

Package: regfusionr (via r-universe)

August 24, 2024

Type Package

Title Registration Fusion in R

Version 0.2.0

Maintainer Tim Schäfer <ts+code@rcmd.org>

Description Implementation of the 'Wu et al. 2018' registration fusion method in R.

License MIT + file LICENSE

Encoding UTF-8

URL <https://github.com/dfsp-spirit/regfusionr>

BugReports <https://github.com/dfsp-spirit/regfusionr/issues>

Imports freesurferformats (>= 0.1.17), gifti, oro.nifti, data.table, pracma

Suggests oce, knitr, rmarkdown, testthat (>= 2.1.0), haze (>= 0.2.0), fsbrain (>= 0.5.3)

Additional_repositories <https://dfsp-spirit.r-universe.dev>

VignetteBuilder knitr

RoxygenNote 7.1.2

Repository <https://dfsp-spirit.r-universe.dev>

RemoteUrl <https://github.com/dfsp-spirit/regfusionr>

RemoteRef HEAD

RemoteSha c1895b0cf25657d4f44c12c2250b46b6ead5aca

Contents

fsaverage_to_vol	2
fsaverage_vertices_to_colin27_coords	4
fsaverage_vertices_to_mni152_coords	5
fsaverage_vertices_to_vol_coords	6
linear_fsaverage_coords_to_MNI152_coords	7

mni152_coords_to_fsaverage	7
mni305_coords_to_colin27_coords	8
mni305_coords_to_mni152_coords	9
vol_coords_to_fsaverage	10
vol_to_fsaverage	11

Index	13
--------------	-----------

fsaverage_to_vol	<i>Project or map per-vertex values from the fsaverage surface to the cortex voxels of an MNI volume.</i>
------------------	---

Description

Applies the Wu et al. regfusion method to obtain MNI volume coordinates, then interpolates values.

Usage

```
fsaverage_to_vol(
    lh_input,
    rh_input,
    target_space = "FSL_MNI152",
    rf_type = "RF_ANTs",
    interp = "linear",
    out_type = "mgz",
    out_dir = NULL,
    fsaverage_path = NULL
)
```

Arguments

lh_input	numerical vector of per-vertex data for the left hemisphere of the template subject. Must contain 163842 values for the fsaverage template. Input for fsaverage6 (40962 values) or fsaverage5 (10242 values) can also be used and will be upsampled using nn_interpolate_kdtree . Automatic upsampling is only supported with <code>interp='linear'</code> .
rh_input	numerical vector of per-vertex data for the right hemisphere of the template subject. Must contain 163842 values for the fsaverage template. See description for lh_input for more details.
target_space	character string, the target template or the space that your output volume should be in. One of 'FSL_MNI152' or 'SPM_Colin27'.
rf_type	the regfusion type to use, one of 'RF_ANTs' or 'RF_M3Z'.
interp	interpolation method, currently only 'linear' and 'nearest' are supported. The performance of the 'linear' method is currently quite bad, and it will be rewritten in C++ when I find the time.

out_type	character string, the format of the output files. One of the following: 'mgz' or 'mgh' for FreeSurfer MGZ/MGH format, 'nii' for NIFTI v1 format. Ignored unless out_dir is not NULL.
out_dir	optional character string, the path to a writable output directory to which the output should be written as volume files. If NULL, no data is written to files. If out_dir is not NULL, the return value additionally contains the following keys: 'out_file' and the output file format at key 'out_format', (and 'out_file_seg'/'out_format_seg' for the respective seg versions).
fsaverage_path	character string or NULL, the file system path to the fsaverage directory (NOT including the 'fsaverage' dir itself). If NULL, defaults to the return value of fsbrain::fsaverage.path() on the system. This path is used to read the spherical surface (both hemisphere meshes) of the template subject.

Value

named list with keys 'projected' and 'projected_seg', each of which holds an fs.volume instance, its 'data' key holds a 256x256x256 array with the projected data. The data in 'projected_seg' is identical to the data in 'projected', with the exception that data values originating from the right hemisphere have been incremented by 1000. See out_dir parameter to easily write results to files. If out_dir is not NULL, the return value additionally contains the following keys: 'out_file' and the output file format at key 'out_format'.

Note

This function requires the packages 'fsbrain' and 'haze', which are optional dependencies. The package 'fsbrain' can be installed from CRAN. For 'haze', see <https://github.com/dfsp-spirit/haze>.

This function is quite expensive computationally, especially when using interp = 'linear'.

Author(s)

Tim Schäfer for the R version, Wu Jianxiao and CBIG for the original Matlab version.

Examples

```
## Not run:
lh_input = rnorm(163842L, 3.0, 0.2);
rh_input = rnorm(163842L, 3.0, 0.2);
res = fsaverage_to_vol(lh_input, rh_input, "FSL_MNI152");
```

```
## End(Not run)
```

fsaverage_vertices_to_colin27_coords

Map fsaverage vertex indices to Colin27 coordinates.

Description

Map fsaverage vertex indices to Colin27 coordinates.

Usage

```
fsaverage_vertices_to_colin27_coords(
  vertices,
  hemis,
  fs_home = Sys.getenv("FS_HOME"),
  simplify = FALSE
)
```

Arguments

vertices	integer vector of vertex indices (1-based), the n fsaverage vertices you want to map.
hemis	vector of character strings, the hemispheres of the query vertices. Each entry in the vector has to be either 'lh' or 'rh'. Length must match length of parameter vertices.
fs_home	character string, path to the FreeSurfer installation. Alternatively, a hemilist of freesurferformats::fs.surface instances like surface = list("lh"=mysurflh, "rh"=mysurfrh). Used to find the surfaces, at <fs_home>/subjects/fsaverage/surf/<hemi>.<surface> where hemi is 'lh' and 'rh'. Can be NULL if 'surface' is a hemilist of fs.surface instances.
simplify	logical, whether to return a vector instead of a single-row matrix in case only a single query vertex is given.

Value

matrix of dim n x 3, the MNI152 coordinates for the query vertices, one row per vertex. Also see the 'simplify' parameter.

See Also

Use the more general function [fsaverage_vertices_to_vol_coords](#) for more options.

`fsaverage_vertices_to_mni152_coords`*Map fsaverage vertex indices to MNI152 coordinates.*

Description

Map fsaverage vertex indices to MNI152 coordinates.

Usage

```
fsaverage_vertices_to_mni152_coords(  
  vertices,  
  hemis,  
  fs_home = Sys.getenv("FS_HOME"),  
  simplify = FALSE  
)
```

Arguments

<code>vertices</code>	integer vector of vertex indices (1-based), the <code>n</code> fsaverage vertices you want to map.
<code>hemis</code>	vector of character strings, the hemispheres of the query vertices. Each entry in the vector has to be either 'lh' or 'rh'. Length must match length of parameter <code>vertices</code> .
<code>fs_home</code>	character string, path to the FreeSurfer installation. Alternatively, a hemilist of <code>freesurferformats::fs.surface</code> instances like <code>surface = list("lh"=mysurflh, "rh"=mysurfrh)</code> . Used to find the surfaces, at <code><fs_home>/subjects/fsaverage/surf/<hemi>.<surface></code> where <code>hemi</code> is 'lh' and 'rh'. Can be NULL if 'surface' is a hemilist of <code>fs.surface</code> instances.
<code>simplify</code>	logical, whether to return a vector instead of a single-row matrix in case only a single query vertex is given.

Value

matrix of dim `n` x 3, the MNI152 coordinates for the query vertices, one row per vertex. Also see the 'simplify' parameter.

See Also

Use the more general function [fsaverage_vertices_to_vol_coords](#) for more options.

 fsaverage_vertices_to_vol_coords

Map fsaverage vertex indices to MNI152 or Colin27 volume coordinates.

Description

Map fsaverage vertex indices to MNI152 or Colin27 volume coordinates.

Usage

```
fsaverage_vertices_to_vol_coords(
  vertices,
  hemis,
  fs_home = Sys.getenv("FS_HOME"),
  simplify = FALSE,
  rf_type = "RF_ANTs",
  template_type = "MNI152_orig"
)
```

Arguments

vertices	integer vector of vertex indices (1-based), the n fsaverage vertices you want to map.
hemis	vector of character strings, the hemispheres of the query vertices. Each entry in the vector has to be either 'lh' or 'rh'. Length must match length of parameter vertices.
fs_home	character string, path to the FreeSurfer installation. Alternatively, a hemilist of freesurferformats: : fs.surface instances like surface = list("lh"=mysurflh, "rh"=mysurfrh). Used to find the surfaces, at <fs_home>/subjects/fsaverage/surf/<hemi>.<surface> where hemi is 'lh' and 'rh'. Can be NULL if 'surface' is a hemilist of fs.surface instances.
simplify	logical, whether to return a vector instead of a single-row matrix in case only a single query vertex is given.
rf_type	the registration fusion type, one of 'RF_ANTs' or 'RF_M3Z'.
template_type	the space into which to map. One of 'MNI152_orig', 'MNI152_norm', 'Colin27_orig', 'Colin27_norm'. Note that the 'RF_ANTs' rf_type must be used for _orig templates, and the 'RF_M3Z' type for _norm templates.

Value

matrix of dim n x 3, the MNI152 or Colin27 coordinates for the query vertices, one row per vertex. Also see the 'simplify' parameter.

`linear_fsaverage_coords_to_MNI152_coords`

Transform MNI305 coords (FreeSurfer fsaverage surface) to MNI152 coordinates using the linear method.

Description

This uses the 4x4 matrix from the FreeSurfer Coordinate Systems documentation.

Usage

```
linear_fsaverage_coords_to_MNI152_coords(vertex_coords)
```

Arguments

`vertex_coords` nx3 matrix of coordinates, e.g., typically from fsaverage surface vertices.

Value

nx3 numerical matrix if MNI152 coords.

Note

This is the opposite of using the *Wu et al.* approach: a linear transformation matrix is used. This approach is mainly implemented in this package to allow you to easily check the difference between the methods.

`mni152_coords_to_fsaverage`

Map MNI152 coords to fsaverage coords and vertices.

Description

Map MNI152 coords to fsaverage coords and vertices.

Usage

```
mni152_coords_to_fsaverage(  
  coords,  
  surface = "white",  
  fs_home = Sys.getenv("FS_HOME"),  
  silent = TRUE  
)
```

Arguments

coords	nx3 numeric matrix, the source RAS coordinates in the input image which must be in MNI152 space. The coords must be within the cortex, otherwise the mapping makes no sense and NaN values are returned for the respective coords.
surface	character string, the fsaverage surface (brain mesh) to load. Must be a valid FreeSurfer surface name like 'white', 'pial', 'orig', 'inflated'.
fs_home	character string, path to the FreeSurfer installation. Alternatively, a hemilist of freesurfer formats: fs.surface instances like surface = list("lh"=mysurflh, "rh"=mysurfrh). Used to find the surfaces, at <fs_home>/subjects/fsaverage/surf/<hemi>.<surface> where hemi is 'lh' and 'rh'. Can be NULL if 'surface' is a hemilist of fs.surface instances.
silent	logical, whether to suppress output messages in case of coords outside of cortex.

Value

named list with entries 'fsaverage_vertices': integer vector of fsaverage surface vertex indices, 'hemi': vector of hemi strings for the vertices, 'fsaverage_coords': nx3 numeric matrix of target coordinates, 'query_mni_coords': copy of input parameter coords, 'query_mni_voxels': the voxel indices at the query RAS coords.

Note

See the more general function [vol_coords_to_fsaverage](#) for more options.

Author(s)

Tim Schäfer for the R version, Wu Jianxiao and CBIG for the original Matlab version.

Examples

```
## Not run:
mni_ras = c(60.0, 0.0, 10.0)
res = mni152_coords_to_fsaverage(mni_ras, surface = "white");
res$fsaverage_vertices; # 9092

## End(Not run)
```

mni305_coords_to_colin27_coords

Find Colin27 coordinate of fsaverage vertex closest to the given MNI305 coordinate.

Description

Find Colin27 coordinate of fsaverage vertex closest to the given MNI305 coordinate.

Usage

```
mni305_coords_to_colin27_coords(
    coords,
    surface = "orig",
    fs_home = Sys.getenv("FS_HOME"),
    simplify = FALSE
)
```

Arguments

coords	nx3 numerical matrix, the MNI305 query coordinates.
surface	a character string defining the fsaverage surface to load (like "white" or "orig"), or a pre-loaded hemilist of surfaces (i.e., freesurferformats::fs.surface instances)
fs_home	character string, path of the FreeSurfer directory from which the fsaverage surfaces should be loaded. Ignored if surface is a hemilist (in that case the surfaces have already been loaded).
simplify	logical, whether to return a vector instead of a single-row matrix in case only a single query vertex is given.

Value

nx3 numerical matrix, the Colin27 coordinates for the vertices closest to the given MNI305 query coordinates. Depending on the distance to the closest vertex, this may be way off. Also see the 'simplify' parameter.

```
mni305_coords_to_mni152_coords
```

Find MNI152 coordinate of fsaverage vertex closest to the given MNI305 coordinate.

Description

Find MNI152 coordinate of fsaverage vertex closest to the given MNI305 coordinate.

Usage

```
mni305_coords_to_mni152_coords(
    coords,
    surface = "orig",
    fs_home = Sys.getenv("FS_HOME"),
    simplify = FALSE
)
```

Arguments

coords	nx3 numerical matrix, the MNI305 query coordinates.
surface	a character string defining the fsaverage surface to load (like "white" or "orig"), or a pre-loaded hemilist of surfaces (i.e., freesurferformats::fs.surface instances)
fs_home	character string, path of the FreeSurfer directory from which the fsaverage surfaces should be loaded. Ignored if surface is a hemilist (in that case the surfaces have already been loaded).
simplify	logical, whether to return a vector instead of a single-row matrix in case only a single query vertex is given.

Value

nx3 numerical matrix, the MNI152 coordinates for the vertices closest to the given MNI305 query coordinates. Depending on the distance to the closest vertex, this may be way off. Also see the 'simplify' parameter.

vol_coords_to_fsaverage

Map MNI152 or Colin27 volume coords to fsaverage coords and vertices.

Description

Map MNI152 or Colin27 volume coords to fsaverage coords and vertices.

Usage

```
vol_coords_to_fsaverage(
  coords,
  surface = "white",
  fs_home = Sys.getenv("FS_HOME"),
  silent = TRUE,
  rf_type = "RF_ANTs",
  template_type = "FSL_MNI152"
)
```

Arguments

coords	nx3 numeric matrix, the source RAS coordinates in the input image which must be in MNI152/Colin27 space. The coords must be within the cortex, otherwise the mapping makes no sense and NaN values are returned for the respective coords.
surface	character string, the fsaverage surface (brain mesh) to load. Must be a valid FreeSurfer surface name like 'white', 'pial', 'orig', 'inflated'.

`fs_home` character string, path to the FreeSurfer installation. Alternatively, a hemilist of freesurferformats: : fs.surface instances like `surface = list("lh"=mysurflh, "rh"=mysurfrh)`. Used to find the surfaces, at `<fs_home>/subjects/fsaverage/surf/<hemi>.<surface>` where hemi is 'lh' and 'rh'. Can be NULL if 'surface' is a hemilist of fs.surface instances.

`silent` logical, whether to suppress output messages in case of coords outside of cortex.

`rf_type` the regfusion type to use, one of 'RF_ANTs' or 'RF_M3Z'.

`template_type` character string, the source template or the space that your input image is in. One of 'MNI152_orig', 'Colin27_orig', 'MNI152_norm', 'Colin27_norm'.

Value

named list with entries 'fsaverage_vertices': integer vector of fsaverage surface vertex indices, 'hemi': vector of hemi strings for the vertices, 'fsaverage_coords': nx3 numeric matrix of target coordinates, 'query_mni_coords': copy of input parameter coords, 'query_mni_voxels': the voxel indices at the query RAS coords.

Author(s)

Tim Schäfer for the R version, Wu Jianxiao and CBIG for the original Matlab version.

Examples

```
## Not run:
mni_ras = c(60.0, 0.0, 10.0)
res = vol_coords_to_fsaverage(mni_ras, surface = "white");
res$fsaverage_vertices; # 9092

## End(Not run)
```

`vol_to_fsaverage` *Project or map values from MNI volume to fsaverage surface.*

Description

Applies the Wu et al. regfusion method to obtain surface coords, then interpolates values.

Usage

```
vol_to_fsaverage(
  input_img,
  template_type,
  rf_type = "RF_ANTs",
  interp = "linear",
  out_type = "curv",
  out_dir = "."
)
```

Arguments

input_img	3D or 4D NIFTI or MGZ image instance of type <code>fs.volume</code> , or a character string that will be interpreted as a file system path to such a volume that should be loaded with <code>freesurferformats::read.fs.volume</code> . If 4D, the 4th dimension is considered the time/subject dimension.
template_type	character string, the source template or the space that your input image is in. One of 'MNI152_orig', 'Colin27_orig', 'MNI152_norm', 'Colin27_norm'.
rf_type	the regfusion type to use, one of 'RF_ANTs' or 'RF_M3Z'.
interp	interpolation method, currently only 'linear' is supported.
out_type	character string, the format of the output files. One of the following: 'curv' for FreeSurfer curv format, 'mgz' for FreeSurfer MGZ format, 'gii' for GIFTI format.
out_dir	character string, the path to a writable output directory. If NULL, the returned named list contains the projected data (instead of the path of the file it was written to) at the keys 'lh' and 'rh', and the parameter 'out_type' is ignored.

Value

named list of 2 character strings, the output files (for the 2 hemispheres) at keys 'lh' and 'rh'. See the documentation for parameter 'out_dir' if you want the data in R instead.

Author(s)

Tim Schäfer for the R version, Wu Jianxiao and CBIG for the original Matlab version.

Index

fsaverage_to_vol, [2](#)
fsaverage_vertices_to_colin27_coords,
[4](#)
fsaverage_vertices_to_mni152_coords, [5](#)
fsaverage_vertices_to_vol_coords, [4](#), [5](#),
[6](#)

linear_fsaverage_coords_to_MNI152_coords,
[7](#)

mni152_coords_to_fsaverage, [7](#)
mni305_coords_to_colin27_coords, [8](#)
mni305_coords_to_mni152_coords, [9](#)

nn_interpolate_kdtree, [2](#)

vol_coords_to_fsaverage, [8](#), [10](#)
vol_to_fsaverage, [11](#)