Package 'gdi'

May 5, 2023

Title Volumetric Analysis using Graphic Double Integration

Version 1.2.0

Date 2023-05-04

Author Darius Nau [aut, cre]

Maintainer Darius Nau <dariusnau@gmx.at>

Description Tools implementing an automated version of the graphic double integration technique first employed by Jerison (1973) <ISBN:9780323141086> and Hurlburt (1999) <doi:10.1080/02724634.1999.10011145>. Graphic double integration is primarily used for volume or mass estimation of (extinct) animals, and the package 'gdi' aims to make this technique as convenient and versatile as possible. The main functions of 'gdi' provide utilities for automatically measuring diameters from digital silhouettes provided as image files, and for calculating volume via graphic double integration with a simple elliptical, superelliptical (following Motani 2001 <doi:10.1666/0094-8373(2001)027%3C0735:EBMFST%3E2.0.CO;2>) or complex cross-sectional model.

Depends jpeg, png

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

R topics documented:

| cscorr | *Measure and analyze cross-sectional geometry* |

Description

Measure and analyze cross-sectional geometry

Usage

```
cscorr(
  image_file,
  threshold = 0.5,
  channel = 4,
  method = "greater",
  return = "area_corr",
  k = 2,
  scale = 1
)
```

Arguments

`image_file`	Image to be read. Images can be jpeg or png files, or a previously read image saved as an object in R.
`threshold`	Reference value for colour criterium after which pixels that are part of the cross-section are differentiated from the background.
`channel`	Colour channel to which to apply the threshold criterium. Default is 4 (alpha channel of rgba image). Channel setting needs to be adjusted depending on the colour mode of the image used (e.g. there are two channels to choose from in a greyscale image, and 3 in an rgb image).
`method`	Method for determining which pixels to count. Default "greater" counts pixels with value greater than threshold (e.g. higher opacity, in the case of an alpha channel). "less" counts pixels with a value less than the threshold. "not" counts all pixels not precisely matching threshold. Any other character string results in only pixels exactly matching the value given as threshold being counted.
`return`	What value to return. Possible values are "area_corr" (Default, returns ratio between measured area and area of ellipse with same horizontal and vertical diameters), "aspect_ratio" (returns aspect ratio), "diameters" (returns diameters) and "area" (returns area). Any other value for this parameter will prompt the function to return a vector containing all of these.
`k`	optional superellipse exponent for the (super)ellipse to which the measurements should be compared (for the "area_corr" setting for the parameter return).
`scale`	Optional scale of the image (for raw area measurements).

Value

Either a numeric of length 1 (depending on the input of the return parameter), defaulting to the area correction factor (if return=="area_corr"), or (if return is left empty or does not match any of the predefined settings) a numeric vector of length 5 containing all the possible outputs (x and y diameters, aspect ratio, area and area correction factor).

Examples

```
fdir <- system.file(package="gdi")
correction_factor <- cscorr(file.path(fdir,"exdata","cross_section.png"))
```

---

fdetect	*Tool to help determine which threshold value and method to use with measuresil() or cscorr(). The function analyzes all pixels along the edges of the image to determine the background colour, to help with deciding on appropriate settings and avoid errors introduced by inappropriate settings*

Description

Tool to help determine which threshold value and method to use with measuresil() or cscorr(). The function analyzes all pixels along the edges of the image to determine the background colour, to help with deciding on appropriate settings and avoid errors introduced by inappropriate settings

Usage

```
fdetect(image_file, threshold = 0.5, channel = 4, plot = FALSE)
```

Arguments

image_file	Image to be read. Images can be jpeg or png files, or a previously read image saved as an object in R.
threshold	Reference value for colour criterium after which pixels that are part of the silhouette should be differentiated from the background.
channel	Colour channel to which to apply the threshold criterium. Default is 4 (alpha channel of rgba image). Channel setting needs to be adjusted depending on the colour mode of the image used (e.g. there are two channels to choose from in a greyscale image, and 3 in an rgb image).
plot	Whether to plot a histogram with the detected colour values (if TRUE) or not (if FALSE, default).

Value

A list()-object containing: $edgetable (a table of the different colour values detected and their respective frequencies), $histogram (a histogram-object of the colour values), $most_common (the most common colour value found), $foreground (a character string, indicating whether the foreground colour value is likely "greater" or "less" than the specified threshold), $result (a character string giving a summary of the results)

Examples

```
fdir <- system.file(package="gdi")
fdetect(file.path(fdir,"exdata","lat.png"))
```

---

gdi	*Estimate volume using Graphic Double Integration.*

Description

Estimate volume using Graphic Double Integration.

Usage

```
gdi(
  lat,
  dors,
  scale = 10,
  sliceL = 1/scale,
  method = "raw",
  k = 2,
  corr = 1,
  smooth.ends = FALSE,
  return = "total"
)
```

Arguments

lat	Measurements of diameter in lateral view/first of two orthogonal views to be used with the gdi. Can be either a numeric vector, or a text file to be scanned. Defaults to "lat.txt".
dors	Measurements of diameter in dorsal view/second of two orthogonal views to be used with the gdi. Can be either a numeric vector, or a text file to be scanned. Must be the same length as lat. Defaults to "dors.txt".
scale	Scale of the data in terms of how many units of the input data are in one side of the desired unit of output volume. Defaults to 10.
sliceL	Length of individual segments to be used in the GDI. Defaults to 1/scale.
method	Method to be used for the GDI. Default "raw" setting calculates each segment as an elliptical cylinder with volume = Area * SliceL. Any other string will result in volume being calculated as an elliptical frustum with base areas based on the measurements of segments i and i+1.
k	Superellipse exponent to be used for the cross-sectional area. Defaults to 2.0 (normal ellipse).

`corr`	Correction factor for area of cross-sections, calculated as the ratio between the actual cross-sectional area and that of a (super)ellipse (depending on the specified exponent k) with the same diameters. This setting enables the function to account for complex, non-elliptical cross-sections. Default value is 1, i.e. no correction. Can be either a single number, or a numeric vector of the same length as lat and dors (in the case of a changing cross-sectional geometry along the length of the body).
`smooth.ends`	If method != "raw", specify whether first and last segments should be left raw, or taper to 0 (i.e. be approximated as cones). Only applies if there are no leading or following zeros in the measurement vectors.
`return`	Determines whether to report the estimated total volume (if default/"total"), or a data.frame with segment radii, areas and volumes (if left empty of any other character string.

Value

Either a single number representing the total volume estimated, or (if return!="total") a data.frame() containing columns with the radii in both dimensions, the estimated elliptical or superelliptical areas, and the segment volumes.

Examples

```
lateral <- rep(2,4) #generate example data
dorsal <- rep(2,4)
gdi(lat=lateral, dors=dorsal, scale=10, method="raw", k=2.0)
gdi(lat=lateral, dors=lateral/2, scale=10, method="smooth", k=2.3)
```

---

imghist	*Simple histogram analysis for all colour values in an input image. Can be used to help assess whether a chosen threshold value is appropriate for differentiating the silhouette from the background, or for general.*

Description

Simple histogram analysis for all colour values in an input image. Can be used to help assess whether a chosen threshold value is appropriate for differentiating the silhouette from the background, or for general.

Usage

```
imghist(
  image_file,
  threshold = 0.5,
  channel = 4,
  breaks = seq(0, 1, 0.05),
  plot = TRUE,
  unique = FALSE
)
```

Arguments

`image_file`	Image to be read. Images can be jpeg or png files, or a previously read image saved as an object in R.
`threshold`	Reference value for colour criterium after which pixels that are part of the silhouette should be differentiated from the background.
`channel`	Colour channel to which to apply the threshold criterium. Default is 4 (alpha channel of rgba image). Channel setting needs to be adjusted depending on the colour mode of the image used (e.g. there are two channels to choose from in a greyscale image, and 3 in an rgb image).
`breaks`	A vector of breaks for the histogram, defaults to a bin width of 0.05 between colour values of 0 and 1.
`plot`	Whether to plot a histogram, defaults to TRUE
`unique`	Whether to return counts for unique colour values, defaults to FALSE.

Value

A plotted histogram (unless plot==FALSE), and a matrix containing the counts from the histogram (default) or the counts for unique colour values (if unique==TRUE).

Examples

```
fdir <- system.file(package="gdi")
imghist(file.path(fdir,"exdata","lat.png"))
```

---

measuresil	*Take pixel-by-pixel measurements of a silhouette in jpeg or png format for use with the gdi function.*

Description

Take pixel-by-pixel measurements of a silhouette in jpeg or png format for use with the gdi function.

Usage

```
measuresil(image_file, threshold = 0.5, channel = 4, method = "greater")
```

Arguments

`image_file`	Image to be read. Images can be jpeg or png files, or a previously read image saved as an object in R.
`threshold`	Reference value for colour criterium after which pixels that are part of the silhouette are differentiated from the background.
`channel`	Colour channel to which to apply the threshold criterium. Default is 4 (alpha channel of rgba image). Channel setting needs to be adjusted depending on the colour mode of the image used (e.g. there are two channels to choose from in a greyscale image, and 3 in an rgb image).

method   Method for determining which pixels to count. Default "greater" counts pixels with value greater than threshold (e.g. higher opacity, in the case of an alpha channel). "less" counts pixels with a value less than the threshold. "not" counts all pixels not precisely matching threshold. Any other character string results in only pixels exactly matching the value given as threshold being counted.

Value

A numeric vector giving the measurements of the silhouette

Examples

```
fdir <- system.file(package="gdi")
lat <- measuresil(file.path(fdir,"exdata","lat.png"))
```

---

sellipse   *Estimate area of a superellipse. Assistant function for gdi.*

Description

Estimate area of a superellipse. Assistant function for gdi.

Usage

```
sellipse(a, b, k)
```

Arguments

a   First radius of the superellipse.

b   Second radius of the superellipse.

k   superellipse exponent.

Value

A single number giving the area of the superellipse

Examples

```
major_radius<-2
minor_radius<-3
exponent<-2.3
sellipse(major_radius, minor_radius, exponent)
```

Index