The code is an implementation of the Nystrӧm-based spectral clustering with the K-nearest neighbour-based sampling (KNNS) method **(Pang et al. 2021)**. It is aimed for individual tree segmentation using airborne LiDAR point cloud data.

**When using the code, please cite as:**

Yong Pang, Weiwei Wang, Liming Du, Zhongjun Zhang, Xiaojun Liang, Yongning Li, Zuyuan Wang (2021) Nystrӧm-based spectral clustering using airborne LiDAR point cloud data for individual tree segmentation, International Journal of Digital Earth

**Code files:**

‘segmentation.py’: the main function, including deriving local maximum from Canopy Height Model (CHM);

‘VNSC.py’: other functions for the algorithm, including mean-shift voxelization, similarity graph construction, KNNS sampling, eigendecomposition, k-means clustering, as well as the computation and writing of individual tree parameters.

**Key parameters:**

When using the code, users can adjust the values of local maximum window, *gap* (the upper limit of the number of final clusters), knn (the number of k-nearest neighbours in the similarity graph) and quantile in meanshift method based specific data characteristics. Currently, the value of local maximum window is 3m ×3m, the value of gap is defined as the 1.5 times of the local maximum detected from CHM. Parameter *knn* can be defined as a constant value (40 in the code) based on the data characteristics, or be determined through the relationship between it and the number of voxels. The default setting of quantile in meanshift method is the average density of point clouds. More details can be found in **Pang et al. (2021).**

**Test data:**

‘ALS\_pointclouds.txt’: point cloud data;

‘ALS\_CHM.tif’: CHM of the point cloud data;

‘Reference\_tree.csv’: field measurements for algorithm validation. The position was measured using differential GNSS. The tree height of each tree in this file is obtained by regression estimation.

**Outputs:**

‘Data\_seg.csv’: coordinate of each point (x, y, z) as well as its cluster label after segmentation;

‘Parameter.csv’: individual tree parameters (TreeID, Position\_X, Position\_Y, Crown, Height) based on the calculation described in **Pang et al. (2021)**.

**Notice:**

The code is only for the preliminary segmentation. Generally, key parameters are adjusted to produce reasonable individual tree or over-segmentation. For complicate forest stand, post-processing (including merging, filtering, and secondary segmentation) is recommended to improve segmentation results.

The codes were tested in Python 3.6. Dependencies: numpy, mahotas, gdal and sklearn.