Package 'htmrenamer'

May 21, 2019

Type Package

Title File Renaming Tools for High Throughput Microscopy

Version 0.1.0

Author Hugo Botelho

Maintainer HugoBotelho <hugobotelho@gmail.com>

Description Enables the systematic renaming of Zeiss and Leica automated microscopes.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Depends gWidgets2tcltk,
reshape2,
tiff,
xlsx,
XML

Suggests testthat

R topics documented:

echo *Print to renamer GUI*

Description

Prints a message to the console and the renamer tool GUI and/or the R console. When printing to the console, just calls the print function.

Usage

```
echo(message, printToGUI = TRUE, printToConsole = TRUE)
```

Arguments

message	character, the message to be printed.
printToGUI	logical, indicating whether or not to print the message to the gtext widget on the renamer GUI.
printToConsole	logical, indicating whether or not to print the message to the console.

getfiles.XML.LRP *Find Leica MatrixScreener XML and LRP files*

Description

Analyzes the Leica MatrixScreener 'AdditionalData' folder and returns the file names of ScanningTemplate files

Usage

```
getfiles.XML.LRP(folder)
```

Arguments

folder	character, Leica MatrixScreener export folder.

Value

character vector with file names.

Note

There must be exactly one copy of these files:

- {ScanningTemplate}templatename.xml
- {ScanningTemplate}templatename.lrp

`newinfile.char` *Create empty infile (character vector)*

Description

Creates an empty microscope infile template, which can be printed to the console and/or saved as a file. The output is structured as a character vector. See the **Note** section for a description of the infile structure.

Usage

```
newinfile.char(numrow = 8, numcol = 12, show = FALSE,
  saveto = character())
```

Arguments

`numrow`	integer, the number of rows in the multiwell plate. The default value is 8 (i.e. 96 well plate).
`numcol`	integer, the number of columns in the multiwell plate. The default value is 12 (i.e. 96 well plate).
`show`	logical, indicating whether or not to print the result to the console.
`saveto`	character. If specified, the empty infile template is saved in a file at this location.

Value

Character vector with empty infile template.

Note

The microscope infile is a text file with the following structure:

```
001--A--01--00--00--data1--data2
002--A--02--01--00--data1--data2
003--A--03--02--00--data1--data2
```

In the infile, metadata fields are separated by a double dash (`--`). The meaning of the metadata fields is the following:

- Well number
- Row label (A, B, ...)
- Column label (01, 02, 03, ...)
- Column label (00, 01, 02, ...)
- Row label (00, 01, 02, ...)
- Experimental data 1
- Experimental data 2

Wells are numbered in ascending order, line by line. In the case of a 96 well plate, wells are numbered as shown:

```
-----1---2---3---4---5---6---7---8---9--10--11--12- ------------------------------------------
A | 001 002 003 004 005 006 007 008 009 010 011 012 B | 013 014 015 016 017 018 019 020 021 022 023 024
C | 025 026 027 028 029 030 031 032 033 034 035 036 D | 037 038 039 040 041 042 043 044 045 046 047 048
E | 049 050 051 052 053 054 055 056 057 058 059 060 F | 061 062 063 064 065 066 067 068 069 070 071 072
G | 073 074 075 076 077 078 079 080 081 082 083 084 H | 085 086 087 088 089 090 091 092 093 094 095 096
```

The experimental data fields may contain any relevant information describing the well contents.

See Also

Function `newinfile.df` creates an empty infile in the form of a data frame.
Function `read.infile.df` reads infile text files from disk.

Examples

```
# Infile template for a 96 well plate
newinfile.char()

# Save 384 well plate infile template to the working directory.
# The TXT file extension is recommended.
newinfile.char(numrow = 16, numcol = 24, saveto = "./infile_template.txt")
```

---

newinfile.df *Create empty infile (data frame)*

Description

Creates an empty microscope infile template, which can be printed to the console and/or saved as a file. The output is structured as a data frame. See the **Note** section for a description of the infile structure.

Usage

```
newinfile.df(numrow = 8, numcol = 12, show = FALSE,
  saveto = character())
```

Arguments

`numrow`	integer, the number of rows in the multiwell plate. The default value is 8 (i.e. 96 well plate).
`numcol`	integer, the number of columns in the multiwell plate. The default value is 12 (i.e. 96 well plate).
`show`	logical, indicating whether or not to print the result to the console.
`saveto`	character. If specified, the empty infile template is saved in a file at this location.

Value

Data frame with empty infile template.

Note

The microscope infile is a text file with the following structure:

```
001--A--01--00--00--data1--data2
002--A--02--01--00--data1--data2
003--A--03--02--00--data1--data2
```

In the infile, metadata fields are separated by a double dash (--). The meaning of the metadata fields is the following:

- Well number
- Row label (A, B, ...)
- Column label (01, 02, 03, ...)
- Column label (00, 01, 02, ...)
- Row label (00, 01, 02, ...)
- Experimental data 1
- Experimental data 2

Wells are numbered in ascending order, line by line. In the case of a 96 well plate, wells are numbered as shown:

```
-----1---2---3---4---5---6---7---8---9--10--11--12- ------------------------------------------
A | 001 002 003 004 005 006 007 008 009 010 011 012 B | 013 014 015 016 017 018 019 020 021 022 023 024
C | 025 026 027 028 029 030 031 032 033 034 035 036 D | 037 038 039 040 041 042 043 044 045 046 047 048
E | 049 050 051 052 053 054 055 056 057 058 059 060 F | 061 062 063 064 065 066 067 068 069 070 071 072
G | 073 074 075 076 077 078 079 080 081 082 083 084 H | 085 086 087 088 089 090 091 092 093 094 095 096
```

The experimental data fields may contain any relevant information describing the well contents.

See Also

Function `newinfile.char` creates an empty infile in the form of a character vector.
Function `read.infile.df` reads infile text files from disk.

Examples

```
# Infile template for a 96 well plate
newinfile.df()

# Save 384 well plate infile template to the working directory.
# The CSV file extension is recommended.
newinfile.df(numrow = 16, numcol = 24, saveto = "./infile_template.csv")
```

read.infile.df *Read microscope infiles*

Description

Imports a text file representing a microscope infile as a data frame. Skips all lines which do not match the infile structure. See newinfile.df or newinfile.char for a description of the infile structure.

Usage

```
read.infile.df(infilepath)
```

Arguments

infilepath	character, the path to the infile.

Value

Data frame with infile.

See Also

Function newinfile.df creates an empty infile in the form of a data frame.
Function newinfile.char creates an empty infile in the form of a character vector.

Examples

```
# Create infile in disk
tempfile <- tempfile()
newinfile.char(saveto = tempfile)

# Read infile
myinfile <- read.infile.df(tempfile)
head(myinfile)

# Delete infile file from disk
file.remove(tempfile)
```

rename_leica *Leica renamer*

Description

Renaming of Leica MatrixScreener files according to a microscope infile. See newinfile.df or newinfile.char for a description of the infile structure.

Usage

```
rename_leica(sourcefolder, targetfolder, infilepath, compress = FALSE,
  move = FALSE, outputDescriptors = TRUE, printToGUI = TRUE)
```

Arguments

sourcefolder	character, the folder with raw images. The folder usually has the folllowing name structure: demoplate_01--2019_01_01_12_00_00
targetfolder	character, folder where to place renamed files. A subfolder wih the same name as the infile will be created by the function and must not exist previously.
infilepath	character, location of the microscope infile
compress	logical, compress raw images?
move	logical, move (rather than copy) raw images?
outputDescriptors	logical, save plain text files with experimental metadata?
printToGUI	logical, print messages to the renamer GUI?

Value

Renamed files, metadata and log file

Null

See Also

rename_leica_gui for interacting with this function using a GUI.

---

rename_leica_gui *Leica renamer GUI*

Description

Displays the renamer tool GUI for Leica LAS X MatrixsSreener exports. The actual renaming process is performed by function rename_leica().

Usage

```
rename_leica_gui()
```

Value

Null

See Also

rename_leica is the function which implements the actual renaming algorithm.
rename_zeiss_gui for the Zeiss renamer GUI.

rename_zeiss *Zeiss renamer*

Description

Renaming of Zeiss TIF files exported with Zen Blue. Files areLeica MatrixScreener files according to a microscope infile. See newinfile.df or newinfile.char for a description of the infile structure.

Usage

```
rename_zeiss(sourcefolder, targetfolder, infilepath, numrow, numcol,
  REGEXvalidimages = "^.*?s.*?t.*?\\.tif$",
  REGEXscenenum = "^.*/.*_s(.*)t.*\\.tif$",
  REGEXtimenum = "^.*/.*_s.*t(.*)\\.tif$", printToGUI = FALSE,
  move = FALSE)
```

Arguments

sourcefolder	character, the folder with raw images. The folder usually has the folllowing name structure: "demoplate_01–2019_01_01_12_00_00"
targetfolder	character, folder where to place renamed files. A subfolder wih the same name as the infile will be created by the function and must not exist previously.
infilepath	character, location of the microscope infile
numrow	integer, number of rows in the multi well plate
numcol	integer, number of columns in the multi well plate
REGEXvalidimages	character, regular expression matching all files which should be renamed
REGEXscenenum	character, regular expression capturing ((\1)) the scene number (i.e. well number)
REGEXtimenum	character, regular expression capturing ((\1)) the time number in a time lapse
printToGUI	logical, print messages to the renamer GUI?
move	logical, move (rather than copy) raw images?

Value

Renamed files, metadata and log file

Null

See Also

rename_leica_gui is the function which implements the actual renaming algorithm.

rename_zeiss_gui *Zeiss renamer GUI*

Description

Displays the renamer tool GUI for Zeiss Zen Blue exports. The actual renaming process is performed by function rename_zeiss().

Usage

```
rename_zeiss_gui()
```

Value

Null

See Also

rename_zeiss is the function which implements the actual renaming algorithm.
rename_leica_gui for the Leica renamer GUI.

Index