

Package: PersomicsArray (via r-universe)

September 16, 2024

Type Package

Title Automated Persomics Array Image Extraction

Version 1.0

Date 2016-09-23

Author John Smestad [aut, cre]

Maintainer John Smestad <smestad.john@gmail.com>

Depends R (>= 3.1.0), grDevices, jpeg, tiff, stringr, raster, utils,
graphics, stats

Description Automated identification of printed array positions from
high content microscopy images and the export of those
positions as individual images written to output as
multi-layered tiff files.

License GPL-3

LazyData TRUE

NeedsCompilation no

Date/Publication 2016-09-26 23:59:25

Repository <https://jsmes123.r-universe.dev>

RemoteUrl <https://github.com/cran/PersomicsArray>

RemoteRef HEAD

RemoteSha 4b28612a79fc2c9a8573141cf2dc8d5dc2dd0d24

Contents

PersomicsArray-package	2
example_annotation	2
example_plate	3
plot_low_res	3
spot_id	4

Index	6
--------------	----------

PersomicsArray-package

PersomicsArray: an R package for automated extraction of annotated images from high-content microscopy images of Persomics plates

Description

The PersomicsArray package contains functions to sequentially read multi-channel jpeg or tiff high content microscopy image files, identify coordinates of printed siRNAs via one of the image channels, and then export individual multi-channel tiff images for each identified array position. The names of the exported images are supplied by an input csv file containing the array annotations. Images are exported as tiff files to automatically-created sub-directories named after the input image files.

Details

Package:	PersomicsArray
Type:	Package
Version:	1.0
Date:	2016-09-23
License:	GPL-3
LazyLoad:	yes

This is version 1.0 of the PersomicsArray package.

Author(s)

John Smestad
<smestad.john@gmail.com>

See Also

[<spot_id>](#) [<plot_low_res>](#) [<example_annotation>](#) [<example_plate>](#)

example_annotation

Example Annotation Data for PersomicsArray Package

Description

This is a very small example of the expected array annotation format taken as input by the `spot_id` function in the PersomicsArray package.

Usage

example_annotation

Format

data.frame

See Also

[<PersomicsArray-package>](#) [<spot_id>](#) [<plot_low_res>](#) [<example_plate>](#)

example_plate	<i>Example Plate Image for PersomicsArray Package</i>
---------------	---

Description

This is a very small example image used for teaching functionality in the PersomicsArray package.

Usage

example_plate

Format

data.frame

See Also

[<PersomicsArray-package>](#) [<spot_id>](#) [<plot_low_res>](#) [<example_annotation>](#)

plot_low_res	<i>plot_low_res</i>
--------------	---------------------

Description

This function generates low resolution plots of pixel values stored in arrays, and is called by function `spot_id`.

Usage

```
plot_low_res(plot.image=NULL, rescale.factor=1000, add=FALSE,
pallette=gray.colors(20),main=NULL)
```

Arguments

plot.image	Array-type object containing image data.
rescale.factor	Numeric value specifying the fold reduction of resolution from original image. Larger values make the function run faster, but at the cost of image resolution.
add	If TRUE, adds the plot to the current active device. If FALSE, a new plot is generated.
pallette	Specifies the color pallette to be used for generating the plot.
main	Character string containing the plot title.

Author(s)

John Smestad
<smestad.john@gmail.com>

See Also

[<PersomicsArray-package>](#) [<spot_id>](#) [<example_annotation>](#) [<example_plate>](#)

Examples

```
# load example image data from package
data("PersomicsPlate")

# isolate single color channel
red <- example_plate[, ,1]

# plot low resolution image of
plot_low_res(red, rescale.factor=10)
```

spot_id

spot_id

Description

This function sequentially reads multi-channel jpeg or tiff high content microscopy image files, identifies coordinates of printed siRNAs via one of the image channels, and then exports individual multi-channel tiff images for each identified array position. The names of the exported images are supplied by an input csv file containing the array annotations. Images are exported to automatically-created sub-directories named after the input image files.

Usage

```
spot_id(files, annotation, channel.num=3, spot.channel=1, smooth.cycle=4,
binary.cut= 0.3, channel.scaling=TRUE, scale.percentiles=c(0.01,0.99))
```

Arguments

files	Vector containing the names of desired files to be read from the current working directory. All array images must have invariable numbers of rows and columns.
annotation	Name of the csv file in the current working directly that contains array annotations. File must have invariable number of rows and columns.
channel.num	Numeric value specifying the number of color channels contained by the input image. jpeg images contain a max of 3. tiff files can contain more than this. The maximum number of channels used by this function is 6.
spot.channel	Numeric value specifying the channel number used for identification of grid array spots.
smooth.cycle	Numeric value specifying the number of iterations of smoothing operation applied to the extracted binary image prior to identification of pixel clusters. Designed for elimination of rogue pixels from binary image.
binary.cut	Numeric value in the range of 0-1 defining the cut-off pixel intensity for defining the binary image.
channel.scaling	If TRUE, all color channels are scaled according to the values supplied to scale.percentiles. This parameter scales the input data to optimize coverage of the dynamic range of displayed pixel intensity values. If FALSE, input data is used "as is" with no scaling operation applied.
scale.percentiles	Vector of two numeric values in the range of 0-1 that specify the percentiles of pixel intensity values used to define min and max for individual channel signal.

Author(s)

John Smestad
<smestad.john@gmail.com>

See Also

[<PersomicsArray-package>](#) [<plot_low_res>](#) [<example_annotation>](#) [<example_plate>](#)

Examples

```
# get names of all plate image files in current working directory
tif.files <- list.files(path = getwd(), pattern = ".tif", all.files = FALSE,
  full.names = FALSE, recursive = FALSE, ignore.case = TRUE,
  include.dirs = FALSE)

# read images, process, and extract individual image files for each array position
spot_id(files=tif.files,ann="annotation.csv",channel.num=3, spot.channel=1,
smooth.cycle=4, binary.cut= 0.3, channel.scaling=TRUE, scale.percentiles=c(0.01,0.99))
```

Index

- * **datasets**

- example_annotation, 2

- example_plate, 3

- * **package**

- PersomicsArray-package, 2

example_annotation, 2, 2, 3–5

example_plate, 2, 3, 3, 4, 5

PersomicsArray-package, 2

plot_low_res, 2, 3, 3, 5

spot_id, 2–4, 4