DISTRIBUTED RANDOM FOREST CLASSIFICATION OF URBAN MOBILE MAPPING DATA

Christian Alis
University College London, UK

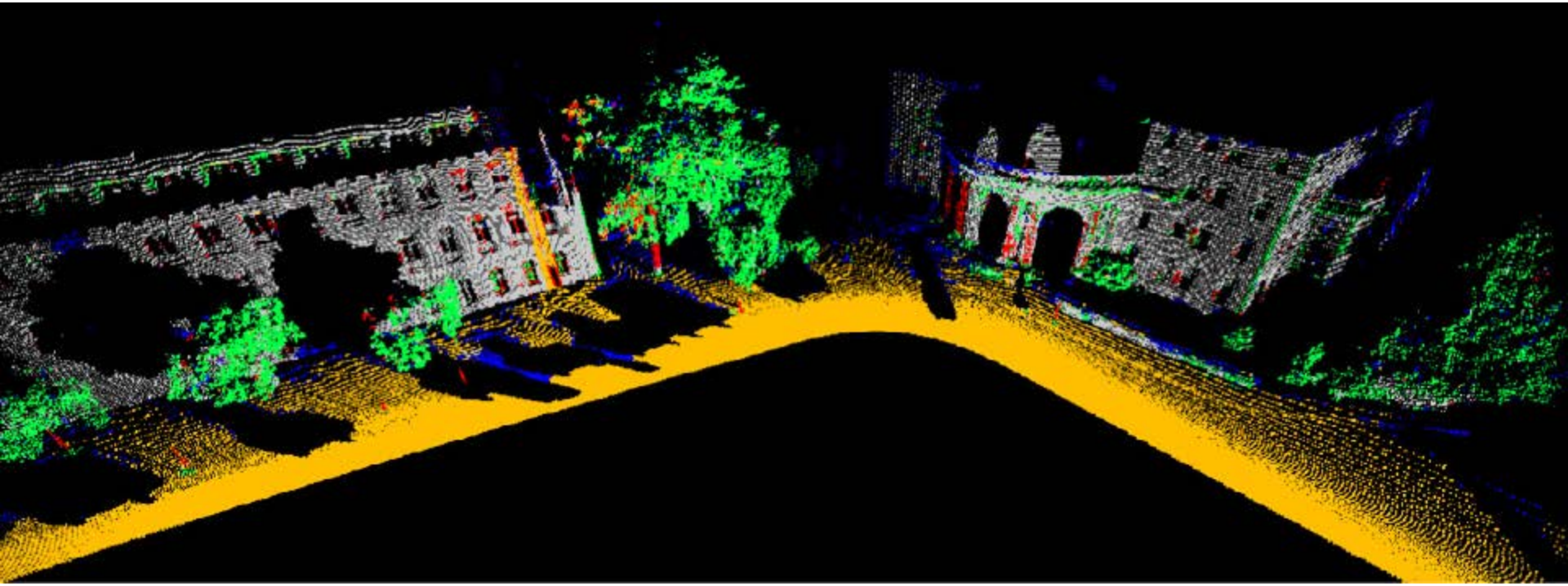
IQmulus, Final Workshop

Bergen, September 21, 2016

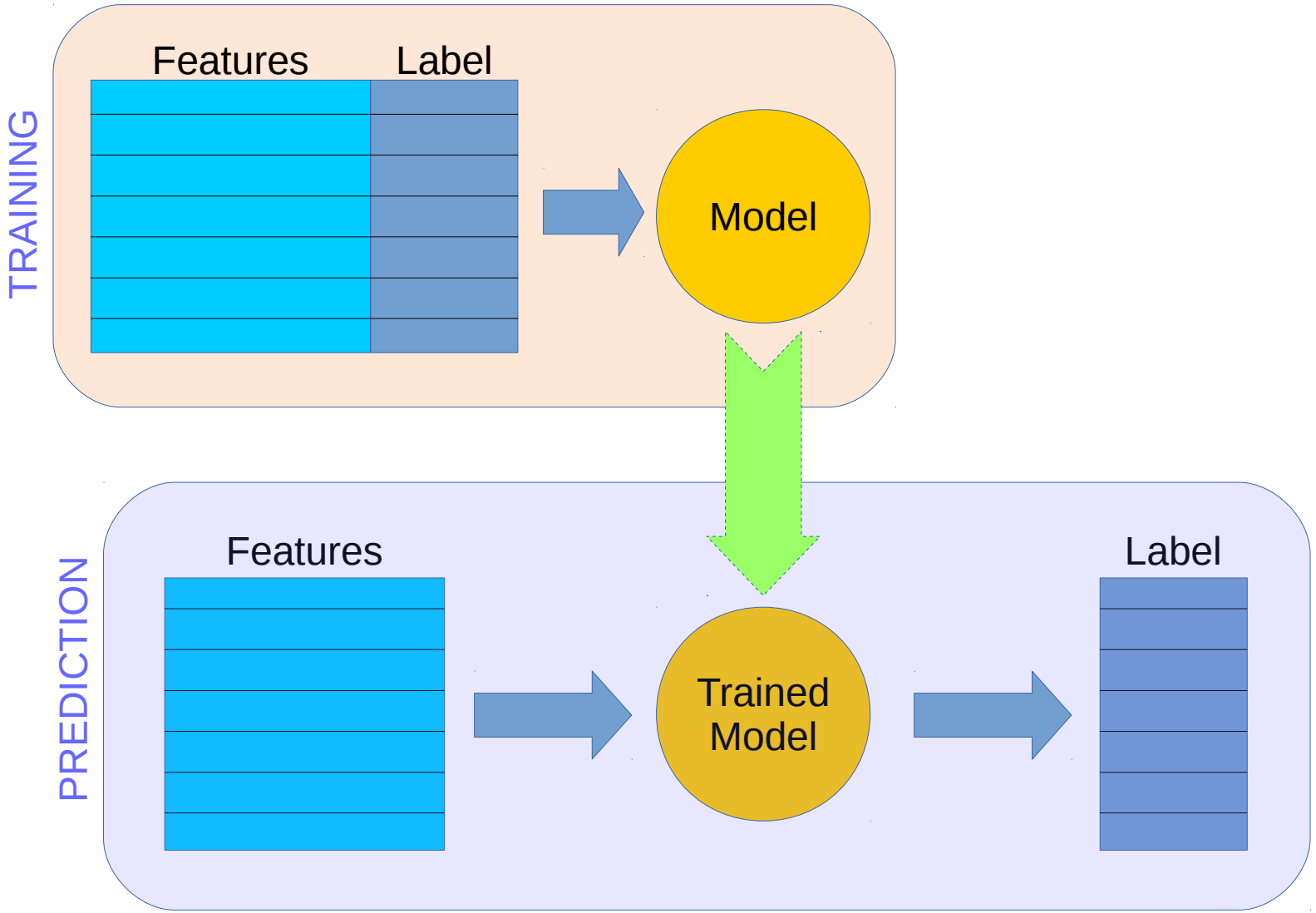
The research leading to these results has received funding from the European Union Seventh Framework Programme (*FP7/2007-2013*) under *grant agreement* n° 318787.

Distributed Random Forest classification of urban mobile mapping data

Christian Alis, Jan Boehm, Kun Liu
Dept of Civil, Environmental and Geomatic Engineering
University College London



M. Weinmann, B. Jutzi, S. Hinz, and C. Mallet, ‘Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers’, ISPRS Journal of Photogrammetry and Remote Sensing, vol. 105, pp. 286–304, Jul. 2015.


TRAINING
Features
Label
Model
PREDICTION
Features
Trained
Model
Label

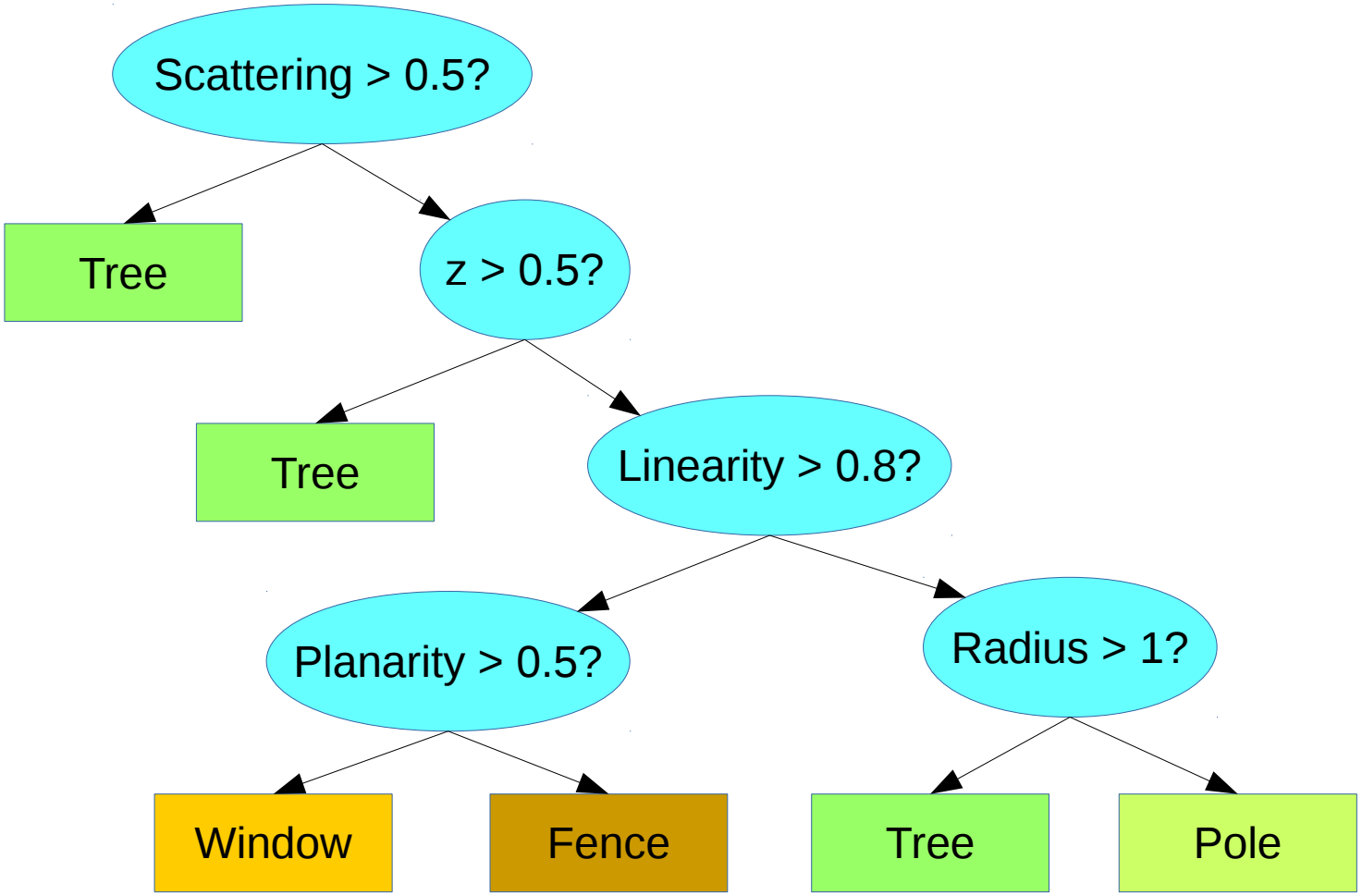
Point cloud classification features

- Linearity
- Planarity
- Scattering
- Omnivariance
- Anisotropy
- Eigenentropy
- Eigenvalue sum
- Change of curvature
- Z value

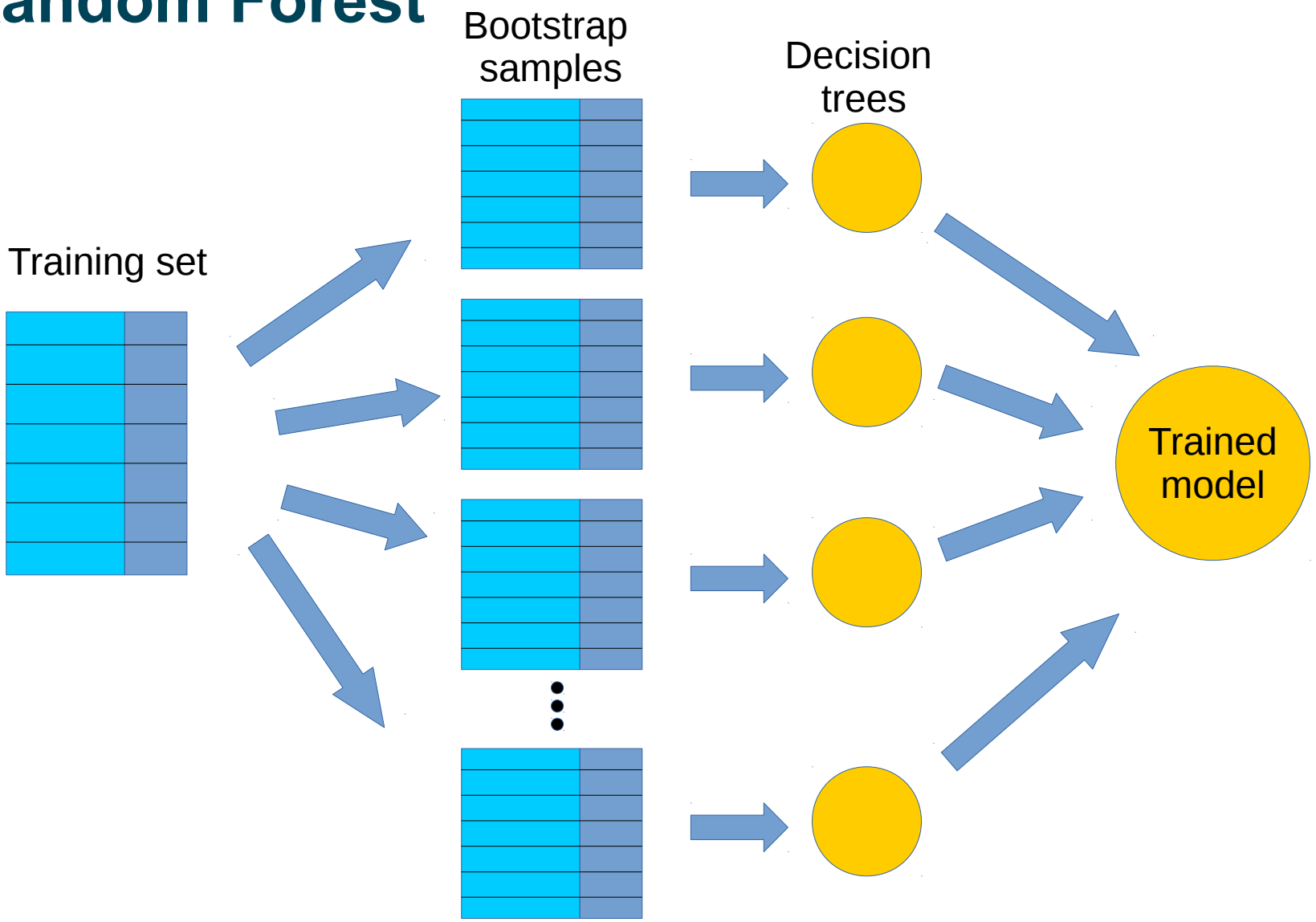
- Radius
- Density
- Verticality
- Z range
- Z standard deviation
- 2D radius
- 2D density
- 2D eigenvalue sum
- 2D eigenvalue ratio

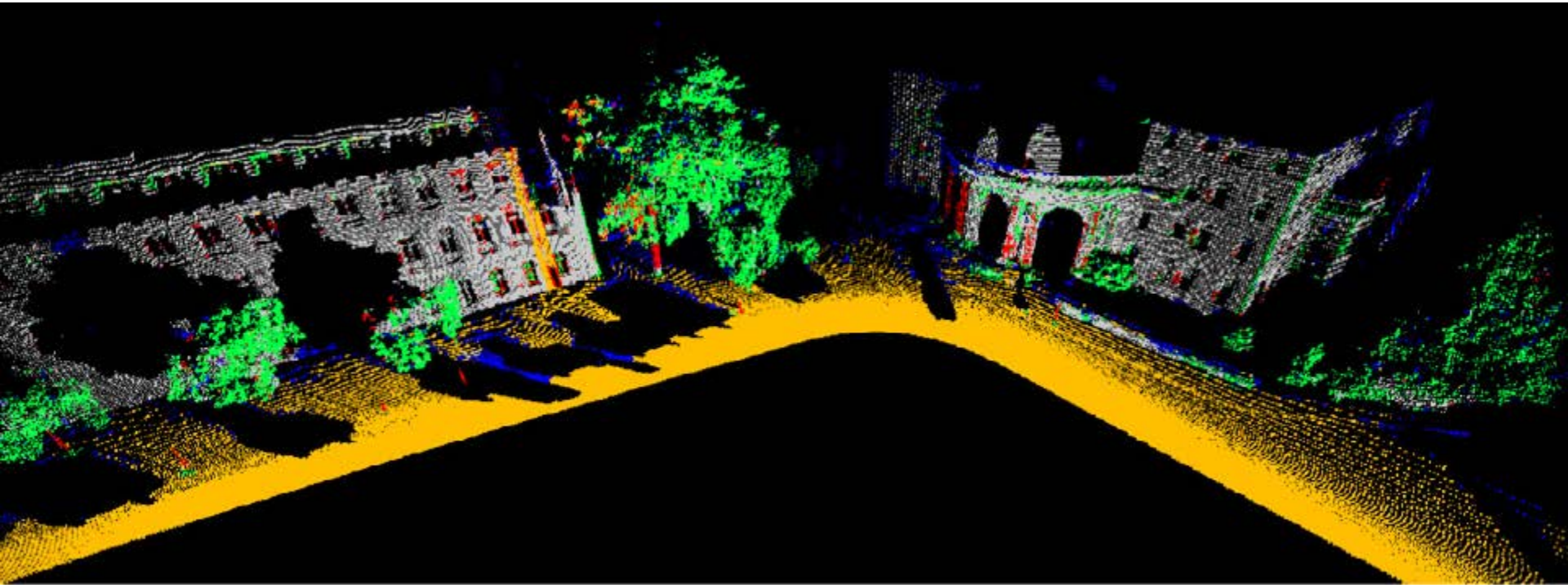
Features are computed based on neighbourhood of the point

Decision Tree



Random Forest





What if the point cloud consists of millions or billions of points?

M. Weinmann, B. Jutzi, S. Hinz, and C. Mallet, ‘Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers’, ISPRS Journal of Photogrammetry and Remote Sensing, vol. 105, pp. 286–304, Jul. 2015.

https://commons.wikimedia.org/wiki/File:BigData_2267x1146_white.png

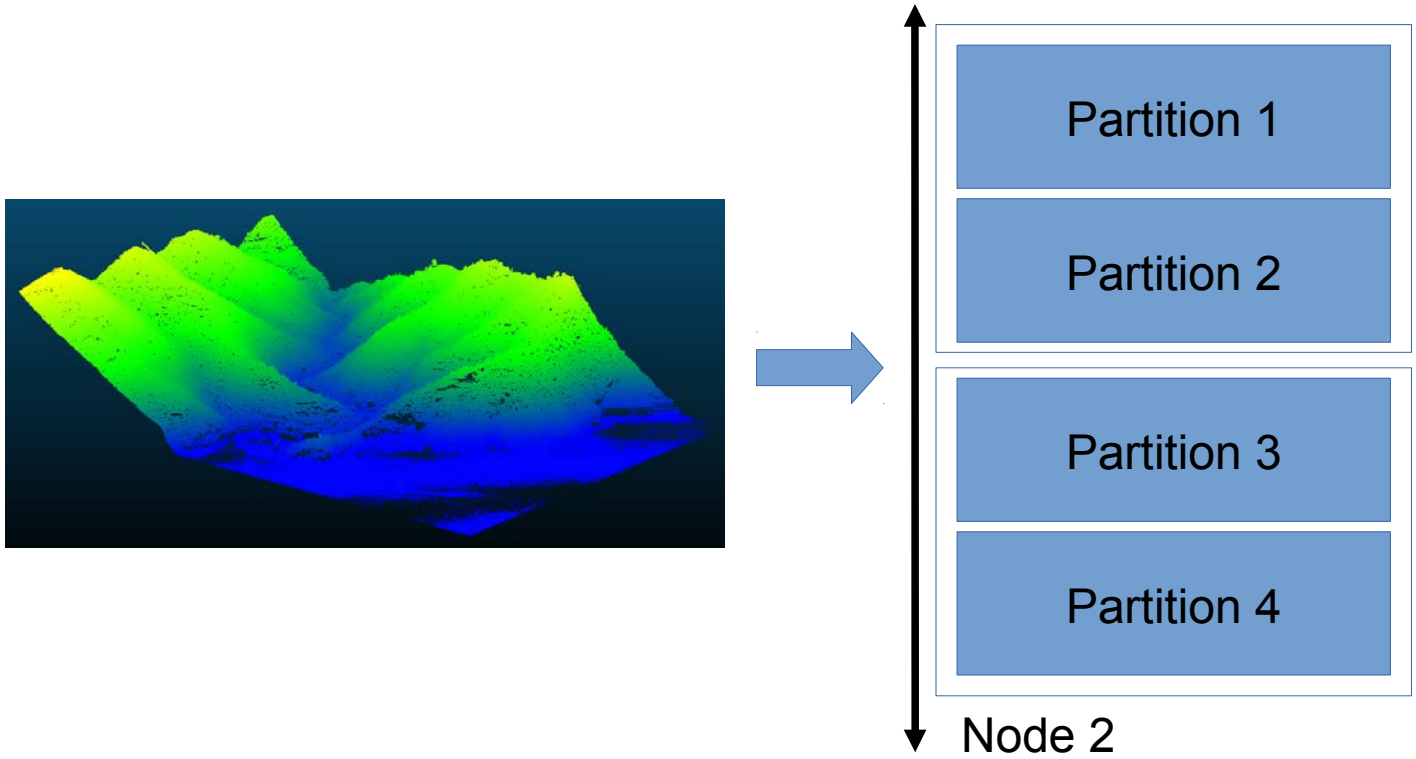
Apache Spark

- A processing engine for distributed computation
- Able to scale from single to multiple machines
- Allows coding using different languages
- Very active development and user community
- Allows processing of non-spatial data
- Has a machine learning library with random forest

- No built-in support for point clouds
- Partitioning

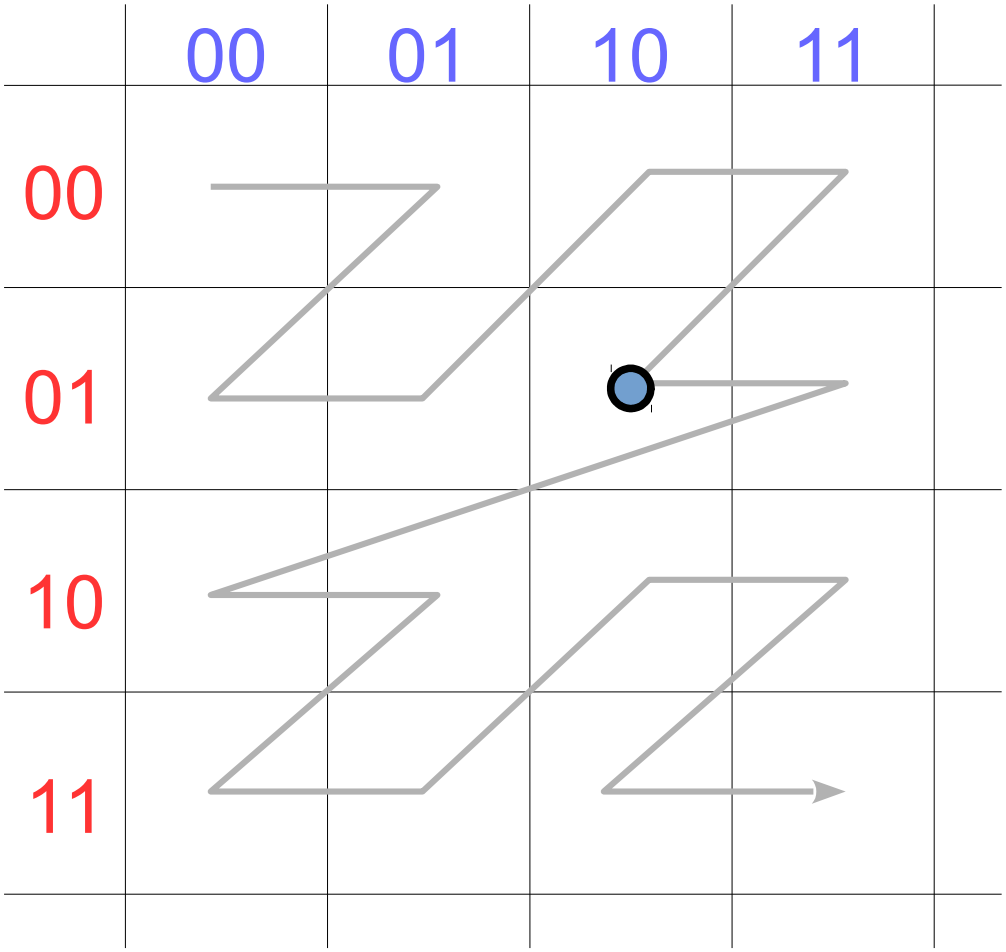
The challenge of data partitioning

- Not all data would fit on a single machine
- Network I/O is too slow to transfer data as needed



Z-order indexing

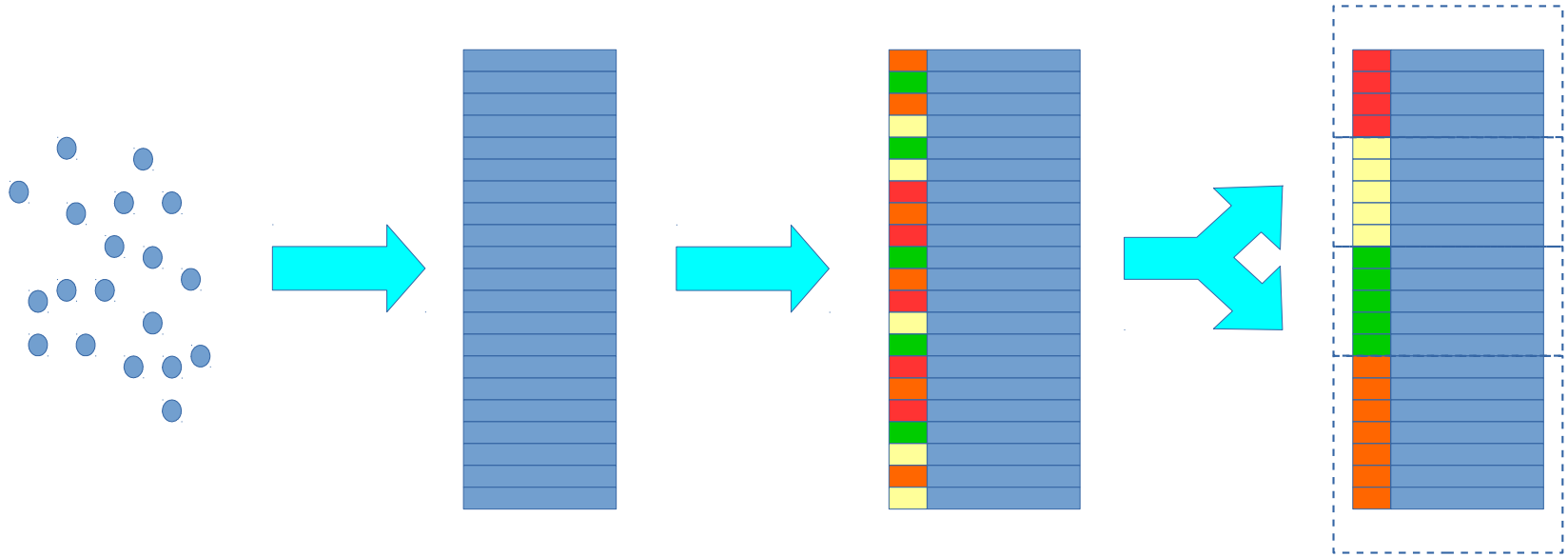
A space-filling curve with adjustable level of detail



1 0
0 1

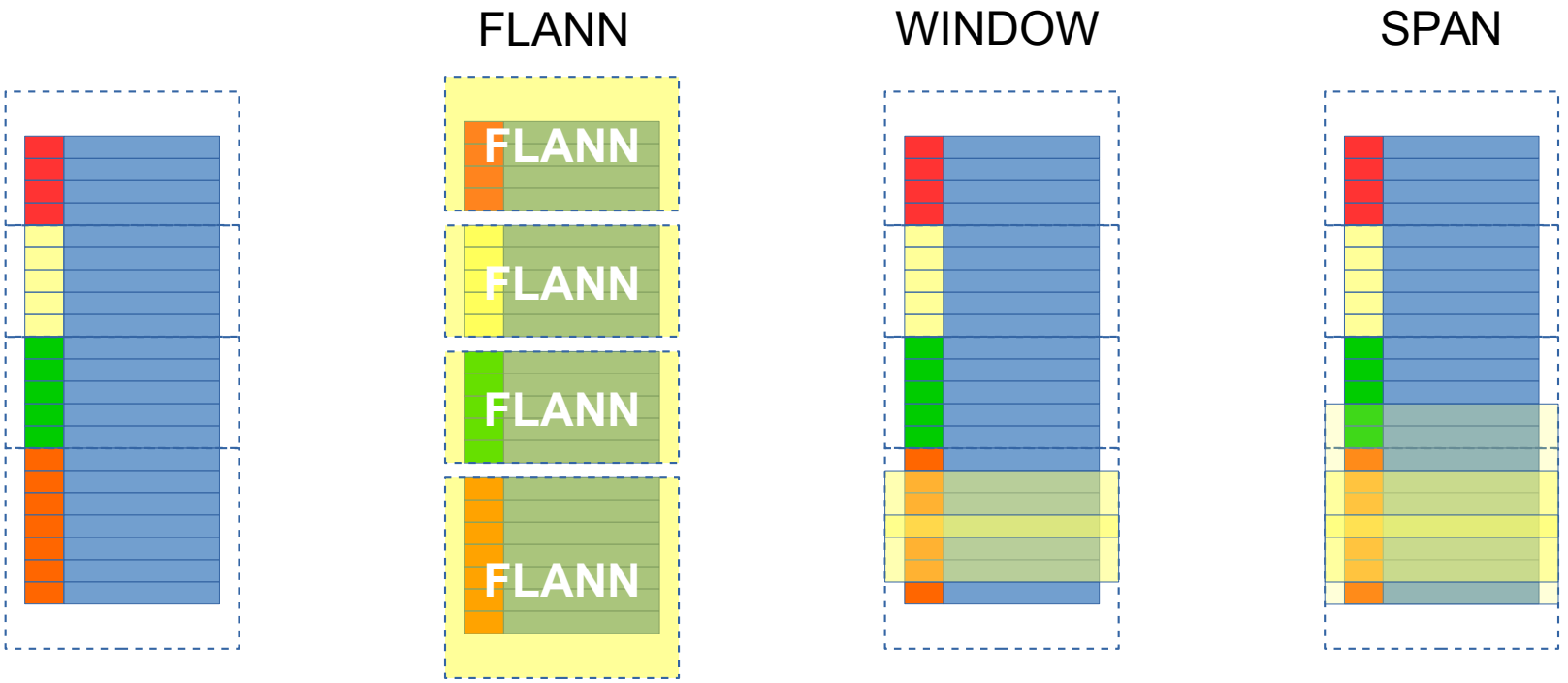
0110 = 6

Partitioning point clouds using z-order

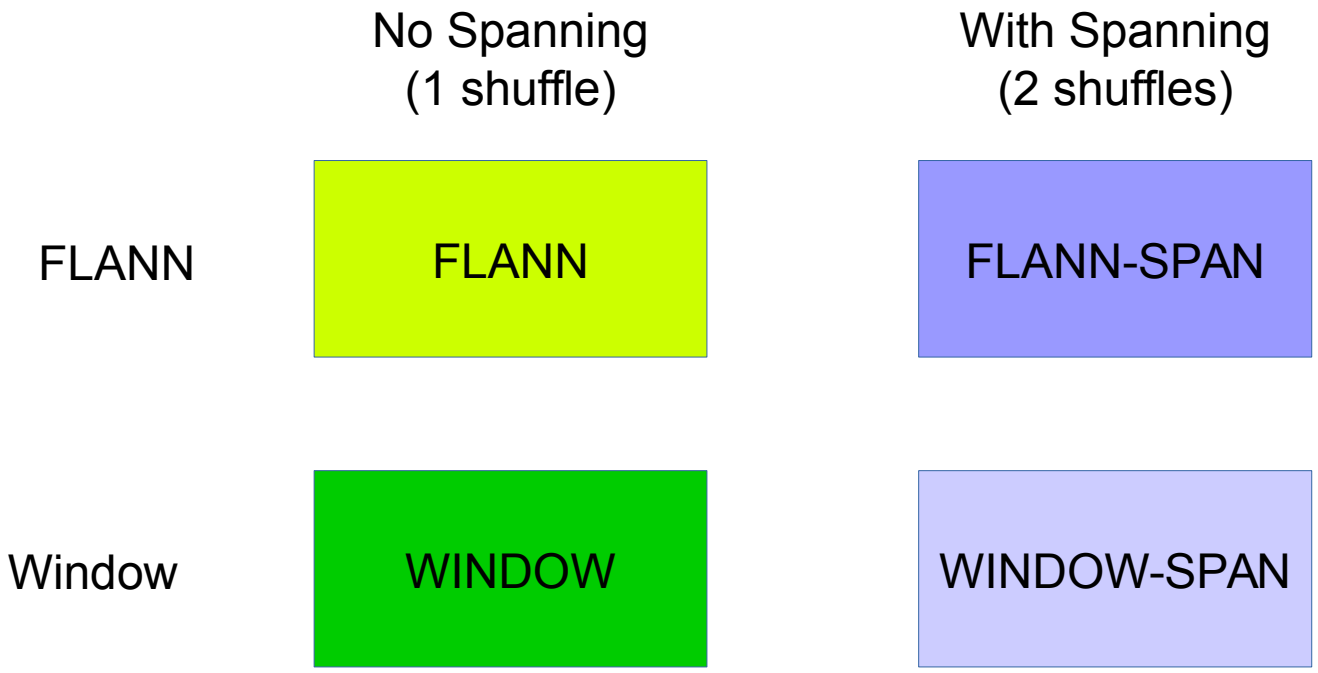


Point cloud | Table of points | Indexed table | Partitioned point cloud

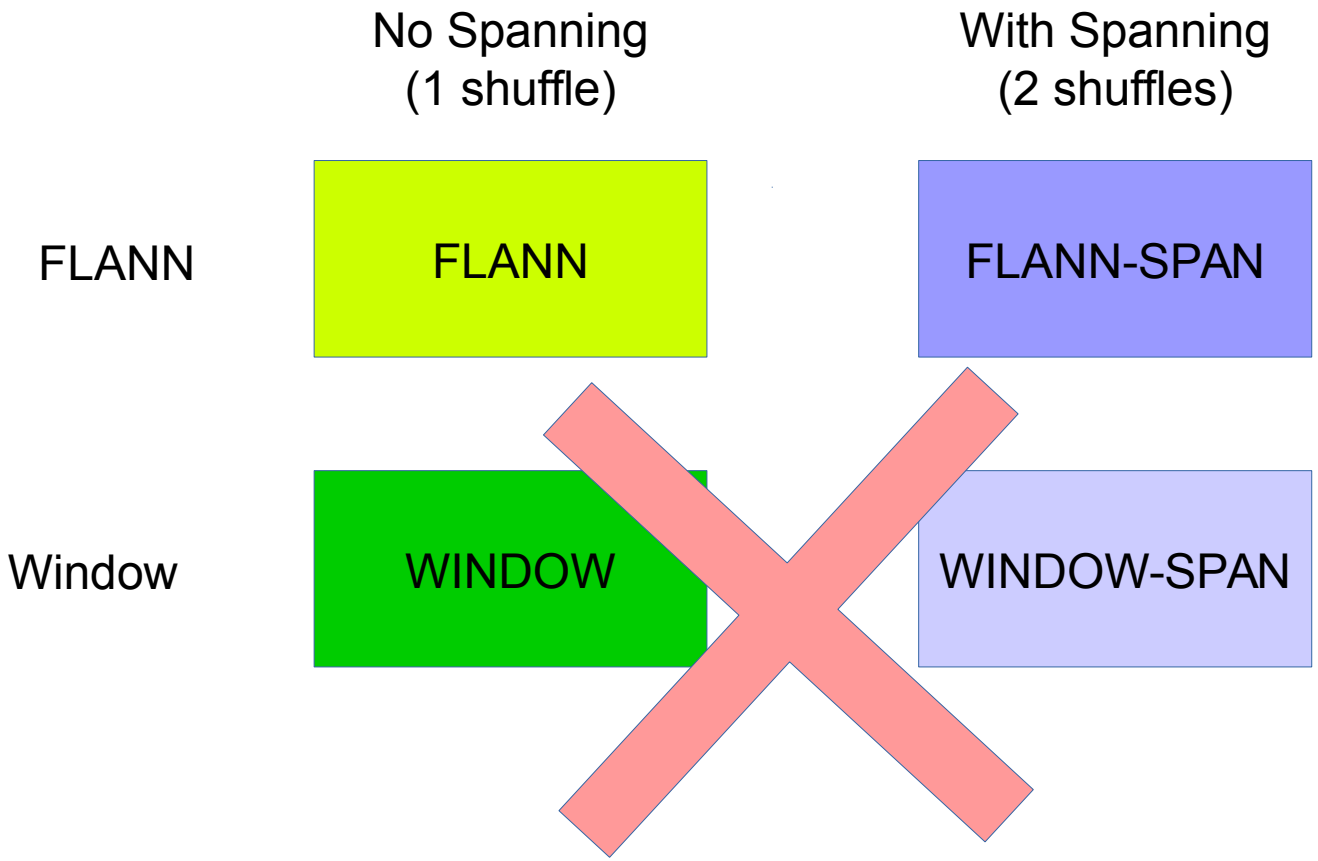
Ways of doing kNN + feature extraction



Workflows

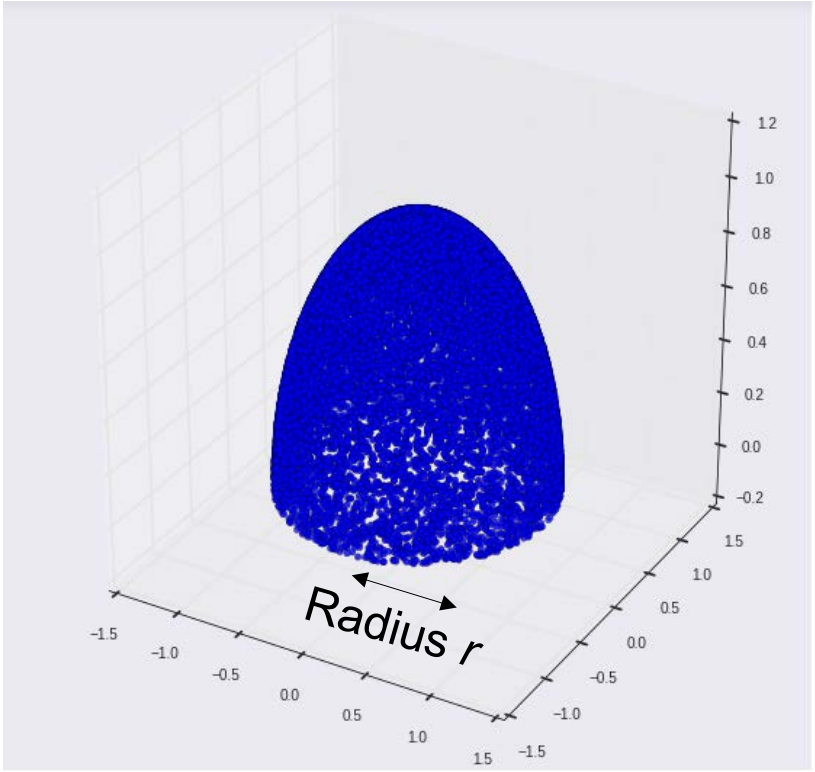

No Spanning
(1 shuffle)
With Spanning
(2 shuffles)
FLANN
FLANN
FLANN-SPAN
Window
WINDOW
WINDOW-SPAN

Workflows

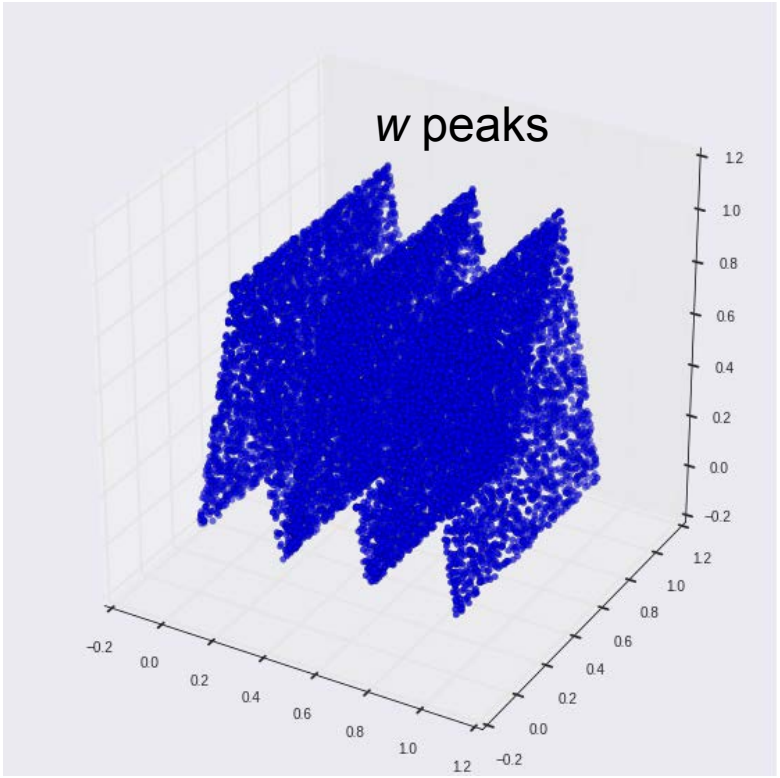

No Spanning
(1 shuffle)
With Spanning
(2 shuffles)
FLANN
FLANN
FLANN-SPAN
Window
WINDOW
WINDOW-SPAN

Test point clouds

Hemisphere

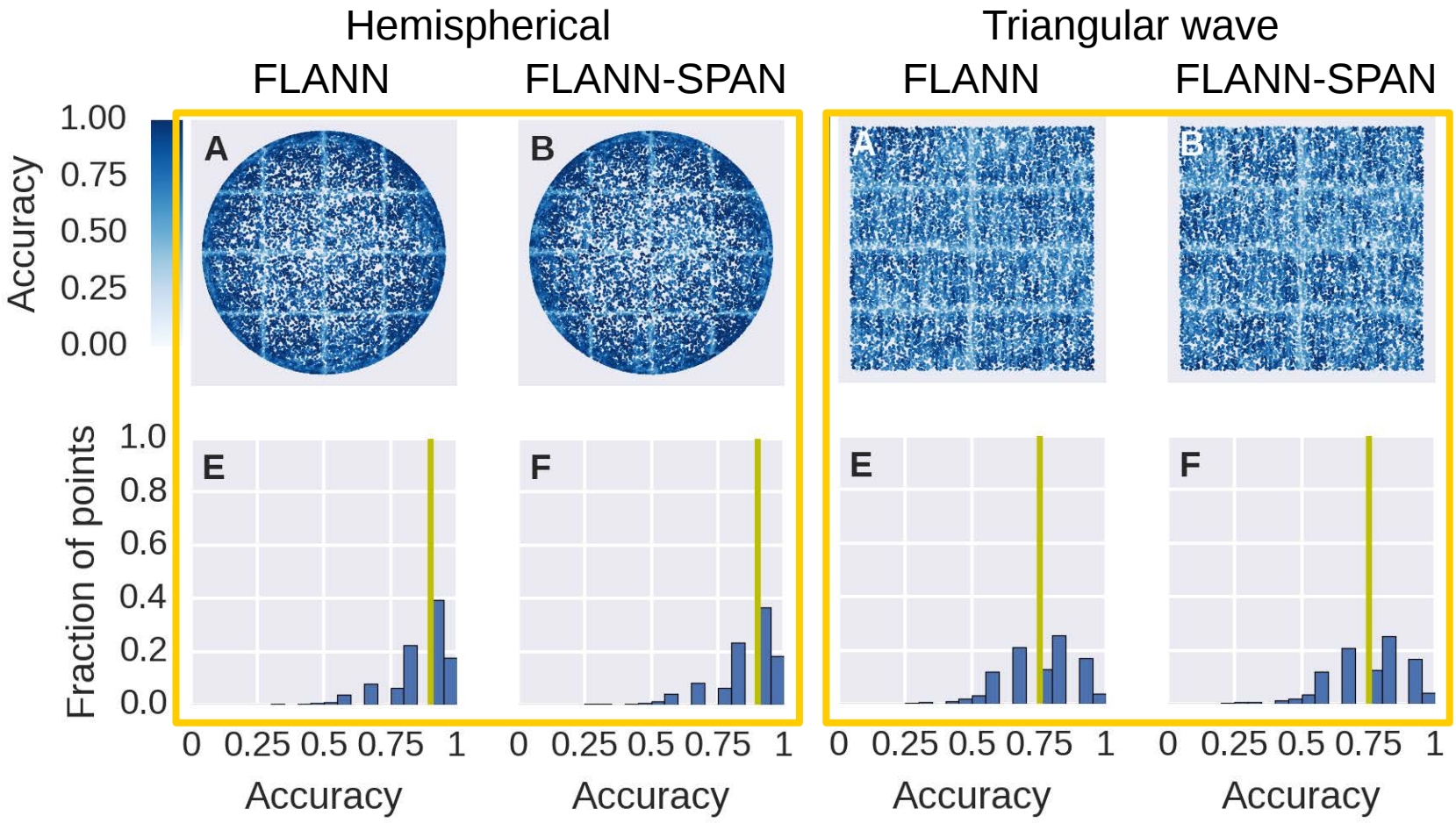


Triangular wave



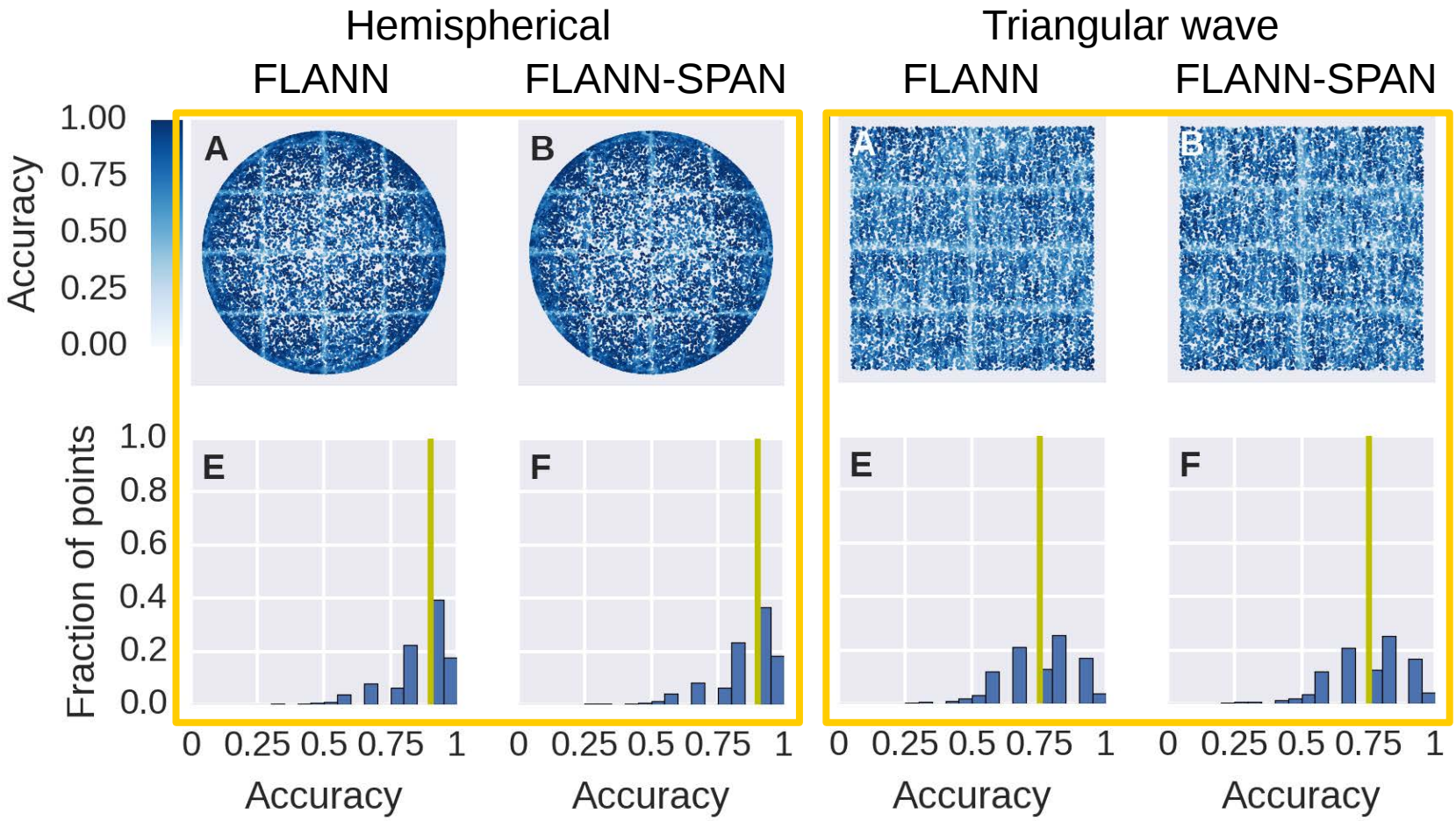
Accuracy at different points

Results with or without spanning are similar



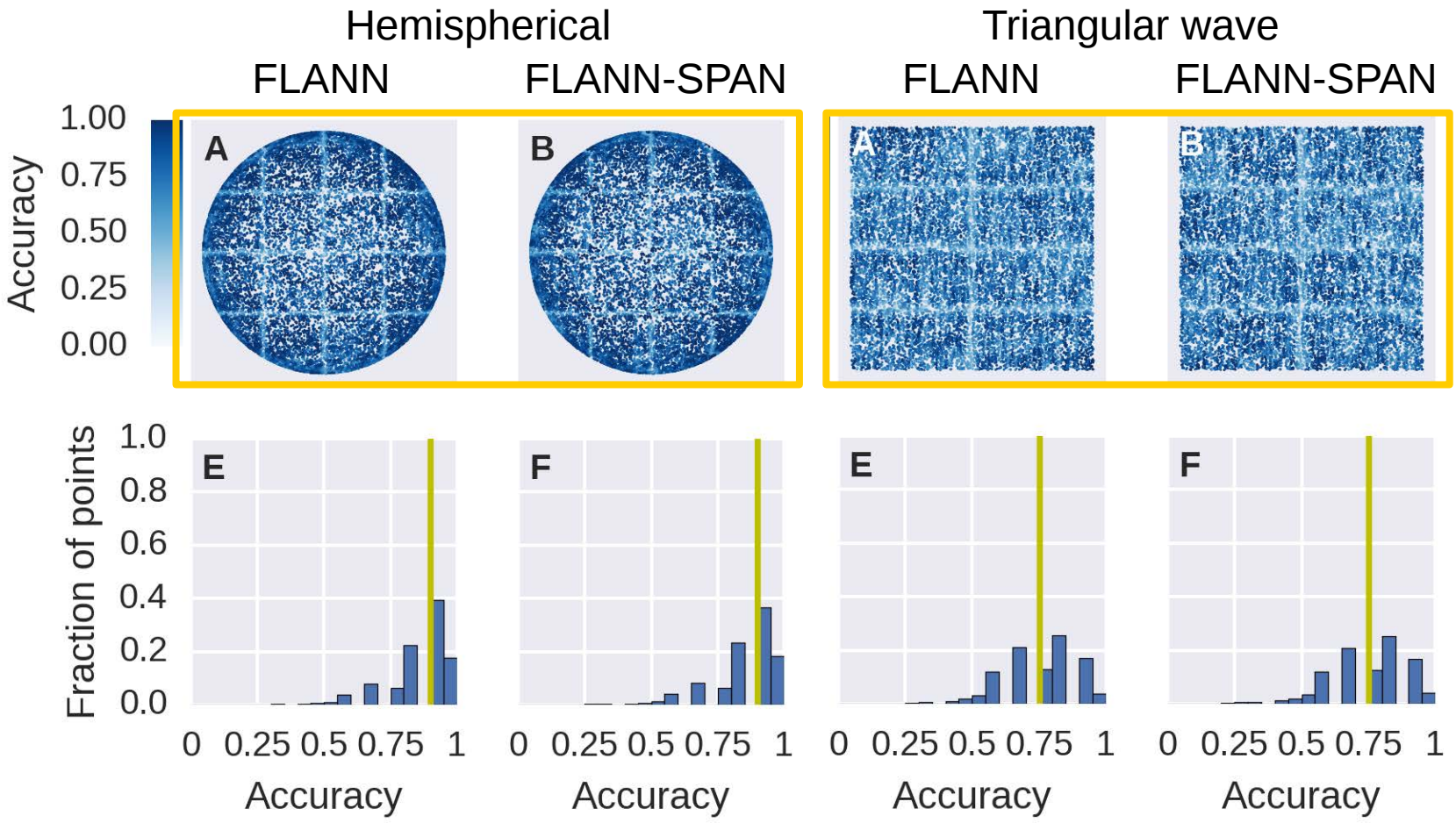
Accuracy at different points

More accurate with smoother point clouds



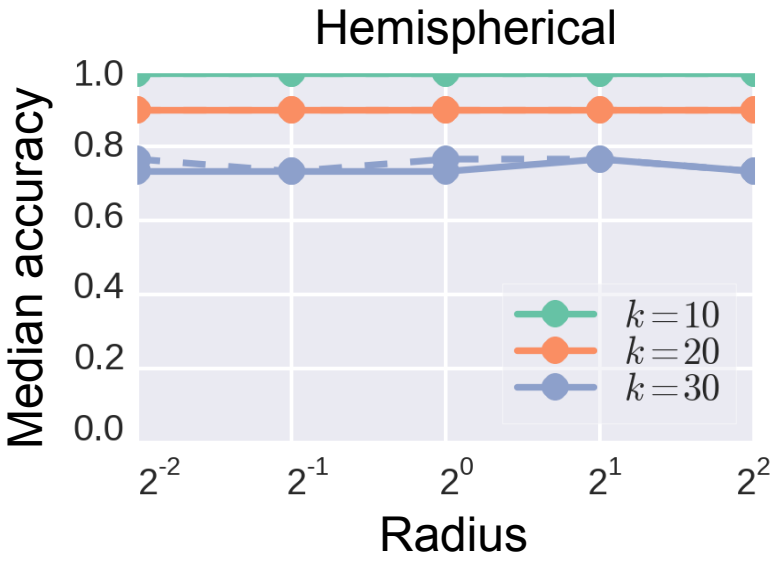
Accuracy at different points

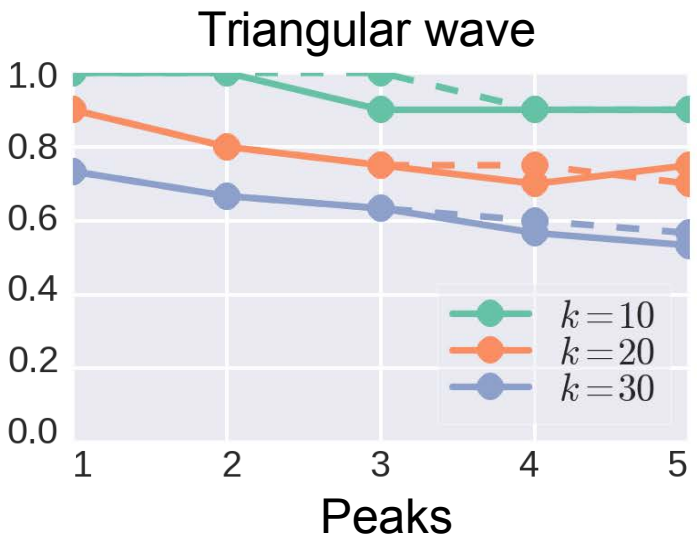
Generally lower accuracy at points near boundary



Accuracy for different neighbourhood sizes

Accuracy gets worse with increasing *k*



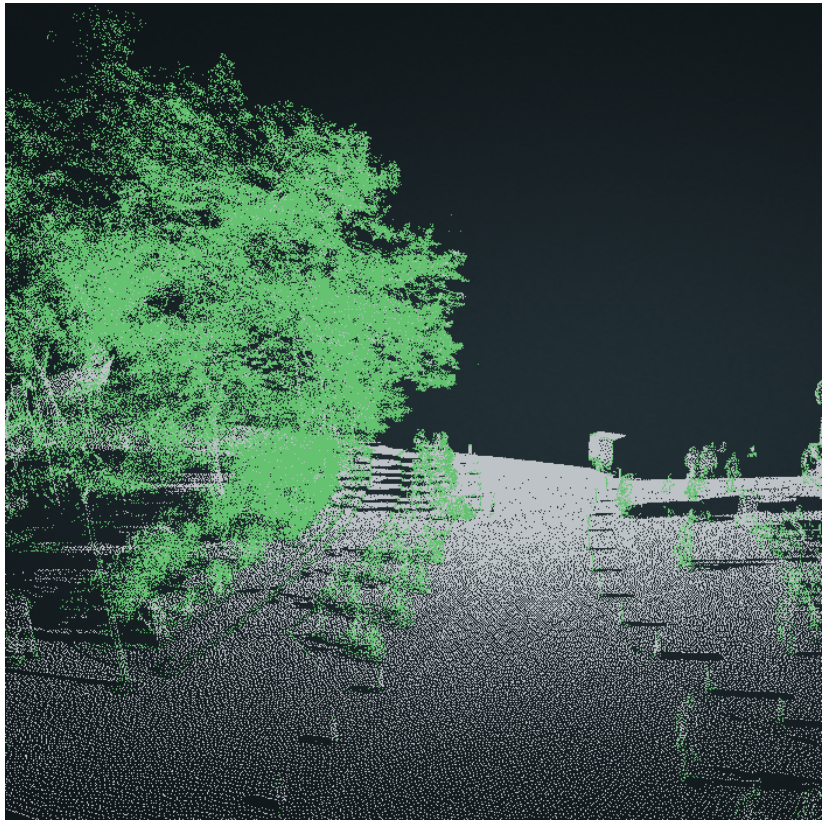


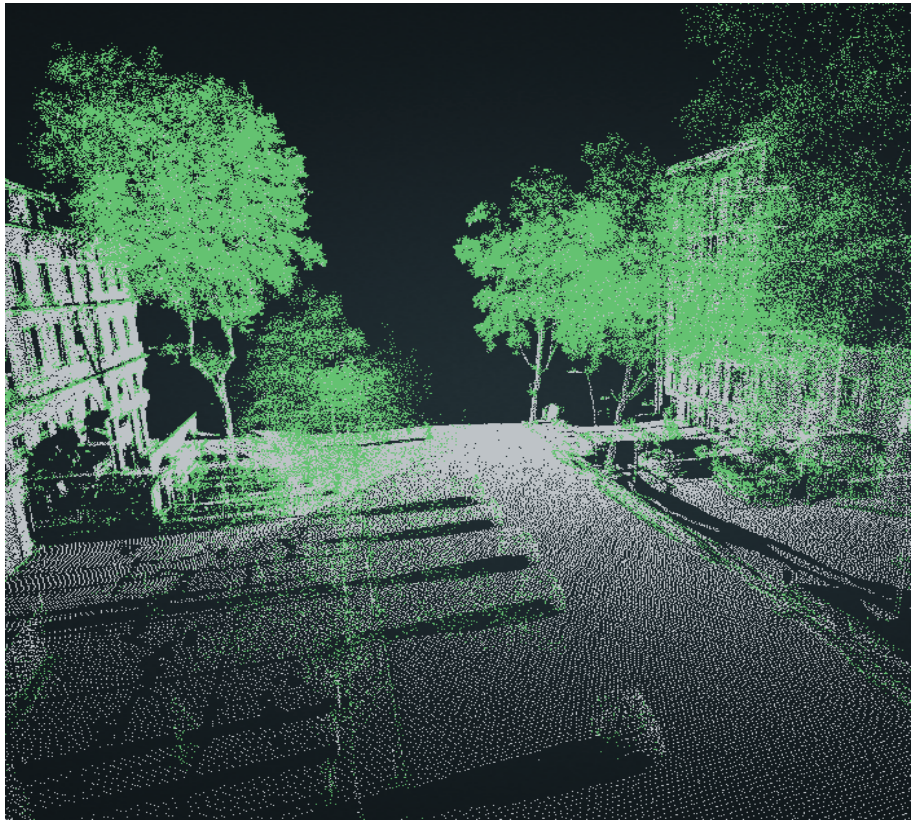
Spark MLlib machine learning library

- Distributed machine learning library
- Supports DT, random forest and GBTree
- Interface similar to scikit-learn
- RDD and DataFrame-based interfaces

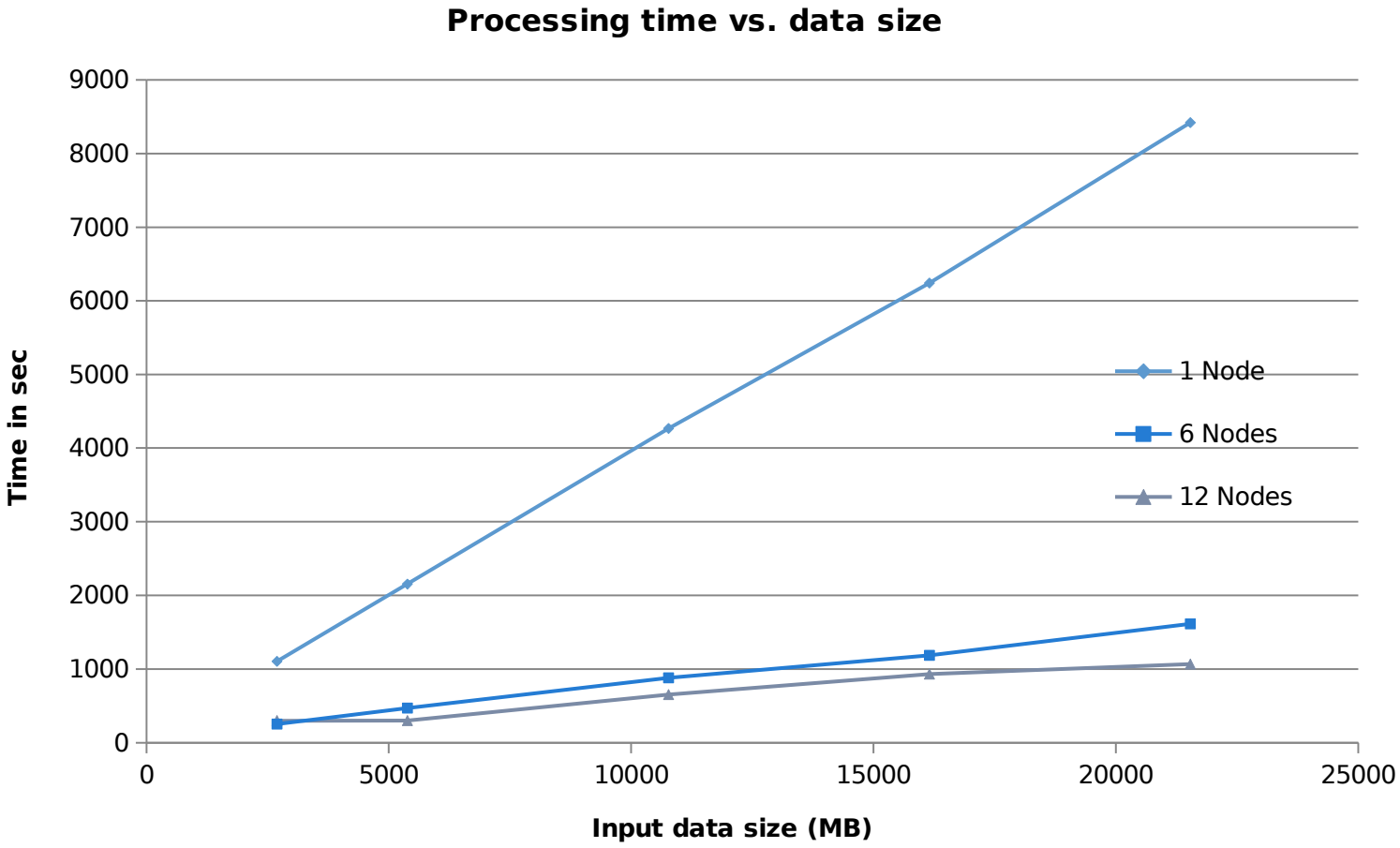
Tree/non-tree classification

- Training set
  - Terramobilita/IQmulus data set (12M points)
  - ~200K tree points
- Testing set
  - Toulouse data set
  - ~3M points per tile

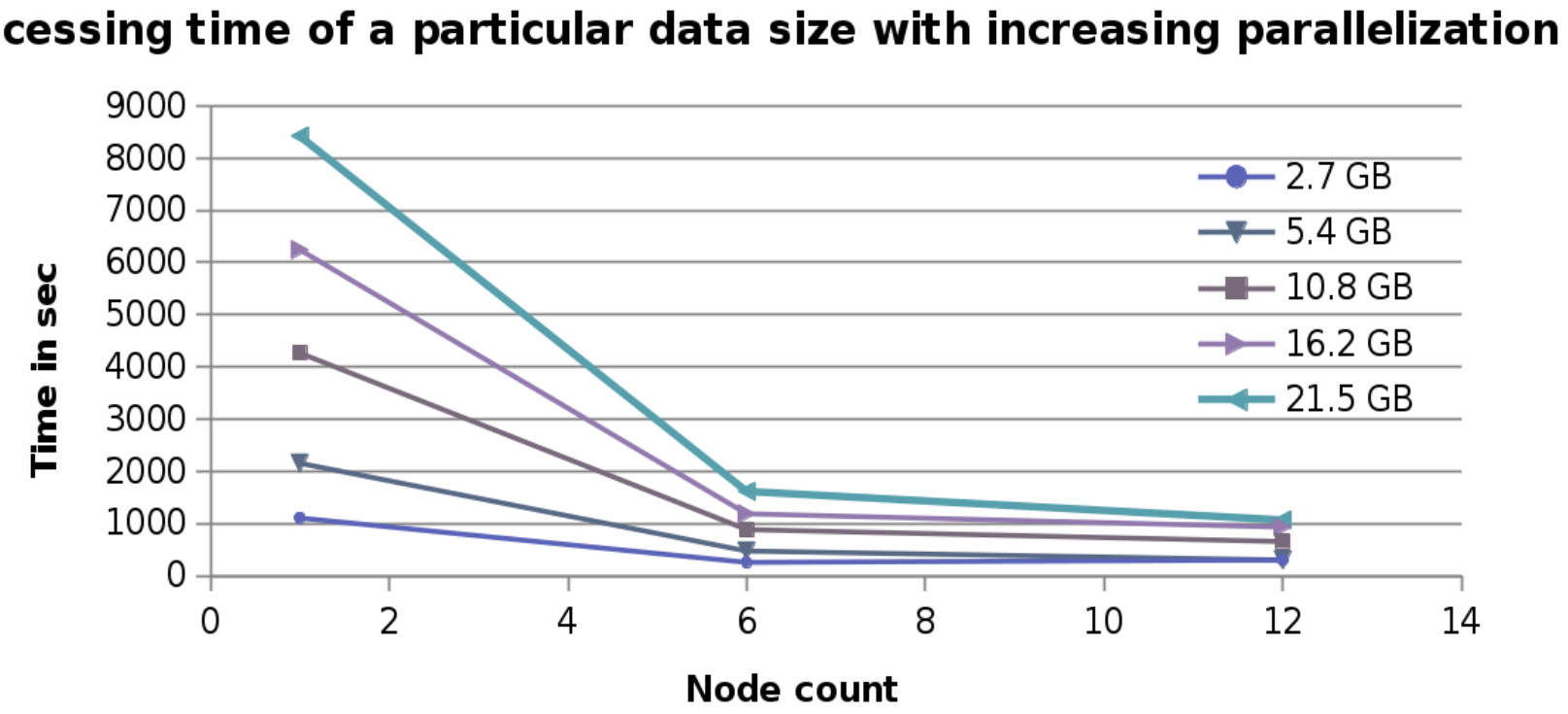




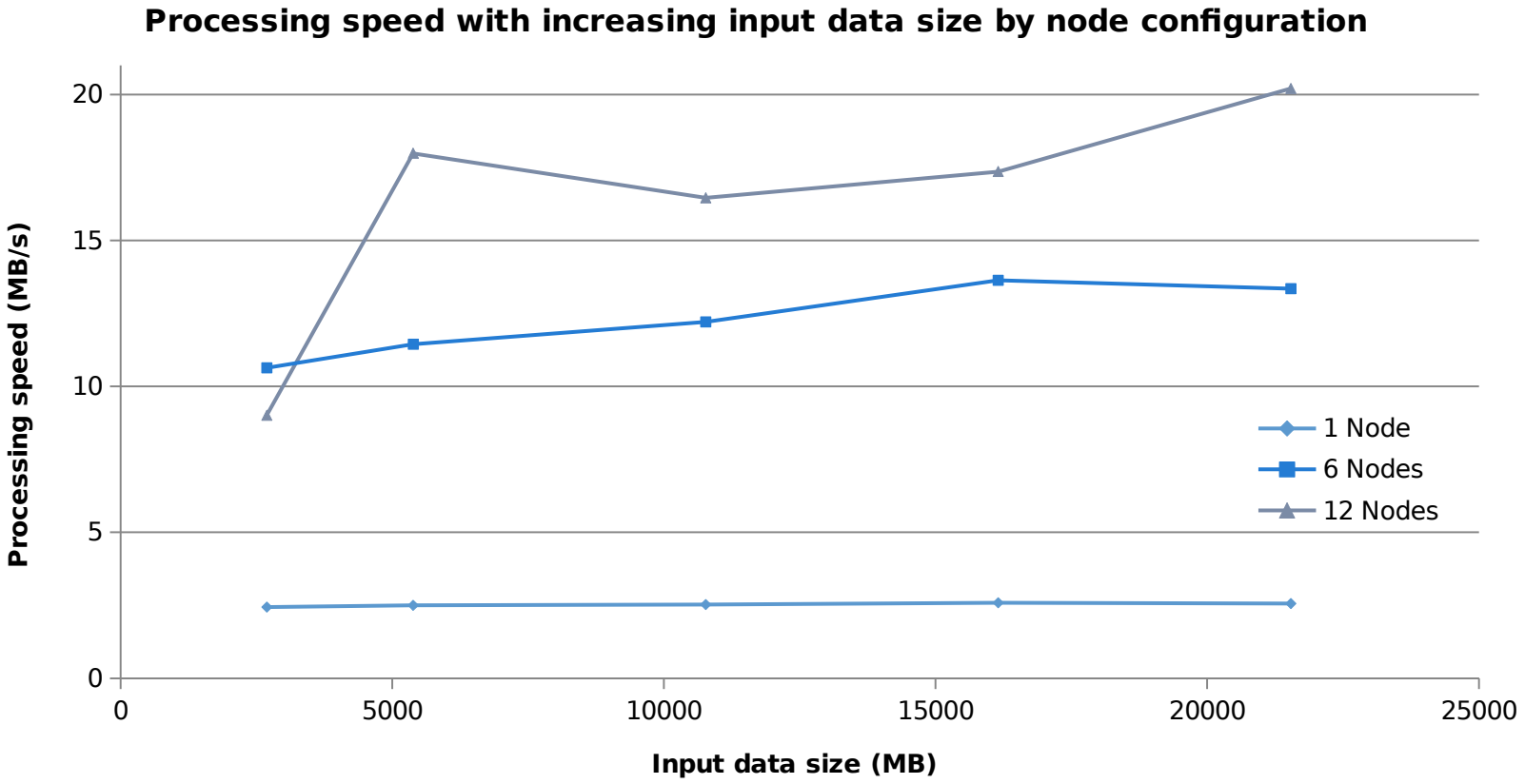
Linear scaling of running time with input data size



Better scalability for larger data size



Better scalability for larger data size

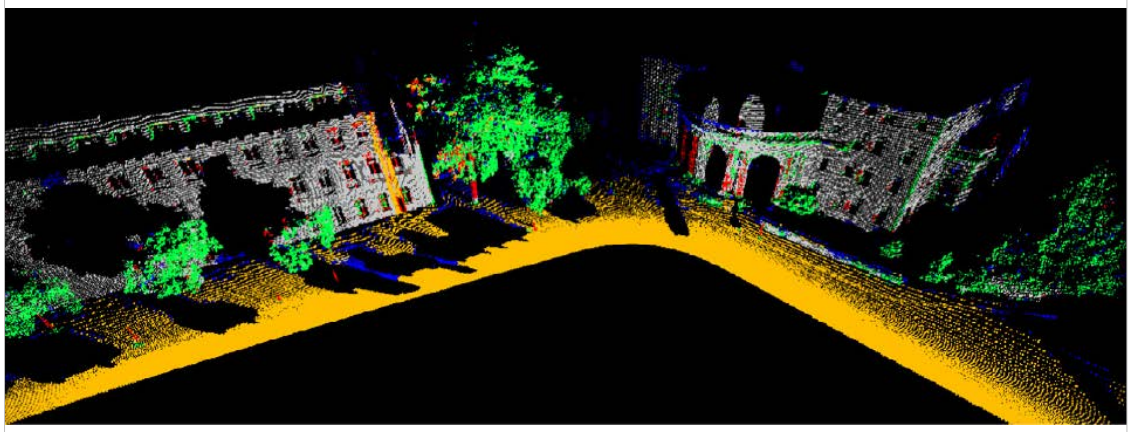


Summary

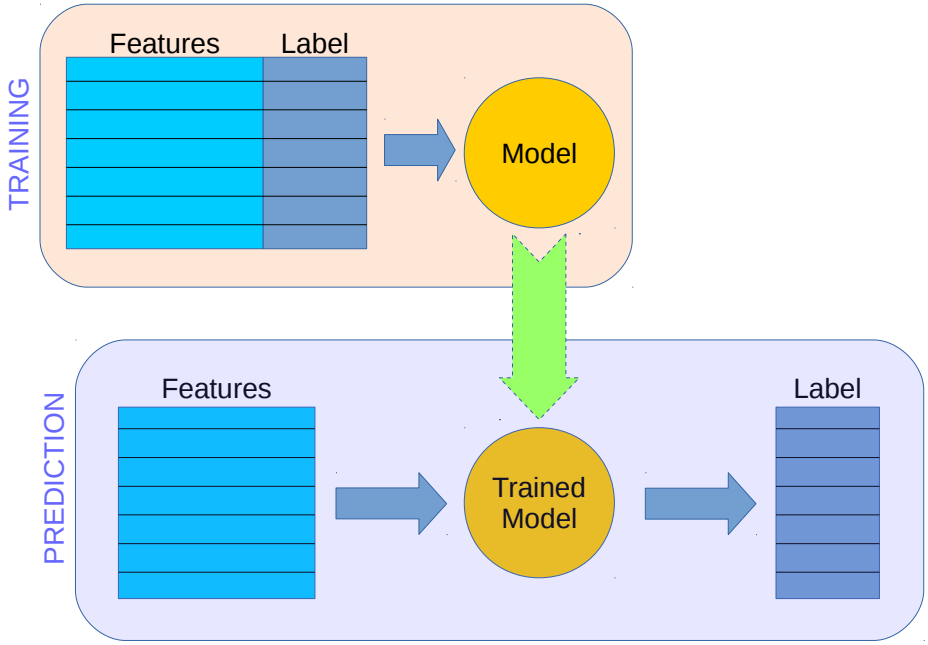
- Random forests can be used to classify point clouds
- Distributed processing can be performed with Spark
- Z-order partitioning mitigates boundary error
- Spark MLlib allows scalable RF classification

Distributed Random Forest classification of urban mobile mapping data

Christian Alis, Jan Boehm, Kun Liu
Dept of Civil, Environmental and Geomatic Engineering
University College London



M. Weinmann, B. Jutzi, S. Hinz, and C. Mallet, 'Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers', ISPRS Journal of Photogrammetry and Remote Sensing, vol. 105, pp. 286–304, Jul. 2015.

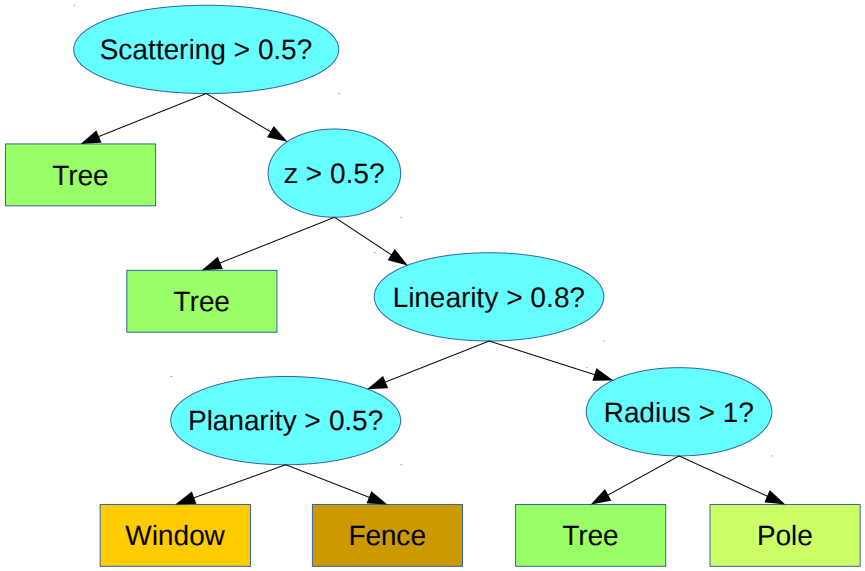

TRAINING
Features
Label
Model
PREDICTION
Features
Trained
Model
Label

Point cloud classification features

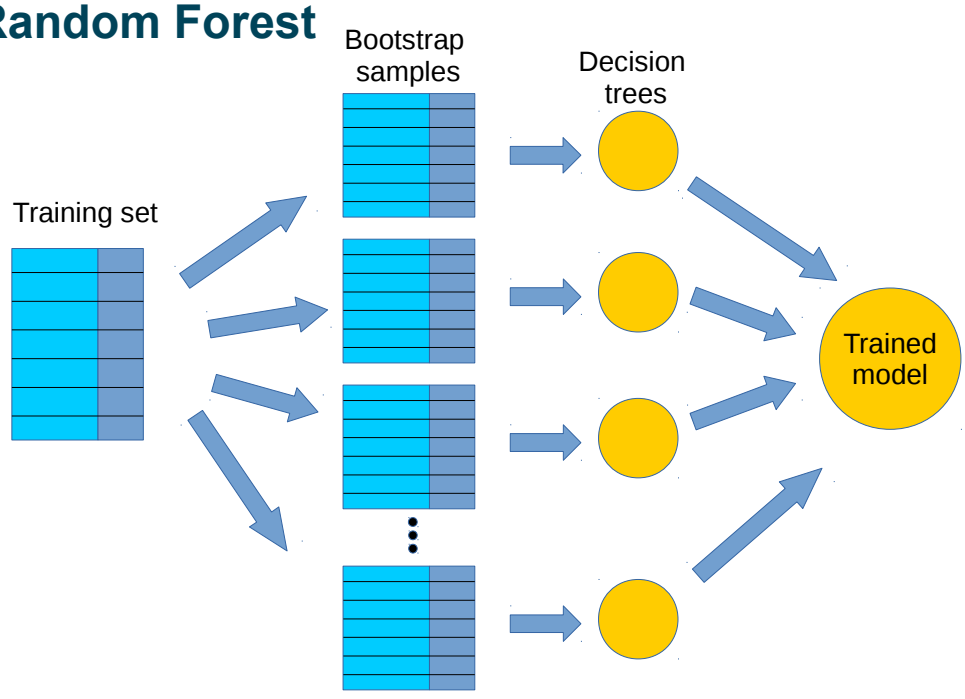
- Linearity
- Planarity
- Scattering
- Omnivariance
- Anisotropy
- Eigenentropy
- Eigenvalue sum
- Change of curvature
- Z value
- Radius
- Density
- Verticality
- Z range
- Z standard deviation
- 2D radius
- 2D density
- 2D eigenvalue sum
- 2D eigenvalue ratio

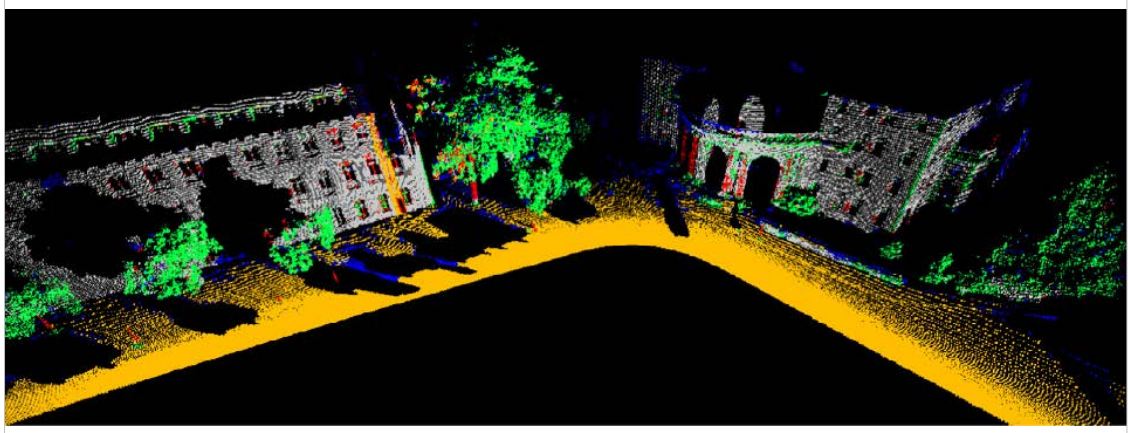
Features are computed based on neighbourhood of the point

Decision Tree



Random Forest





What if the point cloud consists of millions or billions of points?

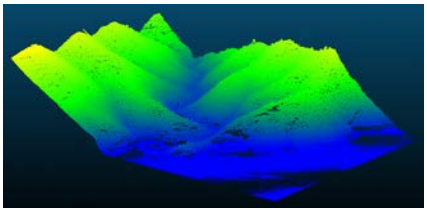
M. Weinmann, B. Jutzi, S. Hinz, and C. Mallet, 'Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers', ISPRS Journal of Photogrammetry and Remote Sensing, vol. 105, pp. 286–304, Jul. 2015.

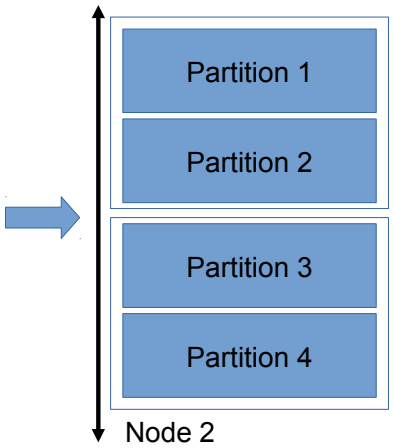
https://commons.wikimedia.org/wiki/File:BigData_2267x1146_white.png

GeoTrellis, SpatialSpark and Magellan provide support for raster and vector data

The challenge of data partitioning

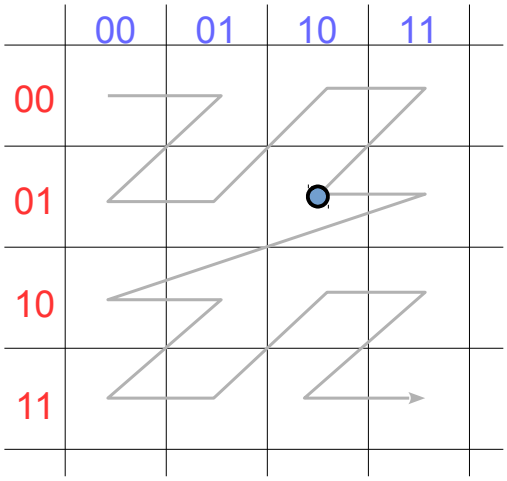
- Not all data would fit on a single machine
- Network I/O is too slow to transfer data as needed





Z-order indexing

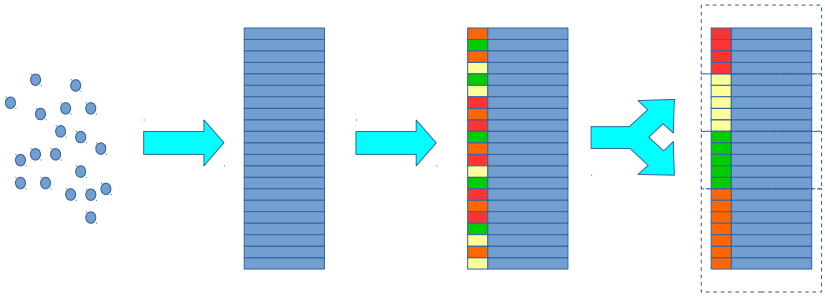
A space-filling curve with adjustable level of detail



1 0
0 1

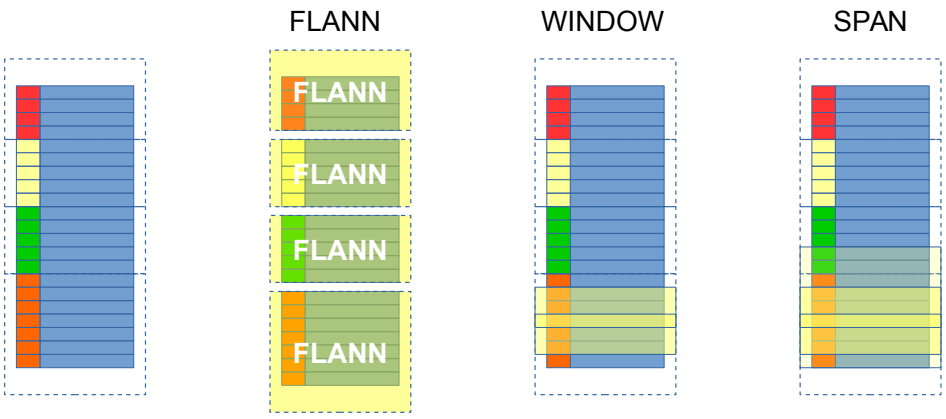
0110 = 6

Partitioning point clouds using z-order

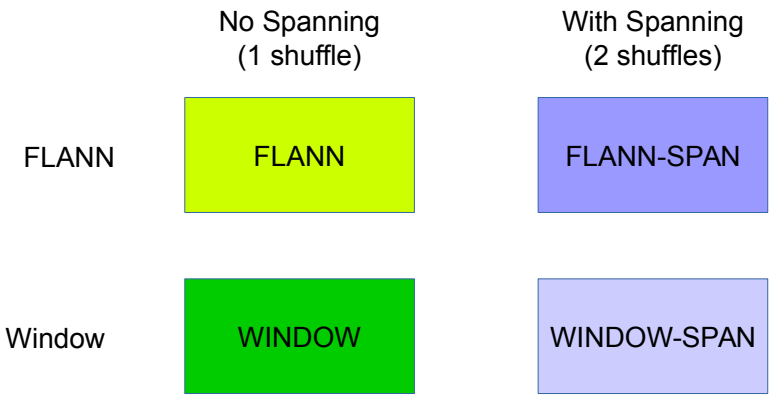


Point cloud | Table of points | Indexed table | Partitioned point cloud

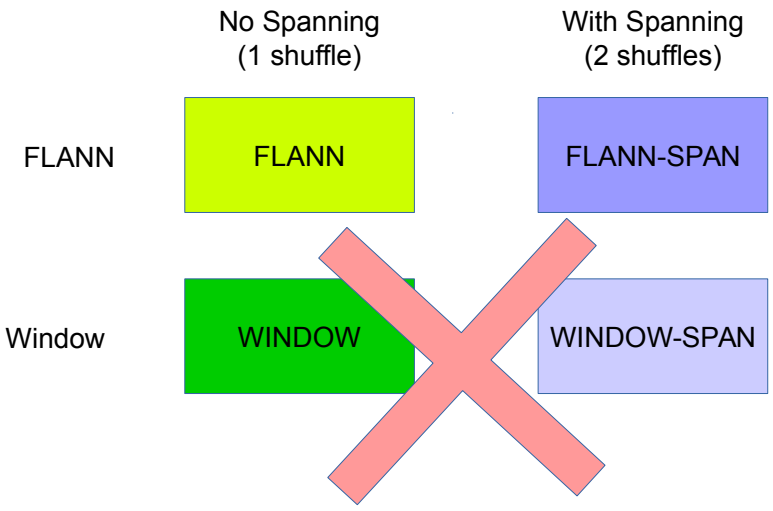
Ways of doing kNN + feature extraction



Workflows

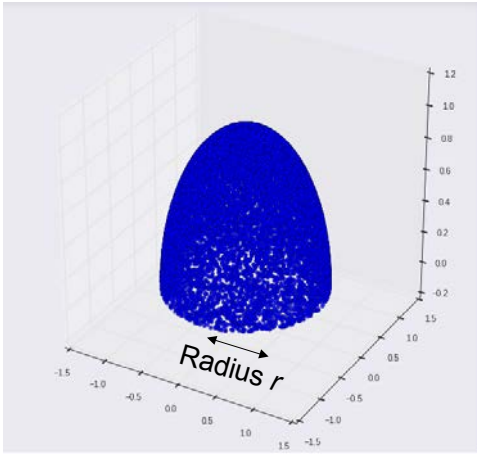

No Spanning
(1 shuffle)
With Spanning
(2 shuffles)
FLANN
FLANN
FLANN-SPAN
Window
WINDOW
WINDOW-SPAN

Workflows

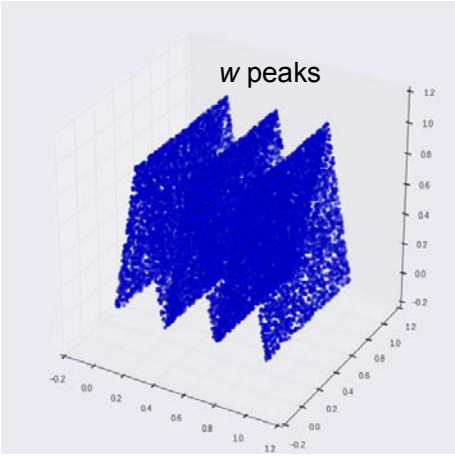

No Spanning
(1 shuffle)
With Spanning
(2 shuffles)
FLANN
FLANN
FLANN-SPAN
Window
WINDOW
WINDOW-SPAN

Test point clouds

Hemisphere

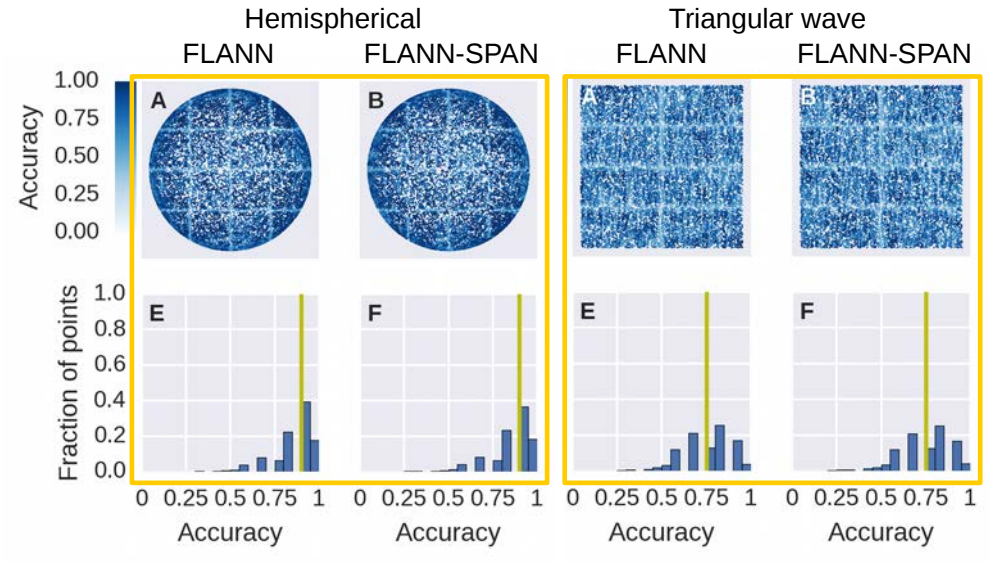


Triangular wave



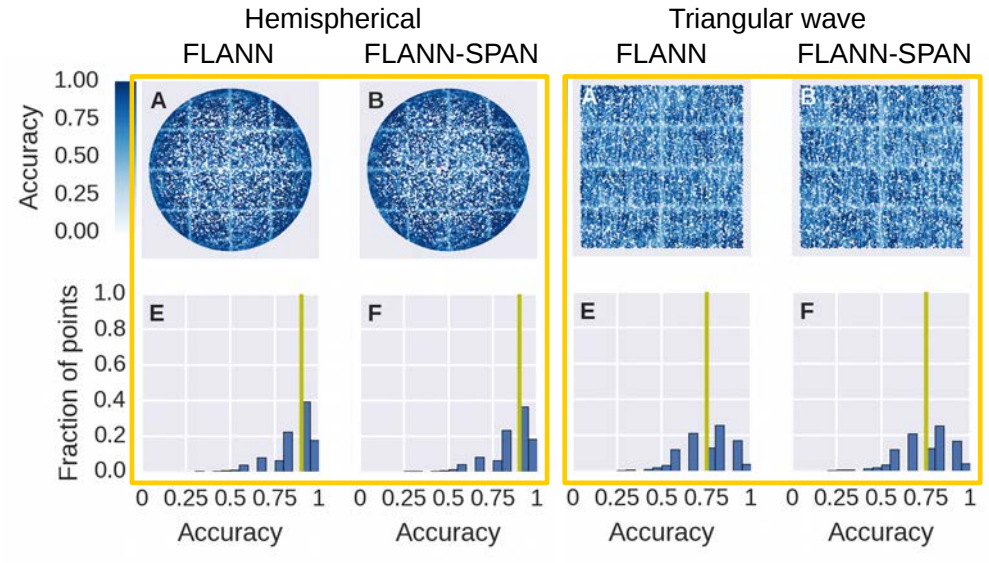
Accuracy at different points

Results with or without spanning are similar



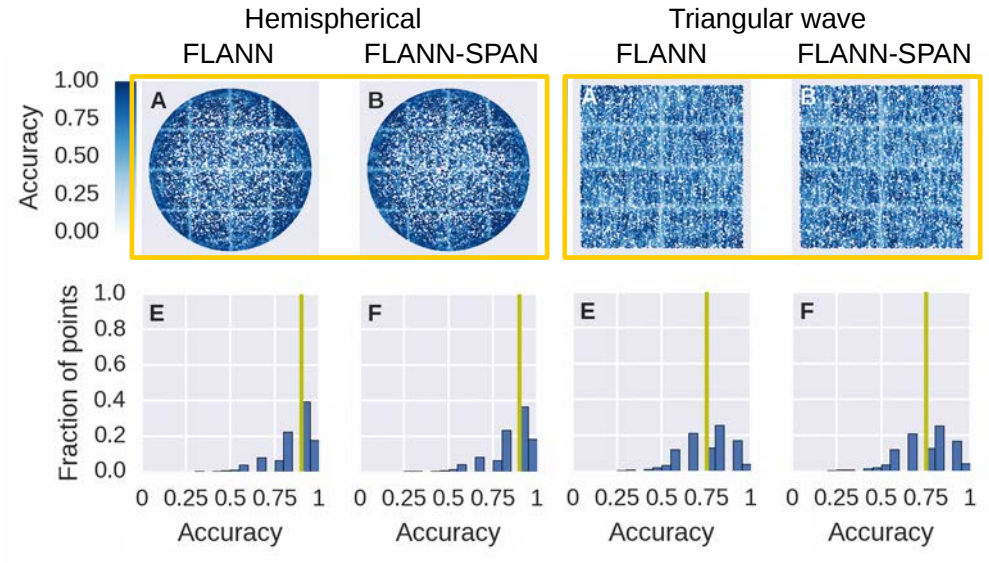
Accuracy at different points

More accurate with smoother point clouds



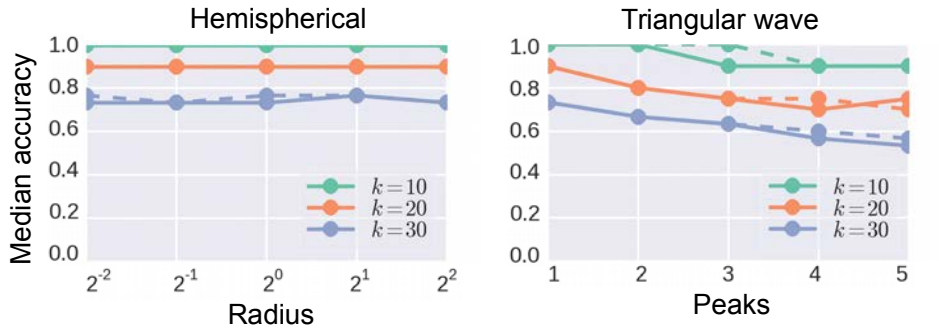
Accuracy at different points

Generally lower accuracy at points near boundary



Accuracy for different neighbourhood sizes

Accuracy gets worse with increasing *k*



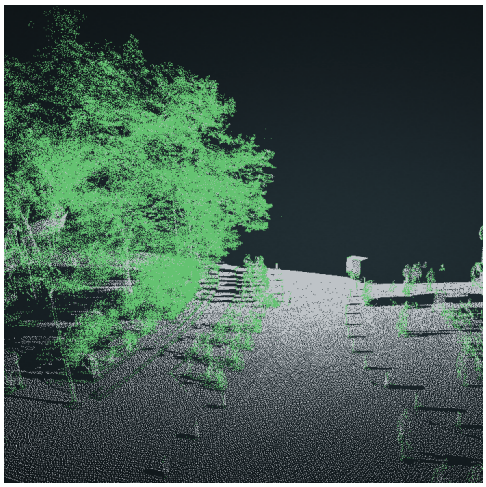
UCL

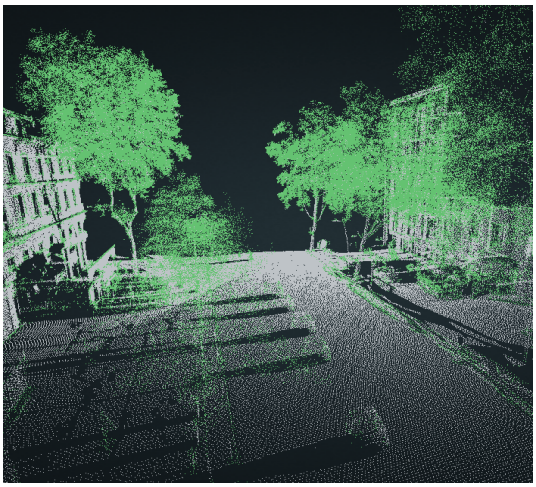
Spark MLlib machine learning library

- Distributed machine learning library
- Supports DT, random forest and GBTree
- Interface similar to scikit-learn
- RDD and DataFrame-based interfaces

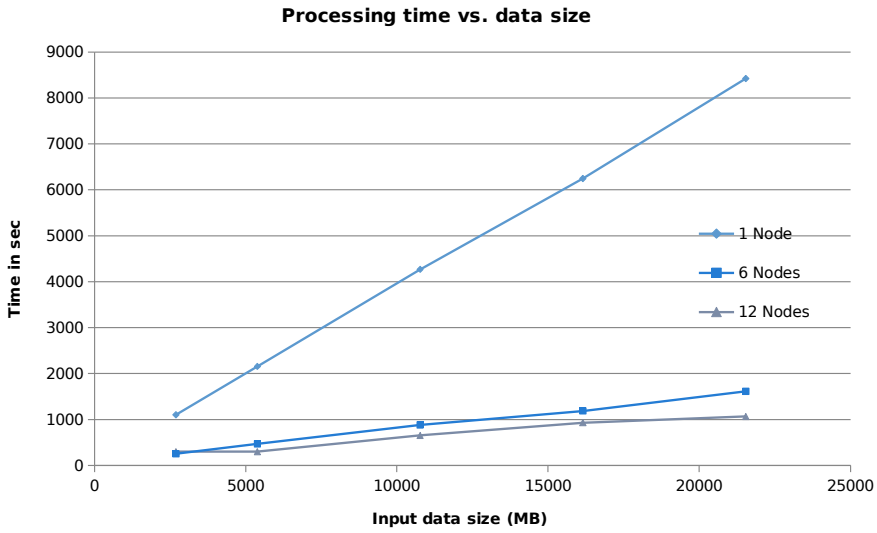
Tree/non-tree classification

- Training set
  - Terramobilita/IQmulus data set (12M points)
  - ~200K tree points
- Testing set
  - Toulouse data set
  - ~3M points per tile

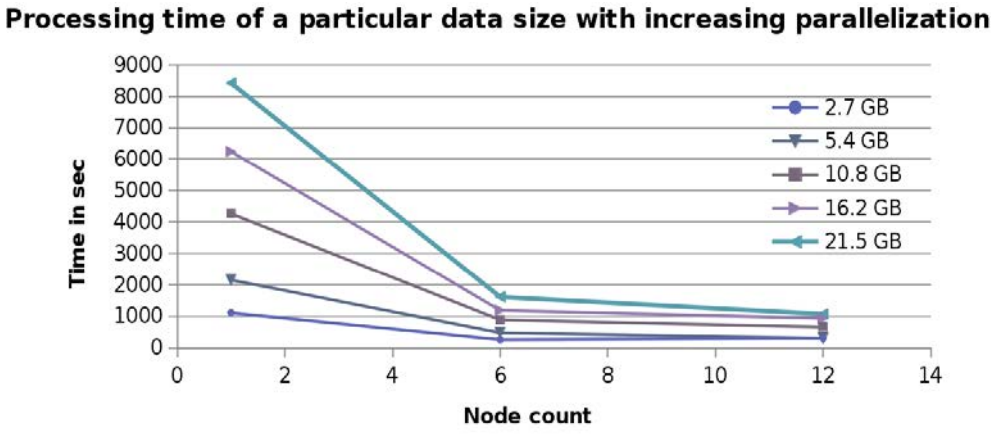




Linear scaling of running time with input data size

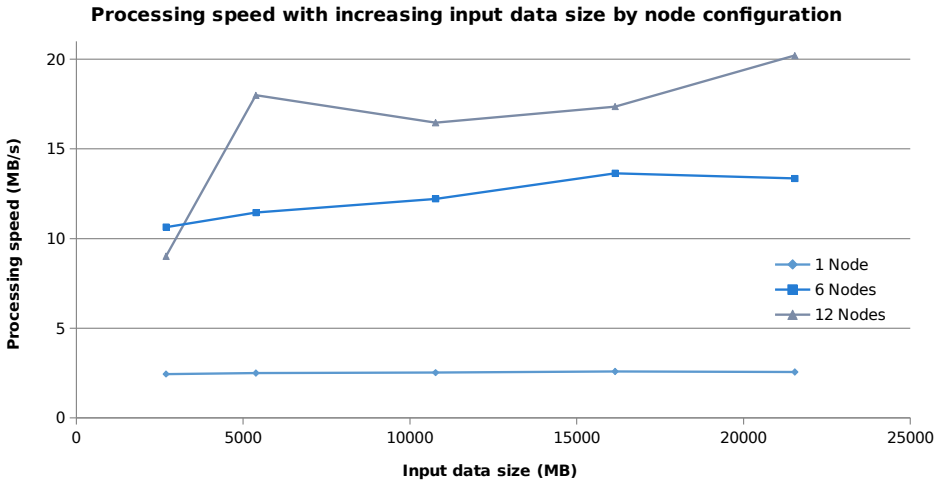


Better scalability for larger data size



UCL

Better scalability for larger data size



Summary

- Random forests can be used to classify point clouds
- Distributed processing can be performed with Spark
- Z-order partitioning mitigates boundary error
- Spark MLlib allows scalable RF classification