

Package: LEEF.measurement.flowcam (via r-universe)

September 12, 2024

Type Package

Title Pre-Process and Extract Flowcam Data

Version 0.8.31

Maintainer Rainer M. Krug <Rainer.Krug@uzh.ch>

BugReports <https://github.com/LEEF-UZH/LEEF.measurement.flowcam/issues>

URL <https://github.com/LEEF-UZH/LEEF.measurement.flowcam>

Description More about what it does (maybe more than one line) Use four spaces when indenting paragraphs within the Description.

License MIT + file LICENSE

Depends R (>= 3.5.0)

Imports data.table, yaml, utils, plyr, dplyr, purrr, magrittr, e1071, tidyselect, stats, loggit, randomForest

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Suggests testthat, openssl

Repository <https://leef-uzh.r-universe.dev>

RemoteUrl <https://github.com/LEEF-UZH/LEEF.measurement.flowcam>

RemoteRef master

RemoteSha 0885ed543995078ec3bb3bb9405d9414e7e40f6d

Contents

add_new_data	2
classify	3
extractor_flowcam	4
extractor_flowcam_classify	4
extractor_flowcam_filter	5

extractor_flowcam_prepare	5
load_parameter	6
par_classifier_constant	6
par_classifier_increasing	6
par_species_tracked	7
par_template	7
pre_processor_flowcam	8
print_parameter	8
raw_data_ok	9
register	9
save_parameter	10
Index	11

add_new_data	<i>Check if data in input folder is OK and move to raw data folder</i>
--------------	--

Description

Check if data in input folder is OK and move to raw data folder

Usage

```
add_new_data(input, output)
```

Arguments

input	The folder, where a folder flowcam is located which contains the new files.
output	A folder, which contains a subfolder called flowcam, i.e. the usually the raw data folder, into which the files will be moved to.

Value

a list which contains the individual results for each file. TRUE if moved, FALSE if an error occurred. Details of the error re in the error files in the input/flowcam directory.

classify	<i>Classify algae_traits nad calculates densities</i>
----------	---

Description

Classify algae_traits nad calculates densities

Usage

```
classify(  
  algae_traits,  
  classifiers_constant,  
  classifiers_increasing,  
  composition,  
  exp_design,  
  species_tracked,  
  timestamp  
)
```

Arguments

algae_traits	algae_traits
classifiers_constant	constant temperature classifier
classifiers_increasing	increasing temperature classifier
composition	composition
exp_design	experimental design
species_tracked	species tracked
timestamp	timestamp to be used to stamp the classified data

Value

list containing two objects:

- algae_traits including species
- algae_densities densities of the differenc particles identifieds

extractor_flowcam *Preprocessor flowcam data*

Description

extract data from all *_classes*_data.csv files

Usage

```
extractor_flowcam(input, output)
```

Arguments

input	directory from which to read the data
output	directory to which to write the data

Value

invisibly TRUE when completed successful

extractor_flowcam_classify
Preprocessor flowcam data

Description

extract data from all *_classes*_data.csv files

Usage

```
extractor_flowcam_classify(input, output)
```

Arguments

input	directory from which to read the data
output	directory to which to write the data

Value

invisibly TRUE when completed successful

extractor_flowcam_filter
Preprocessor flowcam data

Description

extract data from all *_classes*_data.csv files

Usage

extractor_flowcam_filter(input, output)

Arguments

input	directory from which to read the data
output	directory to which to write the data

Value

invisibly TRUE when completed successful

extractor_flowcam_prepare
Preprocessor flowcam data

Description

extract data from all *_classes*_data.csv files

Usage

extractor_flowcam_prepare(input, output)

Arguments

input	directory from which to read the data
output	directory to which to write the data

Value

invisibly TRUE when completed successful

load_parameter	<i>Load parameter from file</i>
----------------	---------------------------------

Description

Load parameter from file

Usage

```
load_parameter(file = "parameter.yaml")
```

Arguments

file	name of parameter file
------	------------------------

Value

invisibly TRUE

par_classifier_constant	<i>Name of classifier file for constant temperature treatment</i>
-------------------------	---

Description

Name of classifier file for constant temperature treatment

Usage

```
par_classifier_constant(value)
```

par_classifier_increasing	<i>Name of classifier file for increasing temperature treatment</i>
---------------------------	---

Description

Name of classifier file for increasing temperature treatment

Usage

```
par_classifier_increasing(value)
```

par_species_tracked	<i>Species tracked</i>
---------------------	------------------------

Description

Species tracked

Usage

par_species_tracked(value)

par_template	<i>Template function to assign value to parameter in the package wide cache</i>
--------------	---

Description

assign the function to a new cvariable and the name of the function will be used for the parameter name. e.g:

- `fps <- par_template`

Usage

par_template(value)

Arguments

value	if missing, the value of the parameter will be returned, NULL if the parameter does not exist; if specified, the parameter will be set to the value
-------	---

Value

the (new) value of the argument

```
pre_processor_flowcam Preprocessor flowcam data
```

Description

Convert all .tif files in flowcam folder to zip compressed TIFF.

Usage

```
pre_processor_flowcam(input, output)
```

Arguments

input	directory from which to read the data
output	directory to which to write the data

Value

invisibly TRUE when completed successful

```
print_parameter Print the flowcam parameter
```

Description

Print the flowcam parameter

Usage

```
print_parameter(print_as_yaml = TRUE, echo = TRUE)
```

Arguments

print_as_yaml	Print in yaml formatted text; ~ stands for NULL
echo	if the results should be shown on the screen or only the cache returned as a list.

Value

invisible returns list of parameter for further processing

raw_data_ok	<i>Check if data in raw data folder is OK</i>
-------------	---

Description

Check if data in raw data folder is OK

Usage

```
raw_data_ok(input)
```

Arguments

input raw data folder containing flowcam data, i.e usually is some/path/flowcam

Value

TRUE if ok, FALSE or list of problems if not

Examples

```
## Not run:  
raw_data_ok()  
  
## End(Not run)
```

register	<i>Register the processing of flowcam data in the LEEF package</i>
----------	--

Description

Register the processing of flowcam data in the LEEF package

Usage

```
register()
```

Value

invisibly TRUE when completed successful

save_parameter	<i>Save parameter into .yaml file</i>
----------------	---------------------------------------

Description

Save parameter into .yaml file

Usage

```
save_parameter(file = "parameter.yaml")
```

Arguments

file	name of parameter file
------	------------------------

Value

invisibly TRUE

Index

`add_new_data`, [2](#)

`classify`, [3](#)

`extractor_flowcam`, [4](#)

`extractor_flowcam_classify`, [4](#)

`extractor_flowcam_filter`, [5](#)

`extractor_flowcam_prepare`, [5](#)

`load_parameter`, [6](#)

`par_classifier_constant`, [6](#)

`par_classifier_increasing`, [6](#)

`par_species_tracked`, [7](#)

`par_template`, [7](#)

`pre_processor_flowcam`, [8](#)

`print_parameter`, [8](#)

`raw_data_ok`, [9](#)

`register`, [9](#)

`save_parameter`, [10](#)