Package 'rTwig'

March 3, 2025

Title Realistic Quantitative Structure Models

Version 1.4.0

Description Real Twig is a method to correct branch overestimation in quantitative structure models. Overestimated cylinders are correctly tapered using measured twig diameters of corresponding tree species. Supported quantitative structure modeling software includes 'TreeQSM', 'SimpleForest', 'Treegraph', and 'aRchi'. Also included is a novel database of twig diameters and tools for fractal analysis of point clouds.

Encoding UTF-8

RoxygenNote 7.3.2

RdMacros Rdpack

Imports R.matlab, rmatio, tidytable, cobs, igraph, rgl, colourvalues, Rdpack, rlang, Rcpp, RcppSimdJson, geometry

License GPL (>= 3)

Depends R (>= 3.5.0)

LazyData true

Suggests knitr, rmarkdown, ggplot2, ggpmisc, ggpubr, gt, yardstick, dplyr, tidyr

VignetteBuilder knitr

NeedsCompilation yes

Maintainer Aidan Morales <moral169@msu.edu>

URL https://aidanmorales.github.io/rTwig/, https://github.com/aidanmorales/rTwig

BugReports https://github.com/aidanmorales/rTwig/issues

LinkingTo Rcpp, RcppArmadillo

Author Aidan Morales [aut, cre, cph] (<https://orcid.org/0009-0000-4513-4908>), David W. MacFarlane [aut, cph]

Repository CRAN

Date/Publication 2025-03-03 17:10:09 UTC

Contents

`box_dimension` *Box Dimension*

Description

R port of Dominik Seidel's fractal analysis "box-dimension" metric.

Usage

```
box_dimension(cloud, lowercutoff = 0.01, rm_int_box = FALSE, plot = FALSE)
```

Arguments

`cloud`	A point cloud matrix size n x 3. Non-matrices are automatically converted to a matrix.
`lowercutoff`	The smallest box size determined by the point spacing of the cloud in meters. Defaults to 1 cm.
`rm_int_box`	Remove the initial box as TRUE or FALSE. Defaults to FALSE.
`plot`	Plot the results. The user can specify "2D", "3D", or "ALL" plots. FALSE disables plotting. Defaults to FALSE.

Value

Returns a list

References

Arseniou G, MacFarlane DW, Seidel D (2021). "Measuring the Contribution of Leaves to the Structural Complexity of Urban Tree Crowns with Terrestrial Laser Scanning." *Remote Sensing*, **13**(14). doi:10.3390/rs13142773.

Mandelbrot BB (1983). *The fractal geometry of nature*. Freeman.

Saarinen N, Calders K, Kankare V, Yrttimaa T, Junttila S, Luoma V, Huuskonen S, Hynynen J, Verbeeck H (2021). "Understanding 3D structural complexity of individual Scots pine trees with different management history." *Ecology and Evolution*, **11**(6), 2561-2572. doi:10.1002/ece3.7216.

Seidel D (2018). "A holistic approach to determine tree structural complexity based on laser scanning data and fractal analysis." *Ecology and Evolution*, **8**(1), 128-134. doi:10.1002/ece3.3661.

Seidel D, Annighöfer P, Stiers M, Zemp CD, Burkardt K, Ehbrecht M, Willim K, Kreft H, Hölscher D, Ammer C (2019). "How a measure of tree structural complexity relates to architectural benefit-to-cost ratio, light availability, and growth of trees." *Ecology and Evolution*, **9**(12), 7134-7142. doi:10.1002/ece3.5281.

Examples

```
## Calculate Box Dimension
file <- system.file("extdata/cloud.txt", package = "rTwig")
cloud <- read.table(file, header = FALSE)
output <- box_dimension(cloud, plot = "ALL")
output
```

---

cluster_cloud *Cluster Cloud*

Description

Cluster a point cloud or simulate a point cloud based on its QSM. If using the input point cloud, the cylinder ids are transferred to the cloud using the nearest neighbors. If simulating a cloud, the points are built directly from the cylinders with user controlled spacing. The cylinder id can be used to easily join the desired variables from the QSM to the cloud. The nearest neighbor search uses the C++ nanoflann library.

Usage

```
cluster_cloud(cylinder, cloud = NULL, spacing = NULL)
```

Arguments

cylinder	A QSM cylinder data frame.
cloud	The input point cloud for the QSM to cluster. If NULL (default), the simulated cloud is returned.
spacing	The point spacing in meters for the simulated cloud. Defaults to 0.02 meters.

Value

A point cloud data frame

References

Blanco JL, Rai PK (2014). "nanoflann: a C++ header-only fork of FLANN, a library for Nearest Neighbor (NN) with KD-trees." https://github.com/jlblancoc/nanoflann.

Examples

```
## Load Data
file <- system.file("extdata/QSM.mat", package = "rTwig")
file2 <- system.file("extdata/cloud.txt", package = "rTwig")

qsm <- run_rtwig(file, twig_radius = 4.23, metrics = FALSE)
cloud <- read.table(file2)

## Clustered Cloud
clustered_cloud <- cluster_cloud(cylinder = qsm, cloud = cloud)

# # Join QSM variables and export
# filename <- tempfile(pattern = "clustered_cloud", fileext = ".txt")
# clustered_cloud %>%
#   left_join(qsm) %>%
#   fwrite(file = filename)

## Simulated Cloud
simulated_cloud <- cluster_cloud(cylinder = qsm, spacing = 0.01)

# Plot Simulated Cloud
plot_qsm(cloud = simulated_cloud)

# # Join QSM variables and export
# filename2 <- tempfile(pattern = "simulated_cloud", fileext = ".txt")
# simulated_cloud %>%
#   left_join(qsm) %>%
#   fwrite(file = filename2)
```

correct_radii *Correct Radii*

Description

Corrects cylinder radii

Usage

```
correct_radii(cylinder, twig_radius, broken_branch = TRUE)
```

Arguments

cylinder	QSM cylinder data frame
twig_radius	Twig radius in millimeters
broken_branch	Enable or disable the broken branch filter. Defaults to enabled (TRUE).

Value

Returns a data frame

Examples

```
## TreeQSM Processing Chain
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file)
cylinder <- qsm$cylinder
cylinder <- update_cylinders(cylinder)
cylinder <- correct_radii(cylinder, twig_radius = 4.23)
str(cylinder)
```

export_mat *Export MAT*

Description

Exports the cylinder data to the format used by TreeQSM

Usage

```
export_mat(cylinder, filename)
```

Arguments

cylinder	QSM cylinder data frame
filename	Desired name of file

Value

Returns a .mat file

Examples

```
## TreeQSM Processing Chain
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file)
cylinder <- qsm$cylinder
cylinder <- update_cylinders(cylinder)

filename <- tempfile(pattern = "TreeQSM_QSM", fileext = ".mat")
export_mat(cylinder, filename)

## SimpleForest Processing Chain
file <- system.file("extdata/QSM.csv", package = "rTwig")
cylinder <- read.csv(file)
cylinder <- update_cylinders(cylinder)

filename <- tempfile(pattern = "SimpleForest_QSM", fileext = ".mat")
export_mat(cylinder, filename)
```

---

export_mesh *Export Mesh*

Description

Exports a QSM cylinder mesh in various formats

Usage

```
export_mesh(
  cylinder,
  filename,
  format = "ply",
  radius = NULL,
  color = NULL,
  palette = NULL,
  facets = 6,
  normals = FALSE,
  alpha = 1
)
```

Arguments

cylinder	QSM cylinder data frame.

filename	File name and path for exporting. The file extension is automatically added if not present.
format	Mesh file format. Defaults to ply. Supported formats include ply, obj, stl, and blender. format = "blender" exports the mesh in the qsm-blender-addons format.
radius	Radius column name either quoted or unquoted. Defaults to the modified radii.
color	Optional cylinder color parameter. color must be a single hex color string, a grDevices::colors(), a vector of hex colors, or a quoted/unquoted column name. Vectors must have the same length as the cylinder data frame. color = "random" will generate a random color applied to all cylinders. Defaults to branching order.
palette	Optional cylinder color palette for numerical data. Palettes include colourvalues::color_palettes() or a user supplied RGB palette matrix with the length of cylinder. It can also be set to "random" to generate a random palette. If combined with color = "random", each cylinder will have a random, distinct color.
facets	The number of facets in the polygon cross section. Defaults to 6. A higher number of facets improves visual smoothness at the cost of plotting speed, performance and memory.
normals	Export surface normals per vertex. Defaults to FALSE.
alpha	Set the transparency of the cylinders used in the "ply" format. Defaults to 1. 1 is opaque and 0 is fully transparent.

Value

A mesh file

Examples

```
## Load QSM
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file)
cylinder <- qsm$cylinder
cylinder <- update_cylinders(cylinder)

# PLY
filename <- tempfile(pattern = "QSM_ply")
export_mesh(
  cylinder = cylinder,
  filename = filename,
  format = "ply",
  color = "distanceToTwig",
  palette = "viridis",
  normals = TRUE
)

# OBJ
filename <- tempfile(pattern = "QSM_obj")
export_mesh(
  cylinder = cylinder,
```

```
  filename = filename,
  format = "obj",
  normals = TRUE
)

# STL
filename <- tempfile(pattern = "QSM_stl")
export_mesh(
  cylinder = cylinder,
  filename = filename,
  format = "stl",
  normals = TRUE
)

# QSM Blender Addons
filename <- tempfile(pattern = "QSM_blender")
export_mesh(
  cylinder = cylinder,
  filename = filename,
  format = "blender",
  normals = TRUE
)
```

`import_leaves` *Import Leaves*

Description

Imports leaves created by the QSM foliage and needles naive insertion (QSM-FaNNI) algorithm. The leaves are stored as a triangular mesh, compatible with both rTwig and RGL functions.

Usage

```
import_leaves(filename, format = "obj", simplify = FALSE)
```

Arguments

`filename`	a QSM-FaNNI .obj file
`format`	.obj file format. Valid formats include `obj` and `obj_ext`. Defaults to `obj`.
`simplify`	simplify the mesh by removing duplicate vertices. This can reduce memory usage at the cost of import speed. Defaults to FALSE.

Value

rgl::mesh3d object

References

Åkerblom M, Raumonen P, Casella E, Disney MI, Danson FM, Gaulton R, Schofield LA, Kaasalainen M (2018). "An algorithm to insert leaves into quantitative structure models of trees." *Interface Focus*, **8**(2), 20170045. ISSN 2042-8898, doi:10.1098/rsfs.2017.0045, 20180216.

Examples

```
## Not run:

## Import FaNNI leaves from the standard obj format
file <- "path_to_leaves.obj"
qsm <- import_leaves(file, format = "obj")

## Import FaNNI leaves from the extended obj format
file <- "path_to_leaves_extended.obj"
qsm <- import_leaves(file, format = "obj_ext")

## End(Not run)
```

`import_qsm` *Import TreeQSM*

Description

`import_qsm()` is deprecated and will be removed in a future version. Use `import_treeqsm()` instead.

Usage

```
import_qsm(...)
```

Arguments

`...`	function inputs

Value

Returns a list

`import_treegraph` *Import Treegraph*

Description

Imports a QSM created by treegraph

Usage

```
import_treegraph(filename)
```

Arguments

`filename`	a treegraph .json file

Value

Returns a list

References

Yang W, Wilkes P, Vicari MB, Hand K, Calders K, Disney M (2024). "Treegraph: tree architecture from terrestrial laser scanning point clouds." *Remote Sensing in Ecology and Conservation*. ISSN 2056-3485, doi:10.1002/rse2.399.

Wilkes P, Shenkin A, Disney M, Malhi Y, Bentley LP, Vicari MB (2021). "Terrestrial laser scanning to reconstruct branch architecture from harvested branches." *Methods in Ecology and Evolution*, **12**, 2487-2500. doi:10.1111/2041210X.13709.

Examples

```
## Not run:

# Import a treegraph QSM
qsm <- import_treegraph("path/to/json/file")

## End(Not run)
```

`import_treeqsm` *Import TreeQSM*

Description

Imports a QSM created by TreeQSM

Usage

```
import_treeqsm(filename, version = "2.x.x")
```

Arguments

`filename`	a TreeQSM .mat MATLAB file
`version`	TreeQSM version. Valid versions include `2.3.0` - `2.4.1` and `2.0`. Defaults to the most recent format of `2.x.x`.

Value

Returns a list

References

Raumonen P, Kaasalainen M, Åkerblom M, Kaasalainen S, Kaartinen H, Vastaranta M, Holopainen M, Disney M, Lewis P (2013). "Fast Automatic Precision Tree Models from Terrestrial Laser Scanner Data." *Remote Sensing*, **5**(2), 491–520. doi:10.3390/rs5020491.

Examples

```
## Read a TreeQSM MATLAB file in the 2.3.x - 2.4.x format
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file, version = "2.x.x")
summary(qsm)

## Read a TreeQSM MATLAB file in the 2.0 format
file <- system.file("extdata/QSM_2.mat", package = "rTwig")
qsm <- import_treeqsm(file, version = "2.0")
names(qsm)
```

plot_qsm *Plot QSM*

Description

Efficiently plot QSMs, point clouds, leaves, and stem triangulation meshes. Plots can be combined and extensively customized.

Usage

```
plot_qsm(
  cylinder = NULL,
  radius = NULL,
  color = NULL,
  palette = NULL,
  alpha = 1,
  facets = 6,
  skeleton = FALSE,
  skeleton_lwd = 1,
  cloud = NULL,
  pt_color = "#000000",
  pt_palette = NULL,
  pt_size = NULL,
  pt_alpha = 1,
  triangulation = NULL,
  tri_color = NULL,
  tri_palette = NULL,
  tri_alpha = 1,
  leaves = NULL,
  lf_color = "#5BA803",
  lf_alpha = 1,
  axes = TRUE,
  axes_color = "#000000",
  grid = FALSE,
  grid_color = "#D3D3D3",
  hover = FALSE,
  bg_color = "#FFFFFF",
  lit = FALSE,
  pan = TRUE,
  normalize = FALSE,
  mesh = NULL
)
```

Arguments

cylinder	QSM cylinder data frame.
radius	Radius column name either quoted or unquoted. Defaults to the modified radii.

color	Optional cylinder color parameter. color must be a single hex color string, a grDevices::colors(), a vector of hex colors, or a quoted/unquoted column name. Vectors must have the same length as the cylinder data frame. color = "random" will generate a random color applied to all cylinders. Defaults to branching order.
palette	Optional cylinder color palette for numerical data. Palettes include colourvalues::color_palettes() or a user supplied RGB palette matrix with the length of cylinder. It can also be set to "random" to generate a random palette. If combined with color = "random", each cylinder will have a random, distinct color.
alpha	Set the transparency of the cylinders. Defaults to 1. 1 is opaque and 0 is fully transparent.
facets	The number of facets in the polygon cross section. Defaults to 6. A higher number of facets improves visual smoothness at the cost of plotting speed, performance and memory.
skeleton	Plot the QSM skeleton instead of cylinders. Defaults to FALSE.
skeleton_lwd	Skeleton line width. Defaults to 1.
cloud	Point cloud data frame or matrix where the first three columns are the x, y, and z coordinates.
pt_color	Optional point cloud color parameter. pt_color must be a single hex color string, a grDevices::colors(), a vector of hex colors, or a quoted/unquoted column name. pt_color = "random" will generate a random color for all points. Vectors must have the same length as the point cloud data frame.
pt_palette	Optional point cloud color palette for numerical data. pt_palette includes colourvalues::color_palettes() or a user supplied RGB palette matrix with the length of the points. It can also be set to "random" to generate a random palette. If combined with color = "random", each point will have a random, distinct color.
pt_size	Size of the points. Defaults to 0.1.
pt_alpha	Set the transparency of the point cloud. Defaults to 1. 1 is opaque and 0 is fully transparent.
triangulation	Stem triangulation mesh from TreeQSM. Defaults to NULL.
tri_color	Color of the triangulation mesh. Accepts hex colors, grDevices::colors(), or "random". Defaults to rainbow z-axis.
tri_palette	Optional triangulation color palette for z values. Supports the same inputs as palette.
tri_alpha	Set the transparency of the triangulation mesh. Defaults to 1. 1 is opaque and 0 is fully transparent.
leaves	Leaf mesh in the rgl::mesh3d() format. Defaults to NULL.
lf_color	Color of the leaves. Defaults to leaf green. Accepts hex colors, grDevices::colors(),"random", or "random_vertex". lf_color = "random" assigns a random color to all leaves, while lf_color = "random_vertex" which assigns a random color to each vertex.
lf_alpha	Set the transparency of the leaves. Defaults to 1. 1 is opaque and 0 is fully transparent.

axes	Show plot axes. Defaults to TRUE.
axes_color	Set the axes color. Defaults to black.
grid	Show plot grid lines. Defaults to FALSE.
grid_color	Set grid lines color. Defaults to grey.
hover	Show cylinder and branch id on mouse hover. Defaults to FALSE.
bg_color	Set the background color of the plot. Accepts hex colors or `grDevices::colors()`. Defaults to white.
lit	Enable light source in plot. Defaults to FALSE.
pan	Use right mouse button to pan the plot. Defaults to TRUE. Panning is disabled when `hover = TRUE`.
normalize	Normalize the cylinders to 0,0,0. Defaults to FALSE.
mesh	Plot any `rgl::mesh3d()` object.

Value

An `rgl::open3d()` plot

Examples

```
## TreeQSM
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file)
cylinder <- qsm$cylinder
cylinder <- update_cylinders(cylinder)
plot_qsm(cylinder)

triangulation <- qsm$triangulation
plot_qsm(triangulation = triangulation)
```

---

prune_qsm *Prune QSM*

Description

Efficiently prune a QSM. The user can prune by cylinder, branch, and segment ids, or by height or diameter classes, individually, or all at the same time, and return either the pruned data, the remaining data, or a binary index of which cylinders are pruned.

Usage

```
prune_qsm(
  cylinder,
  cylinder_ids = NULL,
  branch_ids = NULL,
  segment_ids = NULL,
  height_m = NULL,
  diameter_cm = NULL,
  invert = FALSE,
  index = FALSE
)
```

Arguments

cylinder	QSM cylinder data frame
cylinder_ids	A single or vector of cylinder ids. Everything connected above the cylinder is pruned.
branch_ids	A single or vector of branch ids. Everything connected to the branch is pruned.
segment_ids	A single or vector of segment ids. Everything connected above the segment is pruned.
height_m	Height class in meters below which all cylinders are pruned. Valid inputs are numeric to one decimal.
diameter_cm	Branch diameter class in centimeters below which all cylinders are pruned. Valid inputs are numeric to one decimal.
invert	Return the remaining or pruned data. Defaults to TRUE (the remaining data), but can be set to FALSE.
index	Returns a column index called pruning indicating if the cylinder is pruned (1) or un-pruned (0). Defaults to FALSE, but can be set to TRUE.

Value

a data frame

Examples

```
## Load QSM
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file)
cylinder <- qsm$cylinder
cylinder <- update_cylinders(cylinder)

## Pruning Index
prune1 <- prune_qsm(cylinder, height_m = 2, index = TRUE)
plot_qsm(prune1, color = pruning, palette = "blue2red")

## Remaining
prune2 <- prune_qsm(cylinder, height_m = 2, invert = FALSE)
plot_qsm(prune2)
```

```
## Pruned
prune3 <- prune_qsm(cylinder, height_m = 2, invert = TRUE)
plot_qsm(prune3)
```

`qsm_summary` *QSM Summary*

Description

`qsm_summary` is deprecated and will be removed in a future version. Use `summarise_qsm()` instead.

Usage

```
qsm_summary(...)
```

Arguments

`...`	function inputs

Value

Returns a list

`run_rtwig` *Run Real Twig*

Description

Run the Real Twig correction and calculate tree metrics for supported QSM formats.

Usage

```
run_rtwig(
  filename,
  twig_radius,
  metrics = TRUE,
  version = NULL,
  smooth = TRUE,
  standardise = FALSE,
  broken_branch = TRUE,
  ...
)
```

Arguments

filename	file path to QSM (.mat, .csv, .json)
twig_radius	Twig radius in millimeters
metrics	Calculate tree metrics. Defaults to TRUE.
version	If using a specific version of TreeQSM, the user can specify the version (e.g. 2.4.1, 2.0, etc.).
smooth	Defaults to TRUE if using TreeQSM. Can be set to FALSE.
standardise	Standardise QSM cylinder data. Defaults to FALSE, but can be set to TRUE.
broken_branch	Enable or disable the broken branch filter. Defaults to TRUE.
...	Additional arguments allowing standardise and standardize to be used as synonyms.

Value

Returns cylinder data frame or list if metrics is true.

Examples

```
## TreeQSM
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- run_rtwig(file, twig_radius = 4.23)
str(qsm$cylinder)
```

---

smooth_qsm *Smooth QSM*

Description

Visually smooth a QSM by ensuring the midpoints of all cylinders are connected. Only TreeQSM is supported.

Usage

```
smooth_qsm(cylinder)
```

Arguments

cylinder	QSM cylinder data frame

Value

Returns a data frame

Examples

```
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file)
cylinder <- qsm$cylinder
cylinder <- update_cylinders(cylinder)

## Before Smoothing
plot_qsm(cylinder)

## After Smoothing
cylinder <- smooth_qsm(cylinder)
plot_qsm(cylinder)
```

`standardise_qsm` *Standardise QSM*

Description

All QSM variables are renamed and reordered a standardised format across the supported QSM software for a consistent experience. All internal rTwig functions use these standardised names for consistency.

Usage

```
standardise_qsm(cylinder)

standardize_qsm(cylinder)
```

Arguments

`cylinder`	QSM cylinder data frame

Details

Renames supported QSM software output columns to be consistent. All names are lower case and underscore delimited. See the dictionary vignette for a detailed description of column names. A consistent QSM format ensures maximum compatibility when analyzing QSMs made with different software. This function can be run either before or after `update_cylinders()` has been run, or at any stage.

`standardise_qsm()` and `standardise_qsm()` are synonyms.

Value

Returns a data frame

Examples

```
## TreeQSM Processing Chain
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file)
cylinder <- qsm$cylinder
cylinder <- standardise_qsm(cylinder)
str(cylinder)

## SimpleForest Processing Chain
file <- system.file("extdata/QSM.csv", package = "rTwig")
cylinder <- read.csv(file)
cylinder <- standardise_qsm(cylinder)
str(cylinder)

## aRchi Processing Chain
file <- system.file("extdata/QSM2.csv", package = "rTwig")
cylinder <- read.csv(file)
cylinder <- standardise_qsm(cylinder)
str(cylinder)
```

`summarise_qsm` *Summarise QSM*

Description

Generates a simple QSM summary. The summary includes basic tree metrics, such as DBH, total height, and also volume and surface area arranged by totals and branching order.

Usage

```
summarise_qsm(cylinder, radius, triangulation = NULL)

summarize_qsm(cylinder, radius, triangulation = NULL)
```

Arguments

`cylinder`	QSM cylinder data frame
`radius`	Radius column name either quoted or unquoted.
`triangulation`	Optional QSM triangulation list. Only supports TreeQSM.

Details

`summarise_qsm()` and `summarize_qsm()` are synonyms.

Value

Returns a list

Examples

```
## SimpleForest Processing Chain
file <- system.file("extdata/QSM.csv", package = "rTwig")
cylinder <- read.csv(file)
cylinder <- update_cylinders(cylinder)
summarise_qsm(cylinder, radius)

# TreeQSM Triangulation
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file)
cylinder <- qsm$cylinder
cylinder <- update_cylinders(cylinder)
summarise_qsm(cylinder, radius, triangulation = qsm$triangulation)
```

---

tree_metrics *Tree Metrics*

Description

Calculates tree metrics from a QSM

Usage

```
tree_metrics(cylinder, verify = TRUE)
```

Arguments

`cylinder`	QSM cylinder data frame
`verify`	Verify QSM topology. Defaults to TRUE. If `verify = TRUE`, only topologically connected structures (e.g. a whole a tree or an individual branch) are eligible to run. This ensures all metrics are correct and verified. If `verify = FALSE`, the metrics are run, but not verified. This is strongly discouraged, but can enable the calculation of tree metrics on topologically disconnected structures.

Details

Calculates detailed tree, branch, and segment metrics from a QSM. Valid inputs require a connected QSM, which can be a whole tree or an individual branch. The outputs include all of the standard outputs from TreeQSM, and also additional variables, including, but not limited to, growth length, reverse branch order, branch segment or node relationships, and distances from twigs and the base of the tree, across various distribution metrics. Also included is a simulated point cloud of the tree, based on the QSM cylinder radii. When corrected with Real Twig, this allows for the testing and validation of point cloud diameter overestimation throughout the tree.

Value

Returns a list of tree metric data frames and a synthetic point cloud

References

Raumonen P, Kaasalainen M, Åkerblom M, Kaasalainen S, Kaartinen H, Vastaranta M, Holopainen M, Disney M, Lewis P (2013). "Fast Automatic Precision Tree Models from Terrestrial Laser Scanner Data." *Remote Sensing*, **5**(2), 491–520. doi:10.3390/rs5020491.

Hackenberg J, Spiecker H, Calders K, Disney M, Raumonen P (2015). "SimpleTree —An Efficient Open Source Tool to Build Tree Models from TLS Clouds." *Forests*, **6**(11), 4245–4294. doi:10.3390/f6114245.

Hackenberg J, Bontemps J (2023). "Improving quantitative structure models with filters based on allometric scaling theory." *Applied Geomatics*, **15**. doi:10.1007/s12518023005374.

Yang W, Wilkes P, Vicari MB, Hand K, Calders K, Disney M (2024). "Treegraph: tree architecture from terrestrial laser scanning point clouds." *Remote Sensing in Ecology and Conservation*. ISSN 2056-3485, doi:10.1002/rse2.399.

Smith D, Sperry J, Enquist B, Savage V, McCulloh K, Bentley L (2013). "Deviation from symmetrically self-similar branching in trees predicts altered hydraulics, mechanics, light interception and metabolic scaling." *The New phytologist*, **201**. doi:10.1111/nph.12487.

Examples

```
## TreeQSM Processing Chain
file <- system.file("extdata/QSM.mat", package = "rTwig")
cylinder <- import_treeqsm(file)$cylinder
cylinder <- update_cylinders(cylinder)
metrics <- tree_metrics(cylinder)
names(metrics)

## SimpleForest Processing Chain
file <- system.file("extdata/QSM.csv", package = "rTwig")
cylinder <- read.csv(file)
cylinder <- update_cylinders(cylinder)
metrics <- tree_metrics(cylinder)
names(metrics)
```

---

twigs *Twig Species Database*

Description

Database of twig radii by tree species

Usage

```
twigs
```

Format

`twigs`:

A data frame containing twig radii measurements

scientific_name The tree's genus and species

radius_mm The average twig radius in millimeters

n The twig measurement sample size

min The minimum twig radii from the samples

max The maximum twig radii from the samples

std The standard deviation of twig radii

cv The coefficient of variation of twig radii

`twigs_index` *Twig Index Database*

Description

Database of twig radii by size index

Usage

```
twigs_index
```

Format

`twigs_index`:

A data frame containing twig radii measurements

size_index The twig size index

radius_mm The average twig radius in millimeters

n The twig measurement sample size

min The minimum twig radii from the samples

max The maximum twig radii from the samples

std The standard deviation of twig radii

cv The coefficient of variation of twig radii

`update_cylinders` *Update Cylinders*

Description

Updates the QSM cylinder data in preparation for radii correction

Usage

```
update_cylinders(cylinder)
```

Arguments

`cylinder`	QSM cylinder data frame

Details

Updates and verifies parent-child cylinder relationships and calculates new variables and metrics found throughout the supported QSM software. `update_cylinders()` is required to run all rTwig functions, and is run automatically if rTwig detects a new QSM.

Value

Returns a data frame

References

Hackenberg J, Spiecker H, Calders K, Disney M, Raumonen P (2015). "SimpleTree —An Efficient Open Source Tool to Build Tree Models from TLS Clouds." *Forests*, **6**(11), 4245–4294. doi:10.3390/f6114245.

Hackenberg J, Bontemps J (2023). "Improving quantitative structure models with filters based on allometric scaling theory." *Applied Geomatics*, **15**. doi:10.1007/s12518023005374.

Examples

```
## TreeQSM Processing Chain
file <- system.file("extdata/QSM.mat", package = "rTwig")
qsm <- import_treeqsm(file)
cylinder <- qsm$cylinder
cylinder <- update_cylinders(cylinder)
str(cylinder)

## SimpleForest Processing Chain
file <- system.file("extdata/QSM.csv", package = "rTwig")
cylinder <- read.csv(file)
cylinder <- update_cylinders(cylinder)
str(cylinder)

## aRchi Processing Chain
```

```
file <- system.file("extdata/QSM2.csv", package = "rTwig")
cylinder <- read.csv(file)
cylinder <- update_cylinders(cylinder)
str(cylinder)
```

Index