# Automatic 3D Plane Detection on an Hadoop Computer Cluster

Jifang DUAN and Dr David R. Selviah
jifang.duan.13@ucl.ac.uk d.selviah@ucl.ac.uk www.ee.ucl.ac.uk/odevices
Photonics Research Group, Electronic & Electrical Engineering, UCL

UCL

## Introduction

In most indoor environment, a large number of objects have planar surfaces. There is a clear need for plane detection which identifies planar areas such as ceilings, floors or tables from point cloud data obtained by scanning an interior environment of a building using 3D terrestrial laser scanners. The RANSAC algorithm is a standard method for detecting planar areas. Our aim is to combine this algorithm with the idea of Big Data analysis to solve problems encountered in the real world.

Due to the large quantity (several terabytes) of high resolution point cloud data obtained by 3D laser scanning (a technique used to rapidly capture shapes of objects, buildings and landscapes), it is difficult and very time-consuming to process all the information using conventional single computer. The Hadoop MapReduce programming model provides a solution to this problem. This model comprises of a Hadoop Distributed File System (HDFS) which is highly fault-tolenrant and is designed to be deployed on low-cost hardware. HDFS breaks down large data sets into small pieces, replicate each piece to multiple nodes and distributed throughout a Hadoop cluster. It also provides parallel processing functionality on the cluster.

Figure 1: 3D scanning

Source of Figure 1: http://www.surfaceandedge.com/applications/civil-engineering

## Ransac algorithm

The Random sample consensus (RANSAC) algorithm was first published by Fischler and Bolles at SRI International in 1981. The input of RANSAC algorithm is a set of observed data which contains outliers. The RANSAC describes an iterative method to estimate parameters of a mathematical model from this observed data. The algorithm is able to robust estimate the model parameters even when a significant number of outliers are presented in the dataset. Based on random sampling, RANSAC can be used for detecting planes, spheres, cylinders, cones and torus. (Schnabel 2007)

In the experiment of 3D plane extraction, RANSAC method is applied for detecting large planar areas from point cloud data obtained by laser scanner.

This RANSAC method starts by randomly selecting three points from the point cloud. Given the fact that every three non-collinear points can establish a plane, these selected three points are able to form a hypothetical plane. But we are only interested in those hypothetical planes that coincide with real planar areas that exist in the original data. Normally, it would take a great number of iterations until the ideal points are found as demanded.

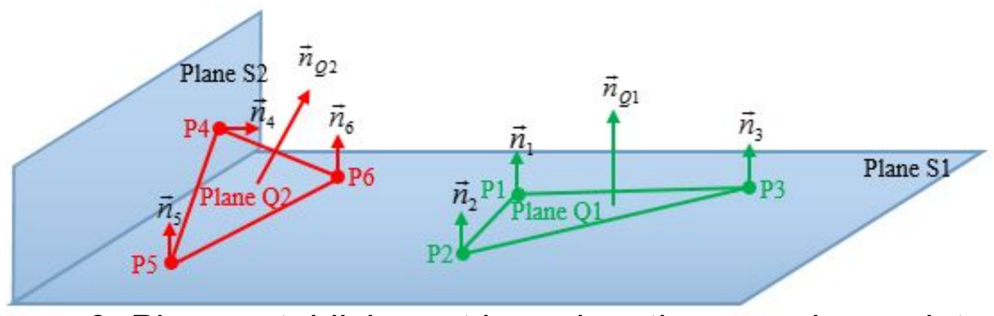


Figure 2: Plane establishment based on three random points

Once a plane model is established, all the other points that lie on this plane are expected to be found. Two conditions are used for determining whether a point belongs to this plane. A point, that is 1) located close enough to this planar area, and 2) whose surface normal is almost parallel to the surface normal of the plane model, can be regarded as a point belonging to this plane.

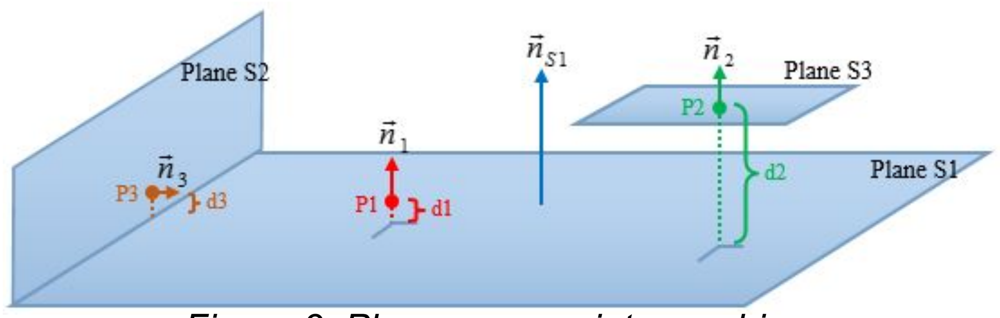


Figure 3: Planar area point searching

Schnabel R., Wahl R. and Klein R. 2007. Efficient RANSAC for Point-Cloud Shape Detection. Computer Graphics Forum 26(2): 214-226.

## RANSAC flowchart and results

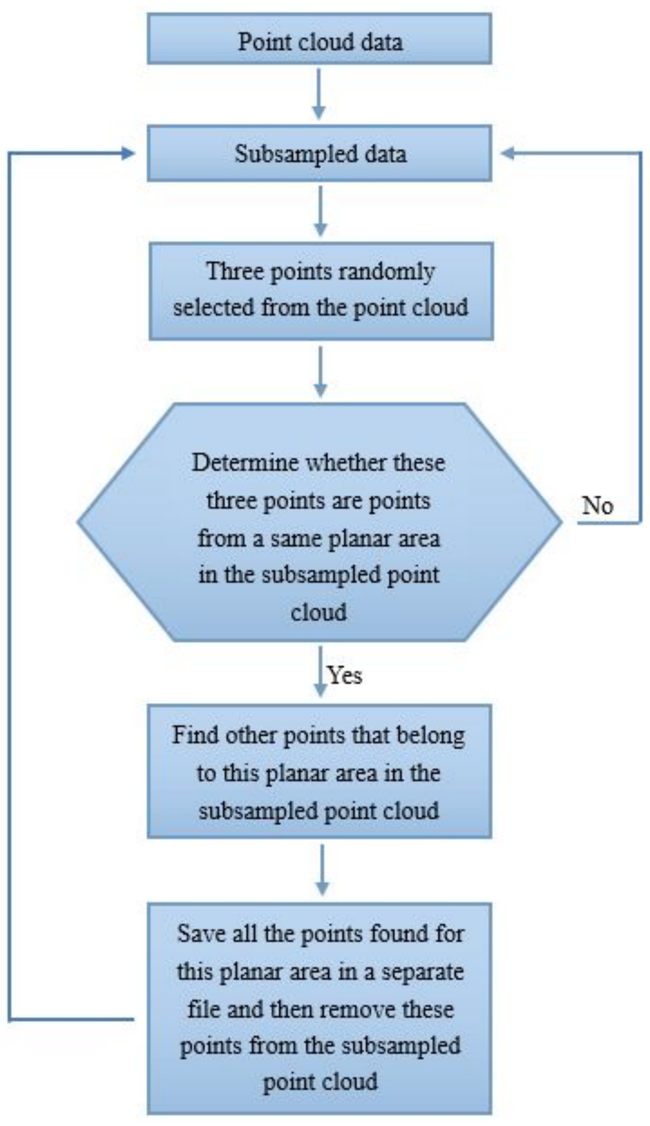


Figure 4: Flowchart of methodology

Figure 5 : Subsampled scan of a corner in the lab

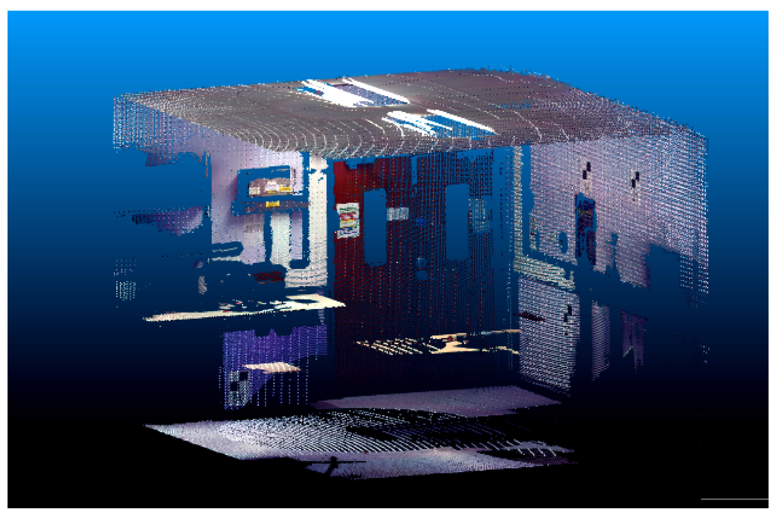

Figure 6: Extracted planes of this scan

## Plane detection on an Hadoop cluster

This experiment uses point cloud data recorded in the lab on 9th floor, Roberts Building. Figure 7 is the overview of the exterior of this lab. This graph is obtained by aligning multiple scans to the same 3D coordinate reference. Figure 7.1, 7.2 and 7.3 are scans captured from different points of this lab. Point cloud data of each scan is stored in separate files. These files are inputs of this MapReduce framework. The output that is expected to be obtained is a list of files, each of which represents one plane of this lab, marked by two parameters: $n_k$ and $d_k$, where $n_k$ is the surface normal of Plane k; $d_k$ is a parameter related to the intersection of Plane k and z coordinate axis.

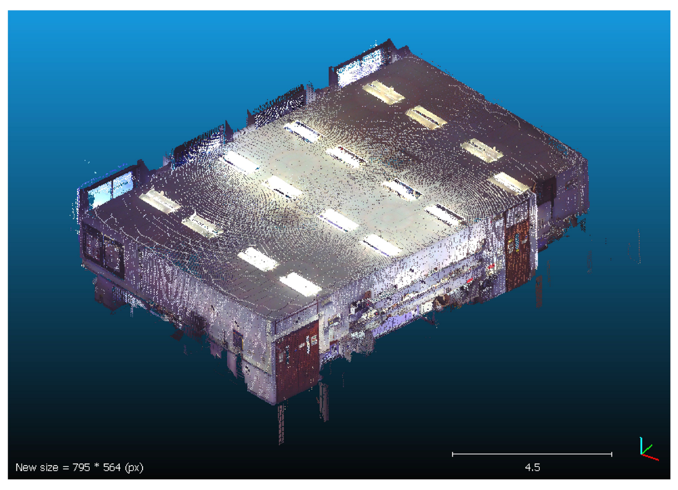

Figure 7: Overview of the lab

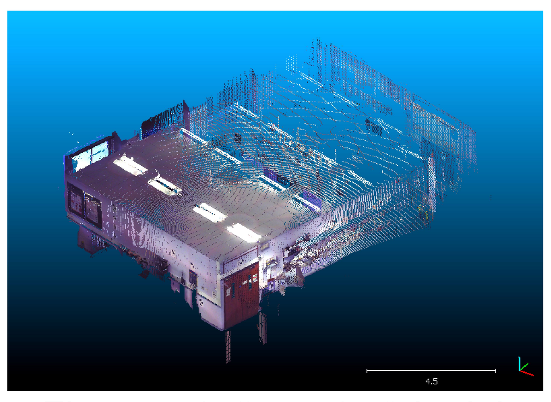

Figure 7.1: Scan 1 of the lab

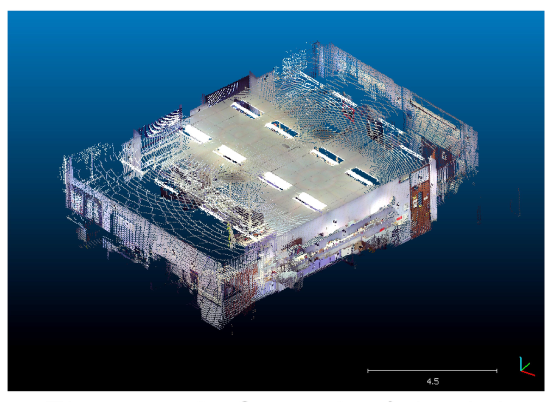

Figure 7.2: Scan 2 of the lab

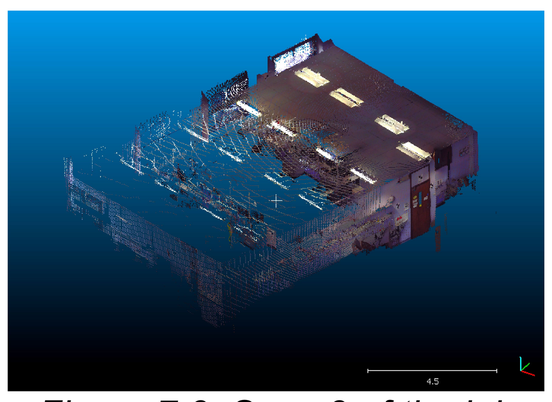

Figure 7.3: Scan 3 of the lab

The plane detection problem is partitioned into three sections while using Hadoop: Map section, Combine section and Reduce section.

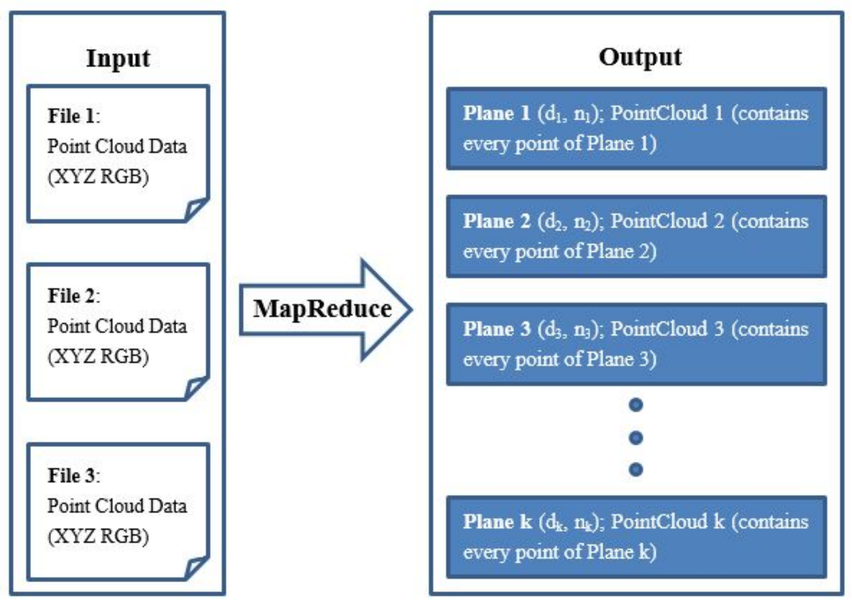


Figure 8: MapReduce overview

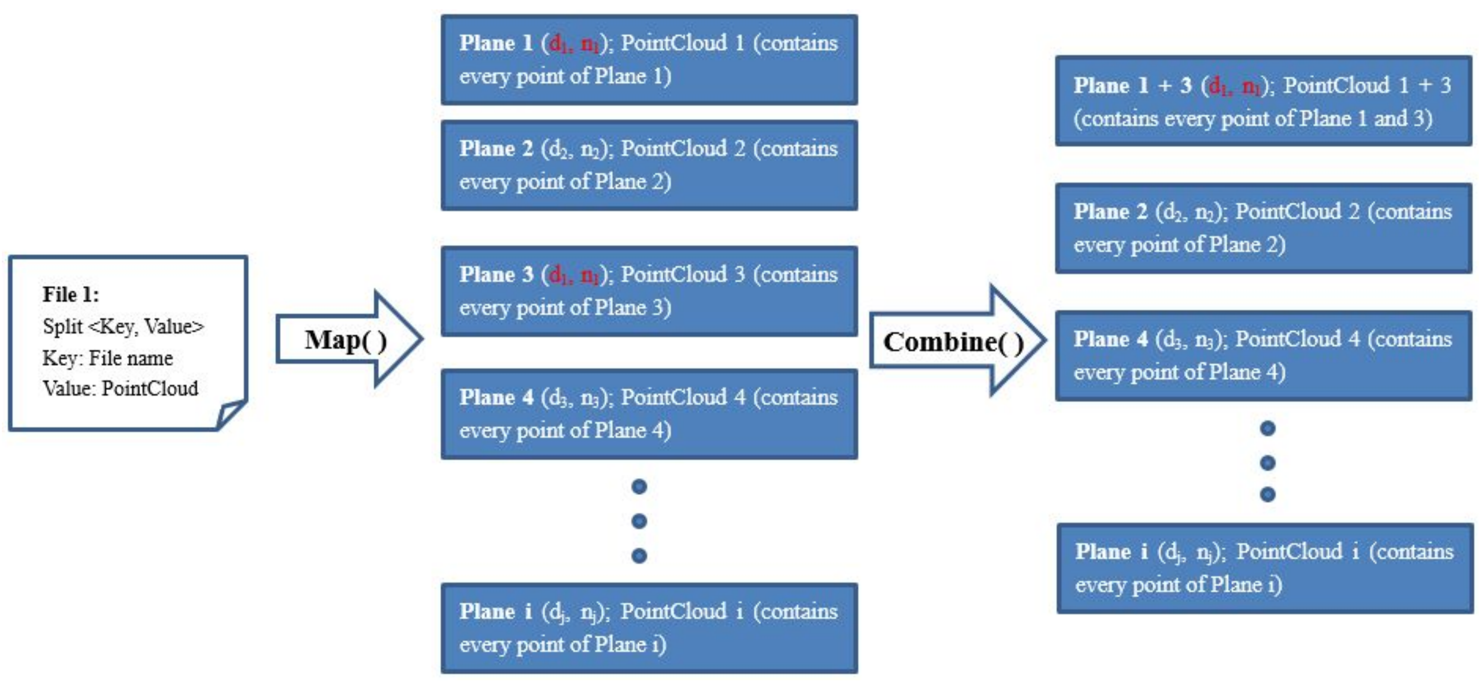


Figure 9: Map & Combine process

Reducer works similar to Combiner, it joins planes which have the same key ($d_k$, $n_k$). The only difference is that these planes that are joint by Reducer come from different files.

## Plane detection results on an Hadoop computer cluster

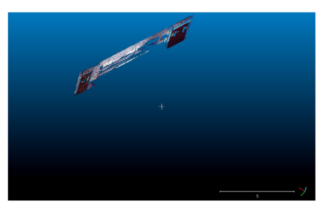

Figure 10.1

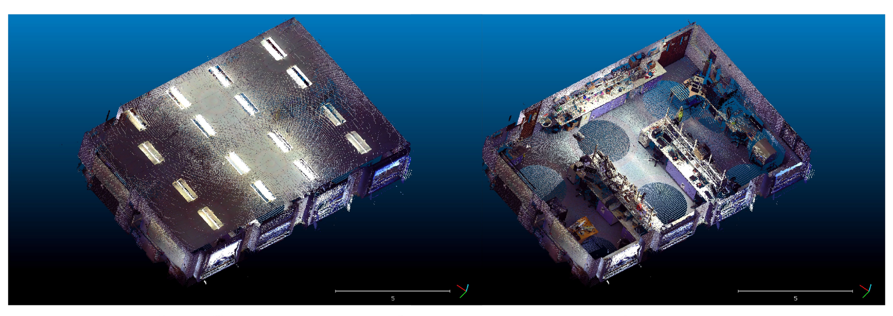

Figure 10 : Overview of the lab with / without ceiling

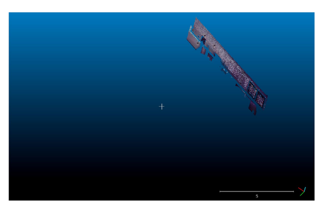

Figure 10.7

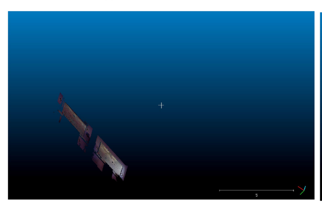

Figure 10.2

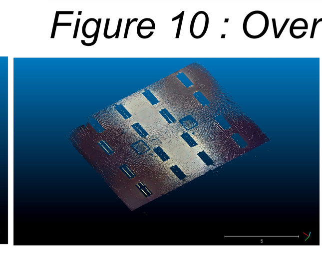

Figure 10.3

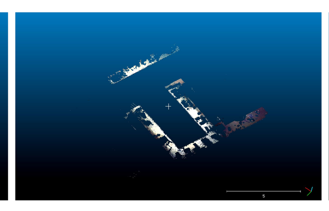

Figure 10.4

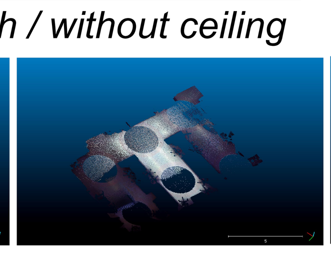

Figure 10.5

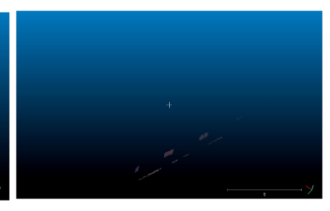

Figure 10.6

Seven large planar areas are extracted from the original data of this lab. Figure 10.3, 10.4 and 10.5 are extracted planes of ceiling, table and floor respectively. Figure 10.1, 10.2, 10.6 and 10.7 are extracted planes of walls all around.

| | File 1 | File 2 | File 3 |
|---|---|---|---|
| Map | 8 planes | 8 planes | 5 planes |
| Combine | 6 planes | 7 planes | 4 planes |
| Reduce | | 7 planes | |
| Number of points | Original: 2.6 million | | Extracted planes: 1.5 million |
| Processing time | | 10 seconds | |

Figure 11 : Experimental results