



main

Prakash-2025_ForestPulse / Box dimension.md

Prakash-Basnet [Update Box dimension.md](#)

f457636 · 1 minute ago



253 lines (156 loc) · 12.2 KB

Preview

Code

Blame



Raw



Measuring Tree/Forest Structural Complexity with (Fractal) Box Dimension (D_b) from 3D Point Clouds

Prepared by Prakash Basnet

prakash.basnet@hawk.de; pbasnet@uni-goettingen.de

● What is this? ...It's a point or dot.

Can you tell the dimensions of this? ...Yes, it's a zero-dimensional object.

— What is this? ...It's a line.

Can you tell the dimensions of this? ...Yes, it's a one-dimensional object.



Now, what are these, and what are the dimensions of these objects? ...Of course, it's a 2-D object.

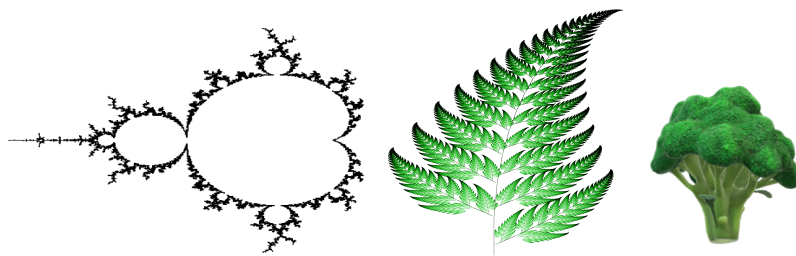
Then, tell me the dimensions of this object.



It's a simple 3-D cube.

Can we measure these objects? ...Yes, by measuring their length, width, and height.

But,

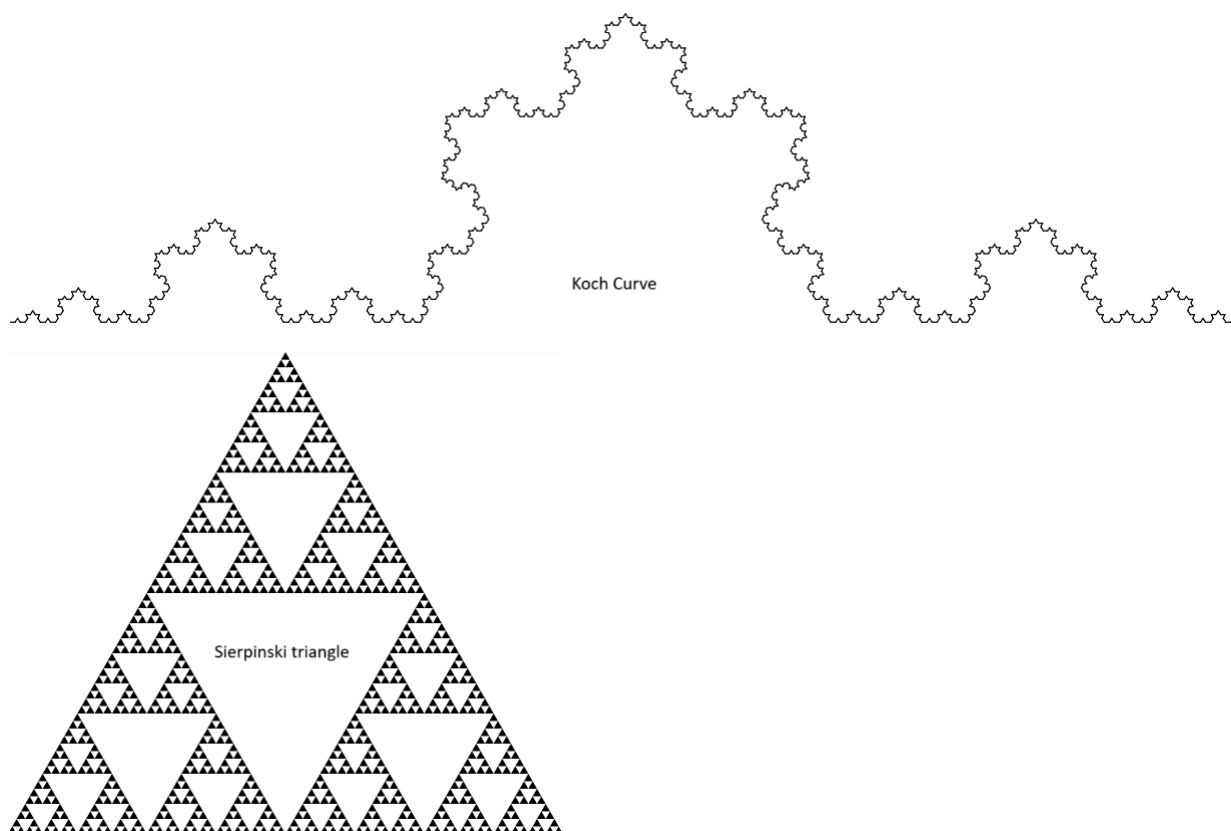


Which dimensional objects are these? How can we measure it?

These are fractal shapes, measured by their fractal dimensions. **But what is a fractal?**

“Clouds are not spheres, mountains are not cones, coastlines are not circles, and bark is not smooth, nor does lightning travel in a straight line.” - Mandelbrot (1983)

A fractal is a geometric object that shows **self-similarity**; its parts resemble the whole, no matter the scale of observation. Imagine zooming in on a fern leaf, a coastline, or a broccoli head: at each magnification, similar patterns reappear. Unlike simple geometric figures, fractals are **irregular and infinitely nested** in structure. They are found widely in nature and are best described not by whole-number dimensions (like 1D lines or 2D surfaces), but by **fractional, non-integer dimensions**, known as the **fractal dimension**. This dimension is crucial for quantifying the complexity of self-similar systems, whether exact (mathematical fractals) or statistical (natural systems).



How do you measure the fractal dimension?

The **fractal dimension** quantifies how complex and space-filling a pattern is. Unlike familiar Euclidean dimensions (1D for a line, 2D for a surface), fractal dimensions can take **non-integer values** between whole numbers.

Several methods exist to estimate fractal dimension, but the **box-counting method** is the most widely used. In this approach, the object is covered with boxes of a certain size, and the number of boxes required to contain the shape is counted. This process is repeated with progressively smaller boxes.

If the logarithm of the number of boxes is plotted against the logarithm of the box size, the result is typically a straight line. The **slope of this line** gives the fractal dimension: steeper slopes correspond to more intricate, space-filling structures.

Why are we discussing this here?

Because **trees are fractal structures**.



Trees exhibit self-similar branching patterns: smaller branches resemble the overall structure of the tree. While we can measure conventional forestry attributes such as height, DBH, basal area, crown dimensions, and volume, these do not capture **overall structural complexity**. To do this, we turn to fractal analysis—specifically the **box dimension**.

From Euclidean to Fractal Dimension (Quick Intuition)

- **0-D:** point
- **1-D:** line
- **2-D:** plane/filled region
- **3-D:** solid volume
- **Fractal dimension** fills the gaps between integers and quantifies **how space-filling** a structure is.

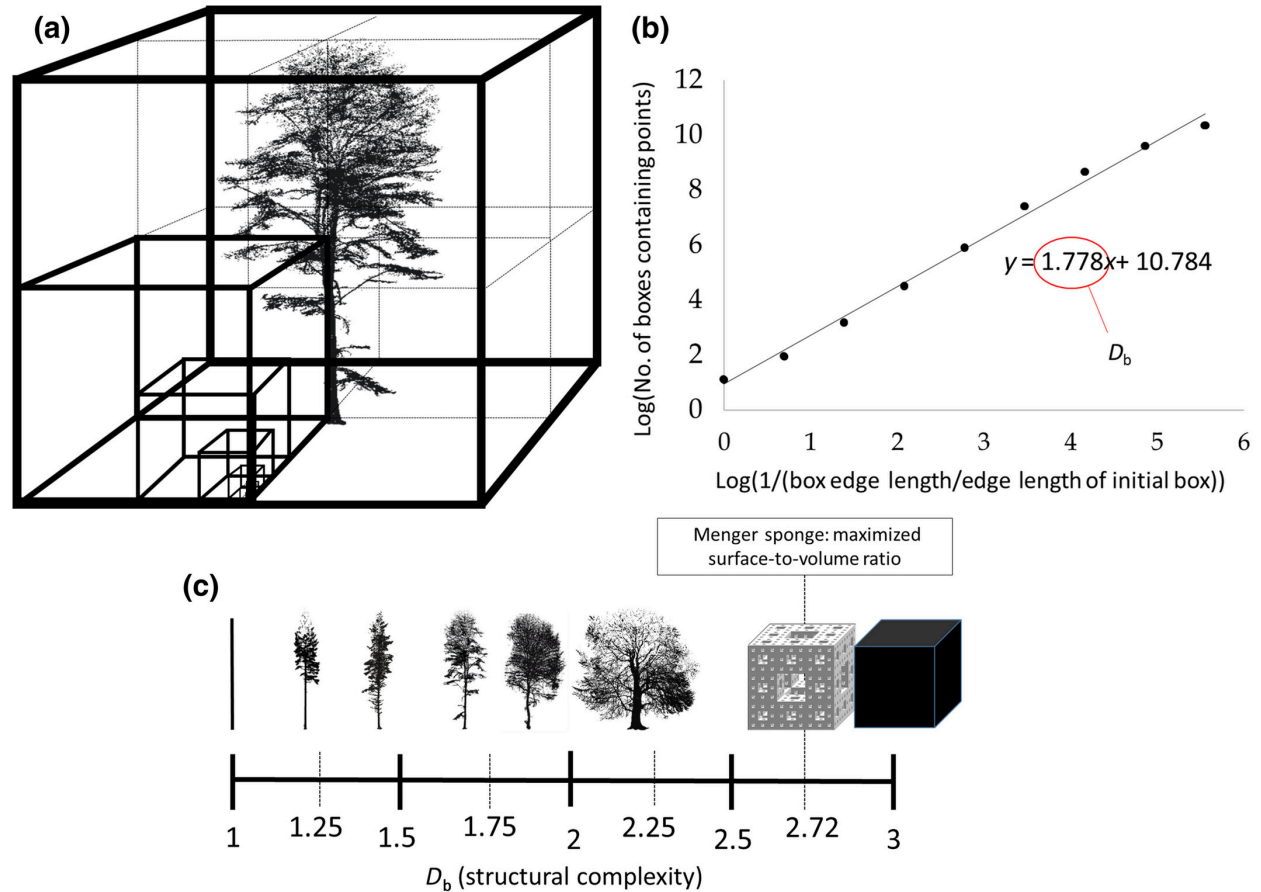
A classic example is the **Menger sponge**, which has a box dimension ≈ 2.7268 . Despite its infinite surface area, it has zero volume, an extreme case of fractal geometry.

Box Dimension

The **box dimension** is a method of fractal analysis used to quantify the *structural complexity* of an object, often derived from 3D point cloud data. The idea is simple:

- Cover the object with boxes of decreasing size.
- Count how many boxes are required to contain the shape.
- Plot the logarithm of the box count against the logarithm of box size.

If the relationship is linear, the **slope** of that line gives the **box dimension**.



Formally, if the box size is (ϵ) and the number of occupied boxes is ($N(\epsilon)$):

$$[N(\epsilon) \propto \epsilon^{-D_b} \Rightarrow \log N(\epsilon) = D_b, \log(1 / \epsilon) + c]$$

Where:

- (D_b) is the **box dimension**.
- For 3D point clouds:
 - ($D_b \approx 1$): line-like structures
 - ($D_b \approx 2$): sheet-like structures
 - ($D_b \approx 3$): volume-filling structures
- Trees typically have (D_b) **between 1.0 and 2.2** (method and data dependent).
- A higher (R^2) of the fit indicates stronger self-similarity across scales.

Data sensitivities

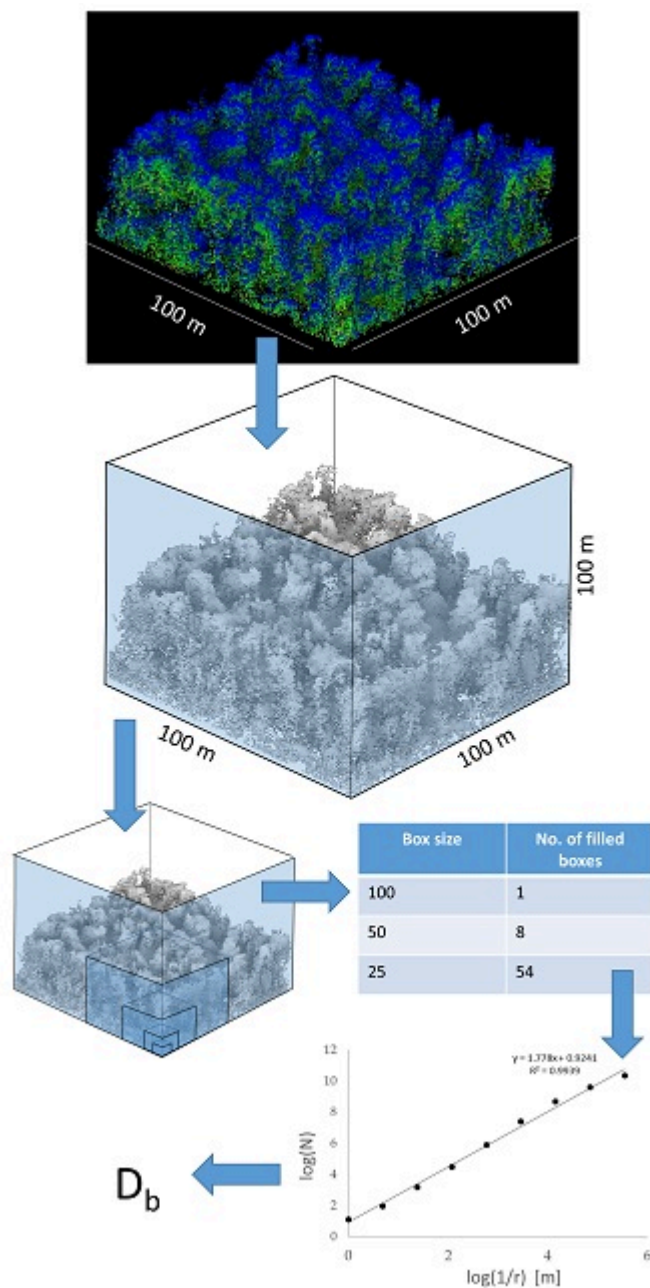
Box dimension estimates depend on:

1. **Point density/resolution**
2. **Occlusion or shadowing**
3. **Extent of the scanned scene**
4. **Scale interval chosen for fitting**

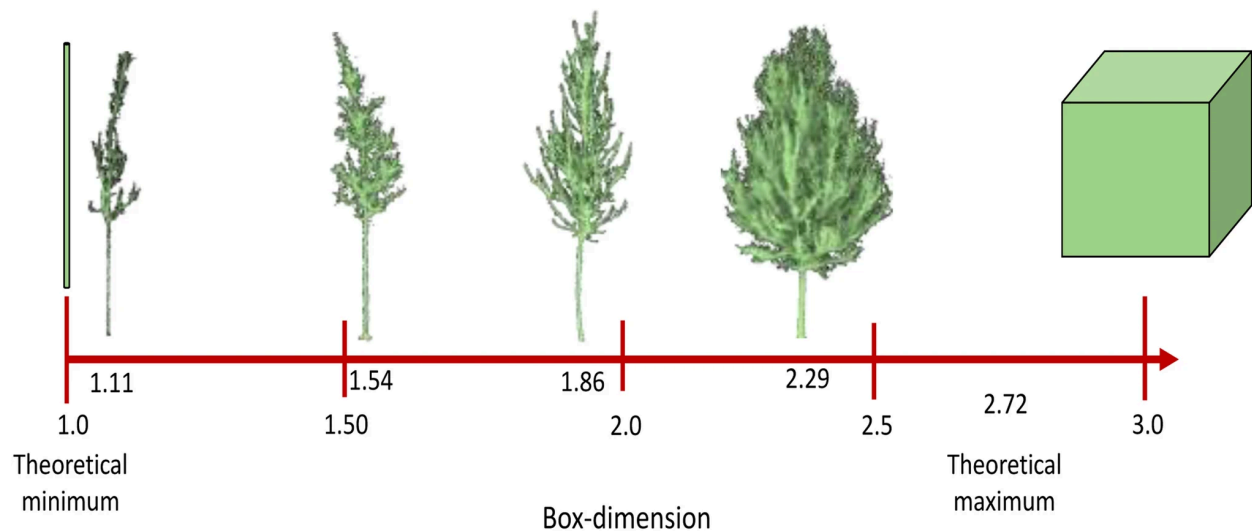
High-quality point clouds (low occlusion, ~0–1 cm spacing) are optimal for robust estimates.

Box Dimension vs. Voxel Counting

In practice, box-counting in 3D is implemented as **voxel counting**. Voxels are the 3D equivalent of pixels, small cubic volume elements. A point cloud is discretized into a voxel grid, and the occupied voxels are counted at multiple scales.



This process converts a 3D model or point cloud into a grid of voxels, each labeled as "occupied" or "empty."



Interpretation

- **Box dimension ranges from 1 (line) to 3 (solid cube).**
- A dimension of **2.72** corresponds to the Menger sponge (infinite surface, zero volume).
- **Most trees fall between 1.0 and 2.2**, reflecting their branching complexity.
- Low resolution → oversimplification (underestimates (D_b)).
- High occlusion → too few boxes at small scales (also biases (D_b)).

Thus, careful preprocessing and high-quality scans are critical for meaningful box-dimension analysis.

How to measure it in reality?

To apply this concept to forests, we calculate the **box-dimension** ((D_b)) as a measure of tree or stand structural complexity using the **rTwig** package in R.

The algorithm for (D_b) was developed initially in *Mathematica* (Wolfram Research, Champaign, IL, USA) (Seidel, 2018; Ehbrecht, 2019; Basnet, 2025). It integrates **all elements of a scanned scene** into a single value, thereby fully leveraging the potential of 3D laser scanning.

In short:

- The point cloud of a tree or stand is enclosed in boxes of decreasing size.
- For each box size, the number of occupied boxes is counted.
- The scaling relationship between box size and number of boxes gives the **box dimension** ((D_b)).

This workflow is implemented in the **rTwig** R package, making it straightforward to estimate structural complexity directly from normalized point cloud data.

rTwig R package

The **rTwig** package provides the function `box_dimension()` to calculate the fractal box dimension ((D_b)) from a 3D point cloud.

Usage

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



Arguments

cloud: A point cloud matrix $n \times 3$ (X, Y, Z). Non-matrices are automatically converted to a matrix

lowercutoff: The smallest box size determined by the point scaping of the cloud in meters. Defaults to 1 cm

rm_int_box: Logical, whether to remove the initial (largest) box from the fit. Defaults to FALSE.

plot: Visualization options: "2D", "3D", or "ALL". FALSE disables plotting. Default is FALSE.

Important preprocessing step: Before running `box_dimension()`, ensure that the point cloud is:

1. In XYZ matrix format.
2. Normalized to ground, meaning the minimum ground Z-value is set to 0.

This can be verified with the `las_check()` function in `lidR`, as shown below.


```
las_check(las)
.....
- Checking attribute population...

  ① 'PointSourceID' attribute is not populated
  ① 'ScanDirectionFlag' attribute is not populated
  ① 'EdgeOfFlightline' attribute is not populated
  Indicates the file, flight line, or sensor source that each point came
  from (important if multiple sensors/flightlines are merged.
  not populated means all points have PointSourceID = 0 (default). This
  means the provider didn't encode the flightline info.

- Checking ground classification... yes

- Checking normalization... no

.....
```



This function reports whether ground classification and normalization have been applied, which is critical for meaningful box-dimension analysis.

```
#### R Script ####
library(lidR)
library(lidRviewer)
library(dplyr)
#install.packages("rTwig")**
library(rTwig)
```




```
# Data access
url <- "https://owncloud.gwdg.de/index.php/s/3z4bYtZ5jdJN8W1/download"
download.file(url, destfile = "uls_goewa.laz", mode = "wb")

# Read data, check, and pre-process with lidR
data <- readLAS("uls_goewa.laz")
print(data)
las_check(data)

las <- normalize_height(las = data,
                        algorithm = tin(),
                        use_class = 2)
las_check(las)

view(las)

las@data[Z<0, ] # Here, options are either to remove all or assign all
to 0; however...

# Forest structural complexity (Box dimension)

cloud = las@data[Z>0.5, 1:3] # Here, all points above 0.5 meter and only
X,Y,z coordinates

db <- box_dimension(cloud = cloud,
                    lowercutoff = 0.01,
                    rm_int_box = FALSE,
                    plot = FALSE )

str(db)

# Box Dimension (slope)
db[[2]]$slope
db[[2]]$r.squared # show similarity

# Visualization
# 2D Plot
box_dimension(las@data[, 1:3], plot = "2D")
# 3D Plot
box_dimension(las@data[, 1:3], plot = "3D")
```

References

- Mandelbrot, B. B. (1983). The Fractal Geometry of Nature. W.H. Freeman.
- Seidel, D. et al. (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. <https://doi.org/10.1002/ece3.5281>

- Seidel, D., Annighöfer, P., Ehbrecht, M., Magdon, P., Wöllauer, S., & Ammer, C. (2020). Deriving Stand Structural Complexity from Airborne Laser Scanning Data—What Does It Tell Us about a Forest? *Remote Sensing*, 12(11), 1854. <https://doi.org/10.3390/rs12111854>
- Dorji, Y., Schuldt, B., Neudam, L., Dorji, R., Middleby, K., Isasa, E., & Körber, K. et al. (2021). *Trees*, 35(4), 1385–1398. <https://doi.org/10.1007/s00468-021-02124-9>
- Basnet, P., Das, S., Hölscher, D., Pierick, K., & Seidel, D. (2025). Drivers of forest structural complexity in mountain forests of Nepal. *Mountain Research and Development*, 45(1), R1–R10. <https://doi.org/10.1659/mrd.2024.00009>
- rTwig vignette (Box Dimension): <https://cran.r-project.org/web/packages/rTwig/vignettes/Box-Dimension.html>