Package 'gpsinterp'

January 25, 2018

Type Package

Title Interpolation of Longitude, Latitude and Direction with JOSM

Version 0.1.0

Depends R (>= 3.3)

Imports dplyr (>= 0.7), purrr (>= 0.2), tibble (>= 1.4), tidyr (>= 0.7), xml2 (>= 1.1)

Suggests testthat

Description Finds coordinates and direction (angle) of a sequence of photos with no GPS exif tags. In JOSM (Java editor for OpenStreetMap), manually identify points that can be located exactly, then interpolate in-between points with R. Repeat the procedure until the approximation is good enough.

URL http://github.com/the-knife/gpsinterp

BugReports http://github.com/the-knife/gpsinterp/issues

License GPL-2

Encoding UTF-8

LazyData true

SystemRequirements JOSM (required), exiftool (optional)

RoxygenNote 6.0.1

NeedsCompilation no

Author Pierre-Yves Berrard [aut, cre]

Maintainer Pierre-Yves Berrard <pierre.yves.berrard@gmail.com>

R topics documented:

interp_cleanup *Clean up after interpolation*

Description

Removes temporary environment (.exif) containing data that was used during interpolation.

Usage

```
interp_cleanup()
```

Details

As closing R session will also delete that environment, this is not mandatory.

interp_init *Initialize data before interpolation*

Description

List photos in input directory and creates an environment for storing data during interpolation iterative process. This is a mandatory prerequisite for interpolating positions.

Usage

```
interp_init(path = ".")
```

Arguments

path	(optional) path where input and output are located. This is not necessary if the working directory is already set to that location.

Details

Search for image files in `input/photo` and stores them in a data.frame.

As the process is iterative (a succession of manual modifications in JOSM and interpolations in R), it is necessary to keep track of intermediate results. Therefore, a temporary environment (named .exif) is created.

Once interpolation is over, it is possible to delete that environment with `interp_cleanup`.

Value

Nothing. Only `.exif$base` is created.

Examples

```
interp_init()
```

interp_josm *Interpolation and JOSM update*

Description

Reads exact positions, linearly interpolates non-exact positions, updates osm file and displays it in JOSM.

Usage

```
interp_josm(path = ".")
```

Arguments

path	(optional) path where input and output are located. This is not necessary if the working directory is already set to that location.

Details

Reads `approx.osm` and detects exact locations (moved manually in JOSM). Then interpotales linearly other points. Updates `approx.osm` and opens it in JOSM (http://localhost:8111 is used to open the file in JOSM, which must be opened beforehand).

Value

Nothing. Only `.exif$base` and `approx.osm` are updated.

Examples

```
# interp_josm()
```

write_coord_csv *Write coordinates and direction in a csv file.*

Description

Write computed longitudes, latitudes (and directions) in a csv file. Images are not modified.

Usage

```
write_coord_csv(path = ".", file = "interp_gps.csv", direction = TRUE,
  ...)
```

Arguments

path	(optional) path where input and output are located. This is not necessary if the working directory is already set to that location.
file	the name of the csv file (default : "interp_gps.csv")
direction	should direction to next photo be calculated and included in the file?
...	other arguments passed to `write.csv`

`write_exiftool` *Writing exif tags with exiftool*

Description

Runs exiftool commands to write computed longitudes, latitudes (and directions) in image files.

Usage

```
write_exiftool(path = ".", direction = TRUE)
```

Arguments

`path`	(optional) path where input and output are located. This is not necessary if the working directory is already set to that location.
`direction`	should direction to next photo be calculated and included in the file?

Details

In order to write image files, **exiftool** must be installed.

Files are not overwritten : a copy of the images including new exif tags are written in the `output` directory.

Index