

# Package ‘RLumSTARR’

September 4, 2020

**Type** Package

**Title** Spatially Resolved Radiofluorescence Analysis

**Version** 0.1.0.9000-31

**Author** Sebastian Kreutzer [aut, cre] (<<https://orcid.org/0000-0002-0734-2199>>),  
Dirk Mittelstrass [aut]

**Maintainer** Sebastian Kreutzer <[sebastian.kreutzer@aber.ac.uk](mailto:sebastian.kreutzer@aber.ac.uk)>

**Description** Analysing spatially resolved radiofluorescence data using ImageJ. A collection of functions to support early work on the subject.

**License** GPL-3

**Depends** R (>= 3.5.0),  
utils,  
methods

**Imports** Luminescence (>= 0.9.8),  
rjags (>= 4-8),  
coda (>= 0.19-1)

**Encoding** UTF-8

**Language** en-GB

**LazyData** true

**RoxygenNote** 7.1.1

## R topics documented:

RLumSTARR-package . . . . .	2
create_RFCurveArray . . . . .	2
extract_TRUELight . . . . .	3
get_MCMCParameter . . . . .	4
run_ImageJ . . . . .	5
<b>Index</b>	7

---

RLumSTARR-package	<i>SpaTiAlly Resolved Radiofluorescence</i>
-------------------	---

---

## Description

A collection of functions to analyse spatially resolved radiofluorescence data

## Details

### Funding

- Sebastian Kreutzer received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 844457.

## Author(s)

Sebastian Kreutzer, Geography & Earth Sciences, Aberystwyth University (United Kingdom)  
Dirk Mittelstrass, TU Dresden (Germany) , RLum Developer Team

## References

##TODO

---

create_RFCurveArray	<i>Create Multidimensional Curve Arrays from File input</i>
---------------------	---

---

## Description

Helper function to create a multidimensional curve array to prepare the Bayesian process

## Usage

```
create_RFCurveArray(files)
```

## Arguments

files [list \(required\)](#): list of .rf files to be imported

## Value

Returns a list with two arrays for the RF\_nat and the RF\_reg curve

## Function version

0.1.0

## How to cite

Kreutzer, S., 2020. create\_RFCurveArray(): Create Multidimensional Curve Arrays from File input. Function version 0.1.0. In: Kreutzer, S., Mittelstrass, D., 2020. RLumSTARR: Spatially Resolved Radiofluorescence Analysis. R package version 0.1.0.9000-31.

**Author(s)**

Sebastian Kreutzer, Geography & Earth Sciences, Aberystwyth University (United Kingdom) ,  
RLum Developer Team

**Examples**

```
##TODO
```

---

extract_TRUELight	<i>Extract True Light from the Camera measurements using a Bayesian Approach</i>
-------------------	--

---

**Description**

Run the Bayesian model to extract the true light from a ROI

**Usage**

```
extract_TRUELight(data, ROI = 1, method_control = list())
```

**Arguments**

`data` [array \(required\)](#): curve array created by [create\\_RFCurveArray](#)  
`ROI` [numeric \(with default\)](#): ROI to be analysed  
`method_control` [list \(optional\)](#): parameter to be passed to `rjags`. Supported are `n.chain`, `n.iter`, `thin`, `variable.names`, `model`

**Value**

Returns a list with an [Luminescence::RLum.Data.Curve](#) object (the RF curve with the true light) and the [rjags::coda.samples](#) output for further processing. *Note: Regardless the observed variable, the parameter `alpha` will be always be used to create the curve*

**Function version**

0.1.0

**How to cite**

Kreutzer, S., 2020. `extract_TRUELight()`: Extract True Light from the Camera measurements using a Bayesian Approach. Function version 0.1.0. In: Kreutzer, S., Mittelstrass, D., 2020. RLum-STARR: Spatially Resolved Radiofluorescence Analysis. R package version 0.1.0.9000-31.

**Author(s)**

Sebastian Kreutzer, Geography & Earth Sciences, Aberystwyth University (United Kingdom) ,  
RLum Developer Team

**Examples**

```
##TODO
```

---

get_MCMCParameter	<i>Extracts a parameter from an MCMC list</i>
-------------------	---

---

## Description

Short cut to extract a parameter from an MCMC list. If more processing is wanted, the 'coda' package can be used

## Usage

```
get_MCMCParameter(mcmc, parameter, prob = 0.95, unlist = TRUE)
```

## Arguments

mcmc	<a href="#">coda::mcmc</a> or <a href="#">coda::mcmc.list</a> ( <b>required</b> ) input
parameter	<a href="#">character</a> ( <b>required</b> ): name of the parameter to be extracted
prob	<a href="#">numeric</a> ( <i>with default</i> ): probability for the HPD calculation
unlist	<a href="#">logical</a> ( <i>with default</i> ): if TRUE the output is a <a href="#">matrix</a> otherwise a <a href="#">list</a>

## Value

Returns a matrix with the parameter value or a [list](#)

## Function version

0.1.0

## How to cite

Kreutzer, S., 2020. `get_MCMCParameter()`: Extracts a parameter from an MCMC list. Function version 0.1.0. In: Kreutzer, S., Mittelstrass, D., 2020. RLumSTARR: Spatially Resolved Radiofluorescence Analysis. R package version 0.1.0.9000-31.

## Author(s)

Sebastian Kreutzer, Geography & Earth Sciences, Aberystwyth University (United Kingdom) ,  
RLum Developer Team

## See Also

[coda::HPDinterval](#)

## Examples

```
##TODO
```

---

run_ImageJ	<i>Run ImageJ SR-RF macro</i>
------------	-------------------------------

---

## Description

The script runs the SR-RF ImageJ macro in batch mode out of R

## Usage

```
run_ImageJ(
  path,
  RF_nat = "default",
  RF_reg = "default",
  bg_rm = "take_from_RF_reg",
  image_group_size = 5,
  image_alignment = TRUE,
  first_slices_rm = FALSE,
  noise_tolorance = 10,
  ROI_size = 10,
  center_x = 0.5,
  center_y = 0.5,
  diameter = 0.9,
  use_predefined_ROIs = FALSE,
  channel_time = 5,
  save_workflow_images = FALSE,
  save_additional_results = FALSE,
  save_signal_decay_videos = FALSE,
  offset_time = 0,
  .ImageJ = "/Applications/Fiji.app/Contents/MacOS/ImageJ-macosx"
)
```

## Arguments

path	<b>character (required)</b> : path to files to be analysed
RF_nat	<b>character (default)</b> : name of the RF_nat file
RF_reg	<b>character (default)</b> : name of the RF_reg file
bg_rm	<b>character (with default)</b> : background subtraction options. Allowed are none (no background subtraction), take_from_RF_reg (takes the last 100 channels from the RF_reg signal: dangerous) or <your file name> (this does not work in batch mode)
image_group_size	<b>numeric (with default)</b> : grouping value for running median to remove outliers
image_alignment	<b>logical (with default)</b> : enable/disable image alignment
first_slices_rm	<b>logical (with default)</b> : remove first slice of each curve set
noise_tolorance	<b>numeric (with default)</b> : noise tolerance parameter
ROI_size	<b>numeric (with default)</b> : ROI size in pixel

center\_x        **numeric** (*with default*): aliquot ROI centre x-coordinate  
 center\_y        **numeric** (*with default*): aliquot ROI centre y-coordinate  
 diameter        **numeric** (*with default*): relative diameter aliquot ROI  
 use\_predefined\_ROIs        **logical** (*with default*): use pre-defined ROIs imported from a file ROIs.zip found in the same folder as the files  
 channel\_time    **numeric** (*with default*): channel time, this parameter was set the moment the sequence was written  
 save\_workflow\_images        **logical** (*with default*): enable/disable writing of additional workflow images  
 save\_additional\_results        **logical** (*with default*): enable/disable writing of additional workflow images  
 save\_signal\_decay\_videos        **logical** (*with default*): enable/disable writing of additional workflow videos  
 offset\_time     **numeric** (*with default*): offset time for the time axis  
 .ImageJ        **numeric** (*with default*): Path to ImageJ (the macro is shipped with the package)

### Value

This functions returns the path of the analysed data

### Function version

0.1.0

### How to cite

Kreutzer, S., 2020. run\_ImageJ(): Run ImageJ SR-RF macro. Function version 0.1.0. In: Kreutzer, S., Mittelstrass, D., 2020. RLumSTARR: Spatially Resolved Radiofluorescence Analysis. R package version 0.1.0.9000-31.

### Author(s)

Sebastian Kreutzer, Geography & Earth Sciences, Aberystwyth University (United Kingdom) ,  
RLum Developer Team

### Examples

##TODO

# Index

## \* **package**

RLumSTARR-package, [2](#)

array, [3](#)

character, [4](#), [5](#)

coda::HPDinterval, [4](#)

coda::mcmc, [4](#)

coda::mcmc.list, [4](#)

create\_RFCurveArray, [2](#), [3](#)

extract\_TRUELight, [3](#)

get\_MCMCParameter, [4](#)

list, [2–4](#)

logical, [4–6](#)

Luminescence::RLum.Data.Curve, [3](#)

matrix, [4](#)

numeric, [3–6](#)

rjags::coda.samples, [3](#)

RLumSTARR (RLumSTARR-package), [2](#)

RLumSTARR-package, [2](#)

run\_ImageJ, [5](#)