Image2Hemiphot(image = "") *convert image into hemiphot image*

Description
Turns a .jpg image into image of class hemiphot

Arguments
image image variable to be converted to hemiphot class

Details
none

Value
Returns an object of class hemiphot and adds a hemiphot circle. Center of this circle (cx, cy) are set by default to center of image and radius at half the height of the image. This can be changed with the function SetCircle()

Examples
```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)
PlotHemiImage(im.hemi)
```

See also
SetCircle(image = "", cx = 0, cy = 0, cr = 0)

SetCircle(image = "", cx = 0, cy = 0, cr = 0) *change circle position and radius*

Description

Changes circle dimensions and location on image of hemiphot class

Arguments

image	image variable whose circle needs adjusting
cx	x coordinate of hemiphot circle
cy	y coordinate of hemiphot circle
cr	radius of hemiphot circle

Details

none

Value

Returns an object of class hemiphot with new hemiphot circle.

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)
PlotHemiImage(im.hemi)
## modify circle
SetCircle(image = im.hemi, cx = 100, cy =100, cr = 100)
PlotHemiImage(im.hemi)
```

PlotHemiImage(image = "", draw.circle = T, channel = “”) *plot hemiphot class image*

Description

Plots an image of class Hemiphot

Arguments

image	image of hemiphot class
DrawCircle	determines if hemiphot circle is drawn on image
Channel	determines the color channel to be shown

Value

If DrawCircle is FALSE no circle is drawn. Default is TRUE. DrawCircle makes use of cx, cy and cr to draw circle. These are set at conversion to Hemiphot class. Values for channel are “R”, “G”, “B” for red, green or blue channel. If no value is passed the image is drawn with all color information available.

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)
PlotHemiImage(im.hemi, DrawCircle = T)
```

SelectRGB(image = "", channel = "")

Description

Selects one of the three color channels of an image of class Hemiphot

Arguments

image	image of hemiphot class
channel	color channel to be plotted “R”, “G”, “B”

Details

Channel can take three values (“R”, “G” or “B”). The chosen channel will be the one returned as value

Value

Returns image of class hemiphot with only one color channel

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")
```

ThresholdImage (image = "", th = 0.5, draw.image = F)

Description

Converts the image to black and white image with pixel values 0 and 1

Arguments

image	image of hemiphot class and 1 color channel only
th	threshold to be chosen, values between 0 and 1

Details

Th can take values between 0 and 1. Generally values should be around 0.5

Value

Returns image of class hemiphot with only one color channel with pixel values of 0 and 1

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")

## Threshold the image
im.hemi.th =  ThresholdImage (image = im.hemi.blue, th = 0.6, DrawImage = T)
```

CalcGapFractions(image = "") *calculate canopy openness fractions*

Description
Calculates canopy openness on 89 circles of image of hemiphot class

Arguments
image image of hemiphot class and 1 color channel only

Details
None

Value
Returns an array of 89 values of canopy openness

Examples
```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")

## Threshold the image
im.hemi.th =  ThresholdImage (image = im.hemi.blue, th = 0.6, DrawImage = T)
##calculate gap fractions and canopy openness
gap.fractions = CalcGapFractions(image.hemi.th)
canopy.openness = CalcOpenness(fractions = gap.fractions); canopy.openness
```

CalcOpenness (fractions) *calculate canopy openness*

Description

Calculates canopy openness above foto site based on partial openness on 89 circles of image of hemiphot class

Arguments

fractions	array of 89 values of partial canopy openness

Details

Fractions should be the result of the function CalcGapFractions()

Value

Returns the canopy openness above the image site

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")

## Threshold the image
im.hemi.th =  ThresholdImage (image = im.hemi.blue, th = 0.6, DrawImage = T)

##calculate gap fractions and canopy openness
gap.fractions = CalcGapFractions(image.hemi.th)
canopy.openness = CalcOpenness(fractions = gap.fractions); canopy.openness
```

CalcLAI(fractions, width = 6) *calculate leaf area index*

Description

Calculates leaf area index (LAI) above photo site based on partial openness on 89 circles of image of hemiphot class. Using 5 rings (7, 23, 38, 53, 68 degrees)

Arguments

fractions	array of 89 values of partial canopy openness
width	width of band (on both sides of ring) in number of pixels

Details

Fractions should be the result of the function CalcGapFractions()

Value

Returns the leaf area index above the image site

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")

## Threshold the image
im.hemi.th =  ThresholdImage (image = im.hemi.blue, th = 0.6, DrawImage = T)
##calculate gap fractions and canopy openness
gap.fractions = CalcGapFractions(image.hemi.th)
lai = CalcLAI(fractions = gap.fractions); lai
```

DrawSolarTracks = function(image ="", lat = 0, lon = 0, time.zone = 0, d = 150, magn.corr = 0, sun.location = F, h = 0) *show sun tracks*

Description

Shows suntracks of image of hemiphot class

Arguments

image	image of class hemiphotimage
lat	latitude of photo site, lat ranges from -90 to 90
lon	longitude of photo site, lon ranges from -180 to 180
d	day
magn.corr	Deviation of magnetic north from true geographical north
sun.location	if the location of the sun should be drawn (small yellow dot) for time = h
h	time for which sun location should be drawn

Details

Day can take a single value of a vector of days

Value

Returns none

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")

## Threshold the image
im.hemi.th =  ThresholdImage (image = im.hemi.blue, th = 0.6, DrawImage = T)

## Show solar tracks on image
DrawSolarTracks(image.hemi.th, lat = 52, d = 320, MagnCorr = 0)
```

CalcPAR.Day(image.th, lat = 0, lon = 0, time.zone = 0, d = 150, tau = 0.6, uoc = 0.15, magn.corr = 0, draw.tracks = T, full.day = T)

Description

Calculates Photosynthetic Active Radiation (PAR) at photo site

Arguments

image	image of class hemiphotimage with 1 color channel
lat	latitude of photo site, lat ranges from -90 to 90.
lon	longitude of photo site
time.zone	Time zone of photo site
tau	atmospheric transmissivity
uoc	percentage of direct radiation that is diffuse
d	day
draw.tracks	Boolean to determine of tracks should be drawn on image
magn.corr	Deviation of magnetic north from true geographical north
full.day	Store PAR values per 2 minute periods or just total for the day

Details

Latitude will be truncated between -89.9 and 89.9 is smaller or larger. Tau is usually set at 0.6 but may vary between 0.4 – 0.6 in tropics. UOC is the amount of direct light that is used as diffuse light in the Uniform Ovecast Sky (UOC). If Fullday is set to TRUE values are stored in 2 minute periods, else PAR is calculated as daily values. FullDay will be ignored if the number of days is more than 1 day.

Value

Returns an array with solar time, actual time (corrected with location in time zone and equation of time), and PAR in 4 portions (direct and diffuse above canopy and direct and diffuse below canopy). If FullDay is TRUE an array with 4 columns will be returned with 4 fractions of PAR in 2 minute periods.

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")

## Threshold the image
im.hemi.th =  ThresholdImage (image = im.hemi.blue, th = 0.6, DrawImage = T)

## Calculate PAR for 1 day in two minute periods
Rad = CalcPAR.Day(image.th, lat = 5, lon = -54, TimeZone = -4,
      d = 150, tau = 0.6, UOC = 0.15,
      MagnCorr = 0, DrawTracks = T, FullDay = T)
```

PlotPAR.Day(radiation = "", real.time = F)

Description

Constructs a plot of Photosynthetic Active Radiation (PAR) at photo site

Arguments

Radiation	array of two minute values of PAR
real.time	Boolean to determine if time used is solar time or actual location time

Details

Radiation should be an array produced by CalcParDay and have two hour columns and 4 radiation columns.

Value

Returns none

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")

## Threshold the image
im.hemi.th =  ThresholdImage (image = im.hemi.blue, th = 0.6, DrawImage = T)

## Calculate PAR for 1 day in two minute periods
Rad = CalcPAR.Day(image.th, lat = 5, lon = -54, TimeZone = -4,
        d = 150, tau = 0.6, UOC = 0.15,
        MagnCorr = 0, DrawTracks = T, FullDay = T)

PlotPAR.Day(radiation = Rad, RealTime = F)
```

CalcPAR.Year(image.th, lat = 0, lon = 0, tau = 0.6, uoc = 0.15, magn.corr = 0)

Description

Calculates Photosynthetic Active Radiation (PAR) at photo site for 365 days

Arguments

image	image of class hemiphotimage with 1 color channel
lat	latitude of photo site, lat ranges from -90 to 90.
lon	longitude of photo site
time.zone	Time zone of photo site
tau	atmospheric transmissivity
uoc	percentage of direct radiation that is diffuse
magn.corr	Deviation of magnetic north from true geographical north

Details

Latitude will be truncated between -89.9 and 89.9 is smaller or larger. Tau is usually set at 0.6 but may vary between 0.4 – 0.6 in tropics. UOC is the amount of direct light that is used as diffuse light in the Uniform Ovecast Sky (UOC). If Fullday is set to TRUE values are stored in 2 minute periods, else PAR is calculated as daily values. FullDay will be ignored if the number of days is more than 1 day.

Value

Returns an array with PAR in 4 portions (direct and diffuse above canopy and direct and diffuse below canopy), calculated for 365 days. First column of the array contains the days

Examples

```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")

## Threshold the image
im.hemi.th =  ThresholdImage (image = im.hemi.blue, th = 0.6, DrawImage = T)

## Calculate PAR for 365 days
Rad = CalcPAR.Year(image.th, lat = 52, tau = 0.6, UOC = 0.15, MagnCorr = 0)
```

PlotPAR.Year(radiation = "")

Description
Constructs a plot of Photosynthetic Active Radiation (PAR) at photo site for 365 days of the year

Arguments
Radiation array of 365 values of PAR

Details
Radiation should be an array produced by CalcParYear and have one day column and 4 radiation columns.

Value
Returns none

Examples
```
# read a sample file
im = readJPEG("DSCN0516.jpg", native = F)

## convert to class HemiphotImage - adding a circle
im.hemi = Image2Hemiphot(im)

## convert to hemiphotImage with only one color channel
im.hemi.blue = SelectRGB(image = im.hemi, channel = "B")

## Threshold the image
im.hemi.th =  ThresholdImage (image = im.hemi.blue, th = 0.6, DrawImage = T)

## Calculate PAR for 1 day in two minute periods
Rad = CalcPAR.Year(image.th, lat = 52, tau = 0.6, UOC = 0.15, MagnCorr = 0)

## Calculate PAR for 365 days
Rad = CalcPAR.Year(image.th, lat = 52, tau = 0.6, UOC = 0.15, MagnCorr = 0)
PlotPAR.Year(radiation = Rad)
```