Package 'vesselr'

April 3, 2017

Type Package

Title Gradient and Vesselness Tools for Arrays and NIfTI Images

Version 0.2.1

Maintainer Jordan D. Dworkin <jdwor@mail.med.upenn.edu>

Description Simple functions for calculating the image gradient, image hessian, volume ratio filter, and Frangi vesselness filter of 3-dimensional volumes.

Imports oro.nifti, parallel, pbmcapply, pbapply

License GPL-2

Encoding UTF-8

LazyData true

URL https://github.com/jdwor/vesselr

RoxygenNote 6.0.1

NeedsCompilation no

Author Jordan D. Dworkin [aut, cre]

Repository CRAN

Date/Publication 2017-04-03 20:40:54 UTC

R topics documented:

blobness3D 2
gradient3D 3
hessian3D 3
vesselness3D 4

Index **6**

blobness3D *3D Volume Blobness*

Description

This function returns a blobness map for a 3D array or NIfTI volume. This blobness measure is based on the volume ratio described by Pierpaoli and Basser (1996).

Usage

```
blobness3D(image, mask, radius = 5, color = "dark", parallel = FALSE,
  cores = 2)
```

Arguments

image	a 3D array or image of class nifti
mask	an array or nifti mask of voxels for which vesselness will be calculated, with more selective masking improving speed significantly. Note that mask should be in the same space as the image volume
radius	an integer specifying radius of the neighborhood (in voxels) for which the blobness should be calculated. Note that this value essentially serves as the scale of the blob objects
color	a string specifying whether blobs will appear darker ("dark") or brighter ("bright") than their surroundings
parallel	is a logical value that indicates whether the user's computer is Linux or Unix (i.e. macOS), and should run the code in parallel
cores	if parallel = TRUE, cores is an integer value that indicates how many cores the function should be run on

Value

A 3D volume of the volume ratio blobness scores.

References

C. Pierpaoli, P.J. Basser (1996). Toward a Quantitative Assessment of Diffusion Anisotropy. Magnetic Resonance in Medicine. 36, pp. 893-906.

Examples

```
## Not run:
library(neurobase)
flair <- readnii('path/to/epi')
mask <- flair!=0
brightspots <- blobness3D(image = flair, mask = mask, radius = 5,
                      color = "bright", parallel = TRUE, cores = 4)
## End(Not run)
```

gradient3D *3D Volume Gradient*

Description

This function returns the gradient images for a 3D array or NIfTI volume.

Usage

```
gradient3D(image, mask = NULL, which = "all", radius = 1)
```

Arguments

image	a 3D array or image of class nifti
mask	an array or nifti mask of voxels for which the gradient will be calculated, if NULL the gradient will be run for the full array. Note that mask should be in the same space as the image volume
which	a string specifying the gradient direction that should be returned; either "all" for a list of x, y, and z gradient volumes, or "x", "y", or "z" for a single volume with the given gradient
radius	an integer specifying radius of the neighborhood (in voxels) for which the gradient should be calculated

Value

Either a list of three gradient volumes or a single gradient volume, in either array or NIfTI format based on what was input.

Examples

```
## Not run:
library(neurobase)
epi <- readnii('path/to/epi')
gradients <- gradient3D(image = epi, which = "all")
## End(Not run)
```

hessian3D *3D Volume Hessian*

Description

This function returns the eigenvalues of the hessian matrices for a 3D array or NIfTI volume.

Usage

```
hessian3D(image, mask, radius = 1, parallel = FALSE, cores = 2)
```

Arguments

image	a 3D array or image of class nifti
mask	an array or nifti mask of voxels for which vesselness will be calculated, with more selective masking improving speed significantly. Note that mask should be in the same space as the image volume
radius	an integer specifying radius of the neighborhood (in voxels) for which the hessian should be calculated
parallel	is a logical value that indicates whether the user's computer is Linux or Unix (i.e. macOS), and should run the code in parallel
cores	if parallel = TRUE, cores is an integer value that indicates how many cores the function should be run on

Value

A list of three eigenvalue volumes.

Examples

```
## Not run:
library(neurobase)
epi <- readnii('path/to/epi')
mask <- epi!=0
hesseigs <- hessian3D(image = epi, mask = mask)
## End(Not run)
```

---

vesselness3D *3D Volume Vesselness*

Description

This function returns a vesselness map for a 3D array or NIfTI volume. This vesseless measure is based on the method described by Frangi (1998).

Usage

```
vesselness3D(image, mask, radius = 1, color = "dark", parallel = FALSE,
  cores = 2)
```

Arguments

image	a 3D array or image of class nifti
mask	an array or nifti mask of voxels for which vesselness will be calculated, with more selective masking improving speed significantly. Note that mask should be in the same space as the image volume

radius	an integer specifying radius of the neighborhood (in voxels) for which the vesselness should be calculated. Note that this value essentially serves as the scale of the vessel objects
color	a string specifying whether vessels will appear darker ("dark") or brighter ("bright") than their surroundings
parallel	is a logical value that indicates whether the user's computer is Linux or Unix (i.e. macOS), and should run the code in parallel
cores	if parallel = TRUE, cores is an integer value that indicates how many cores the function should be run on

Value

A 3D volume of the Frangi vesselness scores.

References

A.F. Frangi, W.J. Niessen, K.L. Vincken, M.A. Viergever (1998). Multiscale vessel enhancement filtering. In Medical Image Computing and Computer-Assisted Intervention - MICCAI'98, W.M. Wells, A. Colchester and S.L. Delp (Eds.), Lecture Notes in Computer Science, vol. 1496 - Springer Verlag, Berlin, Germany, pp. 130-137.

Examples

```
## Not run:
library(neurobase)
epi <- readnii('path/to/epi')
mask <- epi!=0
veins <- vesselness3D(image = epi, mask = mask, radius = 1,
                      color = "dark", parallel = TRUE, cores = 4)
## End(Not run)
```

Index

blobness3D, 2

gradient3D, 3

hessian3D, 3

nifti, *2–4*

vesselness3D, 4