3D Point Cloud (CloudCompare)

Lee Kah Meng[1], Chai Rui Qin[2]
U.G. Student of Electronic Systems Engineering
Universiti Teknologi Malaysia
Kuala Lumpur, Malaysia

***Abstract* — CloudCompare is a programme that processes 3D point clouds such as those created by a laser scanner on the certain projects. The CloudCompare project originated in 2003 with Daniel Girardeau-PhD Montaut's on Change detection on 3D geometric data and was originally conceived as a cooperation between Telecom ParisTech and EDF's R&D division. Its primary objective at the time was to identify changes in 3D high density point clouds recorded with laser scanners in industrial facilities like power plants or construction sites. Following that, it grew into a more wide and powerful 3D data processing application. CloudCompare can handle an unlimited number of scalar fields per point cloud, allowing for the use of a variety of specialised algorithms (smoothing, gradient evaluation, statistics, etc). A dynamic colour rendering technique aids in the efficient visualisation of per-point scalar fields of triangular meshes and calibrated photos**

***Keywords* — CloudComPy, CloudCompare, 3D Point Cloud, python**

1. INTRODUCTION

CloudCompare is a software for editing and processing 3D point clouds and triangular meshes. It was made determined to do coordinate examinations between thick 3D point clouds. It depends on an octree structure that considers magnificent execution when directing this kind of activity. Furthermore, because the majority of point clouds are collected using terrestrial laser scanners, CloudCompare was designed to handle large point clouds on a basic laptop generally over 10 million points. Many other point cloud processing algorithms as well as display enhancement tools followed by registration, resampling, colour/normal vectors/scalar fields management, statistics computation, sensor management, interactive or automatic segmentation, etc.

2. PROBLEM BACKGROUND

2.1 PROBLEM STATEMENT

In the earlier, it was difficult for engineers and architects to characterise the 3D modifications and internal structure of a specific item. As a result, they have encountered several concerns and issues while constructing the project. However, using CloudCompare can instantly detect changes in 3D high density point clouds gathered with laser scanners in industrial facilities or construction sites. CloudCompare also includes editing and rendering capabilities for 3D point clouds and triangular models. It also includes a number of complex processing algorithms for a variety of building applications.

2.2 OBJECTIVES

Our objective to carry out this assignment are

1. To implement the CloudCompare to show the surface of an object and data of any internal features, colour and materials.
2. To understand the fundamental tools for manipulating and rendering 3D point clouds and triangular models manually.
3. To measures based on point cloud projections computed on the 2D spaces onto which the points are projected.
4. To monitoring or optimizing image, construction and systems of the objects.

3. METHODOLOGY OR PROCEDURE

3.1 INSTALLATION OF ANACONDA

Anaconda is software that includes conda and Anaconda Navigator as well as Python and hundreds of scientific packages. Conda works with command line interfaces such as Anaconda Prompt for Windows, Linux, and terminal for macOS. Navigator is a desktop graphical user interface that allows you to launch applications and easily manage conda packages, environments and channels without using command-line commands. There is a certain Python version and a set of packages in an environment as shown in Figure 3.3.1. Hence, Anaconda can create an environment for CloudComPy in Anaconda3 from the ubuntu terminal. Anaconda supports multiple versions of Python and associated packages. A Python or R environment will typically include various packages and a version of Python.

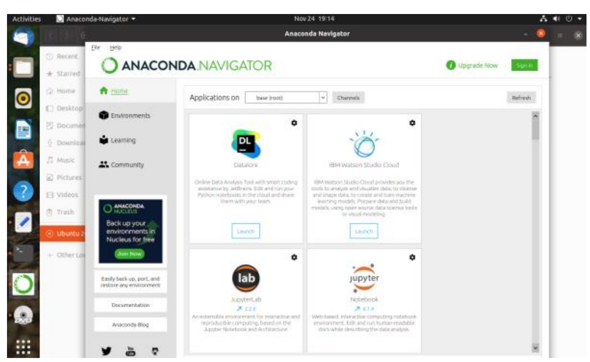

Figure 3.1.1: Installation of Anaconda

3.2 INSTALLATION OF CLOUDCOMPY

CloudCompare is a software that edits and processes 3D point clouds and triangular meshes. It was intended to do coordinate analyses between dense 3D point clouds. The result of conda list command is provided in the sources in building directory will show in Figure 3.2.1. The installation of the binary in the directory is needed based on the requirements. It is necessary to load the new environment before using CloudCompare or CloudComPy. Then, open CloudCompare and import the point cloud file by selecting files and open, depending on the number of points in the point cloud. From a new prompt is replaced then can proceed to the testing of CloudCompare

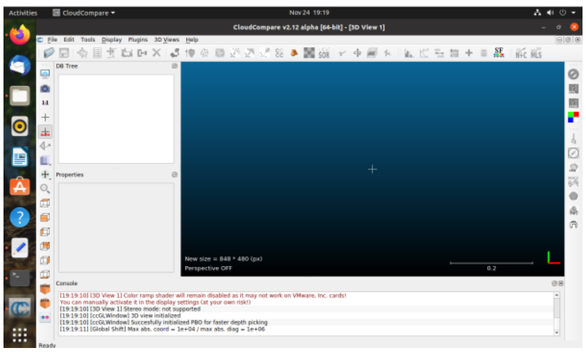

Figure 3.2.1: Installation of CloudCompare

3.3 TESTING CLOUDCOMPY

The test of CloudCompare by open the library of all Python testing folder. The terminal is open by the environment of the CloudCompare as shown in Figure 3.3.1

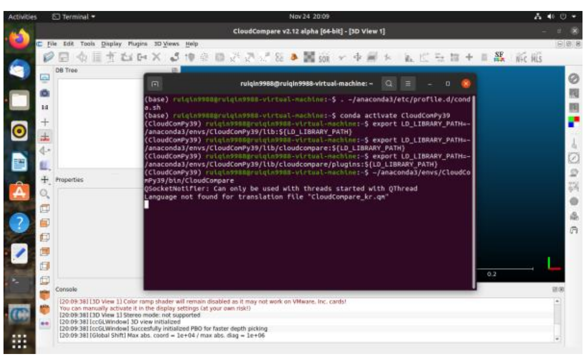

Figure 3.3.1: Testing of CloudCompare

3.3 PYTHON TEST RUN

A Python test is performed by ctest. It takes about a minute or two to load the file and execute all the tests. Then, it is continued with the . envPyCC. This completes the Python path and runs a script.

4. RESULTS AND OBSERVATION

First, three new prompts is replaced in the terminal after the environment is created as shown in Figure 4.1.

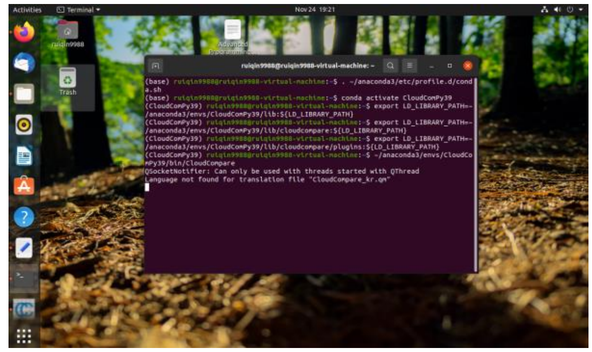

Figure 4.1: Image new prompts in the terminal

After proceeding to the testing of CloudCompare, the 3D electricity pylon (transmission tower) will show as shown in Figure 4.2 and Figure 4.3.

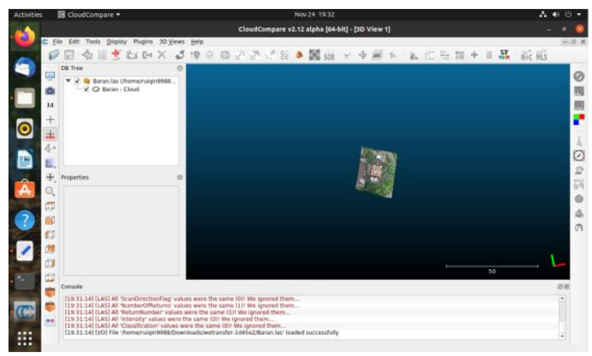

Figure 4.2: 3D Image of electricity pylon (transmission tower) from top view

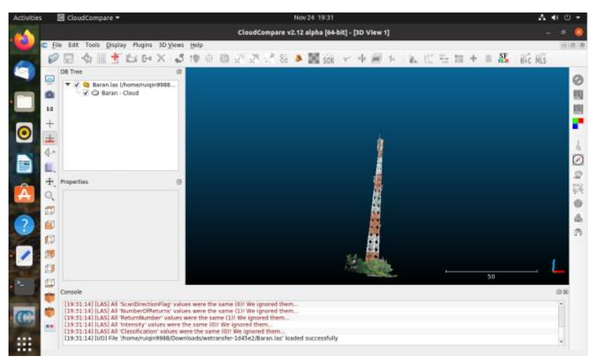

Figure 4.3: 3D Image of electricity pylon (transmission tower) from side view

The PythonAPI_test is open in the terminal and run ctest to execute all the tests. The runtime is reset to obtain the results. The results show as the Figure 4.4 and Figure 4.5.

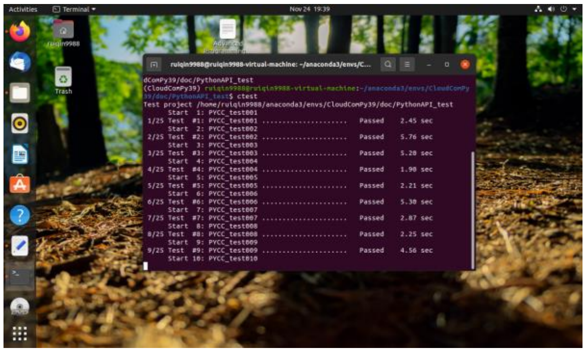

Figure 4.4: The PythonAPI_test and ctest run

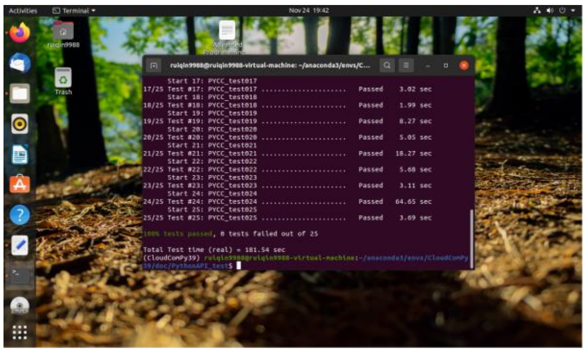

Figure 4.5: The results of PythonAPI_test and ctest run

Then run the . envPyCC and Python test of each. Finally the pythonpath and run a script is completed as the Figure 4.6.

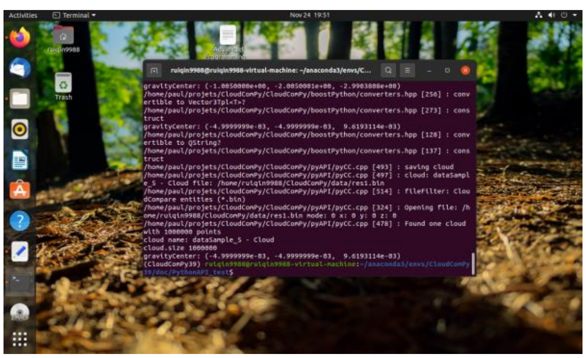

Figure 4.6: The results of example Python test001 files is run

5. CONCLUSION

Based on the assignment, we successfully demonstrated the implement the CloudCompare to show the surface of an object and data of any internal features, colour and materials using Python. At the same time, the basic principle of the fundamental tools for manipulating and rendering 3D point clouds and triangular models is studied to monitoring or optimizing image, construction and systems of the objects. To do that, we can realize the number of test results displayed on the terminal to obtain the data value for each part of the electricity pylon.

6. REFERENCES

[1] Anaconda Installers. Available online: https://www.anaconda.com/products/individual

[2] CloudCompare v2.6.1 - User manual.pdf. Available online: http://www.cloudcompare.org/doc/qCC/CloudCompare%20v2.6.1%20-%20User%20manual.pdf

[3] CloudComPy: Introduction and application use. Available online: https://github.com/CloudCompare/CloudComPy/tree/master/tests

[4] CloudCompare—3D Point Cloud and Mesh Processing Software—Open-Source Project. Available online: http://www.cloudcompare.org