Package ‘hemispheR’

December 4, 2023

Title Processing Hemispherical Canopy Images

Version 1.1.2

Description Import and classify canopy fish-eye images, estimate angular gap fraction and derive canopy attributes like leaf area index and openness. Additional information is provided in the study by Chianucci F., Macek M. (2023) <doi:10.1016/j.agrformet.2023.109470>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

Imports autothresholdr, dplyr, graphics, grDevices, jpeg, dismo, scales, sf, tidyr

Depends R (>= 4.1), terra (>= 1.7-55)

LazyData true

Language en-US

NeedsCompilation no

Author Francesco Chianucci [aut, cre] (<https://orcid.org/0000-0002-5688-2060>), Martin Macek [aut] (<https://orcid.org/0000-0002-5609-5921>)

Maintainer Francesco Chianucci <fchianucci@gmail.com>

Repository CRAN

Date/Publication 2023-12-04 22:30:02 UTC

R topics documented:

binarize_fisheye — *Compute the threshold of a single-channel fisheye image, and return a binary fisheye image of canopy (0) and gap (1) pixels*

Description

The function calculates a single threshold of a single-channel raster image using the autothresholdr::auto_thresh() functionality. The single thresholding is also applied at sub-image level if zonal=TRUE. The available methods are described at https://imagej.net/plugins/auto-threshold. The thresholding value is then used to make a binary raster image of canopy (0) and gap (1) pixels.

Usage

```
binarize_fisheye(
  img,
  method = "Otsu",
  zonal = FALSE,
  manual = NULL,
  display = FALSE,
  export = FALSE
)
```

Arguments

img	SpatRaster. A single layer fisheye image imported by import_fisheye() using the terra::rast() functionality.
method	Character. The method used to threshold the image, using the autothresholdr::auto_thresh() function. For details, see https://imagej.net/plugins/auto-threshold. Default = 'Otsu'.
zonal	Logical. If is set to TRUE, it divides the images in four (N, W, S, E) regions and classify each region separately. Useful in case of uneven illumination condition in the image.
manual	Numeric. It uses a manual thresholding instead of automatic one. If selected, it overrides automatic thresholding.
display	Logical. If is set to TRUE, it plots the classified binary image. Default to FALSE.
export	Logical. If is set to TRUE, it saves the binary fisheye image as tif file. Default to FALSE.

Value

A binary single-layer image (SpatRaster)

See Also

https://imagej.net/plugins/auto-threshold

Examples

```
c.im<-system.file('extdata/circular_coolpix4500+FC-E8_chestnut.jpg',package='hemispheR')

c.im |>
import_fisheye(circ.mask=camera_fisheye('Coolpix4500+FC-E8')) |>
 binarize_fisheye(display=TRUE)

#zonal thresholding:
c.im |>
import_fisheye(circ.mask=camera_fisheye('Coolpix4500+FC-E8')) |>
 binarize_fisheye(zonal=TRUE,display=TRUE)

#manual thresholding:
c.im |>
import_fisheye(circ.mask=camera_fisheye('Coolpix4500+FC-E8')) |>
binarize_fisheye(manual=55,display=TRUE)
```

`camera_fisheye` *Provide circular mask parameters from known camera+fisheye lens models*

Description

Provide circular mask parameters from known camera+fisheye lens models

Usage

```
camera_fisheye(model = NULL)
```

Arguments

`model`	Character. An input camera+lens model

Value

A list of three parameters (xc, yc, rc) of the circular mask

Examples

```
#available camera+lenses:
list.cameras

camera_fisheye(model='Coolpix4500+FC-E8')
```

canopy_fisheye — *Calculate canopy attributes from angular gap fraction data derived from fisheye images*

Description

The function calculate canopy attributes from angular distribution of gap fraction. It returns both the effective (Le) and actual (L) leaf area index following the Miller theorem (1967). The Lang and Xiang (1986) clumping index LX is calculated as the ratio of Le to L; two additional clumping indices (LXG1, LXG2) are derived from ordered weighted average gap fraction as in Chianucci et al. (2019). The mean leaf angle (MTA) and the ellipsoidal x are derived from Norman and Campbell (1989). Canopy openness is also provided as weighted diffuse non-interceptance (DIFN), following the LAI-2200 manual (Li-Cor Inc., Nebraska US).

Usage

```
canopy_fisheye(rdfw)
```

Arguments

`rdfw`	Dataframe. The input dataframe generated from `gapfrac_fisheye()`, which contains gap fraction for zenith and azimuth bins.

Value

A dataframe of canopy attributes from classified fisheye images.

See Also

Chianucci F., Zou J., Leng P., Zhuang Y., Ferrara C. 2019. A new method to estimate clumping index integrating gap fraction averaging with the analysis of gap size distribution. Canadian Journal of Forest Research 49 doi:10.1139/cjfr20180213

LAI-2200C Plant Canopy Analyzer - Instruction Manuals. Licor.

Lang A.R.G., Xiang Y. 1986. Estimation of leaf area index from transmission of direct sunlight in discontinuous canopies. Agricultural and Forest Meteorology 37, 228-243. doi:10.1016/0168-1923(86)90033X

Miller J.B. 1967. A formula for average foliage density. Australian Journal of Botany 15(1) 141 - 144. doi:10.1071/BT9670141 .

Norman J.M., Campbell G.S. 1986. Canopy structure. In: Plant Physiological Ecology, pp. 301-325 doi:10.1007/9789400922211_14.

Examples

```
c.im<-system.file('extdata/circular_coolpix4500+FC-E8_chestnut.jpg',package='hemispheR')
c.im |>
  import_fisheye(circ.mask=camera_fisheye('Coolpix4500+FC-E8')) |>
  binarize_fisheye() |>
  gapfrac_fisheye(lens='FC-E8',nrings=7,nseg=8,endVZA=70) |>
  canopy_fisheye()

#Zenith rings similar to LAI-2000/2200:
c.im |>
 import_fisheye(circ.mask=camera_fisheye('Coolpix4500+FC-E8')) |>
  binarize_fisheye() |>
  gapfrac_fisheye(lens='FC-E8',nrings=5,nseg=8,endVZA=75) |>
  canopy_fisheye()

#The hinge angle method close to 1 radian (57 degree):
c.im |>
 import_fisheye(circ.mask=camera_fisheye('Coolpix4500+FC-E8')) |>
  binarize_fisheye() |>
  gapfrac_fisheye(lens='FC-E8',nrings=1,nseg=8,startVZA=55,endVZA=60) |>
  canopy_fisheye()
```

`gapfrac_fisheye` *Derive angular gap fraction from a classified fisheye image*

Description

The function calculates the gap fraction for a number of zenith annuli (rings) and azimuth sectors (segments). A list of lens is available for correcting for lens distorsion. Type 'list.lenses'.

Usage

```
gapfrac_fisheye(
  img.bw,
  maxVZA = 90,
  lens = "equidistant",
  startVZA = 0,
  endVZA = 70,
  nrings = 7,
  nseg = 8,
  message = FALSE,
  display = FALSE
)
```

Arguments

img.bw	SpatLayer. A binary, single-layer fisheye image generated from binarize_fisheye().
maxVZA	Numeric. The maximum Zenith angle (in degrees) corresponding to the image radius. Default= 90.
lens	Character. The lens type for fisheye-lens correction. A list of lenses is available by typing *list.lenses*. If missing, it is assumed equidistant.
startVZA	Numeric. The minimum Zenith angle (in degrees) considered in the analysis. Default is 0.
endVZA	Numeric. The maximum Zenith angle (in degrees) considered in the analysis. Default is 70.
nrings	Numeric. The number of equiangular zenith rings considered in the analysis. Default is 7.
nseg	Numeric. The number of azimuth segments considered in the analysis. Default is 8.
message	Logical. If set to TRUE, it reports the circular mask used in the analysis.
display	Logical. If set to TRUE, it desplays the zenith rings and azimuth segments overlaid on the fisheye image.

Value

A dataframe of gap fraction (GF) for zenith rings (rows) and azimuth segments (columns).

Author(s)

Francesco Chianucci

See Also

Lens correction functions have been retrieved from the following sources:

Pekin and Macfarlane 2009: doi:10.3390/rs1041298

Paul Bourke:http://www.paulbourke.net/dome/fisheyecorrect/

Hemisfer: https://www.schleppi.ch/patrick/hemisfer/index.php

Examples

```
c.im<-system.file('extdata/circular_coolpix4500+FC-E8_chestnut.jpg',package='hemispheR')

#List of lenses for fisheye projection correction:
list.lenses

#Zenith rings similar to LAI-2000/2200:
c.im |>
 import_fisheye(circ.mask=camera_fisheye('Coolpix4500+FC-E8')) |>
  binarize_fisheye() |>
  gapfrac_fisheye(lens='FC-E8',nrings=5,nseg=8,endVZA=75,display=TRUE)
```

```
#The hinge angle method close to 1 radian (57):
c.im |>
 import_fisheye(circ.mask=camera_fisheye('Coolpix4500+FC-E8')) |>
  binarize_fisheye() |>
  gapfrac_fisheye(lens='FC-E8',nrings=1,nseg=8,startVZA=55,endVZA=60,display=TRUE)
```

`import_fisheye` *Import a fisheye image as a single channel raster, and apply a circular mask*

Description

This function imports fisheye images using `terra::rast()` functionality, by selecting a single channel, or a combination of channels. The default option (blue channel) is generally preferred for canopy image analysis as it enables high contrast between canopy and sky pixels, which ease image thresholding. A circular mask is then applied to mask outside pixel in case of circular fisheye images. It can be manually inserted, or retrieved using the `camera_fisheye()` function. Alternatively, it is automatically calculated. Additional functions include a gamma correction and a contrast stretch.

Usage

```
import_fisheye(
  filename,
  channel = 3,
  circ.mask = NULL,
  circular = TRUE,
  gamma = 2.2,
  stretch = FALSE,
  display = FALSE,
  message = TRUE
)
```

Arguments

`filename`	Character. The input image filename.
`channel`	Character. Either the band number corresponding to an image channel or a mixing channel method (Available options are: 'first','GLA','Luma','2BG','BtoRG','B','GEI','RGB'). Default is 3 (Blue channel).
`circ.mask`	List. The circular mask parameters (xc,yc,rc) to be applied to the image. It can be created from a list of available cameras using the `camera_fisheye()` function. If omitted, it is created automatically in circular images, and corresponds to half the lower image side.
`circular`	Logical. It indicates if the fisheye image is circular (circular=TRUE) or full-frame (circular=FALSE) type. This influences the way the radius is calculated if circ.mask is not inserted. Default is circular.

gamma	Numeric. It indicates the input gamma, which is then back-corrected to unity. Default is 2.2 (typical in jpeg images). If no gamma is required, just set gamma=1.
stretch	Logical. It indicates if a linear stretch should be applied to enhance contrast. Default FALSE.
display	Logical. If is set to TRUE, it plots the image along with the applied mask and a circle radius. Default to FALSE.
message	Logical. If is set to TRUE, it prints the mask used for importing the image. Default to TRUE.

Value

A single-channel image (SpatRaster).

Examples

```
c.im<-system.file('extdata/circular_coolpix4500+FC-E8_chestnut.jpg',package='hemispheR')

#set the circular mask automatically:
import_fisheye(c.im,circ.mask=list(xc=1136,yc=850,rc=754),channel='B',gamma=2.2,display=TRUE)

#list of cameras for circular mask:
list.cameras

#set the circular mask using camera_fisheye():
import_fisheye(c.im,circ.mask=camera_fisheye('Coolpix4500+FC-E8'), gamma=2.2)

#automatic calculating circular mask:
import_fisheye(c.im,channel='B',gamma=2.2,display=TRUE)

#import a fullframe image:
f.im<-system.file('extdata/fullframe_D90_Nikkor-10.5_beech.jpg',package='hemispheR')
import_fisheye(f.im,circular=FALSE,channel='B',gamma=2.2,display=TRUE)
```

---

zonal_mask *Divide a raster image into four stacks which are used as masks.*

Description

This function imports a SpatRaster image using `terra::rast()` functionality, and divide into four masks, using the image centre and borders as vertices. The four zonal masks are then returned as a RasterStack.

Usage

```
zonal_mask(img)
```

Arguments

`img` SpatRaster. The input single layer image generated from `terra::rast()`.

Value

A 4-layers stacks of image masks

Examples

```
image<-system.file('extdata/circular_coolpix4500+FC-E8_chestnut.jpg',package='hemispheR')
zmsk<-zonal_mask(terra::rast(image, lyrs=3))
terra::plot(zmsk,col=gray.colors(5),main=c('N','W','S','E'))
```

Index