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

October 25, 2024

Type Package

Title Prepares Movies for Analysis with Bemovi and Extracts Data

Description Module for the LEEF pipeline to process bemovi data.

Version 0.8.40

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

BugReports `https://github.com/LEEF-UZH/LEEF.measurement.bemovi/issues/`

URL `https://leef-uzh.github.io/LEEF.measurement.bemovi/`,
`https://github.com/LEEF-UZH/LEEF.measurement.bemovi/`

License MIT + file LICENSE

Depends R (>= 3.5.0)

Imports bemovi.LEEF (>= 1.0.1), R.utils, utils, parallel, data.table, dplyr, purrr, tidyselect, magrittr, yaml, rlang, e1071, loggit

Additional_repositories `https://leef-uzh.github.io/drat/`

Encoding UTF-8

LazyData true

RoxygenNote 7.2.1

Suggests testthat, openssl

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

RemoteUrl https://github.com/LEEF-UZH/LEEF.measurement.bemovi

RemoteRef master

RemoteSha 1a2748d6150043a2000991927fd087c253ed441e

Contents

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

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 `bemovi` is located which contains the new files.
`output`	A folder, which contains a subfolder called `bemovi`, i.e. the usually the raw data folder, into which the fioles will be moved to.

Value

a `list` which contains the individual reseults for each file. `TRUE` if moved, `FALSE` if an error occured. Details of the eror re in the error files in the `input/bemovi` directory.

`check_tools_path` *Checks tools_path() if all tools are there or downloads them*

Description

Checks tools_path() if all tools are there or downloads them

Usage

```
check_tools_path(path = tools_path(), download = FALSE)
```

Arguments

`path`	path for the tools. defaults to `tools_path()`
`download`	if `TRUE`, the tools are downloaded. If `FALSE`, the default, the ptools path is only checked.

Value

`TRUE` if all tools are installed. a named list of tools which re not installed.

Examples

```
check_tools_path()
```

`classify` *Classify* `morph_mvt` *nad calculates densities*

Description

Classify `morph_mvt` nad calculates densities

Usage

```
classify(
  bemovi_extract,
  morph_mvt,
  trajectory_data,
  classifiers_constant,
  classifiers_increasing,
  video_description_file,
  composition
)
```

Arguments

`morph_mvt`	merged track data - one row per particle
`classifiers_constant`	constant tempersture classifier
`classifiers_increasing`	increasing temperature classifier

Value

`morph_mvt` with the classified species and probabilities

`extractor_bemovi` *Extractor bemovi data*

Description

Analyse all `.avi` files in `bemovi` folder and save as `.rds` file.

Usage

```
extractor_bemovi(input, output)
```

Arguments

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

Details

This function is executing `extractor_bemovi_particle()`, `extractor_bemovi_trajectory()` and finally `extractor_bemovi_merge()` for all parameter files in the format `bemovi_extract*.yml`

Value

invisibly `TRUE` when completed successful

`extractor_bemovi_classify`

Extractor bemovi to id species and calculate density

Description

This function id's the species and calculates the densities.

Usage

```
extractor_bemovi_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_bemovi_filter`

Extractor bemovi to filter traced particles

Description

This function applies a filter.

Usage

```
extractor_bemovi_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_bemovi_merge` *Extractor bemovi data*

Description

Analyse all `.avi` files in `bemovi` folder and save as `.rds` file.

Usage

```
extractor_bemovi_merge(input, output)
```

Arguments

`input`	only for compatibility - not used
`output`	directory to which to write the data

Details

This function is extracting data to be added to the database (and therefore make accessible for further analysis and forecasting) from `.avi` files.

Value

invisibly `TRUE` when completed successful

`extractor_bemovi_overlay` *Extractor bemovi data trajectories*

Description

This function is calculating the trajectories

Usage

```
extractor_bemovi_overlay(input, output)
```

Arguments

`input`	only for compatibility - not used
`output`	directory to which to write the data

Value

invisibly `TRUE` when completed successful

`extractor_bemovi_particle` *Extractor bemovi data particles*

Description

This function **requires** the following files and directories:

Usage

```
extractor_bemovi_particle(input, output)
```

Arguments

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

Details

- one `input` directory which contains
  - a folder named `bemovi` with
    - the `.avi` files
    - a file `bemovi_extract.yml` containing all the parameter for the analysis. This parameter file will be loaded for the analysis.
  - a folder named `00.general.parameter` at the same level as `input`

This function **creates** the following folder if it does not exist:

- `output\bemovi` in which will contain
  - diverse output as created by `bemovi.LEEF::locate_and_measure_particles()`
  - all metadata et al files in the `input/bemovi` folder which are **not** `.avi` and `.metadata` files

Value

invisibly `TRUE` when completed successful

`extractor_bemovi_trajectory` *Extractor bemovi data trajectories*

Description

This function is calculating the trajectories

Usage

```
extractor_bemovi_trajectory(input, output)
```

Arguments

`input`	only for compatibility - not used
`output`	directory to which to write the data

Value

invisibly `TRUE` when completed successful

`pre_processor_bemovi` *Preprocessor bemovi data*

Description

Convert all `.cxd` files in the `input/bemovi` folder to non-proprietary `avi` format.

Usage

```
pre_processor_bemovi(input, output)
```

Arguments

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

Details

This function **requires** the following files and directories:

- one `input` directory which contains
  - a folder named `bemovi` with
    - the `.cxd` files
    - a file `bemovi_extract.yml` containing all the parameter for the analysis. This parameter file will be loaded for the analysis.
  - a folder named `00.general.parameter` at the same level as `input`

This function **creates** the following folder if it does not exist:

- `output\bemovi` in which will contain
  - the `.avi` files (converted `.cxd` files)
  - the `.metadata` text files containing the extracted metadata from the `.cxd` files
  - the files recursively copied from the `00.general.parameter` folder

Value

invisibly `TRUE` when completed successful

`raw_data_ok` *Check if data in raw data folder is OK*

Description

Check if data in raw data folder is OK

Usage

```
raw_data_ok(input)
```

Arguments

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

Value

`TRUE` if ok, `FALSE` or `list` of problems if not

Examples

```
## Not run:
raw_data_ok()

## End(Not run)
```

`register` *Register the processing of bemovi data in the LEEF package*

Description

Register the processing of bemovi data in the LEEF package

Usage

```
register()
```

Value

invisibly `TRUE` when completed successful

`tools_path` *Read and write tools_path*

Description

Read or write the tools path where the tools are located in. If no parameter is given, the path will be returned as a list.

Usage

```
tools_path(tools_path)
```

Arguments

`tools_path`	`character vector` of length one containing the directory where the tools are located

Value

tools directory. If values have set, the value before the change.

Examples

```
tools_path()

tools_path("./temp")
```

Index