

# **The Little Handbook for MTPPR**

**Multi-Trait Point Pattern Reconstruction**

Chris Wudel

2024-08-07

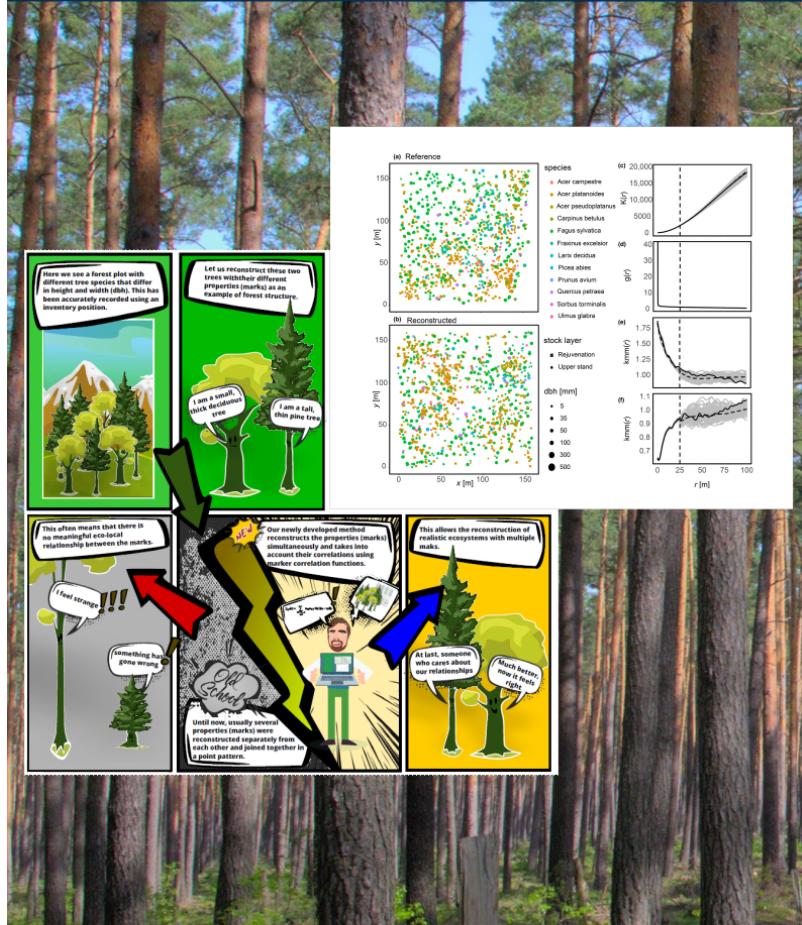
# **Table of contents**

<b>Welcome</b>	<b>3</b>
<b>1 Introduction</b>	<b>5</b>
<b>2 Simple application example</b>	<b>10</b>
<b>3 Further application examples</b>	<b>14</b>
<b>4 Specialized application</b>	<b>18</b>
<b>5 Summary and references</b>	<b>23</b>
<b>References</b>	<b>24</b>

# Welcome

**Multi-Trait Point Pattern Reconstruction (MTPPR)** is an advanced method for analyzing and modeling spatial patterns across various scientific disciplines such as ecology, biology, and geography. MTPPR allows the reconstruction of spatial point distributions, considering multiple associated traits or attributes. This handbook provides an overview of the key concepts, steps, and applications of MTPPR.

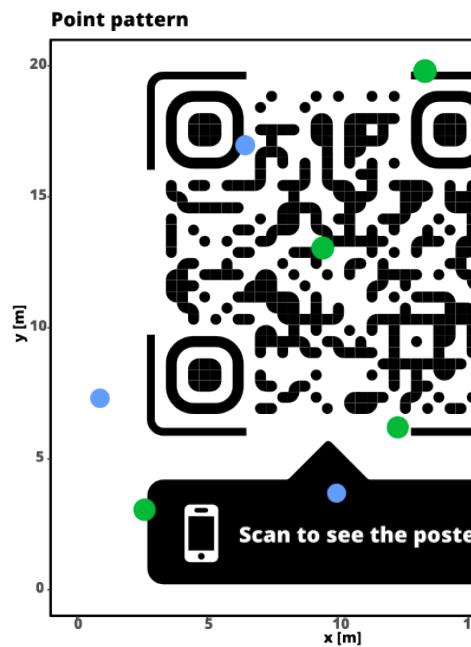
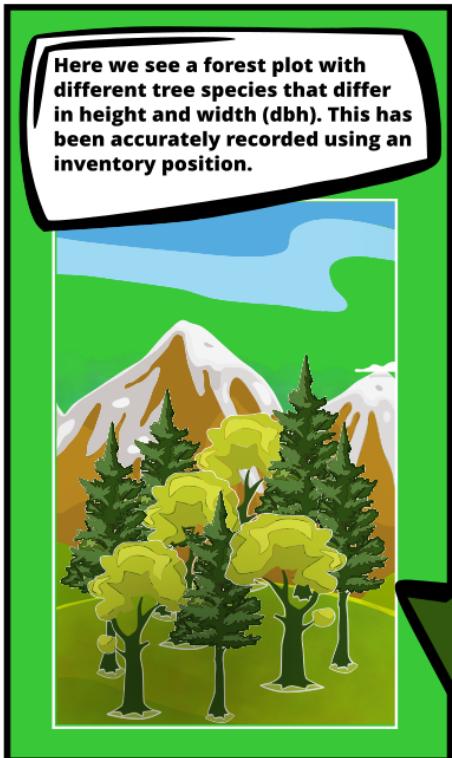
If you would like to find out more about multi-trait point pattern reconstruction, read Wudel et al. (2023) or consult the references.



# 1 Introduction

Multi-Trait Point Pattern Reconstruction (MTPPR) is a statistical method used to analyze and reconstruct spatial patterns. Unlike traditional methods, MTPPR enables the reconstruction of spatial point distributions while considering multiple associated traits (marks). Previous methods for reconstructing point patterns either only consider a single mark or multiple marks independently, neglecting their correlations. MTPPR employs various second-order summary statistics of point pattern analysis, such as the pair correlation function and the mark correlation function.

**Figure:** This illustration depicts the issue in reconstructing dot patterns where the correlations between the marks are not considered. This results in unrealistic proportions, as illustrated here by the disproportionate trees.



This method is crucial in fields like ecology, biology, and geography, where understanding the spatial distribution of entities and their traits is essential. By considering both the spatial locations and the various traits of points, MTPPR allows researchers to uncover complex interactions and dependencies that influence the arrangement and dynamics of the systems under study.

For example, in ecology, MTPPR can be used to reconstruct the distribution of different tree species in a forest, taking into account attributes such as age, height, and health status. This provides insights into ecological processes such as competition, habitat preferences, and the impact of environmental factors. Similarly, in epidemiology, MTPPR can help understand the spread of diseases by correlating spatial data with demographic and environmental traits.

## **Key Concepts**

### **1. Point Patterns:**

- Spatial data points representing the positions of objects or events within a defined area.
- Examples: Locations of trees in a forest, distribution of animal burrows, spread of disease cases.

### **2. Traits:**

- Additional attributes or characteristics associated with each point.
- Examples: Tree species, age, height, health status.

### **3. Multi-Trait Analysis:**

- Simultaneous consideration of multiple attributes to understand their influence on spatial arrangement.
- Can reveal complex interactions and dependencies between traits.

### **4. Reconstruction:**

- Generation of statistically similar point patterns for various applications through reconstruction algorithms.
- Examples: Creating null model patterns for spatial point pattern analysis and constructing artificial datasets suitable for initializing forest ABMs and other stand simulators.

## **Steps in MTPPR**

### **1. Data Collection:**

- Gathering spatial data along with associated traits through field surveys, remote sensing, or other methods.

### **2. Spatial Analysis:**

- Analyzing the spatial arrangement of points using techniques such as Ripley's K-function, pair correlation function, or spatial autocorrelation.
- Assessing the distribution (clustered, random, regular) and the influence of spatial scale.

### **3. Trait Analysis:**

- Evaluating the distribution and correlation of traits using methods like the mark correlation function.

### **4. Model Construction:**

- Developing models that integrate spatial and trait information using statistical models (e.g., spatial point process models).
- Aim: Reconstructing the underlying processes leading to the observed multi-trait point pattern.

### **5. Reconstruction and Validation:**

- Reconstructing point patterns based on the developed models to predict spatial arrangements under various scenarios.
- Validating models by comparing simulated patterns with actual observations.

### **6. Interpretation:**

- Interpreting results to gain insights into ecological, biological, or geographical processes.

## **Applications**

- **Ecology:**

- Generating realistic and statistically similar spatial patterns.
- Understanding the coexistence and competition among plant species.
- Studying the spatial distribution of animal populations and their habitat preferences.

- **Epidemiology:**

- Analyzing the spread of diseases based on environmental factors and population characteristics.

- **Urban Planning:**

- Assessing the spatial distribution of urban features (e.g., green spaces, buildings) and their associated traits (e.g., building types, land use).

## 2 Simple application example

The following example demonstrates a simple application of **Multi-Trait Point Pattern Reconstruction** (MTPPR) by Wudel et al. (2023). It illustrates reconstruction using fictitious datasets that incorporate multiple traits simultaneously. The required libraries must be loaded beforehand.

```
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
library(spatstat)
```

Please note that the maximum number of iterations has been set to  $\text{max\_steps} = 10000$  and  $n_{\text{repetitions}} = 3$  in this example to keep computation time low. No weighting of different summary statistics has been performed, which may be necessary for different applications (e.g., forest stands) to achieve optimal results. For real-world applications, it is advisable to adjust these parameters accordingly. Additionally, in the vignette,  $\text{verbose} = \text{FALSE}$  has been set to minimize print output. We recommend using the default setting  $\text{verbose} = \text{TRUE}$  when running the code to see progress reports.

The next step is to load the point pattern, here is an example of a random point pattern with several marks to show the structure of the data used.

```
xr <- 500
yr <- 1000
N <- 400
y <- runif(N, min = 0, max = yr)
x <- runif(N, min = 0, max = xr)
species <- sample(c("A", "B"), N, replace = TRUE)
diameter <- runif(N, 0.1, 0.4)
random <- data.frame(x = x, y = y, dbh = diameter, species = factor(species))
marked_pattern <- as.ppp(random, W = owin(c(0, xr), c(0, yr)))
```

The point pattern must contain the following data An x and y coordinate, a metric mark (in metres) and a nominal mark defined as a factor. The order must be respected. Now the reconstruction with several marks can be started with the following code. Note that the maximum number of iterations has been set to *max\_steps* = 10000 to keep the computation time for this example to a minimum. For an application, this value should be increased according to the number of points in the pattern.

```
reconstruction <- reconstruct_pattern_multi(marked_pattern, n_repetitions = 1, max_steps = 10000)
```

```
> Progress: || iterations: 0 || Simulation progress: 0% || energy = 0.06578 || energy improvement = 0.06578
> Progress: || iterations: 5000 || Simulation progress: 50% || energy = 0.00218 || energy improvement = 0.00218
> Progress: || iterations: 10000 || Simulation progress: 100% || energy = 0.00216 || energy improvement = 0.00216
```

As a result, you will receive a list containing a variety of information, for example, the reference pattern, the reconstructed pattern, the number of successful actions, the energy development and much more. If you wish to perform several reconstructions of the same reference pattern, you must increase n\_repetitions to the desired number.

```
reconstruction_2 <- reconstruct_pattern_multi(marked_pattern, n_repetitions = 2, max_steps = 10000)
```

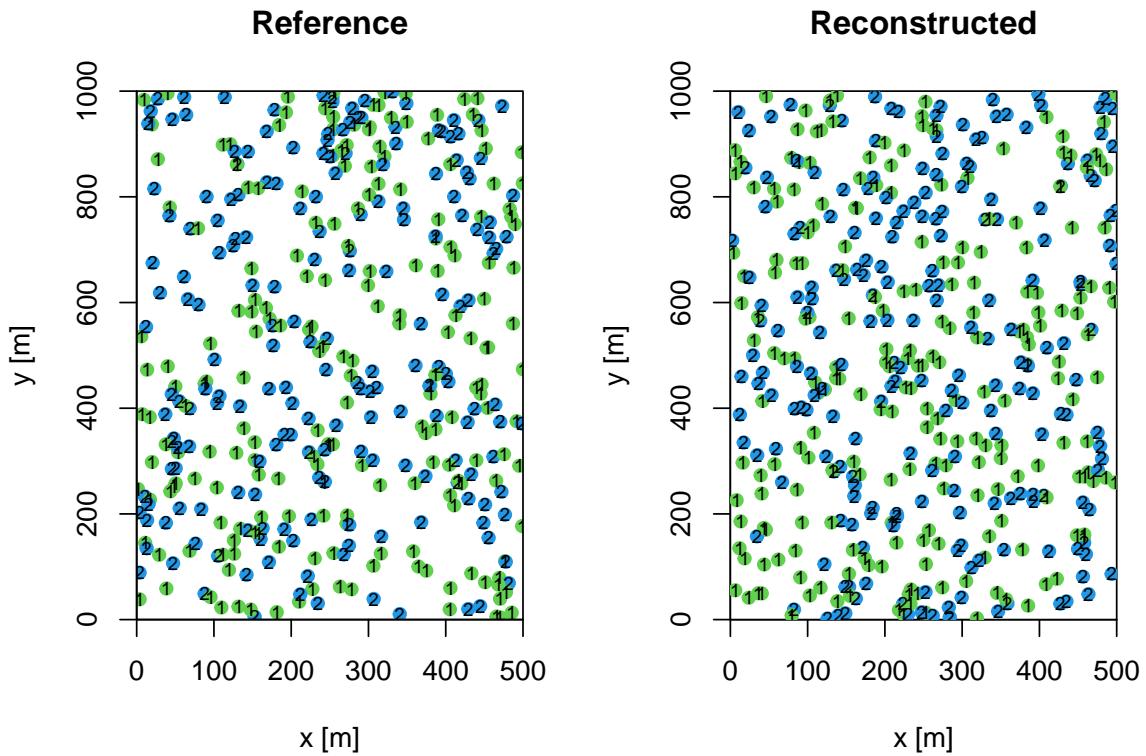
```
> Progress: reconstruction_1 || iterations: 0 || Simulation progress: 0% || energy = 0.13038
> Progress: reconstruction_1 || iterations: 5000 || Simulation progress: 50% || energy = 0.00159
> Progress: reconstruction_1 || iterations: 10000 || Simulation progress: 100% || energy = 0.00148
```

```
> Progress: reconstruction_2 || iterations: 0 || Simulation progress: 0% || energy = 0.0746
> Progress: reconstruction_2 || iterations: 5000 || Simulation progress: 50% || energy = 0.00159
> Progress: reconstruction_2 || iterations: 10000 || Simulation progress: 100% || energy = 0.00148
```

To activate a visualisation of the reconstruction that shows the changes in the pattern at the relevant time, you must proceed as follows.

```
reconstruction_3 <- reconstruct_pattern_multi(marked_pattern, n_repetitions = 1, max_steps = 10000)
```

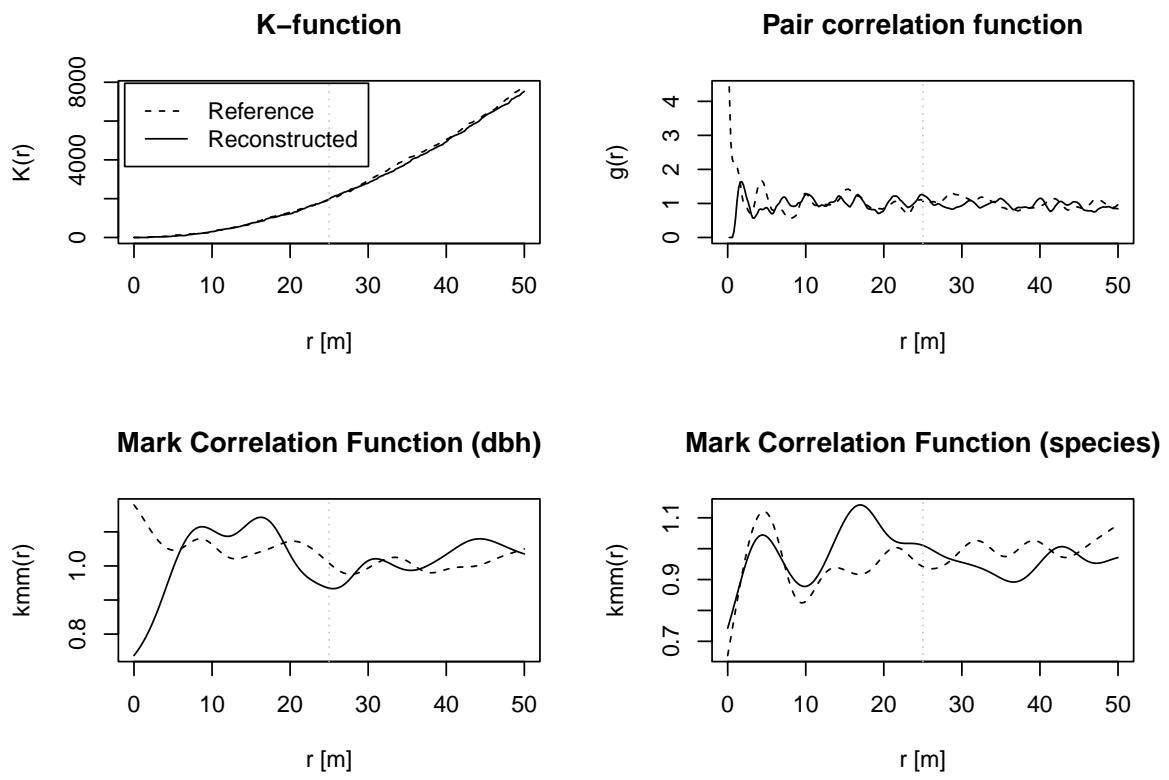
```
> Progress: || iterations: 0 || Simulation progress: 0% || energy = 0.12173 || energy improvement = 0.12173
> Progress: || iterations: 5000 || Simulation progress: 50% || energy = 0.00159 || energy improvement = 0.00159
> Progress: || iterations: 10000 || Simulation progress: 100% || energy = 0.00148 || energy improvement = 0.00148
```



Finally, you can use the following function to view different summary statistics of the reference pattern (black line) compared to the reconstructed pattern (grey line). For this, however, the listed libraries must be loaded additionally.

```
plot(reconstruction)
```

Progress in the creation of the figures: 100%



## 3 Further application examples

In the following, further examples of the simple application of **Multi-Trait Point Pattern Reconstruction** (MTPPR) by Wudel et al. (2023) are presented. These are standard applications, meaning that reconstructions are performed without enlargements or reductions of the reconstruction pattern and without edge correction.

First, the necessary functions and packages will be loaded. If individual R packages are not installed, install them as follows: `install.packages("package name")`.

```
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
library(spatstat)
library(ggplot2)
library(patchwork)
```

Now you need to define which dataset you want to use. There are 2 real datasets and 4 generated datasets available. To select a dataset, define `x` with the name of the dataset you want to use (`x <- 'Dataset Name'`). The datasets are available under 'Records for download'.

```
x <- "random" ## The following sets can be imported:
## Real datasets:
##      "VERMOS_project"
##      "Northwest_German_Forest_Research_Institute"
##      "Marteloscope_data_from_the_by_the_Chair_of_Forest_institute"
## Simulated patterns:
##      "random"
```

```

##      "regular"
##      "cluster_size5"
##      "cluster_size5_and_random"
## to do this, declare x with the corresponding name in "
data <- data_import(x)
W <- data[[2]]
data <- data [[1]]

```

The following parameters are predefined and can be varied arbitrarily in an application file (Application of the Multi-trait Point pattern reconstruction.R) where you can use this code, available for download under the Application folder. It should be noted that for optimal results, the parameter max\_steps should be at least approximately ten times the number of points in the pattern, and the parameter for weights (w\_markcorr) of individual summary statistics needs to be adjusted according to different scenarios. Here, a small number of steps was chosen to save computation time.

```

marked_pattern <- as.ppp(data.frame(data), W = W)
marked_pattern$marks$dbh..mm.<-marked_pattern$marks$dbh..mm.*0.001
xr <- marked_pattern>window$xrange
yr <- marked_pattern>window$yrange

reconstruction <- reconstruct_pattern_multi(
  marked_pattern,
  n_repetitions      = 1,
  max_steps          = 10000,
  no_change          = 5,
  rcount             = 250,
  rmax               = 25,
  issue              = 1000,
  divisor            = "r",
  kernel_arg         = "epanechnikov",
  timing              = TRUE,
  energy_evaluation = TRUE,
  show_graphic       = FALSE,
  Lp                 = 1,
  bw                 = 0.5,
  sd                 = "step",
  steps_tol          = 1000,
  tol                = 1e-4,
  w_markcorr         = c(m_m=1,one_one=0, all=1, m_all=1, all_all=1, m_m0=1, one_one0=0, all0=1),
  prob_of_actions    = c(move_coordinate = 0.4, switch_coords = 0.1, exchange_mark_one = 0.1,
  k                  = 1,

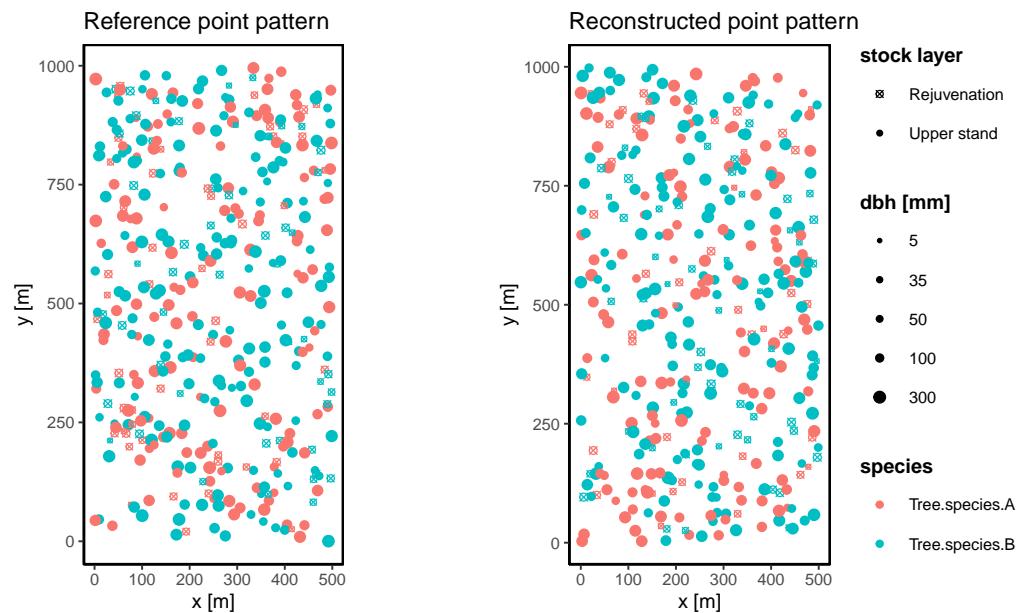
```

```
w_statistics      = c(),
verbose          = TRUE)
```

As a result, you will receive a list with a variety of information, such as the reference pattern, the reconstructed pattern, the number of successful actions, the energy development, and much more. To illustrate the results, first compare the reference pattern with the reconstructed pattern.

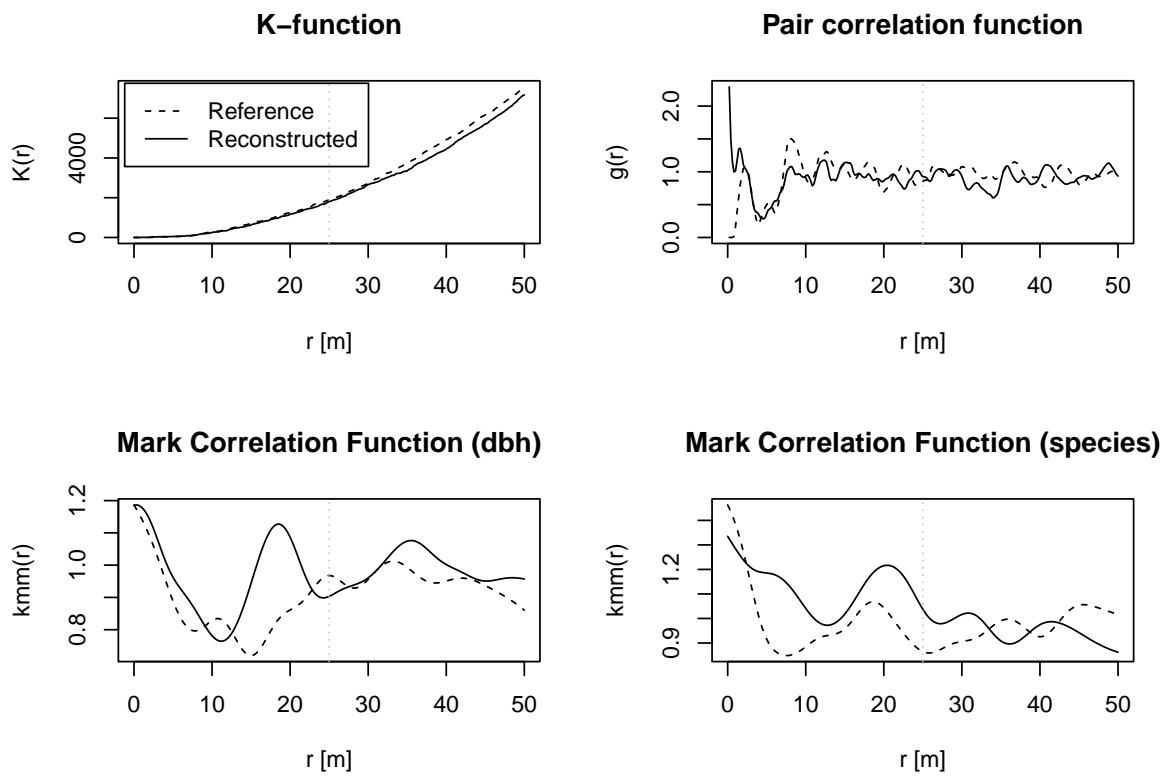
```
vis_pp(reconstruction)
```

Multi-trait point pattern reconstruction  
Reconstruction of two marks (dbh and tree species)



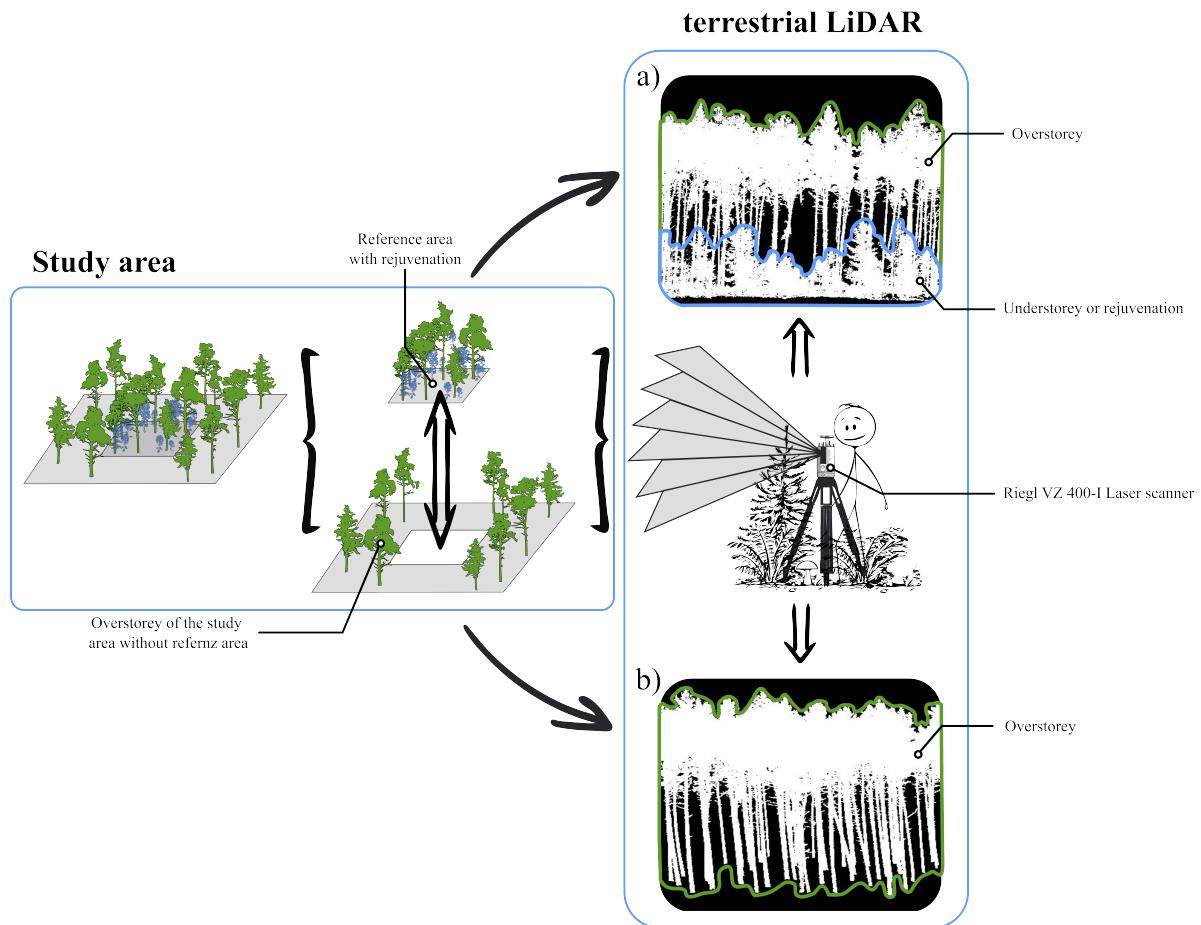
Finally, you can use the following function to compare various summary statistics of the reference pattern (black line) with the reconstructed pattern (grey line). Another function (plot\_sum\_stat) is capable of displaying these summary statistics in a single diagram for multiple repetitions of the reconstructions ( $n\_repetitions > 1$ ), and it is used in the previously mentioned application file (Application of the Multi-trait Point pattern reconstruction.R). For simplicity, this functionality has been omitted here.

```
plot(reconstruction)
```



## 4 Specialized application

This represents a **novel workflow for predicting forest regeneration**. Initially, the canopy of a forest area and a small portion of the regeneration were captured using terrestrial laser scanning. Based on spatial statistical tree data and correlations between tree characteristics, a point pattern reconstruction method was developed, building upon the work of Wudel et al. (2023). This method facilitates the prediction of regeneration across the entire area with high accuracy.



**Figure:** Conceptual presentation of the new innovative workflow for data acquisition and reconstruction of forest regeneration (LiDAR = Light Detection and Ranging). a) Recording

*regeneration on a small area and b) recording of the upper stand trees at the whole forest area.*

The following code can be used to execute the workflow using an available dataset. First, the necessary functions and packages are loaded.

```
source("https://raw.githubusercontent.com/ChrisWudel/Multi-trait-point-pattern-reconstruction")
library(spatstat)
library(ggplot2)
```

Anschließende wird der Datensatz aufgerufen der in Github im Orner Records zu finden ist und published and freely available at the following link: <https://zenodo.org/records/10550778> (Meyer and Wudel, 2024).

```
url <-"C:/Users/admin/Desktop/New Paper/step-by-step application/Data/TLS_core.csv"
data <- read.csv(url,sep = ",", stringsAsFactors= TRUE)
data$dbh <- as.numeric(data$dbh)
```

Below are the parameter settings chosen for this example. These may and should be adjusted for other datasets to achieve good results. Note: the maximum number of steps has also been reduced here to limit computation time.

```
W <- owin(c(0, 100),c(0, 100))
core_window <- owin(c(35, 65),c(35, 65))

marked_pattern <- as.ppp(data.frame(data), W = W)
marked_pattern$marks$dbh <- marked_pattern$marks$dbh*0.001
xr <- marked_pattern>window$xrange
yr <- marked_pattern>window$yrange
obs_window = owin(c(xr),c(yr))
```

```

reconstruction <- reconstruct_pattern_multi(
marked_pattern,
fixed_points      = NULL,
edge_correction   = TRUE,
n_repetitions     = 1,
max_steps         = 10000,
no_change         = 5,
rcount            = 250,
rmax              = 25,
issue              = 5000,
divisor            = "r",
kernel_arg         = "epanechnikov",
timing             = TRUE,
energy_evaluation = TRUE,
show_graphic       = FALSE,
Lp                 = 1,
bw                 = 0.5,
sd                 = "step",
steps_tol          = 10000,
tol                = 1e-4,
w_markcorr        = c(m_m=0,one_one=1500, all=1, m_all=1, all_all=1, m_m0=1500, one_one0=1,
prob_of_actions    = c(move_coordinate = 0.3, switch_coords = 0.1, exchange_mark_one = 0.1, e
k                  = 1,
w_statistics       = c(),
is.fixed           = function(p) 35 <= p$x & p$x <= 65 & 35 <= p$y & p$y <= 65 | p$mark[,"dbh
verbose            = TRUE)

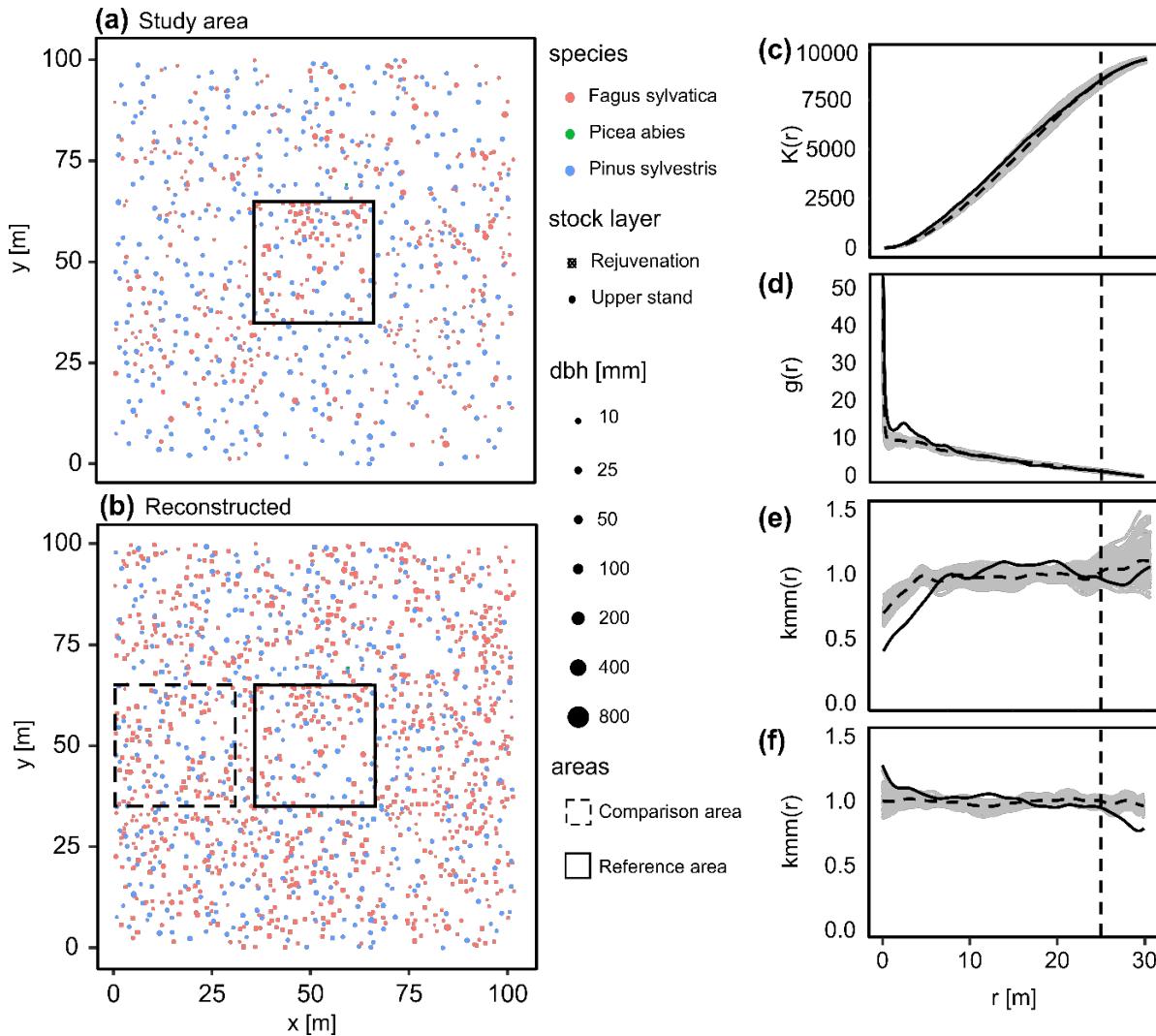
```

```

> Progress: || iterations: 0 || Simulation progress: 0% || energy = 42545.59257 || energy in
> Progress: || iterations: 5000 || Simulation progress: 50% || energy = 17458.88297 || energ
> Progress: || iterations: 10000 || Simulation progress: 100% || energy = 4535.81704 || ener

```

The following figure shows the results of the workflow, which are from an example not derived from the current run. The figure has been created retroactively and is not generated by the function provided. Additionally, in this simulation, significantly more simulation steps were conducted, which would take too long to render for this handbook.



**Figure:** Results of the 100 reconstructions of the dataset described above; (a) reference pattern with core area, black outlined area (b) an example of a reconstruction in which the core area is outlined in black and the comparison area is outlined in black dashed lines. Summary statistics (c)–(f) were generated using the R package ‘spatstat’: (c)  $K$  function; (d)  $pcf$ ; (e)  $mcf$  of the species, where the test function has value if both species coincide and if not; (f)  $mcf$  of the diameter, the test function being the usual product. For distances up to  $= 25$  m (indicated by the vertical dashed line in (c)–(f)). The black solid lines represent the reference curve and the grey solid lines represent the 100 reconstructions. The dashed black line is the mean of the 100 reconstructions.

It is evident that the tree distributions in these two areas are similar. For instance, *Fagus sylvatica* is observed with high intensity only in specific regions of the reference area, and this has been successfully reproduced. Furthermore, it is noticeable that smaller gaps in the

reference area, where canopy trees are spaced similarly, are not occupied by regeneration plants in the comparison area. Also includes the previously mentioned summary statistics of all 100 reconstructions (c)-(f). These statistics demonstrate that the reconstructions perform well (gray lines = reconstructions, solid black line = reference). Within the range up to 25 m, which was considered during reconstruction, deviations from the reference in the K-function (c) are minimal. The pcf (d) shows a slight systematic deviation. The mcf of the species (e) exhibit only minimal discrepancies. The mcf of diameters (f) show smaller discrepancies in the same close range as the pcf. However, deviations become significantly larger and more diverse once the 25 m range is exceeded. This highlights the effectiveness and impact of the mcf's in the reconstruction method.

## 5 Summary and references

This Handbook is intended to facilitate the use of MTPPR and to illustrate its broad application through various examples. The method is still under development and will be regularly updated with changes.

If you have any questions, please contact the author:

**Chris Wudel**, Email: [chris.wudel@tu-dresden.de](mailto:chris.wudel@tu-dresden.de)

Or one of the following co-authors:

- **Dr. Robert Schlicht** (co-developer), Email: [robert.schlicht@tu-dresden.de](mailto:robert.schlicht@tu-dresden.de)
- **Prof. Dr. Uta Berger** (supervised and revised), Email: [uta.berger@tu-dresden.de](mailto:uta.berger@tu-dresden.de)
- **Dr. Franka Huth** (data collection and forestry consultation), Email: [franka.huth@tu-dresden.de](mailto:franka.huth@tu-dresden.de)
- **Nora Meyer** (data collection TLS), Email: [nora.meyer@tu-dresden.de](mailto:nora.meyer@tu-dresden.de)
- **Alexandra Wehnert** (data collection), Email: [alexandra.wehnert@tu-dresden.de](mailto:alexandra.wehnert@tu-dresden.de)

# References

Further references, for example regarding the method description, can be found in the respective publications.

- Fichtner, I., van der Maaten-Theunissen, M., 2022. Marteloscope data of the experimental plot, Naundorf 710 b1 (Saxony, Germany 2020). Zenodo. <https://doi.org/10.5281/zenodo.7147868>
- Meyer, N., Wudel, C., 2024. Terrestrial laser scan data of a experimental plot in Forstamt Billenhagen, 340 a31 (Mecklenburg- Vorpommern, Germany 2023). <https://doi.org/10.5281/zenodo.10550778>
- Wudel, C., Huth, F., Wehnert, A., 2022a. Individual tree data of the temporary test plot Neusorgefeld 5138 - VERMOS project. Zenodo. <https://doi.org/10.5281/zenodo.7157076>
- Wudel, C., Schlicht, R., Berger, U., 2023. Multi-trait point pattern reconstruction of plant ecosystems. Methods Ecol. Evol. 14, 2668–2679. <https://doi.org/https://doi.org/10.1111/2041-210X.14206>
- Wudel, C., Schlicht, R., Berger, U., 2022b. Point-pattern-reconstruction. Zenodo. <https://doi.org/10.5281/zenodo.7228767>