

Package: ahull3D (via r-universe)

July 17, 2026

Title Fast 3D Alpha Hull with Label Propagation

Version 1.0.0

Date 2026-06-26

Description Fast 3D alpha shape (alpha hull) computation with label propagation from input points to hull vertices and faces. Uses 'CGAL' for robust geometric computations. Optimized for Light Detection and Ranging ('LiDAR') processing and tree segmentation workflows. The implementation follows the alpha shape algorithm described in Edelsbrunner et al. (1983) [<doi:10.1007/BF02579193>](https://doi.org/10.1007/BF02579193).

License MIT + file LICENSE

URL <https://github.com/DijoG/ahull3D>

BugReports <https://github.com/DijoG/ahull3D/issues>

Depends R (>= 3.5.0)

Imports Rcpp (>= 1.0.8), rgl (>= 0.100.0), Rvcg (>= 0.18.0)

Suggests RANN (>= 2.6.0), testthat (>= 3.0.0)

LinkingTo Rcpp, RcppEigen, RcppCGAL

SystemRequirements C++17

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Config/pak/sysreqs cmake libfreetype6-dev libglu1-mesa-dev make texlive libpng-dev libuv1-dev libgl1-mesa-dev zlib1g-dev

Repository <https://dijog.r-universe.dev>

Date/Publication 2026-07-08 11:32:45 UTC

RemoteUrl <https://github.com/dijog/ahull3d>

RemoteRef HEAD

RemoteSha ff1f8b0db8c0dbd5ccd81e276d2ef590b0afdc7b

Contents

ahull3D	2
Index	4

ahull3D	<i>Fast 3D Alpha Hull with Label Propagation</i>
---------	--

Description

Computes the alpha hull (alpha shape) of a 3D point cloud with optional label propagation for point classification.

Usage

```
ahull3D(points, alpha, input_truth = NULL, volume = FALSE)
```

Arguments

points	Nx3 matrix of points
alpha	Alpha value (radius parameter)
input_truth	Optional labels (point id) for each input point
volume	Compute volume (default: FALSE)

Value

A mesh object (from `rgl::tmesh3d`) with attributes: - `input_truth`: Propagated labels - `face_truth`: Labels for each face - `alpha`: Alpha value used - `volume`: Volume (if requested)

References

Edelsbrunner, H., Kirkpatrick, D., & Seidel, R. (1983). On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4), 551-559. <doi:10.1007/BF02579193>

Examples

```
# Small example for testing
# Generate random points on a sphere
pts <- matrix(rnorm(100), ncol = 3)
pts <- pts / sqrt(rowSums(pts^2))

# Compute alpha hull
mesh <- ahull3D(pts, alpha = 0.5)

# Visualize
library(rgl)
shade3d(mesh, col = "lightblue")
```

```
# Larger example with visualization (may take longer)
pts <- matrix(rnorm(300), ncol = 3)
pts <- pts / sqrt(rowSums(pts^2))
mesh <- ahull3D(pts, alpha = 0.5)

# Visualize
library(rgl)
shade3d(mesh, col = "lightblue")

# Example with labels
pts <- matrix(rnorm(60), ncol = 3)
pts <- pts / sqrt(rowSums(pts^2))
labels <- rep(1:2, each = 10)
mesh_labeled <- ahull3D(pts, alpha = 0.5, input_truth = labels)
attr(mesh_labeled, "input_truth")

# Compute volume
mesh_vol <- ahull3D(pts, alpha = 0.8, volume = TRUE)
attr(mesh_vol, "volume")
```

Index

ahul13D, [2](#)