto_crop_fast(
  img_path,
  max_cell_area = 20000,
  min_cell_area = 7000,
  mean_pix = 0.08,
  annotation = "off",
  blob_factor = 15,
  bg_blob_factor = 10,
  offset = 0.2,
  final_blob_amp = 10,
  test_amount = 0,
  brush_size_blob = 51,
  sigma_blob = 15,
  channel3_string = "DAPI",
  channel2_string = "SYCP3",
  channel1_string = "MLH3",
  file_ext = "jpeg",
  third_channel = "off",
  cell_aspect_ratio = 2,
  strand_amp = 2,
  path_out = img_path,
  resize_l = 720,
  crowded_cells = "FALSE",
  watershed_radius = 50,
  watershed_tol = 0.2,
  cropping_factor = 1.3
)
```

### Arguments

| | |
|---|---|
| `img_path` | path containing image data to analyse |
| `max_cell_area` | Maximum pixel area of a cell candidate |
| `min_cell_area` | Minimum pixel area of a cell candidate |
| `mean_pix` | Mean pixel intensity of cell crop (in SYCP3 channel) for normalisation |
| `annotation` | Choice to output pipeline choices (recommended to knit) |
| `blob_factor` | Contrast factor to multiply original image by before smoothing/smudging |

| | |
|---|---|
| `bg_blob_factor` | Contrast factor to multiply original image by to take background. Used prior to thresholding. |
| `offset` | Pixel value offset from bg_blob_factor. Used in thresholding to make blob mask. |
| `final_blob_amp` | Contrast factor to multiply smoothed/smudged image. Used in thresholding to make blob mask. |
| `test_amount` | Optional number of first N images you want to run function on. For troubleshooting/testing/variable calibration purposes. |
| `brush_size_blob` | Brush size for smudging the synaptonemal complex channel to make blobs |
| `sigma_blob` | Sigma in Gaussian brush for smudging the synaptonemal complex channel to make blobs |
| `channel3_string` | Optional. String appended to the files showing the channel illuminating cell structures. Defaults to DAPI, if third channel == "on". |
| `channel2_string` | String appended to the files showing the channel illuminating synaptonemal complexes. Defaults to SYCP3 |
| `channel1_string` | String appended to the files showing the channel illuminating foci. Defaults to MLH3 |
| `file_ext` | file extension of your images e.g. tif jpeg or png. |
| `third_channel` | Optional, defaults to "off". Set to "on" if you would also like crops of the third channel. |
| `cell_aspect_ratio` | Maximum aspect ratio of blob to be defined as a cell |
| `strand_amp` | multiplication of strand channel for get_blobs function. |
| `path_out` | user specified output path. Defaults to img_path |
| `resize_l` | length for resized image |
| `crowded_cells` | TRUE or FALSE, defaults to FALSE. Set to TRUE if you have many cells in a frame that almost touch |
| `watershed_radius` | Radius (ext variable) in watershed method used in strand channel. Defaults to 1 (small) |
| `watershed_tol` | Intensity tolerance for watershed method. Defaults to 0.05. |
| `cropping_factor` | size of cropping window square, as factor of characteristic blob radius. Defaults to 1. May need to increase if using watershed. |

## Details

This function takes all images in a directory, and crops around individual cells according to the antibody that stains synaptonemal complexes e.g. SYCP3. First, it increases the brightness and smudges the image with a Gaussian brush, and creates a mask using thresholding (get_blobs). Then it deletes cell candidates in the mask deemed too large, too small, or too long (keep_cells). Using the computeFeatures functions from EBImage to locate centre and radius, the cropping area is determined and the original image cropped. These images are saved in either a user specified directory, or a crops folder at the location of the image files.

### Value

cropped synaptonemal complex and foci channels around single cells, regardless of stage

### Author(s)

Lucy McNeill

### Examples

```
demo_path = paste0(system.file("extdata",package = "synapsis"))
auto_crop_fast(demo_path, annotation = "on", max_cell_area = 30000,
min_cell_area = 7000, file_ext = "tif",crowded_cells = TRUE)
```

---

## `count_foci` *count_foci*

### Description

Calculates coincident foci in synaptonemal complex and foci channel, per cell

### Usage

```
count_foci(
  img_path,
  stage = "none",
  offset_px = 0.2,
  offset_factor = 2,
  brush_size = 3,
  brush_sigma = 3,
  foci_norm = 0.01,
  annotation = "off",
  channel2_string = "SYCP3",
  channel1_string = "MLH3",
  file_ext = "jpeg",
  KO_str = "--",
  WT_str = "++",
  KO_out = "-/-",
  WT_out = "+/+",
  watershed_stop = "off",
  watershed_radius = 1,
  watershed_tol = 0.05,
  crowded_foci = TRUE,
  artificial_amp_factor = 1,
  strand_amp = 2,
  min_foci = -1,
  disc_size = 51,
  modify_problematic = "off",
```

```
  disc_size_foci = 5,
  C1 = 0.02,
  C2 = 0.46,
  C_weigh_foci_number = TRUE
)
```

## Arguments

| | |
|---|---|
| `img_path` | path containing crops to analyse |
| `stage` | meiosis stage of interest. Currently count_foci determines this with thresholding/ object properties in the synaptonemal complex channel by previosly calling the get_pachytene function. Note that if using this option, the count_foci function requires that the input directory contains a folder called "pachytene" with the crops in it. |
| `offset_px` | Pixel value offset used in thresholding of synaptonemal complex channel |
| `offset_factor` | Pixel value offset used in thresholding of foci channel |
| `brush_size` | size of brush to smooth the foci channel. Should be small to avoid erasing foci. |
| `brush_sigma` | sigma for Gaussian smooth of foci channel. Should be small to avoid erasing foci. |
| `foci_norm` | Mean intensity to normalise all foci channels to. |
| `annotation` | Choice to output pipeline choices (recommended to knit) |
| `channel2_string` | String appended to the files showing the channel illuminating synaptonemal complexes. Defaults to SYCP3 |
| `channel1_string` | String appended to the files showing the channel illuminating foci. Defaults to MLH3 |
| `file_ext` | file extension of your images e.g. tiff jpeg or png. |
| `KO_str` | string in filename corresponding to knockout genotype. Defaults to –. |
| `WT_str` | string in filename corresponding to wildtype genotype. Defaults to ++. |
| `KO_out` | string in output csv in genotype column, for knockout. Defaults to -/-. |
| `WT_out` | string in output csv in genotype column, for knockout. Defaults to +/+. |
| `watershed_stop` | Stop default watershed method with "on" |
| `watershed_radius` | Radius (ext variable) in watershed method used in foci channel. Defaults to 1 (small) |
| `watershed_tol` | Intensity tolerance for watershed method. Defaults to 0.05. |
| `crowded_foci` | TRUE or FALSE, defaults to FALSE. Set to TRUE if you have foci > 100 or so. |
| `artificial_amp_factor` | Amplification of foci channel, for annotation only. |
| `strand_amp` | multiplication of strand channel to make masks |
| `min_foci` | minimum pixel area for a foci. Depends on your dpi etc. Defaults to 4 |

| | |
|---|---|
| `disc_size` | size of disc for local background calculation in synaptonemal complex channel |
| `modify_problematic` | option for synapsis to try and "save" images which have likely been counted incorrectly due to a number of reasons. Default settings are optimized for mouse pachytene. Defaults to "off" |
| `disc_size_foci` | size of disc for local background calculation in foci channel |
| `C1` | Default crispness criteria = sd(foci_area)/(mean(foci_area)+1) |
| `C2` | Alternative crisp criteria. |
| `C_weigh_foci_number` | choose crispness criteria- defaults to TRUE to use C1 (weighing with number). Otherwise set to FALSE to use C2 |

### Details

In this function, masks for the synaptonemal complex (SC) and foci channel are created from the saved crops of single/individual cells. These masks are computed using (optional) input parameters related to meiosis stage/ how well spread chromosomes are (for the former) and related to smoothing, thresholding and how "crowded" foci are for the latter. Finally, these two masks are multiplied, and the number of objects found with EBImage's computeFeatures are the colocalizing foci.

The file, cell number, foci count etc. are output as a data frame.

### Value

data frame with foci count per cell

### Author(s)

Lucy McNeill

### Examples

```
demo_path = paste0(system.file("extdata",package = "synapsis"))
foci_counts <- count_foci(demo_path,offset_factor = 3, brush_size = 3,
brush_sigma = 3, annotation = "on",stage = "pachytene")
```

---

## `crop_single_object_fast`
## *crop_single_object_fast*

---

### Description

Creates mask for every individual cell candidate in mask

### Usage

```
crop_single_object_fast(
  retained,
  OOI_final,
  counter_final,
  img_orig,
  img_orig_foci,
  img_orig_DAPI = "blank",
  file_sc,
  file_foci,
  file_DAPI = "blank",
  cell_count,
  mean_pix,
  annotation,
  file_base,
  img_path,
  r_max,
  cx,
  cy,
  channel3_string,
  channel2_string,
  channel1_string,
  file_ext,
  third_channel,
  path_out,
  img_orig_highres,
  resize_l,
  crowded_cells,
  cropping_factor
)
```

### Arguments

| | |
|---|---|
| `retained` | Mask of cell candidates which meet size criteria. After smoothing/smudging and thresholding. |
| `OOI_final` | Objects of interest count. Total number of cell candidates in retained. |
| `counter_final` | Counter for single cell we are focussing on. Remove all other cells where counter_single not equal to counter_final. |
| `img_orig` | description |
| `img_orig_foci` | description |
| `img_orig_DAPI` | description |
| `file_sc` | filename of synaptonemal complex channel image |
| `file_foci` | filename of foci channel image |
| `file_DAPI` | filename of DAPI channel image |
| `cell_count` | counter for successful crops around cells |

| | |
|---|---|
| `mean_pix` | Mean pixel intensity of cell crop (in SYCP3 channel) for normalisation |
| `annotation` | Choice to output pipeline choices (recommended to knit) |
| `file_base` | filename base common to all three channels i.e. without -MLH3.jpeg etc. |
| `img_path` | path containing image data to analyse |
| `r_max` | maximum radius of blob for cropping |
| `cx` | centre of blob x |
| `cy` | centre of blob y |
| `channel3_string` | Optional. String appended to the files showing the channel illuminating cell structures. Defaults to DAPI, if third channel == "on". |
| `channel2_string` | String appended to the files showing the channel illuminating synaptonemal complexes. Defaults to SYCP3 |
| `channel1_string` | String appended to the files showing the channel illuminating foci. Defaults to MLH3 |
| `file_ext` | file extension of your images e.g. tif jpeg or png. |
| `third_channel` | Optional, defaults to "off". Set to "on" if you would also like crops of the third channel. |
| `path_out` | user specified output path. Defaults to img_path |
| `img_orig_highres` | the original strand image with original resolution |
| `resize_l` | length of square to resize original image to. |
| `crowded_cells` | TRUE or FALSE, defaults to FALSE. Set to TRUE if you have many cells in a frame that almost touch |
| `cropping_factor` | size of cropping window square, as factor of characteristic blob radius. Defaults to 1. May need to increase if using watershed. |

### Value

Crops around all candidates in both channels

---

## `get_blobs` *get_blobs*

---

### Description

Makes mask of all objects bright enough to be classified as a cell cadidate

### Usage

```
get_blobs(
  img_orig,
  blob_factor,
  bg_blob_factor,
  offset,
  final_blob_amp,
  brush_size_blob,
  sigma_blob,
  watershed_tol,
  watershed_radius,
  crowded_cells,
  annotation
)
```

### Arguments

| | |
|---|---|
| `img_orig` | Original image |
| `blob_factor` | Contrast factor to multiply original image by before smoothing/smudging |
| `bg_blob_factor` | Contrast factor to multiply original image by to take background. Used prior to thresholding. |
| `offset` | Pixel value offset from bg_blob_factor. Used in thresholding to make blob mask. |
| `final_blob_amp` | Contrast factor to multiply smoothed/smudged image. Used in thresholding to make blob mask. |
| `brush_size_blob` | Brush size for smudging the synaptonemal complex channel to make blobs |
| `sigma_blob` | Sigma in Gaussian brush for smudging the synaptonemal complex channel to make blobs |
| `watershed_tol` | Intensity tolerance for watershed method. Defaults to 0.05. |
| `watershed_radius` | Radius (ext variable) in watershed method used in strand channel. Defaults to 1 (small) |
| `crowded_cells` | TRUE or FALSE, defaults to FALSE. Set to TRUE if you have many cells in a frame that almost touch |
| `annotation` | Choice to output pipeline choices (recommended to knit) have many cells in a frame that almost touch |

### Value

Mask with cell candidates

---

## `get_C1` *get_C1*

---

### Description

calculates the statistic to compare to crisp_criteria, which determines whether the foci count will be reliable

### Usage

```
get_C1(foci_areas, foci_per_cell, C_weigh_foci_number)
```

### Arguments

| | |
|---|---|
| `foci_areas` | pixel area of each foci |
| `foci_per_cell` | foci count for cell |
| `C_weigh_foci_number` | choose crispness criteria- defaults to TRUE to use C1 (weighing with number). Otherwise set to FALSE to use C2 |

### Value

statistic to comapre to crisp_criteria

---

## `get_coincident_foci` *get_coincident_foci*

---

### Description

calculates the statistic to compare to crisp_criteria, which determines whether the foci count will be reliable

### Usage

```
get_coincident_foci(
  offset_px,
  offset_factor,
  brush_size,
  brush_sigma,
  annotation,
  watershed_stop,
  watershed_radius,
  watershed_tol,
  crowded_foci,
  artificial_amp_factor,
```

```
  strand_amp,
  disc_size,
  disc_size_foci,
  img_file,
  cell_count,
  img_orig,
  img_orig_foci,
  stage,
  WT_str,
  KO_str,
  WT_out,
  KO_out,
  C1_search,
  discrepant_category,
  C1,
  C2,
  df_cells,
  C_weigh_foci_number
)
```

## Arguments

| | |
|---|---|
| `offset_px` | Pixel value offset used in thresholding of synaptonemal complex channel |
| `offset_factor` | Pixel value offset used in thresholding of foci channel |
| `brush_size` | size of brush to smooth the foci channel. Should be small to avoid erasing foci. |
| `brush_sigma` | sigma for Gaussian smooth of foci channel. Should be small to avoid erasing foci. |
| `annotation` | Choice to output pipeline choices (recommended to knit) |
| `watershed_stop` | Stop default watershed method with "on" |
| `watershed_radius` | Radius (ext variable) in watershed method used in foci channel. Defaults to 1 (small) |
| `watershed_tol` | Intensity tolerance for watershed method. Defaults to 0.05. |
| `crowded_foci` | TRUE or FALSE, defaults to FALSE. Set to TRUE if you have foci > 100 or so. |
| `artificial_amp_factor` | Amplification of foci channel, for annotation only. |
| `strand_amp` | multiplication of strand channel to make masks |
| `disc_size` | size of disc for local background calculation in synaptonemal complex channel |
| `disc_size_foci` | size of disc for local background calculation in foci channel |
| `img_file` | cell's file name |
| `cell_count` | unique cell counter |
| `img_orig` | original strand crop |
| `img_orig_foci` | cropped foci channel |

| | |
|---|---|
| `stage` | meiosis stage of interest. Currently count_foci determines this with thresholding/ object properties in the synaptonemal complex channel by previosly calling the get_pachytene function. Note that if using this option, the count_foci function requires that the input directory contains a folder called “pachytene” with the crops in it. |
| `WT_str` | string in filename corresponding to wildtype genotype. Defaults to ++. |
| `KO_str` | string in filename corresponding to knockout genotype. Defaults to –. |
| `WT_out` | string in output csv in genotype column, for knockout. Defaults to +/+. |
| `KO_out` | string in output csv in genotype column, for knockout. Defaults to -/-. |
| `C1_search` | TRUE or FALSE whether the image is still being modified until it meets the crispness criteria |
| `discrepant_category` | estimated number of foci that are NOT on a strand. |
| `C1` | Default crispness criteria = sd(foci_area)/(mean(foci_area)+1) |
| `C2` | Alternative crisp criteria. |
| `df_cells` | current data frame |
| `C_weigh_foci_number` | choose crispness criteria- defaults to TRUE to use C1 (weighing with number). Otherwise set to FALSE to use C2 |

## Value

data frame with new row with most recent foci per cell appended

---

`get_foci_per_cell` *get_foci_per_cell*

## Description

creates mask for coincident foci

## Usage

```
get_foci_per_cell(
  img_file,
  offset_px,
  stage,
  strands,
  watershed_stop,
  foci_label,
  annotation,
  cell_count,
  img_orig,
  img_orig_foci,
  artificial_amp_factor,
  coincident_foci
)
```

### Arguments

| | |
|---|---|
| `img_file` | cell's file name |
| `offset_px` | Pixel value offset used in thresholding of synaptonemal complex channel |
| `stage` | meiosis stage of interest. Currently count_foci determines this with thresholding/ object properties in the synaptonemal complex channel by previosly calling the get_pachytene function. Note that if using this option, the count_foci function requires that the input directory contains a folder called "pachytene" with the crops in it. |
| `strands` | black white mask of strand channel |
| `watershed_stop` | Stop default watershed method with "on" |
| `foci_label` | black and white mask of foci channel |
| `annotation` | Choice to output pipeline choices (recommended to knit) |
| `cell_count` | unique cell counter |
| `img_orig` | original strand crop |
| `img_orig_foci` | cropped foci channel |
| `artificial_amp_factor` | amplification factor |
| `coincident_foci` | mask of coincident foci |

### Value

number of foci per cell

---

## `get_overlap_mask` *get_overlap_mask*

### Description

creates mask for coincident foci

### Usage

```
get_overlap_mask(
  strands,
  foci_label,
  watershed_stop,
  img_orig_foci,
  watershed_radius,
  watershed_tol
)
```

### Arguments

| | |
|---|---|
| `strands` | black white mask of strand channel |
| `foci_label` | black and white mask of foci channel |
| `watershed_stop` | Stop default watershed method with "on" |
| `img_orig_foci` | cropped foci channel |
| `watershed_radius` | Radius (ext variable) in watershed method used in foci channel. Defaults to 1 (small) |
| `watershed_tol` | Intensity tolerance for watershed method. Defaults to 0.05. |

### Value

mask with coincident foci on strands

---

## `get_pachytene` *get_pachytene*

### Description

Identifies crops in pachytene

### Usage

```
get_pachytene(
  img_path,
  species_num = 20,
  offset = 0.2,
  ecc_thresh = 0.85,
  area_thresh = 0.06,
  annotation = "off",
  channel2_string = "SYCP3",
  channel1_string = "MLH3",
  file_ext = "jpeg",
  KO_str = "--",
  WT_str = "++",
  KO_out = "-/-",
  WT_out = "+/+",
  path_out = img_path,
  artificial_amp_factor = 3,
  strand_amp = 2,
  resize_l = 120
)
```

## Arguments

| | |
|---|---|
| `img_path` | path containing crops analyse |
| `species_num` | number of chromosomes in the species |
| `offset` | Pixel value offset used in therholding for the synaptonemal complex (SYCP3) channel |
| `ecc_thresh` | The minimum average eccentricity of all objects in mask determined by computefeatures, for a cell to be pachytene. |
| `area_thresh` | The minimum ratio of pixels included in mask to total, for a cell to be classified as pachytene. |
| `annotation` | Choice to output pipeline choices (recommended to knit) |
| `channel2_string` | String appended to the files showing the channel illuminating synaptonemal complexes. Defaults to SYCP3 |
| `channel1_string` | String appended to the files showing the channel illuminating foci. Defaults to MLH3 |
| `file_ext` | file extension of your images e.g. tiff jpeg or png. |
| `KO_str` | string in filename corresponding to knockout genotype. Defaults to –. |
| `WT_str` | string in filename corresponding to wildtype genotype. Defaults to ++. |
| `KO_out` | string in output csv in genotype column, for knockout. Defaults to -/-. |
| `WT_out` | string in output csv in genotype column, for knockout. Defaults to +/+. |
| `path_out` | user specified output path. Defaults to img_path |
| `artificial_amp_factor` | Amplification of foci channel, for RGB output files. Deaults to 3. |
| `strand_amp` | multiplication of strand channel. |
| `resize_l` | length of resized square cell image. |

## Details

This function takes the crops make by auto_crop fast, and determines the number of synaptonemal complex candidates by considering the local background and using EBImage functions. In general, very bright objects which contrast highly with the background will be classified as the same object. Dim objects will likely be classified as many different objects. If the number of objects is too high compared to the species number (species_num) then the cell is determined to not be in pachytene. Note that this function has been optimized for mouse cells which can be very well spread / separated.

## Value

Pairs of foci and synaptonemal channel crops for pachytene

## Author(s)

Lucy McNeill

## Examples

```
demo_path = paste0(system.file("extdata",package = "synapsis"))
SYCP3_stats <- get_pachytene(demo_path,ecc_thresh = 0.8, area_thresh = 0.04, annotation = "on")
```

---

# `keep_cells` *keep_cells*

## Description

Deletes objects in mask which are too small, large, oblong i.e. unlikely to be a cell

## Usage

```
keep_cells(
  candidate,
  max_cell_area,
  min_cell_area,
  cell_aspect_ratio,
  crowded_cells,
  annotation
)
```

## Arguments

| | |
|---|---|
| `candidate` | Mask of individual cell candidates |
| `max_cell_area` | Maximum pixel area of a cell candidate |
| `min_cell_area` | Minimum pixel area of a cell candidate |
| `cell_aspect_ratio` | Maximum aspect ratio of blob to be defined as a cell |
| `crowded_cells` | TRUE or FALSE, defaults to FALSE. Set to TRUE if you |
| `annotation` | Choice to output pipeline choices (recommended to knit) have many cells in a frame that almost touch |

## Value

Mask of cell candidates which meet size criteria

---

## `make_foci_mask` *make_foci_mask*

---

### Description

creates foci mask for foci channel crop

### Usage

```
make_foci_mask(
  offset_factor,
  bg,
  crowded_foci,
  img_orig_foci,
  brush_size,
  brush_sigma,
  disc_size_foci
)
```

### Arguments

| | |
|---|---|
| `offset_factor` | Pixel value offset used in thresholding of foci channel |
| `bg` | background value- currently just mean pixel value of whole image |
| `crowded_foci` | TRUE or FALSE, defaults to FALSE. Set to TRUE if you have foci > 100 or so. |
| `img_orig_foci` | cropped foci channel |
| `brush_size` | size of brush to smooth the foci channel. Should be small to avoid erasing foci. |
| `brush_sigma` | sigma for Gaussian smooth of foci channel. Should be small to avoid erasing foci. |
| `disc_size_foci` | size of disc for local background calculation in foci channel |

### Value

foci mask

---

## `make_strand_mask` *make_strand_mask*

---

### Description

creates strand mask for strand channel crop

### Usage

```
make_strand_mask(
  offset_px,
  stage,
  img_orig,
  disc_size,
  brush_size,
  brush_sigma
)
```

### Arguments

| | |
|---|---|
| `offset_px` | Pixel value offset used in thresholding of synaptonemal complex channel |
| `stage` | meiosis stage of interest. Currently count_foci determines this with thresholding/ object properties in the synaptonemal complex channel by previosly calling the get_pachytene function. Note that if using this option, the count_foci function requires that the input directory contains a folder called “pachytene” with the crops in it. |
| `img_orig` | original strand crop |
| `disc_size` | size of disc for local background calculation in synaptonemal complex channel |
| `brush_size` | size of brush to smooth the foci channel. Should be small to avoid erasing foci. |
| `brush_sigma` | sigma for Gaussian smooth of foci channel. Should be small to avoid erasing foci. |

### Value

strand mask

---

## `remove_XY` *remove_XY*

### Description

applies new row to data frame

### Usage

```
remove_XY(foci_label, foci_candidates, foci_areas)
```

### Arguments

| | |
|---|---|
| `foci_label` | black and white mask of foci channel |
| `foci_candidates` | computeFeatures data frame of foci channel |
| `foci_areas` | the areas of the foci objects |

### Value

mask with XY blob removed

# Index

annotate_foci_counting, 2
annotate_foci_counting_adjusted, 3
append_data_frame, 4
auto_crop_fast, 6

count_foci, 8
crop_single_object_fast, 10

get_blobs, 12
get_C1, 14
get_coincident_foci, 14
get_foci_per_cell, 16
get_overlap_mask, 17
get_pachytene, 18

keep_cells, 20

make_foci_mask, 21
make_strand_mask, 21

remove_XY, 22