Package 'detectorchecker'

March 29, 2019

Type Package

Title An package for analysing pixel damage in CT Scanners

Version 0.1.10

Description The description of a package is usually long,
spanning multiple lines. The second and subsequent lines
should be indented, usually with four spaces.

Depends R (>= 3.5.0)

Imports spatstat, tools, tiff, hdf5r, raster, igraph, plyr, dplyr, readr, dbscan

Suggests testthat, knitr, rmarkdown, roxygen2, devtools

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

VignetteBuilder knitr

R topics documented:

`.assign_module` *Pixel module Function assign a module to each dead pixel*

Description

Pixel module Function assign a module to each dead pixel

Usage

```
.assign_module(detector)
```

Arguments

`detector`	Detector object

Value

dead_modules

.assign_pixel_matrix *Assign dead pixels to a detector*

Description

Assign dead pixels to a detector

Usage

```
.assign_pixel_matrix(detector, pix_matrix)
```

Arguments

detector	Detector object
pix_matrix	A pixel matrix

.check_select *Checks if the selected row and column are within the boundaries of the detector*

Description

Checks if the selected row and column are within the boundaries of the detector

Usage

```
.check_select(detector, row, col)
```

Arguments

detector	Detector object
row	module row
col	module col

Value

Boolean

.classify_clump *Clasifies a clump*

Description

Clasifies a clump

Usage

```
.classify_clump(detector, x, y)
```

Arguments

detector	Detector object
x	vector containing the x coordinates of a clump
y	vector containing the y coordinates of a clump

Value

the class of a clump (1 - singleton, 2 - double, 3 - triplet, 4 - larger cluster, unless it actually has the shape of a line, 5 (6): vertical line where closest edge is the upper (lower) one, 7 (8): horizontal line where closest edge is the right (left) one)

.clump_module *Identifying modules for clumps*

Description

Identifying modules for clumps

Usage

```
.clump_module(detector, rrc)
```

Arguments

detector	Detector object
rrc	raster clumps objects

Value

data frame of the modules relating the clump

.create_ppp_edges_col *This is the ppp_edges_col creation function*

Description

This is the ppp_edges_col creation function

Usage

```
.create_ppp_edges_col(detector)
```

Arguments

detector	Detector object

Value

Point pattern dataset

.create_ppp_edges_row *This is the create_ppp_edges_row creation function*

Description

This is the create_ppp_edges_row creation function

Usage

```
.create_ppp_edges_row(detector)
```

Arguments

detector	Detector object

Value

Point pattern dataset

.create_ppp_gaps_col *Creates ppp object of horizontal gaps*

Description

Creates ppp object of horizontal gaps

Usage

```
.create_ppp_gaps_col(detector)
```

Arguments

detector	Detector object

Value

Point pattern dataset

.create_ppp_gaps_row *Creates ppp object of vertical gaps*

Description

Creates ppp object of vertical gaps

Usage

```
.create_ppp_gaps_row(detector)
```

Arguments

detector	Detector object

Value

Point pattern dataset

`.Default_Detector` *Detector module A S3 class to represent a detector.*

Description

Detector module A S3 class to represent a detector.

Usage

```
.Default_Detector(name = "Default", date = NA, detector_width = NA,
  detector_height = NA, module_col_n = NA, module_row_n = NA,
  module_col_sizes = NA, module_row_sizes = NA, gap_col_sizes = NA,
  gap_row_sizes = NA, module_edges_col = NA, module_edges_row = NA,
  detector_inconsistency = NA, pix_matrix = NA, pix_dead = NA,
  dead_stats = NA, pix_dead_modules = NA, clumps = NA)
```

Arguments

`name`	detector's name
`date`	date
`detector_width`	detector's width
`detector_height`	detector's height
`module_col_n`	number of columns in the grid of modules
`module_row_n`	number of rows in the grid of modules
`module_col_sizes`	vector with widths of the modules
`module_row_sizes`	vector with heights of the modules
`gap_col_sizes`	vector with widths of the gaps
`gap_row_sizes`	vector with heights of the gaps
`module_edges_col`	vector of columns that contain edges of modules
`module_edges_row`	vector of rows that contain edges of modules
`detector_inconsistency`	counts inconsistencies found in parameters entered
`pix_matrix`	pixel matrix
`pix_dead`	dead pixels coordinates
`dead_stats`	dead pixel statistics
`pix_dead_modules`	assigned module for each dead pixel
`clumps`	clumps data (xyc_df data frame with pixels and their clump ID's, xyc_events data frame with clusters (clumps) and their clump ID's and centre coordinates)

Value

Detector object

`.derive_detector` *Deriving additional detector elements Conditions additional elements of Detector object that are frequently used later They are calculated from parameters defined in examples Matrices that contains xy coordiantes of edges of modules By definition, edges are part of modules (not part of gaps) i.e. for each module two pairs: first/last col and first/last row.*

Description

Deriving additional detector elements Conditions additional elements of Detector object that are frequently used later They are calculated from parameters defined in examples Matrices that contains xy coordiantes of edges of modules By definition, edges are part of modules (not part of gaps) i.e. for each module two pairs: first/last col and first/last row.

Usage

```
.derive_detector(detector)
```

Arguments

`detector`	Detector object

Value

Detector object

`.detector_edges` *Defines the coordinates of detector's edges using module and gap sizes Function is in 1d context to be applied to rows and cols separately. Edges are inside the modules (first/last row/col of module).*

Description

Defines the coordinates of detector's edges using module and gap sizes Function is in 1d context to be applied to rows and cols separately. Edges are inside the modules (first/last row/col of module).

Usage

```
.detector_edges(m, g)
```

Arguments

`m`	vector of module sizes
`g`	vectors of gap sizes

Value

Matrix with the information about the edges

.dist_closest_edge *A function to calculate closest distance to an edge for a pixel*

Description

A function to calculate closest distance to an edge for a pixel

Usage

```
.dist_closest_edge(x, size)
```

Arguments

x	Coordinate of pixel
size	Size of module

Value

distance to closest edge

.dist_edge *Function returns distance of a pixel to module edges.*

Description

Function returns distance of a pixel to module edges.

Usage

```
.dist_edge(xy, module_edges)
```

Arguments

xy	Coordinate of pixel
module_edges	vector of edges of a module

Value

tmp Distance to edges

.extract_detector_parameter *Checks whether a detector parameter is in the file string*

Description

Checks whether a detector parameter is in the file string

Usage

```
.extract_detector_parameter(file_string, parameter)
```

Arguments

file_string	String of a file context
parameter	Detector parameter

Value

parameter value

.extract_number *Internal function to convert string values to numbers*

Description

Internal function to convert string values to numbers

Usage

```
.extract_number(s)
```

Arguments

s	String expression?

Value

Numeric value

.getmode *Returns the mode of a set of data*

Description

Returns the mode of a set of data

Usage

```
.getmode(v)
```

Arguments

v	set of data

Value

uniqv the value of the mode

.get_clump_event_ppp *Creates ppp object for damaged detector events*

Description

Creates ppp object for damaged detector events

Usage

```
.get_clump_event_ppp(detector, incl_event_list = NA, height = NULL,
  width = NULL)
```

Arguments

detector	Detector object
incl_event_list	a list of events to be included
height	Detector height
width	Detector width

Value

ppp object for damaged detector events

.get_clump_pixel_ppp *Creates ppp object for damaged detector pixels*

Description

Creates ppp object for damaged detector pixels

Usage

```
.get_clump_pixel_ppp(detector, incl_event_list = NA)
```

Arguments

detector	Detector object
incl_event_list	a list of events to be included

Value

ppp object for damaged detector pixels

.get_detector_ppps *Generate detector ppps for edges and gaps*

Description

Generate detector ppps for edges and gaps

Usage

```
.get_detector_ppps(detector)
```

Arguments

detector	Detector object

Value

a list of ppps for edges and gaps

.get_ppp_dead_module *Generates ppp for the dead pixels for a selected module*

Description

Generates ppp for the dead pixels for a selected module

Usage

```
.get_ppp_dead_module(detector, row, col)
```

Arguments

detector	Detector object
row	module row number
col	module column number

Value

ppp of dead pixels

.mask_to_events *Converts mask (dead pixels) to events*

Description

Converts mask (dead pixels) to events

Usage

```
.mask_to_events(detector, dead_pix_mask, row = NA, col = NA)
```

Arguments

detector	Detector object
dead_pix_mask	Dead pixels mask
row	Module row number
col	Module col number

Value

list of pixels and events

.matrix_from_hdf *Reads in hdf file(s) and returns a pixel matrix*

Description

Reads in hdf file(s) and returns a pixel matrix

Usage

```
.matrix_from_hdf(detector, file_path)
```

Arguments

detector	Detector object
file_path	A list of paths to hdf files. Must be in the correct order.

Value

Data of a combined dataset from hdf files

.matrix_from_tiff *I/O module Reads in tiff file and returns a pixel matrix*

Description

I/O module Reads in tiff file and returns a pixel matrix

Usage

```
.matrix_from_tiff(detector, file_path)
```

Arguments

detector	Detector object
file_path	Path to the tiff file

Value

Pixel matrix with dead pixels flagged with 1

.matrix_from_xml *Reads in xml file and returns a pixel matrix*

Description

Reads in xml file and returns a pixel matrix

Usage

```
.matrix_from_xml(detector, file_path)
```

Arguments

detector	Detector object
file_path	Path to the xml file

Value

Data from an xml file

.norm_vec *Estimates the norm of a vector*

Description

Estimates the norm of a vector

Usage

```
.norm_vec(v)
```

Arguments

v	vector

Value

norm of the vector v

.plot_pixel *Plots pixel distance analysis*

Description

Plots pixel distance analysis

Usage

```
.plot_pixel(data, width, height, file_path = NA)
```

Arguments

data	Matrix containing pixel analysis data
width	Plot width
height	Plot height
file_path	Output path with an extension

.tr *Utils module Calculates the trace value of a square matrix*

Description

Utils module Calculates the trace value of a square matrix

Usage

```
.tr(m)
```

Arguments

m	A square matrix

Value

tr The trace value

`.xyc_pixels2events` *Modifying clusters to events (consisting of 1 pixel representing the cluster) Make into a point pattern of just events rather than pixels. Using xyc_ply object. Collapse in one point using centres for clusters, but end points for lines, type dependend: type 5 (closest to upper edge): ymin type 6 (closest to lower edge): ymax type 7 (closest to right edge): xmin type 8 (closest to left edge): xmax This is inspired by Perkin Elmer Detector and be replaced by other choices if desired.*

Description

Modifying clusters to events (consisting of 1 pixel representing the cluster) Make into a point pattern of just events rather than pixels. Using xyc_ply object. Collapse in one point using centres for clusters, but end points for lines, type dependend: type 5 (closest to upper edge): ymin type 6 (closest to lower edge): ymax type 7 (closest to right edge): xmin type 8 (closest to left edge): xmax This is inspired by Perkin Elmer Detector and be replaced by other choices if desired.

Usage

```
.xyc_pixels2events(xyc_ply)
```

Arguments

`xyc_ply`	clums data frame

Value

events

`.xyc_ply_func` *Clasifies clumps with respect to xy coordinates.*

Description

Clasifies clumps with respect to xy coordinates.

Usage

```
.xyc_ply_func(detector, xyc_pixel_df)
```

Arguments

`detector`	Detector object
`xyc_pixel_df`	xyc_pixel_df

Value

data frame with clasification results

available_detectors *A list of available detectors*

Description

A list of available detectors

Usage

```
available_detectors
```

Format

An object of class `character` of length 5.

check_detector_avail *Checks whether specified detector is available*

Description

Checks whether specified detector is available

Usage

```
check_detector_avail(detector_name)
```

Arguments

`detector_name`	The name of the detector

Value

True or False

create_module *Checks whether detector is available, if so, creates a Detector object*

Description

Checks whether detector is available, if so, creates a Detector object

Usage

```
create_module(detector_name)
```

Arguments

`detector_name`	The name of the detector

Value

Detector object

dead_pix_coords *Extracts a table of dead pixel coordinates from a pixel matrix*

Description

Extracts a table of dead pixel coordinates from a pixel matrix

Usage

```
dead_pix_coords(pix_matrix)
```

Arguments

pix_matrix	pixel matrix with dead pixels flagged with 1

Value

Table containing dead pixel coordinates

Dead_Stats *Analysis module A S3 class to represent dead pixels statistics summary*

Description

Analysis module A S3 class to represent dead pixels statistics summary

Usage

```
Dead_Stats(dead_n = NA, module_n = NA, module_count_arr = NA,
  module_count = NA, avg_dead_mod = NA, Chisq_s = NA,
  Chisq_df = NA, Chisq_p = NA)
```

Arguments

dead_n	Total number of damaged pixels:
module_n	Total number of modules
module_count_arr	Count of dead pixels in each quadrat
module_count	Count of dead pixels in each quadrat
avg_dead_mod	Average number of damaged pixels per module
Chisq_s	The Chi-Squared test statistic value
Chisq_df	Chi-Squared degrees of freedom
Chisq_p	Chi-Squared p-value

Value

Dead_Stats object

`dead_stats_summary` *Summary of damaged pixels*

Description

Summary of damaged pixels

Usage

```
dead_stats_summary(detector)
```

Arguments

`detector`	Detector object

Value

A string with damaged pixels overall statitics

`detector_consist_check` *Basic checks if parameters entered (slightly redundant on purpose) add up*

Description

Basic checks if parameters entered (slightly redundant on purpose) add up

Usage

```
detector_consist_check(detector = NA)
```

Arguments

`detector`	Detector object

detector_summary *Generates a string with the detector summary*

Description

Generates a string with the detector summary

Usage

```
detector_summary(detector)
```

Arguments

detector	Detector object

Value

String with the detector summary

dist_corner *A function to calculate pixel distances from corners*

Description

A function to calculate pixel distances from corners

Usage

```
dist_corner(detector)
```

Arguments

detector	Detector object

Value

Matrix containing pixel distances from corners

dist_edge_col *A function to calculate pixel horizontal distance to module edge*

Description

A function to calculate pixel horizontal distance to module edge

Usage

```
dist_edge_col(detector)
```

Arguments

detector	Detector object

Value

distance matrix

dist_edge_min *A function to calculate L-infinity distance to module edge*

Description

A function to calculate L-infinity distance to module edge

Usage

```
dist_edge_min(detector)
```

Arguments

detector	Detector object

Value

distance matrix

dist_edge_row *A function to calculate pixel vertical distance to module edge*

Description

A function to calculate pixel vertical distance to module edge

Usage

```
dist_edge_row(detector)
```

Arguments

detector	Detector object

Value

distance matrix

Excalibur_Detector *A S3 class to represent the Excalibur detector.*

Description

A S3 class to represent the Excalibur detector.

Usage

```
Excalibur_Detector()
```

Value

Excalibur detector object

find_clumps *Locates and clasifies clumps of a damaged detector*

Description

Locates and clasifies clumps of a damaged detector

Usage

```
find_clumps(detector, row = NA, col = NA)
```

Arguments

detector	Detector object
row	Module row number
col	Module column number

Value

Detector with events matrix

---

get_dead_pix_mask *Creates a mask matrix of dead pixels*

Description

Creates a mask matrix of dead pixels

Usage

```
get_dead_pix_mask(detector)
```

Arguments

detector	Detector object

Value

dead pixel mask

---

get_dead_stats *Generate summary of damaged pixels*

Description

Generate summary of damaged pixels

Usage

```
get_dead_stats(detector)
```

Arguments

detector	Detector object

Value

Dead_Stats object

get_events_mask *Generates events matrix (a matrix with pixels as 0 and events as 1)*

Description

Generates events matrix (a matrix with pixels as 0 and events as 1)

Usage

```
get_events_mask(detector)
```

Arguments

detector	Detector object

Value

events mask

get_ppp_dead *Generates ppp for the dead pixels*

Description

Generates ppp for the dead pixels

Usage

```
get_ppp_dead(detector)
```

Arguments

detector	Detector object

Value

ppp of dead pixels

glm_pixel_ctr_eucl *Fits pixel distance from the centre using glm*

Description

Fits pixel distance from the centre using glm

Usage

```
glm_pixel_ctr_eucl(detector)
```

Arguments

detector	Detector object

Value

Fitted model

glm_pixel_ctr_linf *Fits pixel parallel maxima from the centre using glm*

Description

Fits pixel parallel maxima from the centre using glm

Usage

```
glm_pixel_ctr_linf(detector)
```

Arguments

detector	Detector object

Value

Fitted model

`glm_pixel_dist_edge_col` *Fits pixel distances from the module edges by column using glm*

Description

Fits pixel distances from the module edges by column using glm

Usage

```
glm_pixel_dist_edge_col(detector)
```

Arguments

`detector`	Detector object

Value

Fitted model

`glm_pixel_dist_edge_row` *Fits pixel distances from module edges by row using glm*

Description

Fits pixel distances from module edges by row using glm

Usage

```
glm_pixel_dist_edge_row(detector)
```

Arguments

`detector`	Detector object

Value

Fitted model

`ini_graphics` *Starts the graphics device driver for producing graphics with respect to a chosen format*

Description

Starts the graphics device driver for producing graphics with respect to a chosen format

Usage

```
ini_graphics(file_path)
```

Arguments

`file_path`	Output path with an extension

`load_pix_matrix` *A function to load pixel data*

Description

A function to load pixel data

Usage

```
load_pix_matrix(detector, file_path)
```

Arguments

`detector`	The name of the detector to be used
`file_path`	Path(s) to the file(s) containing dead pixel information

Value

Detector object

perform_glm *Performs glm fitting on the specified symbolic expression*

Description

Performs glm fitting on the specified symbolic expression

Usage

```
perform_glm(symb_expr, family = binomial(link = logit))
```

Arguments

symb_expr	symbolic description of the linear predictor
family	a description of the error distribution

Value

Fitted model

glm_git fitted model

PerkinElmerCropped1600_Detector *A S3 class to represent the PerkinElmerCropped1600 detector.*

Description

A S3 class to represent the PerkinElmerCropped1600 detector.

Usage

```
PerkinElmerCropped1600_Detector()
```

Value

PerkinElmerCropped1600 detector object

`PerkinElmerFull_Detector` *A S3 class to represent the PerkinElmerFull detector.*

Description

A S3 class to represent the PerkinElmerFull detector.

Usage

```
PerkinElmerFull_Detector()
```

Value

PerkinElmerFul detector object

`PerkinElmerRefurbished_Detector` *A S3 class to represent the PerkinElmerRefurbished detector.*

Description

A S3 class to represent the PerkinElmerRefurbished detector.

Usage

```
PerkinElmerRefurbished_Detector()
```

Value

PerkinElmerRefurbished detector object

`Pilatus_Detector` *A S3 class to represent the PerkinElmerRefurbished detector.*

Description

A S3 class to represent the PerkinElmerRefurbished detector.

Usage

```
Pilatus_Detector()
```

Value

Pilatus detector object

pixel_dist_ctr_eucl *A function to calculate euclidean distance from the centre for each pixel*

Description

A function to calculate euclidean distance from the centre for each pixel

Usage

```
pixel_dist_ctr_eucl(detector)
```

Arguments

detector	Detector object

Value

Matrix of euclidean distances

pixel_dist_ctr_linf *A function to calculate parallel maxima from the centre for each pixel*

Description

A function to calculate parallel maxima from the centre for each pixel

Usage

```
pixel_dist_ctr_linf(detector)
```

Arguments

detector	Detector object

Value

Matrix of parallel maxima

plot_angles *Plots NN angles*

Description

Plots NN angles

Usage

```
plot_angles(ppp_obj, caption, file_path = NA)
```

Arguments

ppp_obj	ppp object
caption	caption of the figure
file_path	file path

plot_arrows *Plots NN oriented arrrows*

Description

Plots NN oriented arrrows

Usage

```
plot_arrows(ppp_obj, caption, file_path = NA)
```

Arguments

ppp_obj	ppp object
caption	caption of the figure
file_path	file path

plot_counts *Plots dead pixel counts*

Description

Plots dead pixel counts

Usage

```
plot_counts(module_count_arr, caption, file_path = NA)
```

Arguments

module_count_arr	Counts per array
caption	caption of the figure
file_path	file path

plot_density *Plots module Plots density*

Description

Plots module Plots density

Usage

```
plot_density(ppp_obj, caption, file_path = NA, adjust = 1)
```

Arguments

ppp_obj	ppp object
caption	caption of the figure
file_path	file path
adjust	Kernel density bandwidth

plot_detector *Plot detector*

Description

Plot detector

Usage

```
plot_detector(detector, file_path = NA, caption = TRUE)
```

Arguments

detector	Detector object
file_path	Output file path
caption	Flag to turn on/off figure caption

plot_detector_angles *A function to plot NN angles of dead pixels of detector or module*

Description

A function to plot NN angles of dead pixels of detector or module

Usage

```
plot_detector_angles(detector, file_path = NA, row = NA, col = NA,
  caption = TRUE)
```

Arguments

detector	Detector object
file_path	Output file path
row	Module row number
col	Module column number
caption	Flag to turn on/off figure caption

plot_detector_arrows *A function to plot NN oriented arrrows of dead pixels of detector or module*

Description

A function to plot NN oriented arrrows of dead pixels of detector or module

Usage

```
plot_detector_arrows(detector, file_path = NA, row = NA, col = NA,
  caption = TRUE)
```

Arguments

detector	Detector object
file_path	Output file path
row	Module row number
col	Module column number
caption	Flag to turn on/off figure caption

plot_detector_cnt_mod *A function to plot detector with dead pixel counts per module*

Description

A function to plot detector with dead pixel counts per module

Usage

```
plot_detector_cnt_mod(detector, file_path = NA, row = NA, col = NA,
  caption = TRUE)
```

Arguments

detector	Detector object
file_path	Output file path
row	Module row number
col	Module column number
caption	Flag to turn on/off figure caption

plot_detector_damaged *A function to plot detector with damaged pixels*

Description

A function to plot detector with damaged pixels

Usage

```
plot_detector_damaged(detector, file_path = NA, caption = TRUE)
```

Arguments

detector	Detector object
file_path	Output file path
caption	Flag to turn on/off figure caption

plot_detector_density *A function to plot densities of dead pixels of detector or module*

Description

A function to plot densities of dead pixels of detector or module

Usage

```
plot_detector_density(detector, file_path = NA, adjust = 1, row = NA,
  col = NA, caption = TRUE)
```

Arguments

detector	Detector object
file_path	Output file path
adjust	Kernel density bandwidth
row	Module row number
col	Module column number
caption	Flag to turn on/off figure caption

`plot_detector_kfg` *Plots K, F, G functions*

Description

Plots K, F, G functions

Usage

```
plot_detector_kfg(detector, func, file_path = NA, row = NA, col = NA,
  caption = TRUE)
```

Arguments

`detector`	Detector object
`func`	Function name
`file_path`	Output file path
`row`	module row number
`col`	module column number
`caption`	Flag to turn on/off figure caption

`plot_detector_module_damaged` *A function to plot detector module with damaged pixels*

Description

A function to plot detector module with damaged pixels

Usage

```
plot_detector_module_damaged(detector, col, row, file_path = NA,
  caption = TRUE)
```

Arguments

`detector`	Detector object
`col`	Module column number
`row`	Module row number
`file_path`	Output file path
`caption`	Flag to turn on/off figure caption

plot_events *Plots damaged detector events*

Description

Plots damaged detector events

Usage

```
plot_events(detector, file_path = NA, caption = TRUE,
  incl_event_list = NA, plot_edges_gaps = TRUE)
```

Arguments

detector	Detector object
file_path	Output file path
caption	Flag to turn on/off figure caption
incl_event_list	a list of events to be included
plot_edges_gaps	Plot edgees gaps

plot_events_angles *Plots angles graph of events of a detector or module*

Description

Plots angles graph of events of a detector or module

Usage

```
plot_events_angles(detector, file_path = NA, row = NA, col = NA,
  caption = TRUE, incl_event_list = NA)
```

Arguments

detector	Detector object
file_path	Output file path
row	Module row number
col	Module column number
caption	Flag to turn on/off figure caption
incl_event_list	a list of events to be included

plot_events_arrows *Plots arrows graph of events of a detector or module*

Description

Plots arrows graph of events of a detector or module

Usage

```
plot_events_arrows(detector, file_path = NA, row = NA, col = NA,
  caption = TRUE, incl_event_list = NA)
```

Arguments

detector	Detector object
file_path	Output file path
row	Module row number
col	Module column number
caption	Flag to turn on/off figure caption
incl_event_list	a list of events to be included

plot_events_count *Plots events count per detector or module*

Description

Plots events count per detector or module

Usage

```
plot_events_count(detector, file_path = NA, row = NA, col = NA,
  caption = TRUE, incl_event_list = NA)
```

Arguments

detector	Detector object
file_path	Output file path
row	Module row number
col	Module column number
caption	Flag to turn on/off figure caption
incl_event_list	a list of events to be included

plot_events_density *Plots density graph of events of a detector or module*

Description

Plots density graph of events of a detector or module

Usage

```
plot_events_density(detector, file_path = NA, adjust = 0.25,
  row = NA, col = NA, caption = TRUE, incl_event_list = NA)
```

Arguments

detector	Detector object
file_path	Output file path
adjust	Kernel density bandwidth
row	Module row number
col	Module column number
caption	Flag to turn on/off figure caption
incl_event_list	a list of events to be included

plot_events_kfg *Plots K, F, G functions of a detector or module*

Description

Plots K, F, G functions of a detector or module

Usage

```
plot_events_kfg(detector, func, file_path = NA, row = NA, col = NA,
  caption = TRUE, incl_event_list = NA)
```

Arguments

detector	Detector object
func	Function name
file_path	Output file path
row	Module row number
col	Module column number
caption	Flag to turn on/off figure caption
incl_event_list	a list of events to be included

plot_kfg *Plots K, F, G functions*

Description

Plots K, F, G functions

Usage

```
plot_kfg(ppp_obj, func, file_path = NA, caption = TRUE)
```

Arguments

ppp_obj	ppp object
func	Function name
file_path	Output file path
caption	Flag to turn on/off figure caption

plot_module_events *Plots damaged detector module events*

Description

Plots damaged detector module events

Usage

```
plot_module_events(detector, col, row, file_path = NA, caption = TRUE,
  incl_event_list = NA)
```

Arguments

detector	Detector object
col	Module column number
row	Module row number
file_path	Output file path
caption	Flag to turn on/off figure caption
incl_event_list	a list of events to be included

plot_pixel_ctr_eucl *Calculates and plots pixel euclidean distance from the centre*

Description

Calculates and plots pixel euclidean distance from the centre

Usage

```
plot_pixel_ctr_eucl(detector, file_path = NA)
```

Arguments

detector	Detector object
file_path	Output file path

plot_pixel_ctr_linf *Calculates and plots pixel parallel maxima from the centre*

Description

Calculates and plots pixel parallel maxima from the centre

Usage

```
plot_pixel_ctr_linf(detector, file_path = NA)
```

Arguments

detector	Detector object
file_path	Output file path

plot_pixel_dist_corner *Calculates and plots pixel distances from corners*

Description

Calculates and plots pixel distances from corners

Usage

```
plot_pixel_dist_corner(detector, file_path = NA)
```

Arguments

detector	Detector object
file_path	Output file path

plot_pixel_dist_edge *Calculates and plots L-infinity distances from the module edges*

Description

Calculates and plots L-infinity distances from the module edges

Usage

```
plot_pixel_dist_edge(detector, file_path = NA)
```

Arguments

detector	Detector object
file_path	Output file path

plot_pixel_dist_edge_col *Calculates and plots horizontal distances from the module edges*

Description

Calculates and plots horizontal distances from the module edges

Usage

```
plot_pixel_dist_edge_col(detector, file_path = NA)
```

Arguments

detector	Detector object
file_path	Output file path

plot_pixel_dist_edge_row *Calculates and plots vetical distances from the module edges*

Description

Calculates and plots vetical distances from the module edges

Usage

```
plot_pixel_dist_edge_row(detector, file_path = NA)
```

Arguments

detector	Detector object
file_path	Output file path

`readin_detector` *Reads in a user defined detector from a file*

Description

Reads in a user defined detector from a file

Usage

```
readin_detector(file_path)
```

Arguments

`file_path`	A path to the user defined detector file

Value

Detector object

`remove_high_density_cluster` *Remove high density cluster of dead pixels Recalculates dead statistics and clumps if they were present in the Detector object*

Description

Remove high density cluster of dead pixels Recalculates dead statistics and clumps if they were present in the Detector object

Usage

```
remove_high_density_cluster(detector, min_pts = 30, eps_adjust = 0.05)
```

Arguments

`detector`	Detector object
`min_pts`	minimum points argument of dbscan function
`eps_adjust`	adjust eps

Value

detector object with high density cluster of pixels removed

which_module — *Module module Returns row or column of a module that a dead pixel belongs to*

Description

Module module Returns row or column of a module that a dead pixel belongs to

Usage

```
which_module(coo, me)
```

Arguments

coo	x or y coordinate of a dead pixel
me	module edges

Value

row or column number

which_module_idx — *Function returns both col and row number of a dead pixel.*

Description

Function returns both col and row number of a dead pixel.

Usage

```
which_module_idx(x, y, module_edges_col, module_edges_row)
```

Arguments

x	pixel x coordinate
y	pixel y coordinate
module_edges_col	vector of columns that contain edges of modules
module_edges_row	vector of rows that contain edges of modules

Value

tmp

Index