Cartography M.Sc.

Strategies for a Reduction of Indoor Point Clouds to 'Purified' Room Geometries and their Interactive Presentation

E. Sathish Raymond
(Author)

Dr.rer.nat. Nikolas Prechtel
(Supervisor)

25/09/2020

Introduction

- LIDAR stands for Light Detection And Ranging, which is an active Remote Sensing technique used to examine the object present on the surface of the earth.

- Based on the acquisition and platform type, the usage of LIDAR technology varies from topographic and bathymetric mapping.

- Indoor Mapping is a terrestrial based acquisition technique used to map indoors such as rooms, balcony, basement etc of a building.

- The Building Information Model (BIM) can be created using LIDAR technology, that gives architecture, civil engineer, planners and construction professionals the insight and tools to more efficiently plan, design, construct, manage buildings and infrastructure.

Motivation

- The motivation of this research work is to provide an innovative path/idea of using the recorded geometric 3D information to a semantically tagged 3D models with an interactive visualization.

- The research is carried out to provide a sematic meaning that the lidar information can be segregated/segmented based on the user needs and purpose.

Research Objectives

The Main research objective is:

- To Create a purified 3D room geometry model with an interactive visualization

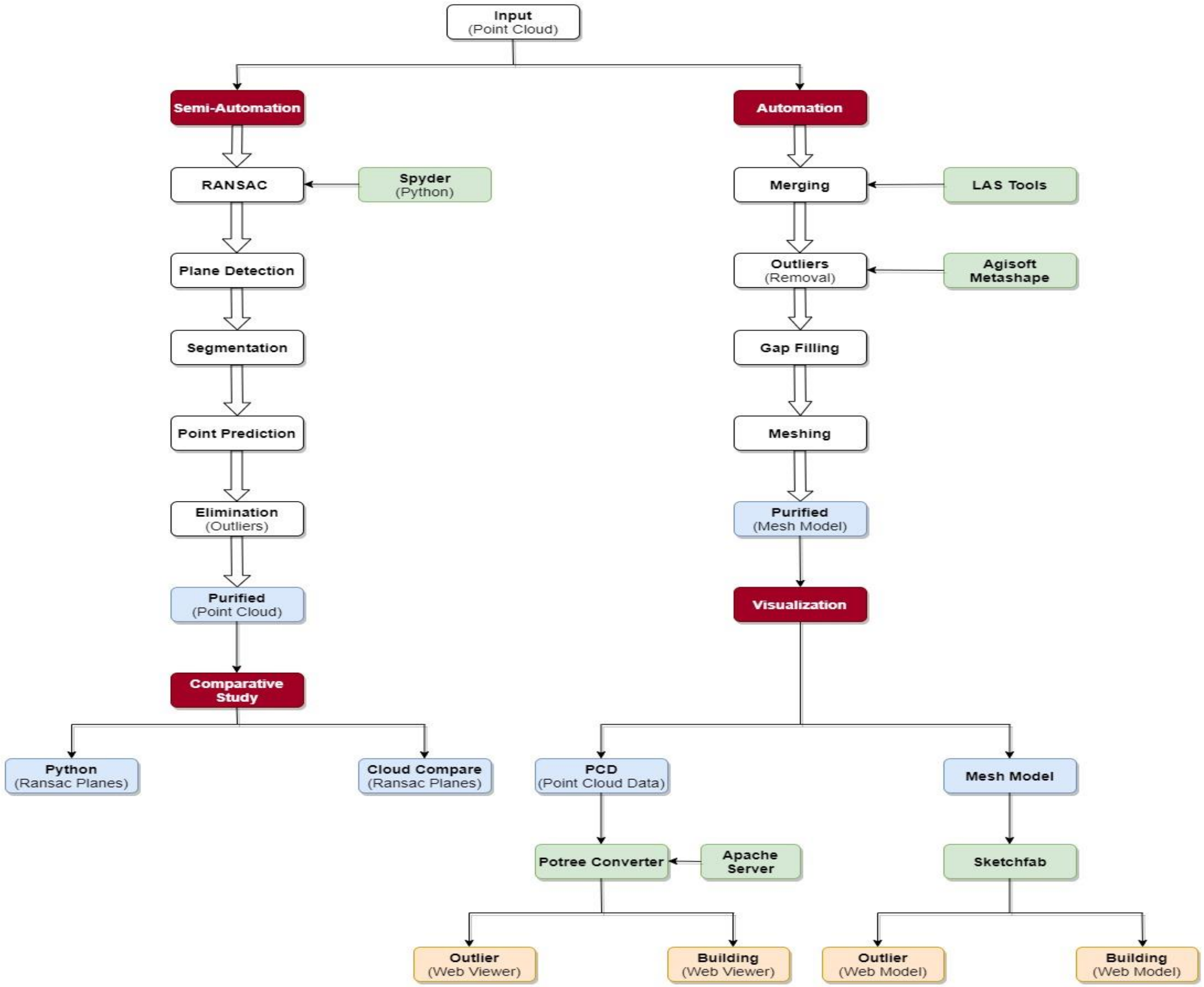
The Sub-objectives are:

- Reduction/Segmentation of indoor point clouds using Semi-automation technique
- Reduction/Segmentation of indoor point clouds using Automation technique
- To create a 3D mesh object using Automation technique
- Comparison of finding planes using Python and Cloud Compare
- To create an interactive 3D model using web technology

Research Questions [RQ]

- What kind of algorithm and model will be used for plane fitting on 3D points?
- What are the different criteria to make a plane a wall plane?
- What kind of plane points will be extruded in the semi-automated technique?
- What is the effort and certainty in interactively performing the automated task?

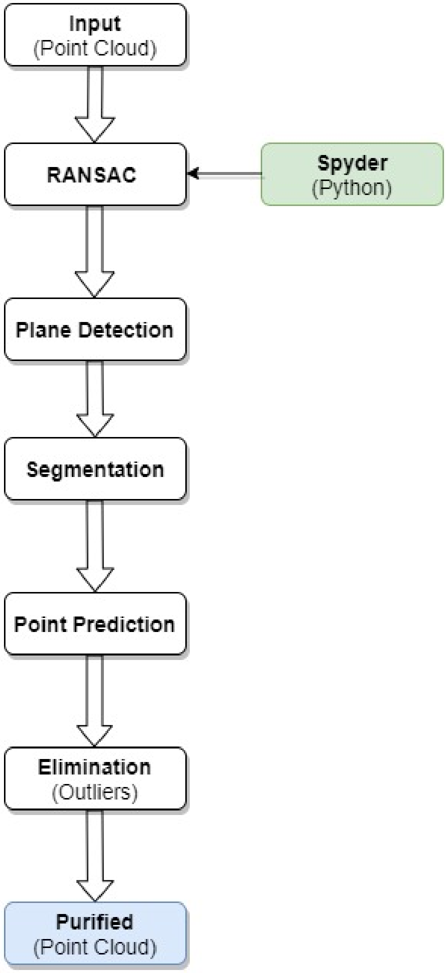
Methodology



Part -1

Semi-Automation

Methodology



RANSAC (Random Sample Consensus)

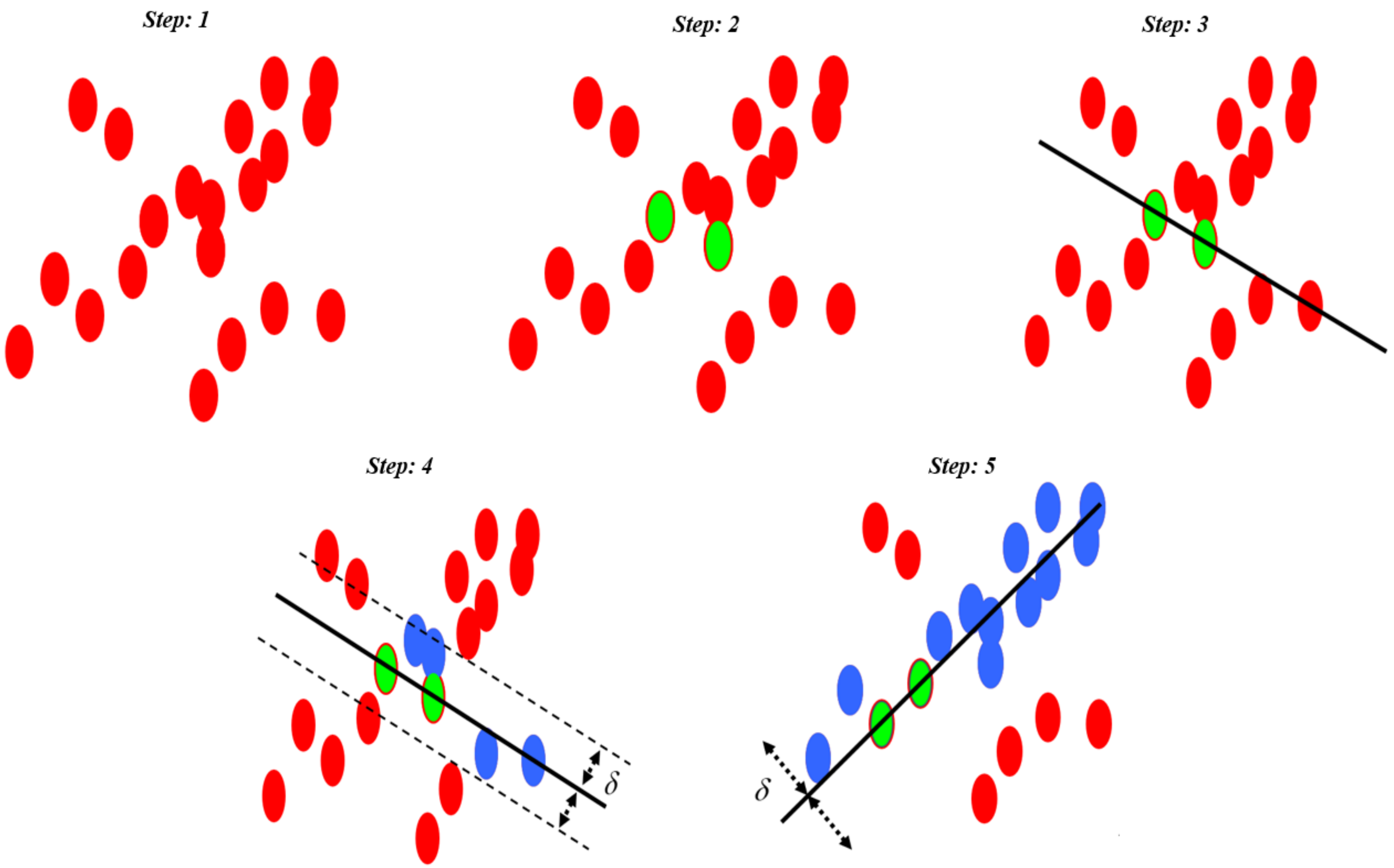
- Ransac stands for Random Sample Consensus is a robust iterative method used to estimate parameters of a mathematical model from a set of observed points which contains outliers and noises.

- Based on the assigned parameters and values the algorithm works. The working principle of the algorithm is so simple.

- For 2D – Line (Uses LSM, LRM)

- For 3D – Plane (Uses LSM, LRM)

RQ-1

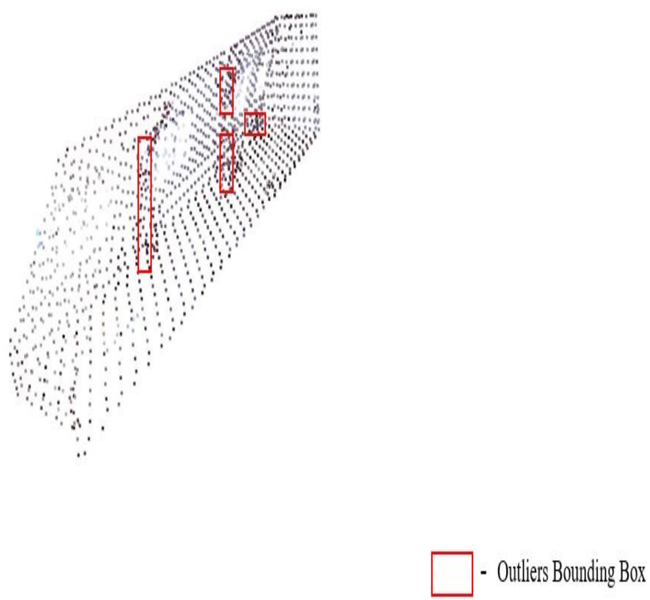


Source: D. Hoiem

Sample 1

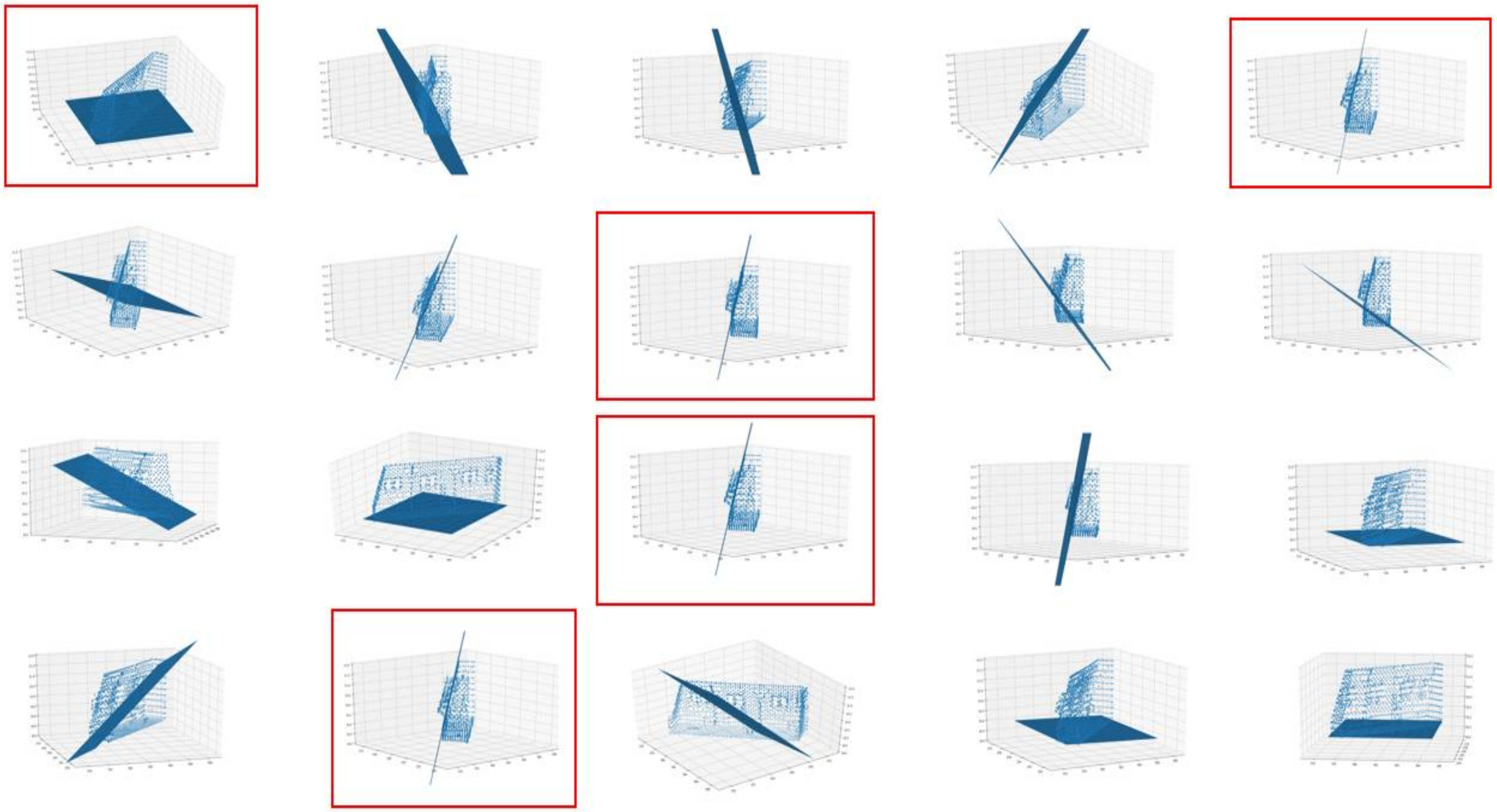
- 5,480,351 Points
- High densely crowded data
- Includes texture color
- Includes vertex normal
- PLY (Polygon File Format)

Down Sampled PCD



- ➢ 1623 Points
- ➢ Includes texture color
- ➢ Includes vertex normal
- ➢ PLY (Polygon File Format)

Iterations



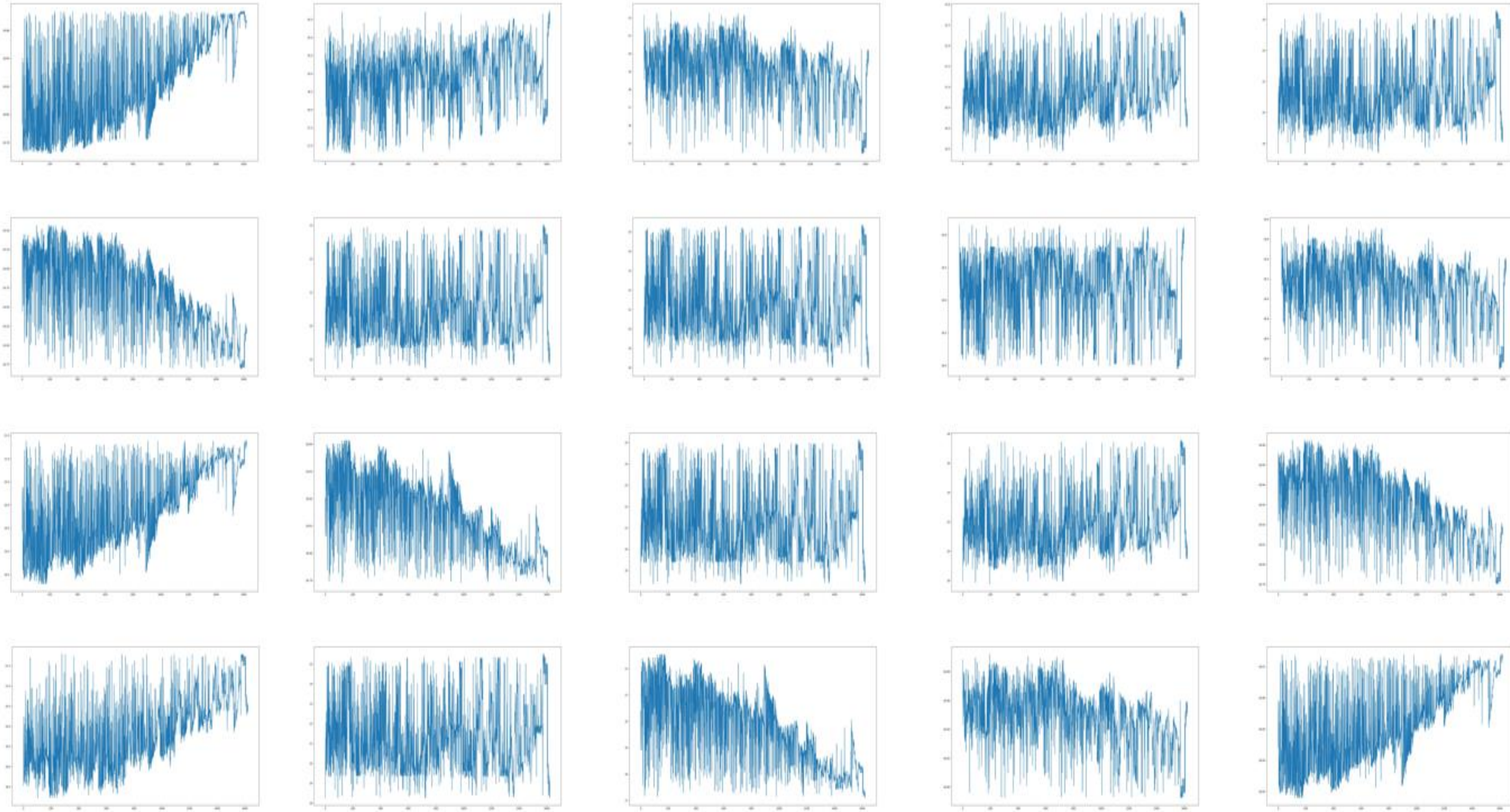
Plane Criteria & Point Prediction

Research Defined Criteria	Iterations
High Favourable Planes	1, 5, 8, 13, 17
Medium Favourable Planes	7, 12, 15, 19, 20
Low Favourable Planes	2, 3, 4, 6, 9, 10, 11, 14, 16, 18

Consideration: High Favorable Planes based on suitable iterations

RQ-2

Graphs



Predicted Points

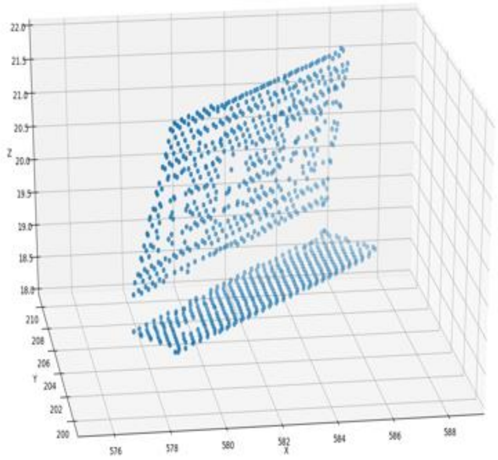
XYZ_Values - Notepad

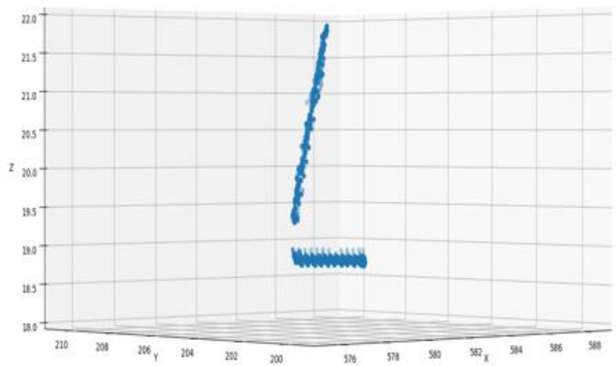
File Edit Format View Help

586.3786532	207.5022181	18.80230232
586.6089998	207.5752209	18.80234361
586.0969425	207.4971569	18.80255065
586.3078998	207.2989081	18.80263167
585.788183	207.7982148	18.8033144
586.0731647	207.8079429	18.80351988
586.0827813	207.224109	18.8035522
585.7856814	207.50632	18.80443239
586.680961	207.8012796	18.80446763
586.3856066	207.8047539	18.80480102
585.7895502	208.1165499	18.80530404
585.7748355	207.1982755	18.805547
585.9927285	206.9901602	18.80568713
586.6781831	208.1111554	18.80595955
586.087028	208.1189925	18.80598434
586.9353776	207.8830708	18.80620978
585.4731639	207.8033124	18.80630295
585.4709784	207.5056199	18.80666857
585.4765233	208.1106108	18.80671911
586.3807067	208.1040644	18.8070605
585.7774911	208.3981638	18.80718033
585.4844038	207.2084555	18.80718961
586.3904568	208.3934781	18.80751575
585.7782705	206.9286649	18.80759412
585.5013869	208.3907625	18.8079367
583.0774201	204.5170554	18.80797445
586.0799372	208.3979056	18.80817383
585.1728859	206.3368985	18.80817635
584.8861734	206.307295	18.80822779
583.0807425	204.7872638	18.80838748
582.7943873	204.7955728	18.80845579
586.0843658	208.7079941	18.80862807
585.179236	208.7097438	18.80876317
585.7839189	208.6963602	18.80877004
585.6897125	206.7050324	18.8089295
582.4851392	205.1035909	18.80907019
584.5824729	206.002941	18.80938301
584.5849058	206.3039035	18.80939702
584.2758712	206.0009231	18.80943317
585.3783098	206.410188	18.80946185
582.7904472	204.5075195	18.80952153
582.4853883	204.8043868	18.80959524
584.5502399	205.7304477	18.80961445
585.4855297	208.7037558	18.8096716

Ln 1, Col 1 100% Windows (CRLF) UTF-8

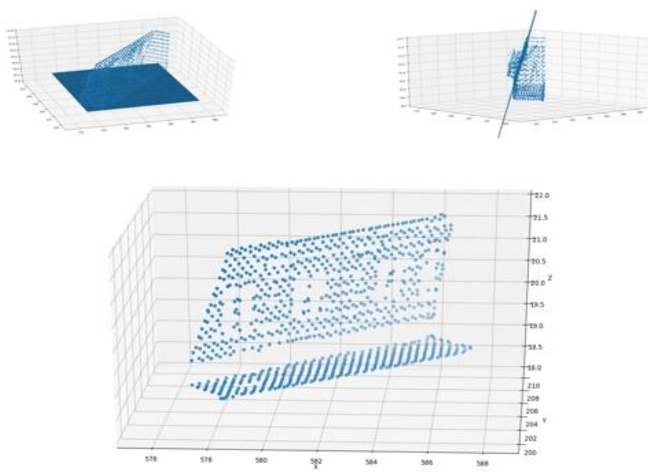
Output



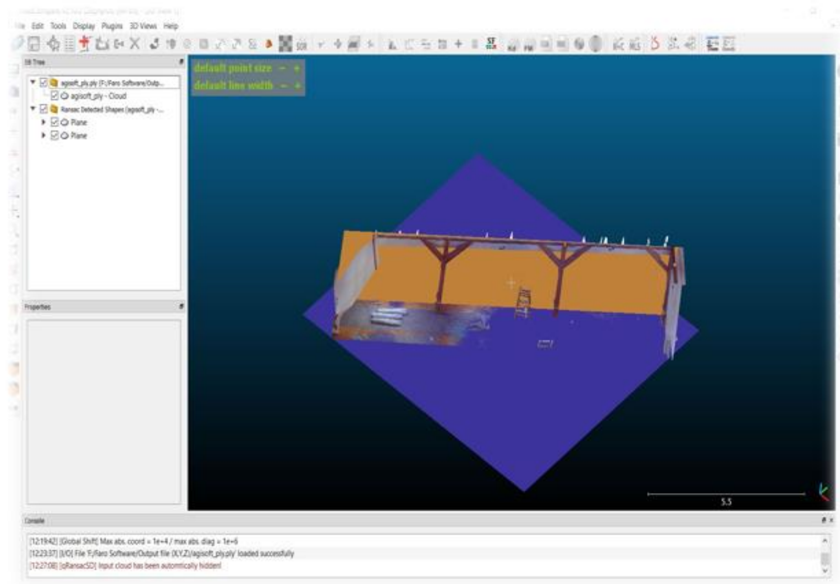


- 910 Points
- Iterations considered – 1, 5, 8, 13, 17
- Outlier Removal – 100%
- Major Planes - 2

Comparative Study (Sample 1)



Python



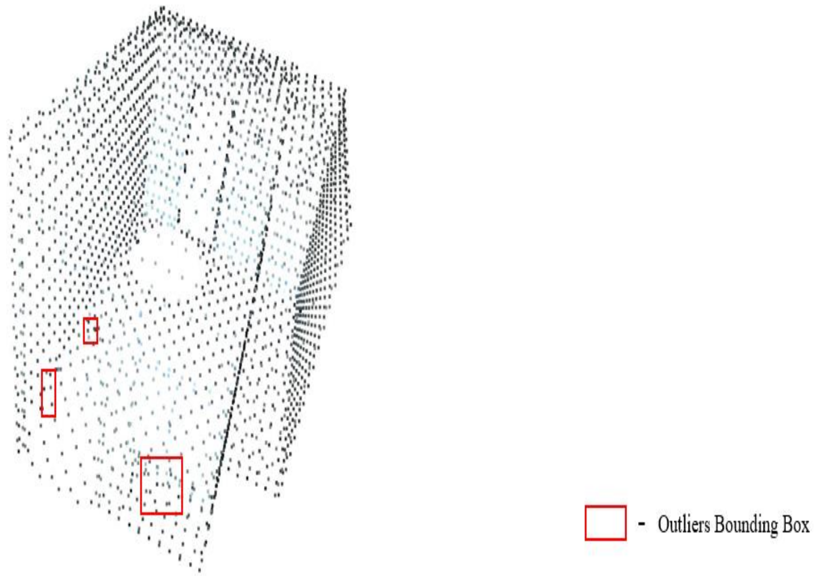
Cloud Compare (CC)

- ➢ Two major planes found
- ➢ Result output intersection (Visual Cross Validation) is 100%
- ➢ Outlier/Obstacles removal is 100%

Sample 2

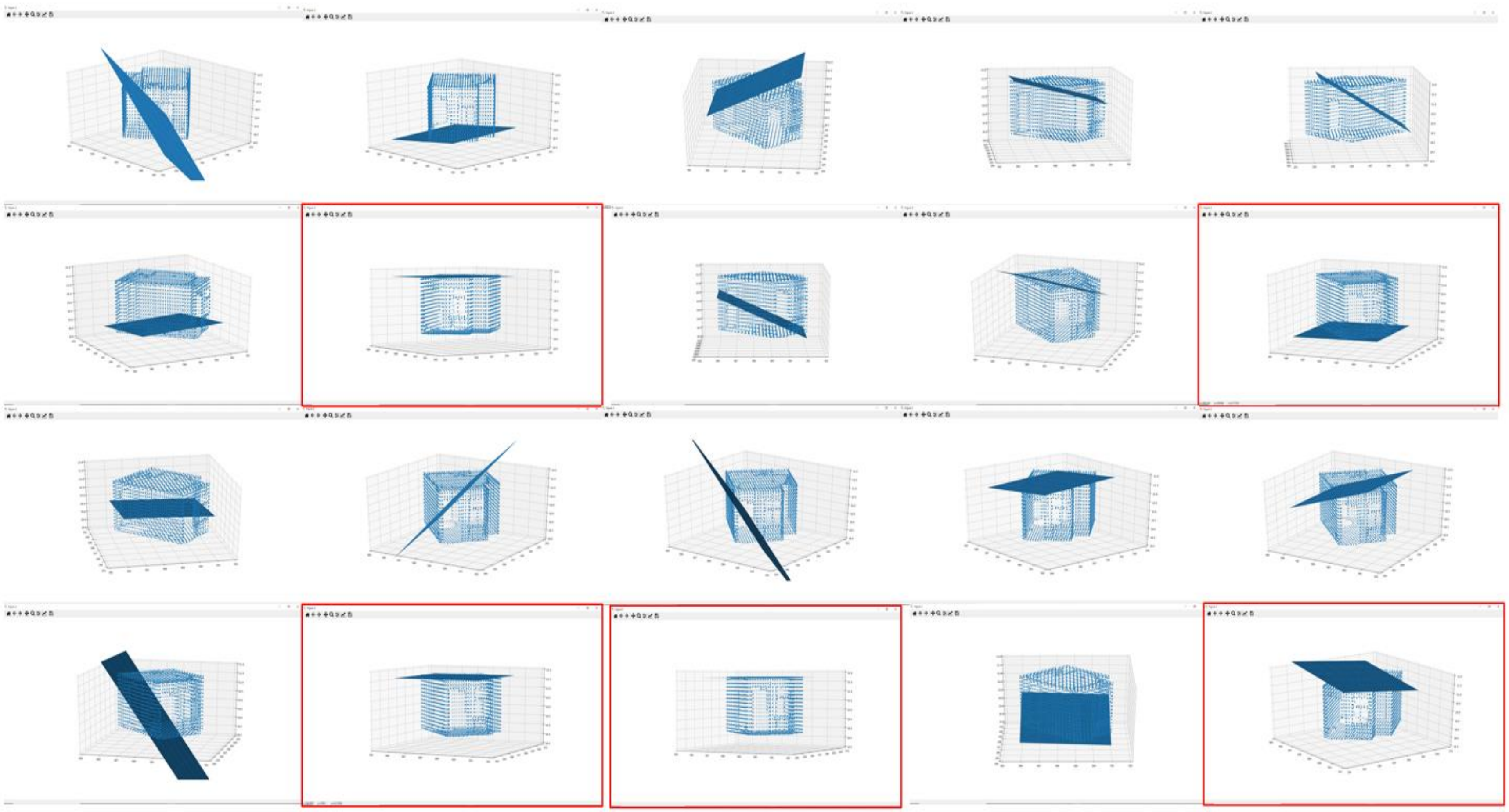
- 40,549,076 Points
- High densely crowded data
- Includes texture color
- Includes vertex normal
- PLY (Polygon File Format)

Down Sample PCD



- 2,524 Points
- Includes texture color
- Includes vertex normal
- PLY (Polygon File Format)

Iterations



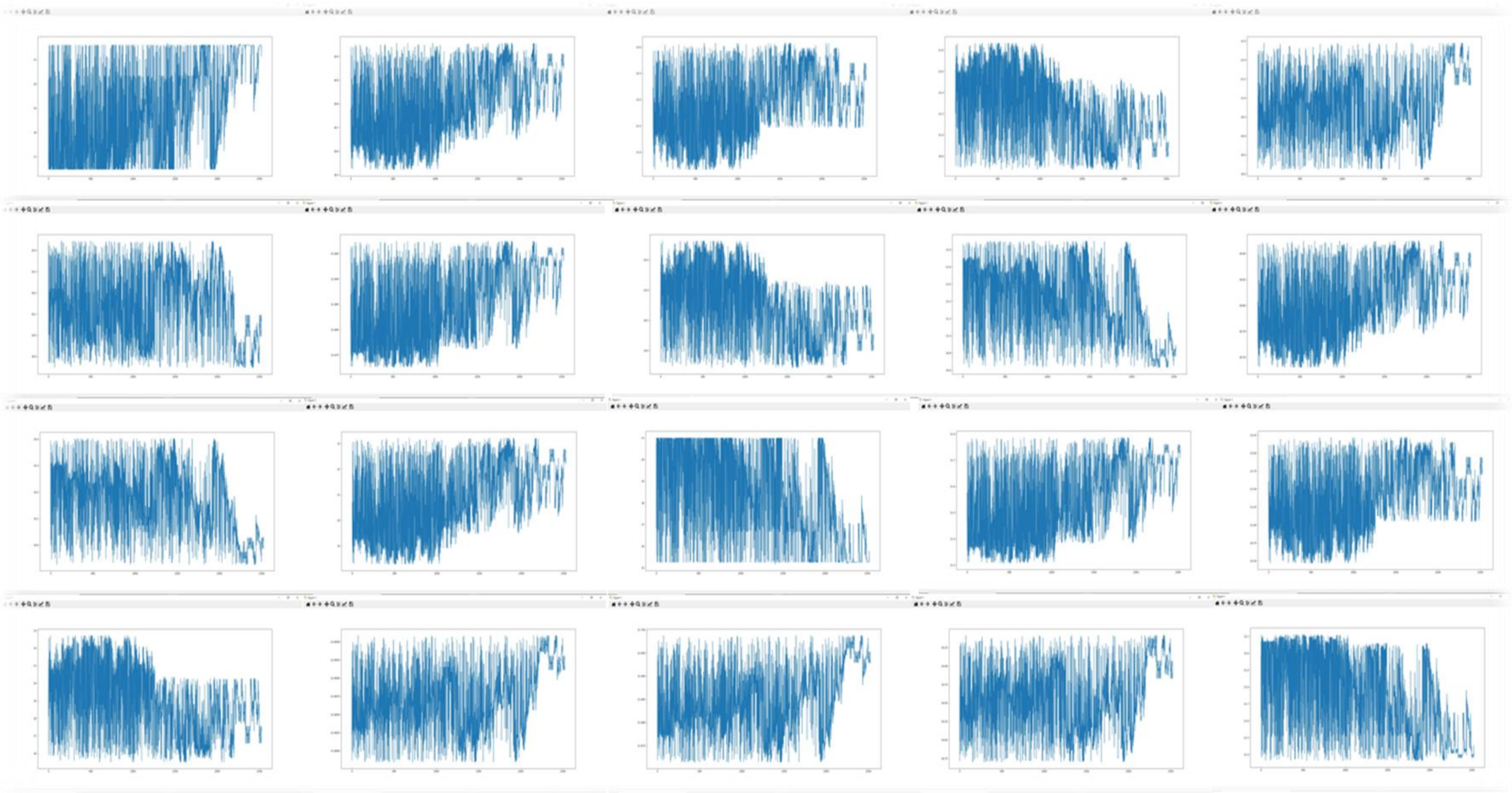
Plane Criteria & Point Prediction

Research Defined Criteria	Iterations
High Favourable Planes	7, 10, 17, 18, 20
Medium Favourable Planes	2, 6, 14
Low Favourable Planes	1, 3, 4, 5, 8, 9, 11, 12, 13, 15, 16, 19

Consideration: High Favorable Planes based on suitable iterations

RQ-3

Graphs



Predicted Points

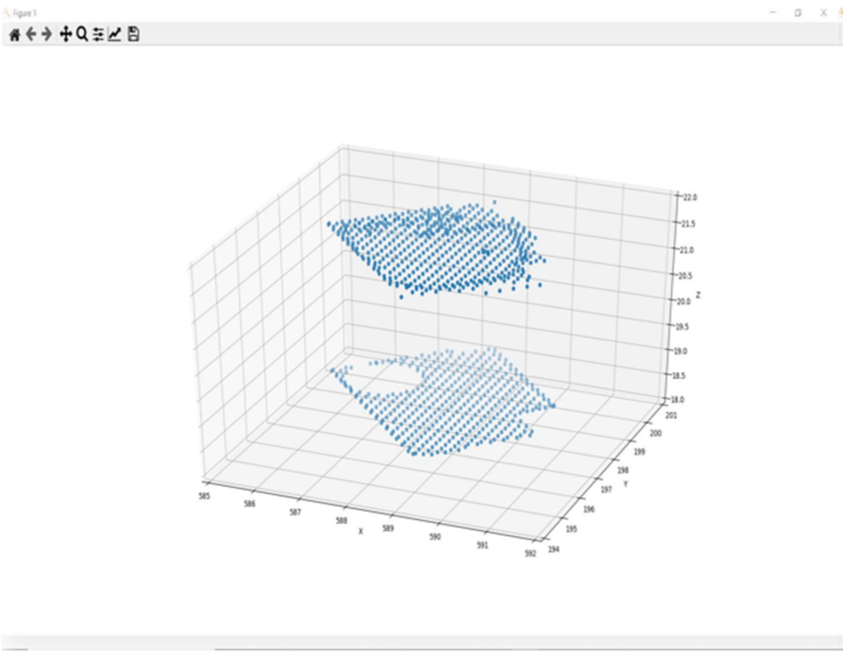
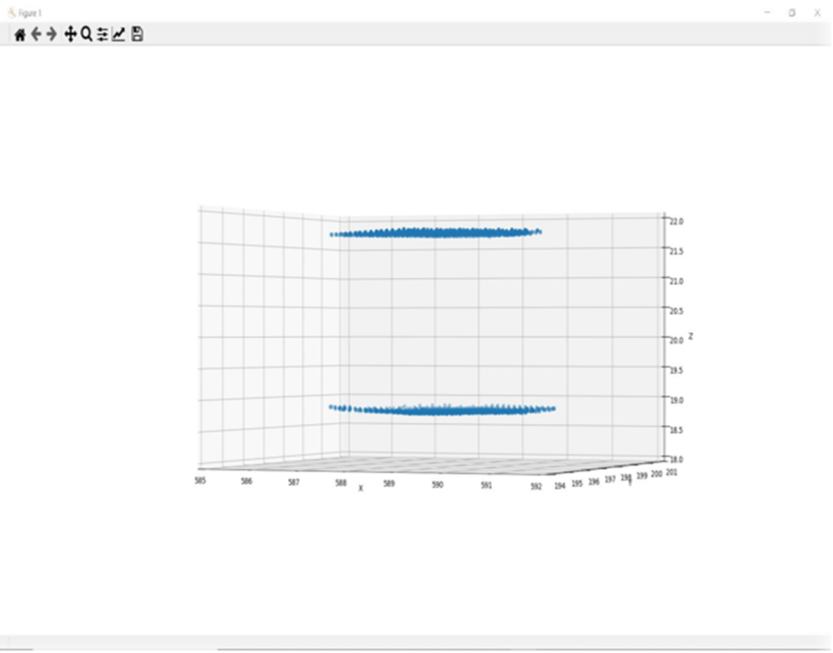
XYZ_Values - Notepad

File Edit Format View Help

588.5075623	198.5751222	18.79316313
588.1062539	197.9768265	18.79320145
588.1060664	198.1758458	18.79346467
588.5068852	198.3755128	18.79349011
587.939721	197.9770897	18.79355202
588.3065781	198.1754785	18.79367729
588.3066405	198.3748266	18.79413832
588.7076354	198.5750307	18.79419643
588.5042173	198.7809559	18.79471626
587.9237653	198.1817903	18.79473102
587.926232	197.7720012	18.79477685
588.3065	197.976928	18.79483572
588.7074617	198.775253	18.79501881
588.5068685	198.1756138	18.79503136
588.1065139	198.3747322	18.79520546
588.306439	198.5753906	18.79522833
588.7070705	198.3761357	18.79547423
588.1060849	197.7777963	18.79572865
588.5077706	197.3784693	18.79593312
588.507423	197.1782534	18.79603266
587.9037722	198.3776303	18.7962673
588.5076832	196.9792058	18.79628876
588.5064905	197.9770001	18.7963328
588.5068362	197.5773708	18.79644736
588.3078063	196.9790085	18.79647216
587.7770586	197.5333951	18.79652005
588.3065753	197.777616	18.79665771
588.7071515	197.5773062	18.79668397
588.9075755	198.7752901	18.79674386
588.7078027	196.9784846	18.79678355
588.7072968	197.3781079	18.79679198
588.3079722	198.7697104	18.79680714
588.7071354	197.7767956	18.79681051
588.5067352	197.7773157	18.79685069
587.906239	197.5785583	18.79690485
588.7072613	198.1759636	18.79693009
588.3081892	196.7794498	18.79693691
588.5084581	196.5794411	18.79693691
588.50789	196.7791327	18.79695208
588.7071386	197.1786096	18.79695512
588.7083523	196.5787006	18.79701489
588.7083149	196.7793943	18.79705963
588.3074509	197.1793063	18.79707991
588.108458	196.7797244	18.79727801

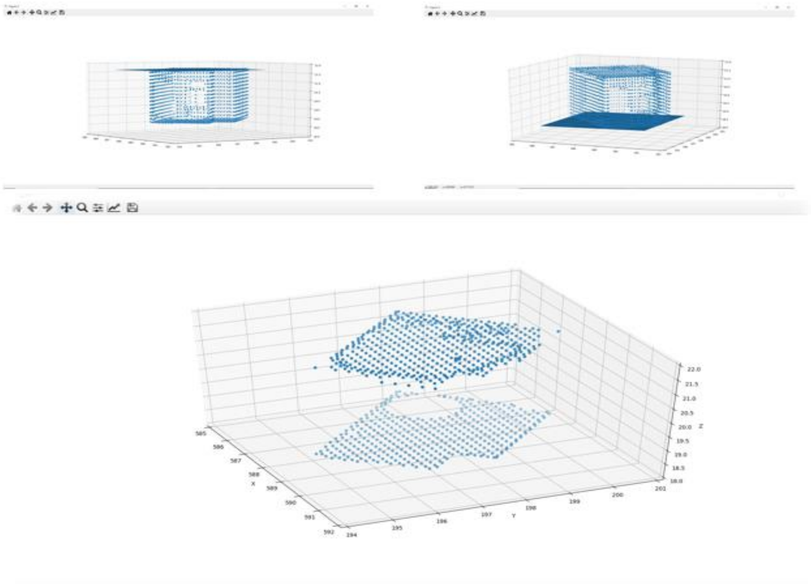
Ln 897, Col 2 | 100% | Windows (CRLF) | UTF-8

Output

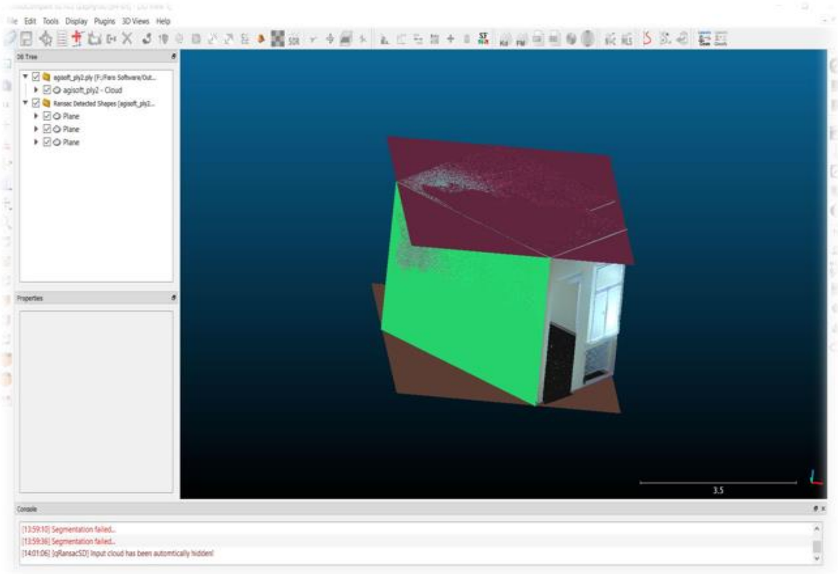



- ➢ 897 Points
- ➢ Iterations considered – 7, 10, 17, 18, 20
- ➢ Outlier Removal – 50%
- ➢ Major Planes - 2

Comparative Study (Sample 2)



Python



Cloud Compare (CC)

- Two major planes found instead of 4 planes
- Result output intersection (Visual Cross Validation) is 50%
- Vertical wall plane couldn't be identified in Python **(Algorithm Limitation)**

Algorithm Limitations

- The Ransac planes are difficult to predict for the 90° vertical walls. The algorithm uses arctan2 math for the plane angle, but failed to predict vertical geometry planes (walls).

- Each iteration for Ransac algorithm is done manually, due to the absence of loop concept in the algorithm.

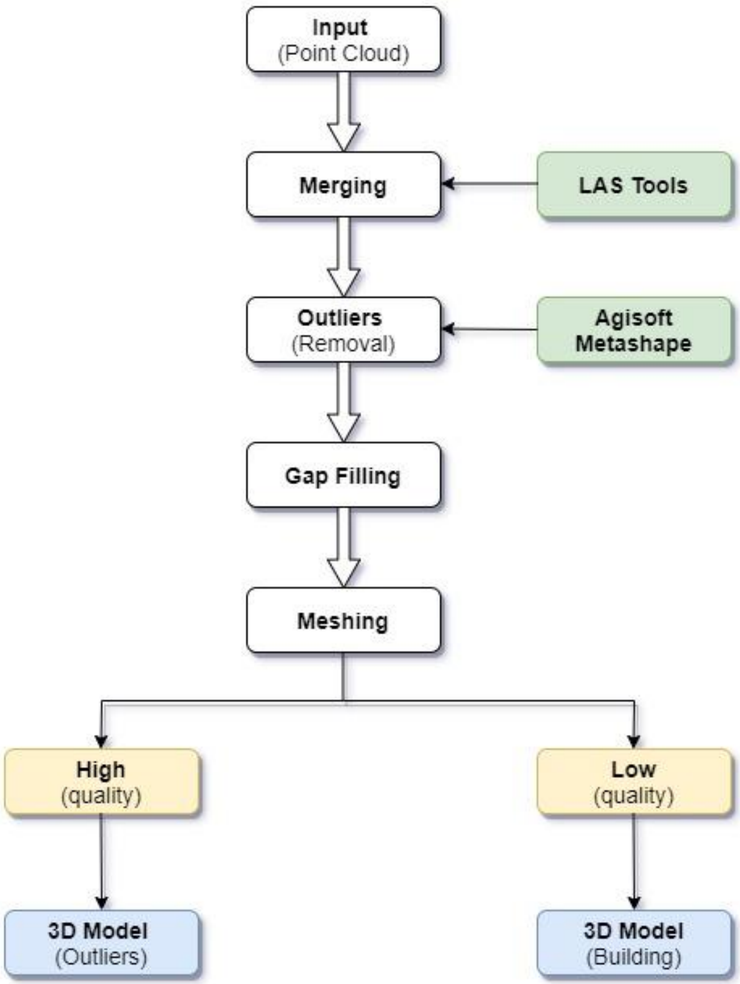
- Each iteration can fit only one single Ransac plane on the Point cloud data.

- Multiple plane fitting on the point cloud data using the defined Ransac algorithm is not possible.

Part -2

Automation

Methodology



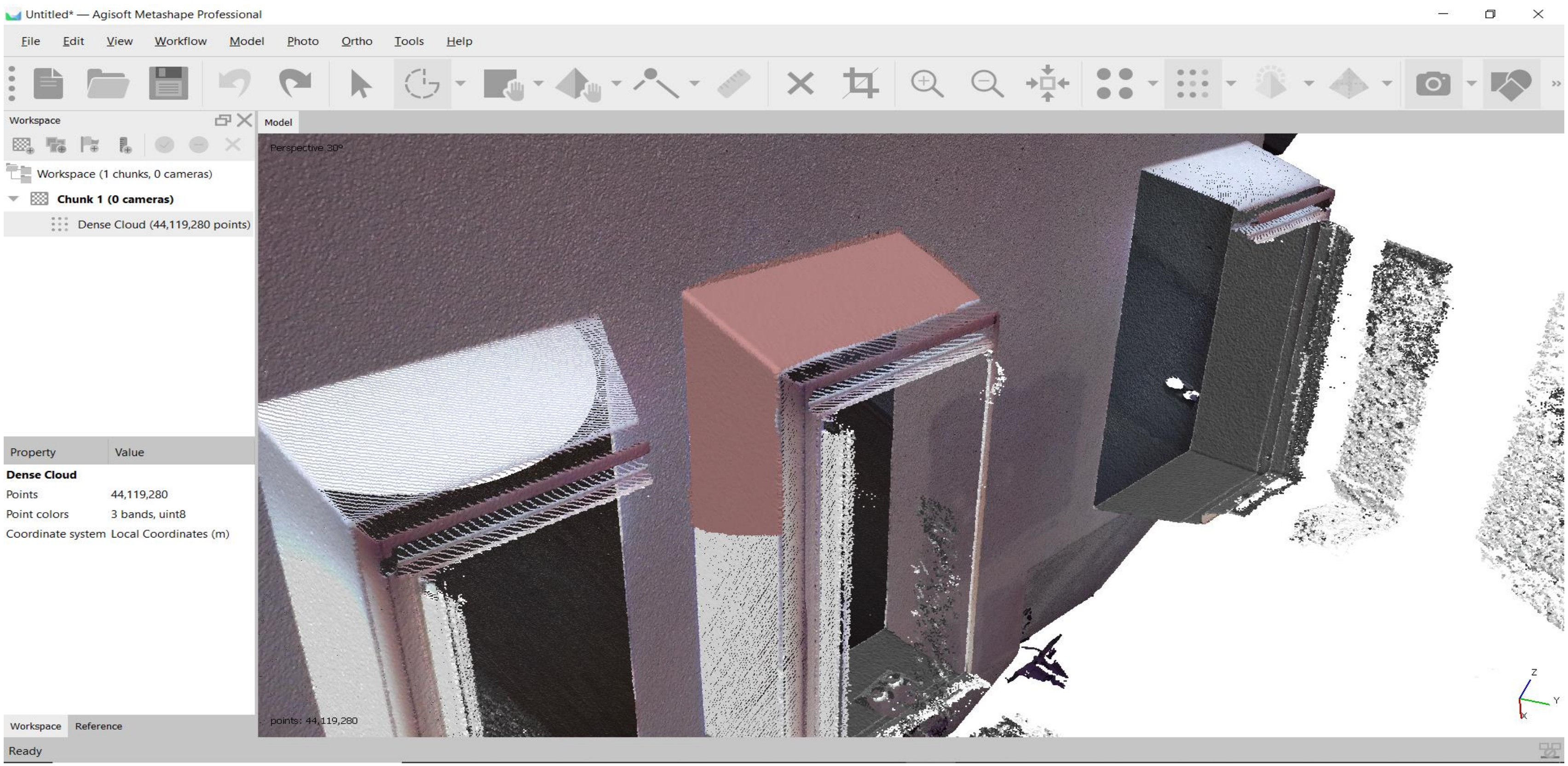
Merging PCD

```
Select M:\Rajasthan\New folder (4)\results\results\New folder\bin\lasmerge.exe
Note that not all of LAStools is "free" (see http://lastools.org/LICENSE.txt)
contact 'martin.isenburg@rapidlasso.com' to clarify licensing terms if needed.
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
GeographicTypeGeoKey: look-up for 0 not implemented
lasmerge -lof file_list.19452.txt -o "merged.laz"
```

LAS Tools (Las Merge)

Visualization of Merged Point cloud data

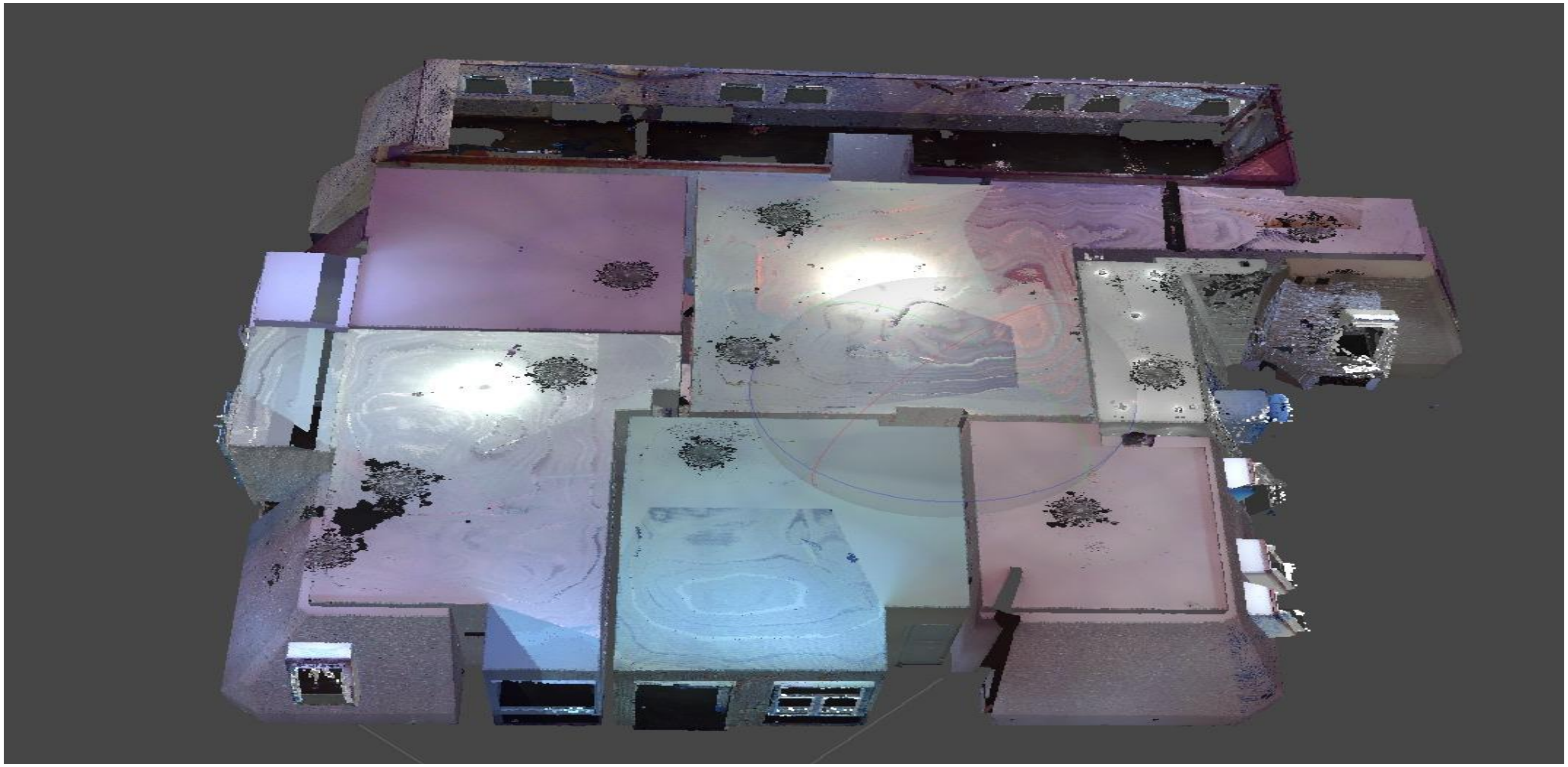
Outlier Removal



Outlier Selection (PCD)

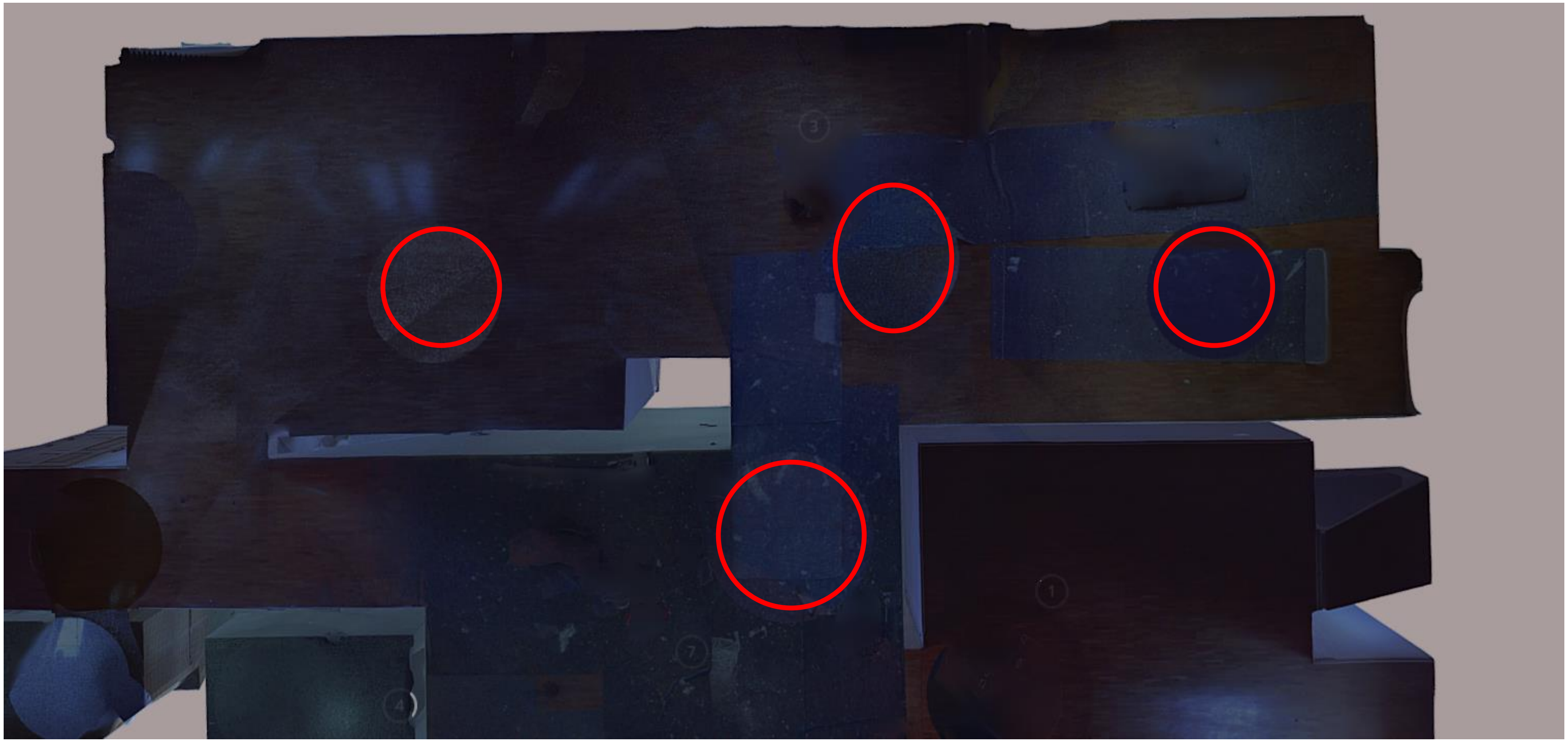
Visualization of Point Cloud Outliers

Purified PCD



Visualization of purified point cloud

Gap Filling

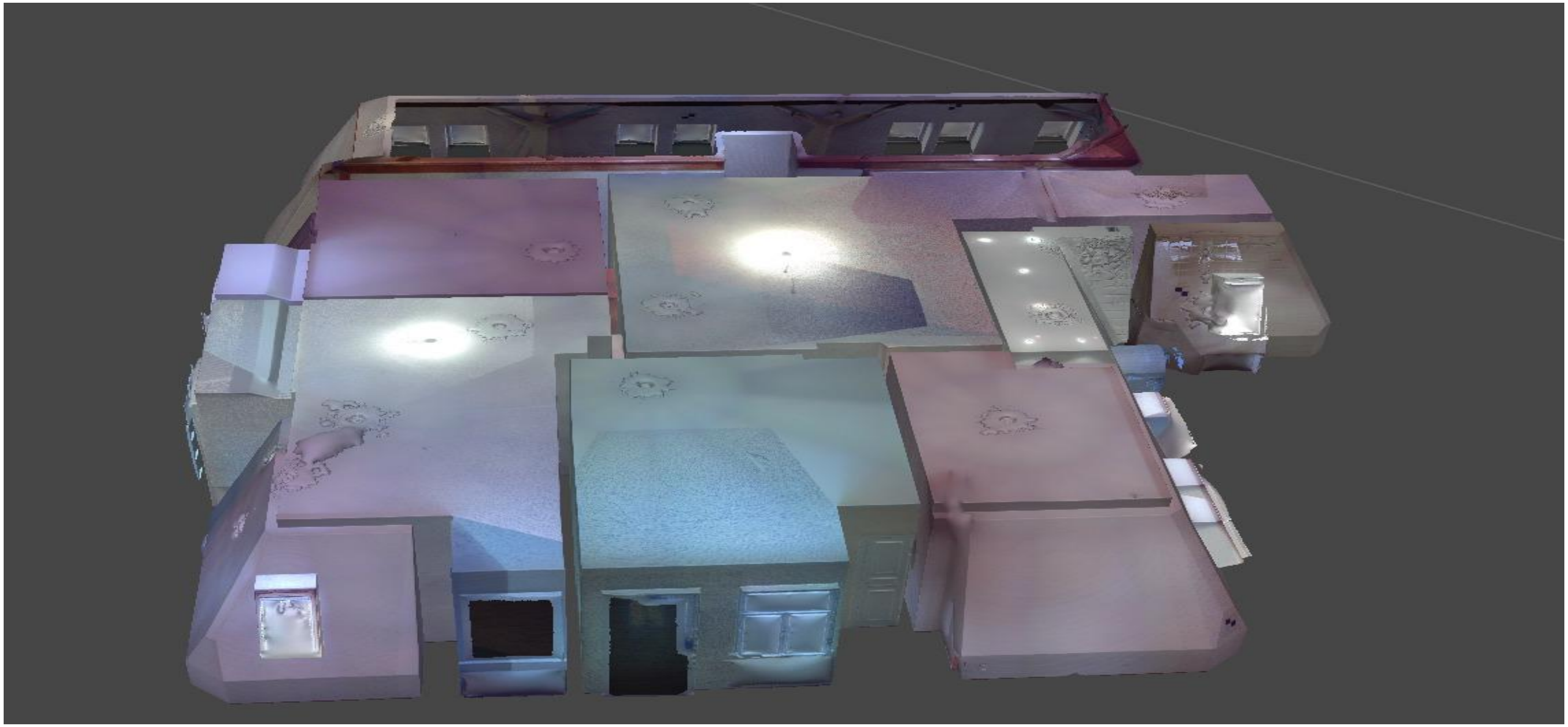


Visualization of filled Tripod gaps (Trilinear Interpolation)

Mesh Model

Visualization of static Mesh model (Outliers)

Mesh Model



Visualization of Static Mesh Model (Building)

RQ-4

Part -3

Visualization

Part -A

Point Cloud Visualization

Potree Converter

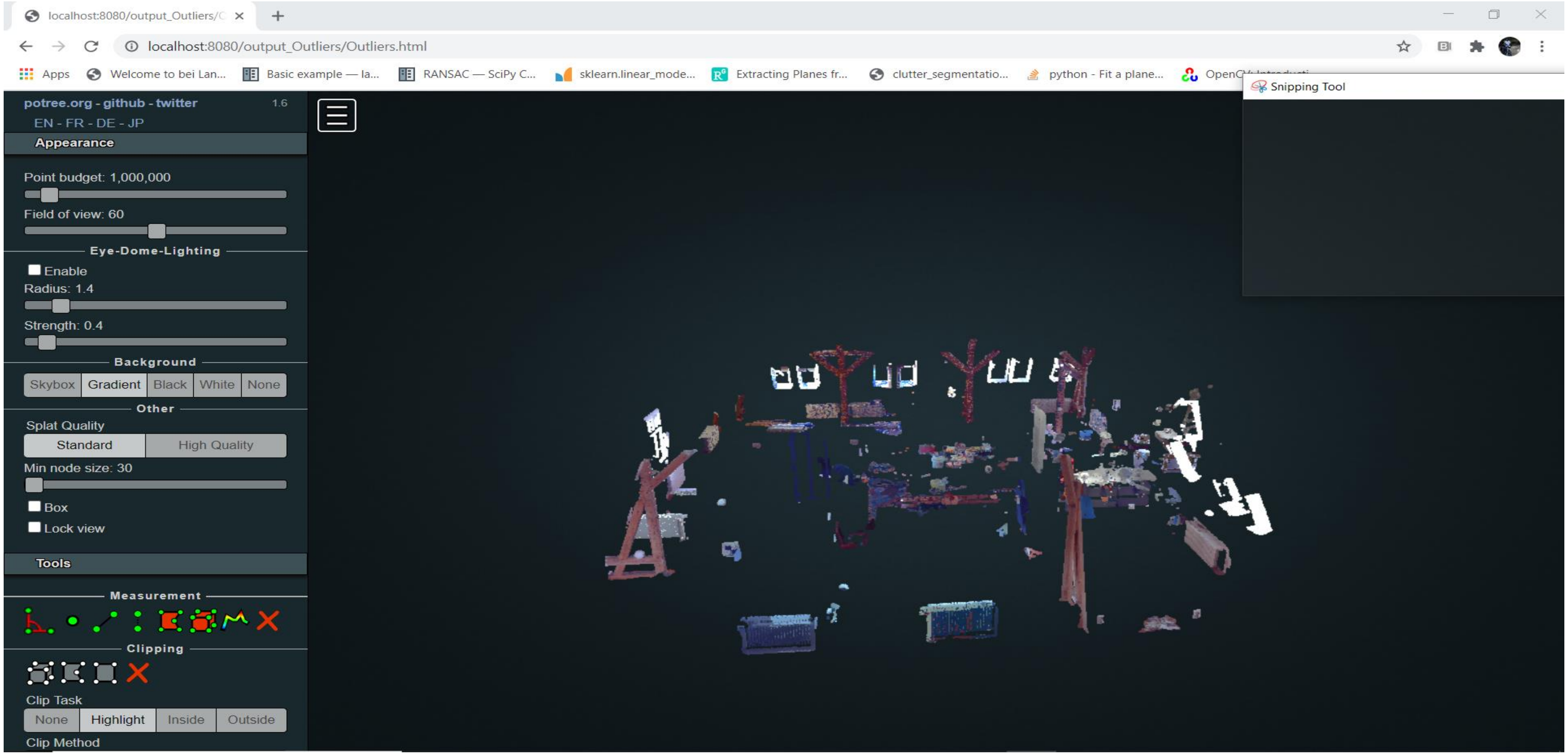
- The Potree Converter is an open source online based point cloud viewer used for measuring, projecting and visualizing the point clouds on web.
- The Potree Converter normally converts point clouds to a format that is compatible with the Potree viewer.
- The Potree viewer helps to visualize the converted point clouds on web with high-end functionalities and operations.
- The Potree Viewer uses Apache server as a backend for accessing the point cloud data.

Source: Potree.org

Visualization Elements

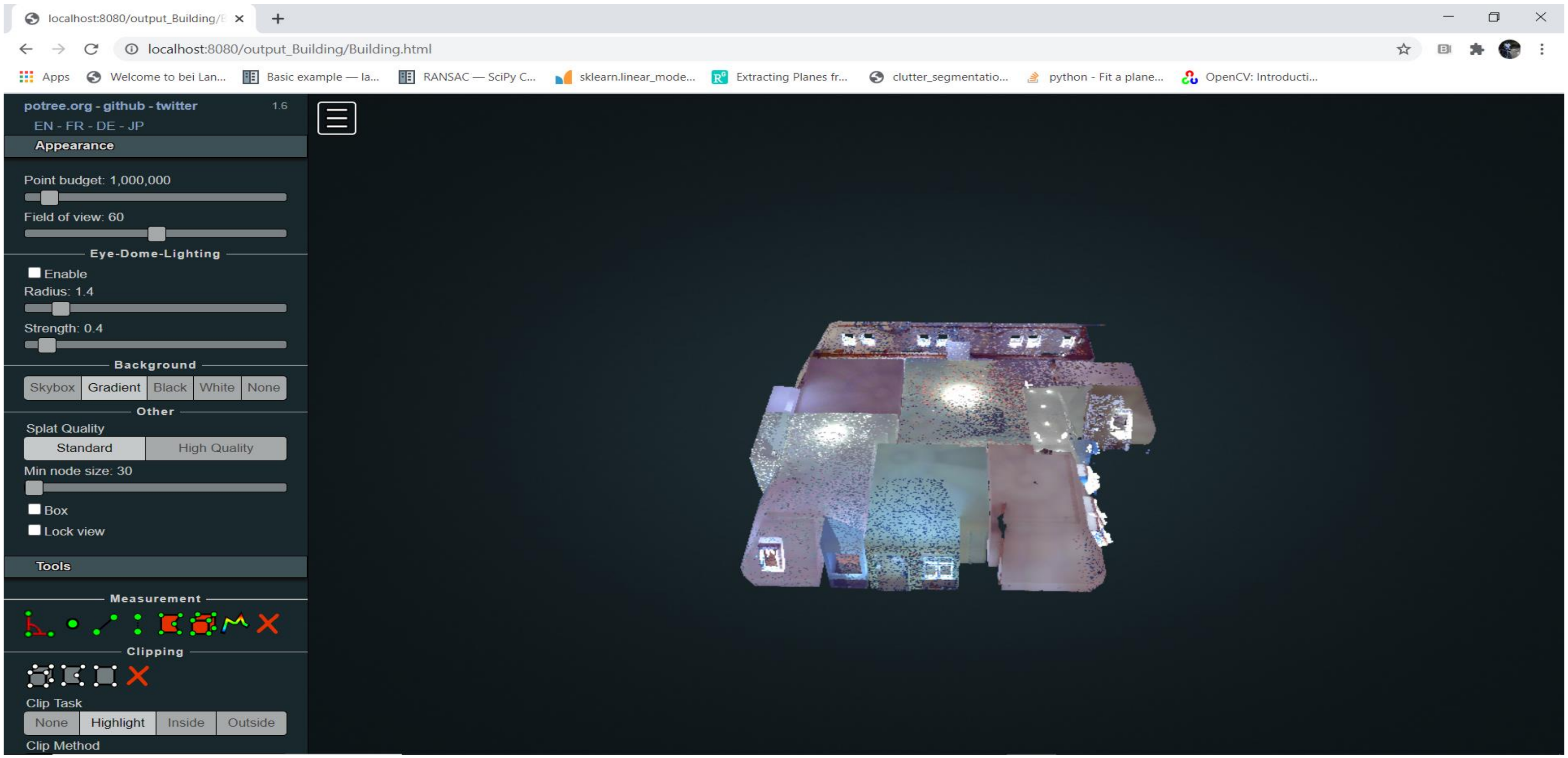
Visualization Elements	Description
Appearance	Provides functionalities on point cloud budget and field of view angle
Eye-Dome-Lighting	Provides functionalities on dome lighting effect on point clouds with radius and strength parameter
Background	Provides functionalities on background colour change
Splat Quality	Provides functionalities on point cloud quality and node size
Measurement Tools	Provides functionalities on point cloud measurement (point, line, height, volume and area)
Clipping Tools	Provides functionalities on clipping of point clouds (box clip, polygon clip and volume clip)
Navigation	Provides functionalities on point cloud navigation (earth control, fly control, orbit control etc)
Projection	Provides functionalities on different camera projection of point clouds (perspective & orthographic)
Scene Export	Provides functionalities on export of point clouds as a scene (JSON, DXF)
Classification Filter	Provides functionalities on visualizing the classified point cloud data (building, trees, forest, roads etc)

PCD Web Viewer



Visualization of PCD (Outliers)

PCD Web Viewer



Visualization of PCD (Buildings)

Part -B

Mesh Model Visualization

Sketchfab

- Sketchfab is an online platform to publish, share, discover, buy and sell 3D, VR and AR model contents.

- It provides a 3D model viewer based on the WebGL and WebVR technologies that allows users to display 3D models on the web.

- The models can be viewed on any mobile browser, desktop browser or Virtual Reality headset.

Source: Wiki

Visualization Elements

Visualization Elements	Description
Navigation	Provides functionalities on model navigation (orbit view & first-person view)
Model Inspector	Provides functionalities to inspect the models (geometry, Material channels, render, viewport & object colour)
Annotations	Provides functionalities to annotate the mesh models using numbering technique
Virtual Reality	Provides functionalities to visualize the model using Virtual Reality technology
Help	Provides functionalities on control options help settings

Model Web Viewer

Visualization of Mesh Model (Outliers)

Link to Access: https://skfb.ly/6TAwr

Model Web Viewer

Visualization of Mesh Model (Building)

Link to Access: **https://skfb.ly/6TCqT**

Conclusion

- The application use of both semi-automation and automation technique on point clouds has shown an innovative and ingenious idea for solving the research tasks and objectives of the research in a very easy and highly efficient way.

- For Semi-automation technique, the use of defined computer vision algorithm with point cloud data (limitations) provides less accuracy and correctness when compared to the inbuilt software algorithms (automation).

- For Automation technique, the use of proper and valid combination of software’s and tools provides high accuracy and correctness on detecting and extracting the purified room geometries from outliers and presenting the output models with high interactive visualization elements.

- From the research, the automation technique has gained an upper hand over the semi-automation technique due to its algorithm limitations.

Future Works

- The prediction of vertical wall planes and extraction of predicted wall points from suitable planes.

- The implementation of loop concepts and automatic extraction of points from suitable planes.

- The implementation of Deep Neural Networks (DNN) concept for high accuracy and correctness on fast and correct plane extraction.

References

- Li, L., Yang, F., Zhu, H., Li, D., Li, Y., & Tang, L. (2017). An Improved RANSAC for 3D Point Cloud Plane Segmentation Based on Normal Distribution Transformation Cells. Remote Sensing, 9(5), 433. doi: 10.3390/rs9050433.

- Bool, D. L., Mabaquiao, L. C., Tupas, M. E., & Fabila, J. L. (2018). Automated Building Detection Using Ransac From Classified Lidar Point Cloud Data. ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII-4/W9, 115–121. doi: 10.5194/isprs-archives-xlii-4-w9-115-2018.

- Kurban, R., Skuka, F., & Bozpolat, H. (2015). Plane Segmentation of Kinect Point Clouds using RANSAC. *The 7th International Conference on Information Technology*. doi: 10.15849/icit.2015.0098.

- Yang, M. Y., & Förstner, W. (2010). *Plane Detection in Point Cloud Data*. (pp. 1-16). (IGG : Technical Report ; Vol. 1, 2010). Bonn: University of Bonn.

References

- Lan, J., Tian, Y., Song, W., Fong, S., & Su, Z. (2018). A Fast Planner Detection Method in LiDAR Point Clouds Using GPU-based RANSAC. *UMCit@KDD*.

- Zeineldin, R.A., & El-Fishawy, N.A. (2016). Fast and accurate ground plane detection for the visually impaired from 3D organized point clouds. *2016 SAI Computing Conference (SAI)*, 373-379.

Any Questions???


ITC
UNIVERSITY OF TWENTE.
TECHNISCHE
UNIVERSITÄT
DRESDEN
Technical
University
of Munich
TUM
TU
WIEN
TECHNISCHE
UNIVERSITÄT
WIEN
Vienna University of Technology