Localization Analyzer for Nanoscale Distributions (LAND)

Manual

Under construction

Contents

1 Introduction

LAND is a software package to quantitatively analyze single molecule localization microscopy (SMLM) data. The package includes algorithms to analyze the clustering of molecules and their spatial distribution. In addition, it contains algorithms specifically designed to analyze the nuclear nanostructure. LAND is designed to evaluate large sample sizes and data with high emitter densities. It is intended to support researchers as a tool

- for a quick initial analysis of the experiments
- to optimize the analysis parameters
- for routine applications with an optimized set of parameters
- to proceed after inital analysis with additional specialized algorithms.

Due to the large number of possible SMLM experiments, custom-tailored pre- and post-processing routines may be necessary. Examples for different applications are shown in **??**. For the experienced user who intend to extend the capabilities of LAND with custom routines, we provide an interface to access its core functionality (**??**).

1.1 Overview

LAND is programmed in Matlab. However, to speed up processing some functions are implemented in C++. As input it takes single molecule coordinates provided as a .csv, .txt or .mat file. These coordinates can be obtained with any localization software that generates a list of coordinates from the raw SMLM image stack (overview of available software packages: *http://bigwww.epfl.ch/smlm/software/index.html*).
Outputs are saved as .mat file or can be exported as .csv file. A detailed explanation of the design of the software is given in **??**.
Implemented routines:

- nearest neighbor analysis (NN-Analysis)
- distance analysis
- Ripley's function
- radial density function (RDF)
- Density-Based Spatial Clustering of Applications with Noise (DBSCAN)
- conformation and texture of nuclear nanostructure (SMLM-ConText)

1.2 Requirements

To run LAND the following programs and toolboxes are required:

- Matlab R2014b or newer
  - Statistics and Machine Learning Toolbox
  - Image Processing Toolbox

External dependencies:

- multiWaitbar (*link to repository at Matlab File Exchange*)
- (optional) Andrea Tagliasacchi's kdtree (*link to repository at Github*)

A copy of *multiWaitbar* is included in this distribution, a separate download and installation is not necessary. For faster computation LAND's algorithm utilize a k-d tree included in Matlab's Statistics and Machine Learning Toolbox. Since it's performance is quite low we recommend to install Andrea Tagliasachhi's kdtree.

Hardware:
Beside the minimum system requirements to run Matlab, a minimum of 8 GByte RAM are recommended.

1.3 Installation

- download the software package from https://github.com/Jan-NM/LAND
- extract LAND-master.zip
- copy the generated LAND-master directory into your default work folder
- right click on LAND-master in Matlab's current folder panel, go to *Add to Path* and click on *Selected Folders and Subfolders*
- type *startClusterAnalysis* into Matlab's command window to start the user interface or just use the command window as described in **??**.

To install Andrea Tagliasacchi's kdtree, perform the following steps:

- download the following files from https://github.com/ataiya/kdtree/tree/master/toolbox
  - KDTree.h
  - kdtree_ball_query.cpp
  - kdtree_build.cpp
  - kdtree_delete.cpp
  - MyHeaps.h
- copy these a files into a newly created folder *...\LAND-master\utilities \ataiya_kdtree \...*
- download and install a Matlab-supported C++ compiler and configure the MEX environment (for details see https://de.mathworks.com/help/matlab/matlab_external/what-you-need-to-build-mex-files.html)

- open *...\LAND-master\utilities \ataiya_kdtree \...* in Matlab's current folder panel
- execute the following commands
  - mex -setup C++
  - mex kdtree_build.cpp
  - mex kdtree_ball_query.cpp
  - mex kdtree_delete.cpp

1.4 Quick Start Guide

LAND can be used via an user interface or directly via the command line. This sections serves as a quick start guide for both of them. We recommend to use the user interface for batch processing and the command line for data exploration and parameter optimization. A detailed description is shown in **??**.

1.4.1 Input Format

LAND can read .mat, .txt or .csv files. The files should be formatted column-wise i.e. each row should represent a localization signal. The data should contain at minimum the 2D or 3D coordinates of the localization signals. Localization precision and frame numbers are optional. A default column order can be entered. Internally LAND uses the following format: x-position (column 1), y-position (column 2), z-position (column 3), x-localization precision (column 4), y-localization precision (column 5), z-localization precision (column 6), frame number (column 7). By default (without any file conversion) LAND can read .mat files in the Orte format: x-position (column 2), y-position (column 3), z-position (column 11), x-localization precision (column 4), y-localization precision (column 5), z-localization precision (column 12), frame number (column 9).

1.4.2 User Interface

To start the user interface, execute *startClusterAnalysis* in the command line. The user interface will pop up. In the file settings panel select the location of your files. The files should be in .mat, .txt or .csv format and should contain at minimum the column-wise 2D or 3D coordinates of the single molecules. Optionally the input files may contain the corresponding localization precision and a frame ID. Since many column formats exists, the corresponding columns can be selected using the *Column Selector* button. Next, click in the *File Selector* list on the files you would like to evaluate. Set up your evaluation settings i.e. select data dimension, random data evaluation, the algorithm and its parameters. Finally, press the *Start* button. When the calculation is finished the data will be saved as .mat file in folder denoted in the *Output Directory* panel. To load the files into Matlab, ensure that LAND is added to the path and double-click on the .mat file.

1.4.3 Command Line

To use LAND from the command line, load a .mat file in the Orte format (subsection 1.4.1) into Matlab or use the supplied FileConverter class. Type

```
variableName = ClusterAnalysis(localizationFile, additionalArguments);
```

into the command line. The first argument *localizationFile* is mandatory. ClusterAnalysis can take up to 5 optional input arguments:

- 1. argument: name or number of the sample (default: '001')
- 2. argument: dimension (default: 2)
- 3. argument: algorithm to calculate the local density (default: 'averageDensity')
- 4. argument: additional parameter for 3. argument (default: [])
- 5. argument: random data with complete spatial randomness (default: true)

For example, if your localization file is called testExperiment in the workspace folder, type
`cell01 = ClusterAnalysis(testExperiment);`
into the command window. This will prepare everything with default settings for further evaluation. An object with the name *cell01* will appear in the workspace. Double-click on the name will open the object and its associated settings. To execute one of the evaluation routines type
`variableName.functionName(parameters);`
For example
`cell01.DBSCAN(30, 6, 0, 1000, 0);`
will execute DBSCAN on the dataset with radius=30 nm, minimum number of points=6, random data=false, maximum diameter of clusters=1000 nm, show plot=false.