

# Package ‘CTRing’

September 10, 2024

**Type** Package

**Title** Density Profiles of Wood from CT Scan Images

**Version** 0.1.0

**Maintainer** Robert Schneider <robert\_schneider@uqar.ca>

**Description** Computerized tomography (CT) can be used to assess certain wood properties when wood disks or logs are scanned. Wood density profiles (i.e. variations of wood density from pith to bark) can yield important information used for studies in forest resource assessment, wood quality and dendrochronology studies. The first step consists in transforming grey values from the scan images to density values. The packages then proposes a unique method to automatically locate the pith by combining an adapted Hough Transform method and a one-dimensional edge detector. Tree ring profiles (average ring density, earlywood and latewood density, ring width and percent latewood for each ring) are then obtained.

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**Suggests** knitr, rmarkdown

**VignetteBuilder** knitr

**Depends** xRing, functional, oro.dicom

**NeedsCompilation** no

**Author** Dipak Mahatara [aut],  
Robert Schneider [cre, aut]

**Repository** CRAN

**Date/Publication** 2024-09-10 09:40:07 UTC

## Contents

addRingFromImage . . . . .	2
addRingFromProfile . . . . .	3
addYears . . . . .	4
calcAvgDens . . . . .	5
checkProfile . . . . .	6

deleteRingFromImage . . . . .	7
deleteRingFromProfile . . . . .	8
densityDataFrame . . . . .	9
detect_pith . . . . .	10
extractProfile . . . . .	11
getEwLw . . . . .	12
getImageInfo . . . . .	13
grayToDensity . . . . .	14
imageToMatrix . . . . .	14
locatePathEnd . . . . .	15
pithCoordinates . . . . .	16
plotImageProfile . . . . .	16
plotProfile . . . . .	17
relToPixel . . . . .	18
removeLastYear . . . . .	19
verifyPith . . . . .	20
xBitTo8Bit . . . . .	20

<b>Index</b>	<b>22</b>
--------------	-----------

---

addRingFromImage	<i>Add ring to pith to bark profile from CT scan image</i>
------------------	--

---

### Description

Add ring to pith to bark profile from CT scan image

### Usage

```
addRingFromImage(n = 1, densProfile, im)
```

### Arguments

n	Number of rings to add
densProfile	Density profile
im	Density matrix

### Value

Corrected density profile with new ring(s) added and blue bar in plot of added ring

**Examples**

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
image_info <- getImageInfo(hdr = hdr_df)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens,
                          n_segments = 12,
                          pixel = TRUE,
                          toPlot = FALSE)

endPath <- c(472, 284)

densPath <- extractProfile(im_dens,
                          image_info,
                          pith_coord,
                          endPath,
                          k = 2, r = 5,
                          threshold = 0.002)

newPath2 <- addRingFromImage(n = 1, densPath, im_dens)

```

---

addRingFromProfile      *Add ring to pith to bark profile from profile plot*

---

**Description**

Add ring to pith to bark profile from profile plot

**Usage**

```
addRingFromProfile(n = 1, densProfile)
```

**Arguments**

n	Number of rings to add
densProfile	Density profile

**Value**

Corrected density profile with new ring(s) added and blue bar in plot of added ring

**Examples**

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

path <- extractProfile(im_dens, image_info, pith_coord, endPath, k = 2, r = 5, threshold = 0.002)

plotProfile(path)
newPath <- addRingFromProfile(n = 1, path)

```

---

addYears

*Add years to series*


---

**Description**

Add years to series

**Usage**

```
addYears(lastYear, densProfile)
```

**Arguments**

lastYear	Last year of series
densProfile	Density profile

**Value**

Density profile with years

**Examples**

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

```

```

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

path <- extractProfile(im_dens, image_info, pith_coord, endPath, k = 2, r = 5, threshold = 0.002)

path <- addYears(2021, path)

```

---

calcAvgDens	<i>Calculate average wood, earlywood and latewood density for every ring</i>
-------------	--

---

### Description

Calculate average wood, earlywood and latewood density for every ring

### Usage

```
calcAvgDens(densProfile)
```

### Arguments

densProfile     Density profile

### Value

List with several vectors

### Examples

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

```

```

path <- extractProfile(im_dens, image_info, pith_coord, endPath, k = 2, r = 5, threshold = 0.002)

pathEwLw <- getEwLw(path)
plotProfile(pathEwLw)
path_avgDens <- calcAvgDens(pathEwLw)
names(path_avgDens)

```

---

checkProfile	<i>Verify position of ring transitions of a density profile</i>
--------------	---

---

### Description

Verify position of ring transitions of a density profile

### Usage

```
checkProfile(profile_with_borders, totRings)
```

### Arguments

profile_with_borders	xRing profile with transitions between rings located
totRings	Total number of rings of the disk

### Value

xRing profile with corrected ring location

### Examples

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
image_info <- getImageInfo(hdr = hdr_df)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens,
                          n_segments = 12,
                          pixel = TRUE,
                          toPlot = FALSE)

endPath <- c(472, 284)

```

```
densPath <- extractProfile(im_dens,
                          image_info,
                          pith_coord,
                          endPath,
                          k = 2, r = 5,
                          threshold = 0.002)

newPath <- checkProfile(densPath, 26)
```

---

deleteRingFromImage    *Add ring to pith to bark profile from CT scan image*

---

### Description

Add ring to pith to bark profile from CT scan image

### Usage

```
deleteRingFromImage(n = 1, densProfile, im)
```

### Arguments

n	Number of rings to remove
densProfile	Density profile
im	Density matrix

### Value

Corrected density profile with ring(s) removed and red bar in plot of deleted ring

### Examples

```
library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual

densPath <- extractProfile(im_dens,
                          image_info,
                          pith_coord,
```

```

                                endPath,
                                k = 2, r = 5,
                                threshold = 0.002)

newPath2 <- addRingFromImage(n = 1, densPath, im_dens)
oldPath2 <- deleteRingFromImage(n = 1, densPath, im_dens)

```

---

deleteRingFromProfile *Delete ring from a pith to bark profile*

---

### Description

Delete ring from a pith to bark profile

### Usage

```
deleteRingFromProfile(n = 1, densProfile)
```

### Arguments

n	Number of rings to remove
densProfile	Density profile

### Value

Corrected density profile with ring(s) removed and red bar in plot of deleted ring

### Examples

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

densPath <- extractProfile(im_dens,
                           image_info,
                           pith_coord,
                           endPath,
                           k = 2, r = 5,

```

```

threshold = 0.002)

plotProfile(densPath)
newPath <- addRingFromProfile(n = 1, densPath)
oldPath <- deleteRingFromProfile(n = 1, newPath)

```

---

densityDataFrame	<i>Convert to dataframe</i>
------------------	-----------------------------

---

## Description

Convert to dataframe

## Usage

```
densityDataFrame(densProfile, sampleID = "NoID", addTransitionType = FALSE)
```

## Arguments

densProfile	Density profile
sampleID	Sample ID
addTransitionType	add transition type to dataframe

## Value

Dataframe with cambial age, density, years, transition type

## Examples

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

path <- extractProfile(im_dens, image_info, pith_coord, endPath, k = 2, r = 5, threshold = 0.002)

pathEwLw <- getEwLw(path)
plotProfile(pathEwLw)

```

```
path_avgDens <- calcAvgDens(pathEwLw)
densityDf <- densityDataFrame(pathEwLw)
```

---

detect\_pith                      *Automatically detect pith in a CT scan image*

---

### Description

Automatically detect pith in a CT scan image

### Usage

```
detect_pith(
  im,
  toPlot = TRUE,
  n_segments = 25,
  flag = TRUE,
  x_0 = 0.5,
  y_0 = 0.5,
  n_run_max = 15,
  threshold = 0.1,
  pixel = TRUE
)
```

### Arguments

im	Matrix of the CT scan image
toPlot	Boolean to plot the location of the pith on the image
n_segments	Number of segments used to locate pith
flag	FALSE if pith location is known
x_0	Estimate of pith location in x
y_0	Estimate of pith location in y
n_run_max	Maximum number of iterations
threshold	Threshold value for identifying ring transition points
pixel	If TRUE, returns x,y coordinates in pixel numbers, else FALSE returns x,y coordinates in relative values of x and y

### Value

x,y pith coordinates

**Examples**

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

```

---

extractProfile	<i>Get profile between two points of the CTScan image matrix</i>
----------------	--

---

**Description**

Get profile between two points of the CTScan image matrix

**Usage**

```

extractProfile(
  im,
  imHeader,
  beginPath,
  endPath,
  r = 10,
  k = 2,
  threshold = 0.01
)

```

**Arguments**

im	Density matrix
imHeader	image header
beginPath	X,Y coordinates of the start point of the path
endPath	X,Y coordinates of the start point of the path
r	Profile width
k	Rolling window width, integer
threshold	Threshold value between maximum and minimum density to establish change of ring

**Value**

Density profile

**Examples**

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

path <- extractProfile(im_dens, image_info, pith_coord, endPath, k = 2, r = 5, threshold = 0.002)

```

---

getEwLw

*Establish the transition point from earlywood to latewood for a series of rings*


---

**Description**

Establish the transition point from earlywood to latewood for a series of rings

**Usage**

```
getEwLw(densProfile)
```

**Arguments**

densProfile     Density profile

**Value**

xRingList with EW to LW transition points with transition type added (1: low number of points in ring; 2: inflexion point estimated by polynomial; 3: min or max are out of range; 4: inflexion point close to min or max; 5: convex-concave)

**Examples**

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

```

```
im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

path <- extractProfile(im_dens, image_info, pith_coord, endPath, k = 2, r = 5, threshold = 0.002)

pathEwLw <- getEwLw(path)

densityDf <- densityDataFrame(path)
```

---

getImageInfo	<i>Extract from header of CT scan image grayscale number of bits and pixel size</i>
--------------	---

---

### Description

Extract from header of CT scan image grayscale number of bits and pixel size

### Usage

```
getImageInfo(hdr)
```

### Arguments

hdr                   Header dataframe

### Value

List with grayscale values, and pixel size

### Examples

```
library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
getImageInfo(hdr = hdr_df)
```

grayToDensity                    *Convert from 8bit gray scale to density*

---

**Description**

Convert from 8bit gray scale to density

**Usage**

```
grayToDensity(im, a = -0.1321, b = 0.01834)
```

**Arguments**

im	Matrix of CT scan image in 8bit gray scale
a	Intercept of the calibration curve
b	Slope of the calibration curve

**Value**

Matrix of density values

**Examples**

```
library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
range(im_8bit)

im_dens <- grayToDensity(im_8bit)
range(im_dens)
```

---

imageToMatrix                    *Convert dicom image to matrix*

---

**Description**

Convert dicom image to matrix

**Usage**

```
imageToMatrix(img)
```

**Arguments**

img                   Dicom image

**Value**

Matrix of image

**Examples**

```
library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
dim(im)
image(im)
```

---

locatePathEnd

*Get coordinates of the end of the path on a CT scan image*

---

**Description**

Get coordinates of the end of the path on a CT scan image

**Usage**

```
locatePathEnd(im, pithCoord)
```

**Arguments**

im                    CT scan image  
pithCoord            X,Y coordinates of the pith

**Value**

Coordinates of the end of the path

**Examples**

```
library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
```

```

im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

```

---

pithCoordinates      *convert pith coordinates from pixels to length units*

---

### Description

convert pith coordinates from pixels to length units

### Usage

```
pithCoordinates(pith_coord, pixel_size_x, pixel_size_y)
```

### Arguments

pith_coord	Pith coordinates in pixels
pixel_size_x	Pixel size in x
pixel_size_y	Pixel size in y

### Value

Pixel coordinates in length units

---

plotImageProfile      *Plot scan image, profile path and ring limits*

---

### Description

Plot scan image, profile path and ring limits

### Usage

```
plotImageProfile(densProfile, im)
```

### Arguments

densProfile	Density profile
im	Density matrix

**Value**

Plot

**Examples**

```
library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual

path <- extractProfile(im_dens, image_info, pith_coord, endPath, k = 2, r = 5, threshold = 0.002)

plotProfile(path)

plotImageProfile(path, im_dens)
```

---

plotProfile

*Plot density profile*

---

**Description**

Plot density profile

**Usage**

plotProfile(densProfile)

**Arguments**

densProfile    Density profile

**Value**

Figure

**Examples**

```

library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

path <- extractProfile(im_dens, image_info, pith_coord, endPath, k = 2, r = 5, threshold = 0.002)

plotProfile(path)

```

---

relToPixel

*Change from relative to fixed pixel coordinate system*


---

**Description**

Change from relative to fixed pixel coordinate system

**Usage**

```
relToPixel(pith_coord, im)
```

**Arguments**

pith_coord	Pith coordinates in relative space (x, y)
im	Density matrix

**Value**

Pixel coordinates in number of pixels (x, y)

---

removeLastYear	<i>Remove the last year of a profile</i>
----------------	--

---

## Description

Remove the last year of a profile

## Usage

```
removeLastYear(densProfile)
```

## Arguments

densProfile     Density profile

## Value

Density profile with the last year removed

## Examples

```
library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

endPath <- c(472, 284) # manual
# not run - endPath <- locatePathEnd(im_dens, pith_coord) # using the image

path <- extractProfile(im_dens, image_info, pith_coord, endPath, k = 2, r = 5, threshold = 0.002)
path_last_year_2021 <- addYears(2021, path)
path_last_year_2020 <- removeLastYear(path_last_year_2021)
```

---

verifyPith	<i>Check if pith location is correct</i>
------------	--

---

**Description**

Check if pith location is correct

**Usage**

```
verifyPith(im, pith_coord)
```

**Arguments**

im	Density matrix of image
pith_coord	Pith coordinates

**Value**

Corrected pith coordinates

**Examples**

```
library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
im_8bit <- xBitTo8Bit(im, image_info$grayScale)
im_dens <- grayToDensity(im_8bit)

pith_coord <- detect_pith(im_dens, n_segments = 12, pixel = TRUE, toPlot = FALSE)

pith_coord_checked <- verifyPith(im_dens, pith_coord)
```

---

xBitTo8Bit	<i>Convert gray scale from measured bits to 8bit</i>
------------	--

---

**Description**

Convert gray scale from measured bits to 8bit

**Usage**

```
xBitTo8Bit(im, bits)
```

**Arguments**

<code>im</code>	Matrix of values in x bits
<code>bits</code>	Number of bits of the original gray scale

**Value**

Matrix of gray scale values in 8bits

**Examples**

```
library(oro.dicom)
file_path <- system.file("extdata", "disk.dcm", package = "CTRing")
dcm <- readDICOM(file_path)
hdr_df <- dcm$hdr[[1]]
image_info <- getImageInfo(hdr = hdr_df)

im <- imageToMatrix(dcm$img)
range(im)

im_8bit <- xBitTo8Bit(im, image_info$grayScale)
range(im_8bit)
```

# Index

[addRingFromImage](#), 2  
[addRingFromProfile](#), 3  
[addYears](#), 4

[calcAvgDens](#), 5  
[checkProfile](#), 6

[deleteRingFromImage](#), 7  
[deleteRingFromProfile](#), 8  
[densityDataFrame](#), 9  
[detect\\_pith](#), 10

[extractProfile](#), 11

[getEwLw](#), 12  
[getImageInfo](#), 13  
[grayToDensity](#), 14

[imageToMatrix](#), 14

[locatePathEnd](#), 15

[pithCoordinates](#), 16  
[plotImageProfile](#), 16  
[plotProfile](#), 17

[relToPixel](#), 18  
[removeLastYear](#), 19

[verifyPith](#), 20

[xBitTo8Bit](#), 20