مجلة كلية التربية الاساسية
كلية التربية الاساسية – الجامعة المستنصرية

Journal of the college of basic education Vol.29 (NO. 118) 2023, pp. 37-52

The Preprocessing operation for 3D Indoor and Outdoor dataset

Amani Y. Noori
Mustansiriyah University, Iraq.
amani.yousif@uomustansiriyah.edu.iq

Abstract:

This paper interested with 3D data forms , 3D point cloud are introduced strong realistic description of big different 3D scenes and object , point cloud is extremely high owing jam caused by huge quantity of data generated from laser scanning device, noise points will be appear in scanning devices by factors of environments. All these noise and unwanted data can be very affected modeling. The point cloud preprocessing is analyzing and changing points cloud to optimize its store , transportation and quality using different computational and mathematical algorithm like compression point cloud, which aid to decrease the storage space or cost and time of transmission, Point cloud clustering aid to eliminate noise and outlier from three dimension points cloud, the point cloud downsample is aim to reduce the points number and KDTree used for structure point cloud in efficient order to find nearest neighbors points, The software process 29 million points while still enabling the user to visualize and navigation smoothly through the point cloud This research working on semantic3D outdoor dataset and Stanfored3D indoor dataset, programming with python 3.7, cloud compare free software v2.6.3 and computer with core i7MP , RAM 16 , NVidia RTX 2060.

Keywords: 3D point cloud, Kdtree algorithm, downsample operation, DB scan cluster algorithm, Remove hidden point's algorithm, Open3 library.

Note: the research is based on a doctoral thesis.

1- Introduction

The real word appearance of information is three dimension if visual information is projected into 2D images will miss depth information and distances among objects in truth world, which makes it little appropriate for applications that need locations and depth information, The three dimensions datasets are increasing in recent computerized ecosystem because increasing number of devices capture real world and represents it in different forms of 3D format like 3d point cloud, RGB-D, meshes and multi view [1].

Point clouds are created almost by using "3 dimensions laser scanner and LIDAR (light detection and ranging)" technology, point clouds datasets contains many point of space or objects, for each point three geometric coordinates (X,Y,Z), representing the geometry of a scene. Point clouds are simple to display, filter and edit, it's easy to rotate and scale. The position and color of point is important [2].

Point cloud can be represent using binary and ASCII code, binary files store directly data in form of binary code either ASCII file. It generally contain of billions or millions of points, they are represent real word with accurate digital format. These coordinates equal to the real object size it can do measurements, inventories, and inspections for these values. This technology is particularly well suited for mapping large scale infrastructures such as electrical corridors, highways, railways, urban environments, among others [3]. In Points cloud , "cloud" mean theses point is unstructured and unordered, every point contain both attribute and geometry information, the information of geometry give the point location inform as coordinates (X,Y,Z) Cartesian coordinate system, the information of attribute expressed the visual look for each point and appear at difference forms[3].

The most famous used are (R,G, B) color value and vector (Vk, Vy, Vz) additionally to information of reflectance as 3d point cloud, many operations can done on points cloud like transformations which mean can multiply the points in linear transformation arrays within the point list, combinations means merging points to reshaping objects, Rendering Reconstructed objects through drawing the all points into image plane [4],but the point cloud have some disadvantages as irregular point cloud data as in figure (1.a) the points not divided (sampled) with each regions equally, unstructured which mean each points are examine individually and the spacing between point and the neighboring point is variable not fixed so it is not regularity grid, this contrast with 2D image where the distance between each point and its neighboring is fixed [4] as in figure (1.b).

Unordered, the scene which stored as point cloud is stored as text file and must store in a form not change the represent of scene but it is invariant to permutation as in figure (1.c) [2]. The contribution which introduced in this paper contains:

1- Using two combined algorithms the first one Hidden points remove and Voxel downsampling on 3D point cloud data.
2- Using two combined algorithms the first one Hidden points remove and DBscan cluster on 3D point cloud data.

3- Using two combined algorithms the first one Hidden points remove and KDTree with nearest neighbor on 3D point cloud data.

2- Related Works

In last years, the preprocessor for 3D point cloud dataset increasing and diversity in methods. According to (X. D. Zhang et al., 2004) proposed preprocessing for three dimensions color point cloud dependent on characteristics of laser scanning of data ,to remove noise and redundant data applied manual polygon choosing and Automatic system decision methods . A data less depending on Grid decreasing method that take care of color-boundary, that avert color and shape distortion for reconstruction of model[5]. In (Chao Cao et al., , 2019) the authors, expose the compression methods for 3D point cloud like the 2D approaches and 1D traversal , the 1D traversal used to transform geometry data into a one dimension.

The 2D technique is used multi view representations by projecting the three dimensions point cloud into many images. There is no information about connectivity between points cloud but provide the correlation of geometry among points for helping in predicates adjacent points [4].

In (Zunran Wang et al., 2017) The Author interest with the problem of transmitting the 3D point cloud data and trying to decrease the network data volume. He used filters are to eliminate the noise and three dimensions image background and introduce segmentation algorithm to obtain the three dimensions point cloud data of interest. The experiment result shows the results with reduced data volume in network and information with high-quality image [6].

In (Tian Qing- guoet al., 2010) the Authors introduce algorithms for point cloud preprocessing based on divide and conquer method. It adopts topological spatial neighborhood theory. In begging distribute point cloud is to less cubes , in line with cube's space 26-neighbor relationship, the Maximum Connected Region (MCR) with scanning line point cloud is created and delete the data out of MCR regarding as noises. Presented these algorithms need only to setup the length of cube edged and the time consumed for preprocesses [7].

In (Kazuo Sugimoto et al., 2017) he used MPEQ compression method for point clouds it started a standardization way on three dimensions point cloud compression ,Technologies for acquisition, rendering and compression of 3D point clouds have advanced rapidly in last years, and the applications working on 3D point clouds is prod casted fast [8].

In (Siheng Chen et al.,2020) introduce a review on 3D point cloud learning a processing for autonomous driving.3D point cloud tools for learning and processing used for map localization , creation, perception modules in an applications of autonomous vehicle [9].

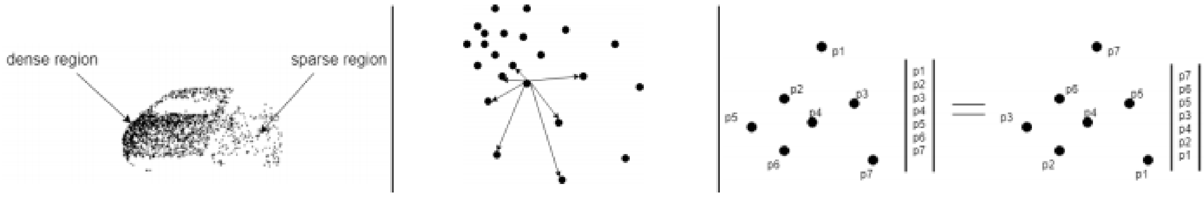


Figure (1): point cloud data a. irregular b. unstructured c. unordered

3-Point cloud file format

The graphics of computers and geometry computation created many format to represent point clouds and polygons generated by laser scanners as shown in table (1) few of these formats include [10]:

1- PLY - file format to describe polygon, developed by Stanford University
2- STL - a file format belongs to the stereo lithography CAD software generated by three dimensions system.
3- OBJ - is developed by wave front is a type of a geometry definition file format
4- X3D – is a type of file format to form 3D computer graphics data based on ISO standard XML
5- PCD is a file formats type have a lot of versions numbers, according to the appearance of Point Cloud Library (PCL) version 1.0.

Can change any of the previous file format extension to another type using cloud compare software by storing it with another extension the cloud compare program introduced.

4- Cloud compare program

Cloud Compare is software used for process 3D point cloud and triangular mesh process. It has been created to perform compare between two dense three dimensions point's clouds or between a triangular mesh and point cloud. Also can doing compression using octree structure on specific data. Then, it has been updated to a more operations on point cloud, including many advanced algorithms (sensor management, registration, resampling, color/normal/scalar processing, and compute of statistics, interactive segmentation, and enhancement of display). It can be downloading for free [11].

Table (1) point cloud file format [12]

Type	Extensions	Descripe	ASCII or Binary	No. of point clouds	No. of Mesh	Features
BIN	.bin	Format for Cloud Compare	Binary	>1	>1	Color(R,G,B) , Normals,scalar fields (>1)
ASCII	.txt,asc,xyz, and Pts	ASCII (X,Y,Z,R,G,B)	ASCII	1		
LAS	.las	Lidar point clouds	Binary	1		Scalar fields and colors(RGB)
PTX	.ptx	point cloud export format	Ascii	>1		normals and Robust processing at loading time
PCD	.PCD	(PCL) point cloud library	Binary	>1		Normals,colors (RGB), scalar fields (>1)
PLY	.ply	Stanford three dimensions geometry format (cloud or mesh)	Both	1	1	normals, colors (RGB or I), one or several scalar fields, a one texture

OBJ	.obj	mesh, wavefront	Ascii	1	>1	normals, textures and materials
STL	.stl	stereoLithography file format(mesh)	Ascii		1	normals

5- Point cloud preprocessing operations

Point cloud is collecting from scanning devices, huge points will generated, many point coordinates are redundant, and they may contain artifact and noise that must to remove. To improve the points which available, will do downsample, DBscan clustering, KDTree and visualization, the points cloud are processing to optimized, reduced, obtaining more consistent results after involving these data corrections [13].

A- Voxel downsampling

Point cloud downsample to choose subset of point loud that can sample to size handle easily. Take a voxel downsampling need a regular voxel grid to create for input points cloud a uniformly downsampled point cloud. It is almost used as a preprocessing step for several point cloud processing applications and it is beneficial in storing data.

Impose $S = \{xi \in Rt\}$ N, i=1 , N= number of points cloud, H operator for downsample select M points from S and M <N , the downsampleing work SM H(S). Where the algorithm operates in two steps:

1- Points are boated into voxels.

2- Each occupied voxel create exactly one point by compute the mean for all points inside.[14]

B- KD Tree

K dimensional tree is a structuring of data can be used as binary search tree in computer applications for ordering the points in a space with k dimensions. It is a binary search tree. K-d trees are very efficient in finding range and nearest neighbor searches. In this paper deals with three dimensions point clouds, so all the k-d trees will be three dimensions. Each level of a kdtree divides every child along a deterministic dimension, using a hyperplane that is straight to the chosen axis. At the root all its children will be fractal based on the first dimension. For example, in a 3D tree, the tree root have an x-aligned plane, the root children both have y-aligned planes, root grandchildren all have z-aligned planes [15]. The canonical technique of k-d tree construction has the following steps [16]:

Step1: choose point to be the tree root.

Step2: choose axis based on depth so that axis rotates through all points. If: axis ==(depth % 2) , where depth root= 0, axis equal (0/1) then suitably to select (x/y) axis.

Step3: reorder point list by axis and select median as axle element. If point. Coordinate less than root put in branch left, if greater put in branch right

Step4: pass over tree until cover all nodes, and then allocate every point to a node.

Step5: iterate step 2-4 recursively until it processes every points in tree.

C- Nearest neighbor on a k-d tree[16] [17]

After building KDtree can be used for search for Nearest Neighbor Search (NSS), NSS problem can be express in the following way: given a set of points N = n1, n2, . . . , nm in a metric space S, the points can be preprocessed in a to generate new points put in a query point Q ∈ S, for facility and quickly selecting the points in N that is nearest to Q. In three dimensional coordinate system the distance between two points can be calculated by the following equation:

$$D = ((X_2 - X_1)^2 + (Y_2 - Y_1)^2 + (Z_2 - Z_1)^2)^{1/2} \qquad (1)$$

Whither D = distance (m, inches ...), X, Y, Z = coordinates [16]

D- Clustering

Is selecting the closest (similar) points are grouped together. Clustering algorithms are useful in the popular cases where it is expensive to label data. Take the example of annotating a large point cloud. Annotate every point by which shape will contain can be a long and boring job; can generated error through negligence or weariness. It is more efficient and cheaper to let a clustering algorithm collected similar points at the same cluster and then can a human process when allocate a label to the cluster [18].

E- Density Based Clustering Algorithms

Is base method for density based clustering, it unsupervised learning algorithm that discover distinguishing groups(clusters) in the data, Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is a It can create clusters of various size and shapes from a big amount of data, which is may be involving outliers and noise[19].

The Density Based Clustering algorithm needs two parameters [19]:

Inputs: minPts: is the "threshold" lowest number of points in a region to be clustered to be considered dense. eps (ε): A distance measure used for finding neighbors of any points.

Step1: The algorithm working by arbitrarily picking up every point in the dataset until complete points.

Step2: the points to be sub of one cluster. If there are at least 'minPoint' points with a radius of 'ε' to the point then group all these points to be on the same cluster.

Step3: repeated compute the neighboring point until complete points and create cluster.

6- **Open3D python library**

Is an open source library that manipulates 3D data, this library exposes a set of selected data structures and algorithms in python and C++ programming language. Open3d afford structure of data for three types of representations: point clouds, RGB-D images, meshes. For each many processing algorithm like Input/ Output, visualization, sampling and data conversation. Point cloud type has structure of three data fields :(point cloud. points) used to save coordinates, (point cloud. normal) used to save normal, (point cloud. colors) used to save colors. The master field is (point cloud. points) the remaining two data fields considering valid if when together have the equal number of records as master field [20].

7- **Dataset**

The three dimensions data set which recorded by 3D scanners like LIDAR be classified into outdoor and indoor [3]:

1- Semantic3D

It is high density quality outdoor dataset contain points is more than four billion points. It classifies 8 semantic classes like cars, high plant, and low plant, and handmade soil, artifacts of scanning and natural soil [21].

2- S3DIS

It is large- scale 3D indoor dataset contain 215 million points. It generated by Stanford University [22].

8- The Result

In this paper will used semantic3D outdoor dataset and S3DIS indoor dataset, choosing from semantic3D: Marketplace Feldkirch_station4 file, Bildstein_station1_xyz_intensity_rgb.txt,StGallenCathedral_station6_rgb_intensity-reduced data and from Stanfored3D choose Hallway_4, conferenceRoom_1, Sofa_1, Chair all that files is Asscii files contain (X, Y,Z,R,G,B) with extension txt each file express in table (2), the preprocessing steps as following:

A- Voxel Downsample

Use function voxel_down_sample in open3d python library to compute the mean of points with voxel_size=0.05 , the output for each dataset file is as shown if figure(3)and table(2) bellow explore the numbers of point after voxel downsample operation for outdoor and indoor dataset the downsample operation very useful in point reduction.

Table (2) voxel downsample operation

Type of dataset	File	Size	No. of point of original file	No. of points after voxel downsample	The reduction percentage
Outdoor-semantic3D	MarketplaceFeldkirch_station4	700MB	10 538 633	1740491	16.5%
Semantic3D	StGallenCathedral_station6_rgb_intensity-reduced	963MB	14 608 690	3104 418	21.2%
Semantic3D	Bildstein_station1_xyz_intensity_rgb	1.20Gb	29 697 591	2177040	7.3%
Stanfored3D	Hallway_4	88.8MB	1542355	65656	4.3%
Stanfored3D	conferenceRoom_1	67.5MB	1136617	49719	4.4%
Stanfored3D	Sofa_1	1.98MB	35397	1836	5.2%
Stanfored3D	Chair	396KB	6729	405	6%

B- DB scan cluster

The function “cluster_dbscan “have two paremter: eps =0.03(the distance between neighbors in cluster) and min_points: min numbers of points to form cluster, if the labels =-1 indicates noise, the output as shown in table (3) and figure(2).

Table (3) DBscan operation

Type of dataset	File	Size	No. of point of original files	No. of cluster after DBscan
Outdoor-semantic3D	MarketplaceFeldkirch_station4	700MB	10538633	24768
Semantic3D	StGallenCathedral_station6_rgb_intensity-reduced	963MB	14 608 690	17944
Semantic3D	Bildstein_station1_xyz_intensity_rgb.txt	1.20Gb	29697591	26578
Stanfored3D	Hallway_4	88.8MB	1542355	120
Stanfored3D	conferenceRoom_1	67.5MB	1136617	192
Stanfored3D	Sofa_1	1.98MB	35397	6
Stanfored3D	Chair_1	396KB	6729	12

C- KDTree

Need to build KDTree first then finding nearest neighbors points

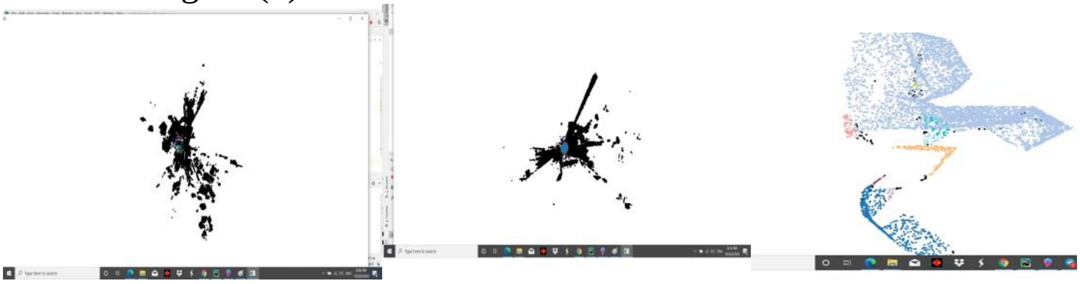
a- Built KDTree

Read the files of 3d point clouds preprocessor it by finding and builds a KDTree for it by using function “KDTreeflann”

b- Finding nearest neighbors points

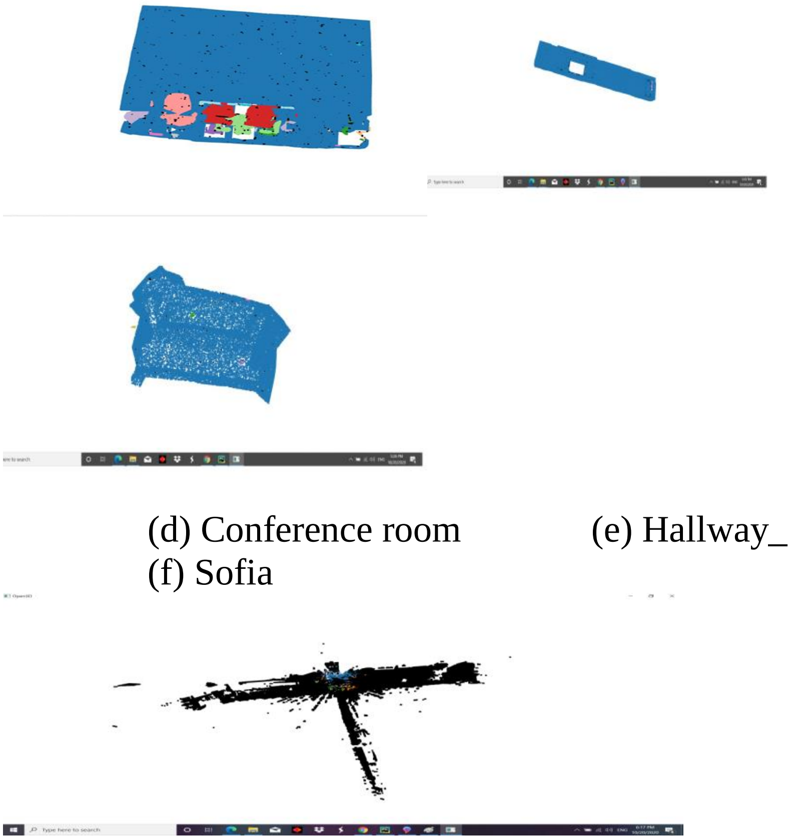
- Take the first 1500 points to be as “anchor point”
- Detriment the reduce 0.3
- Finding distance using equation(1)

Used the function “ Search_knn_vector_3d” for compute the k nearest neighbors of anchor point Query all points and compute distance and finding neighbors with distance less than 0.3 and visualize the points with red color as seen in figure (4)



(a) Bildstein_station1 (b) StGallenCathedral

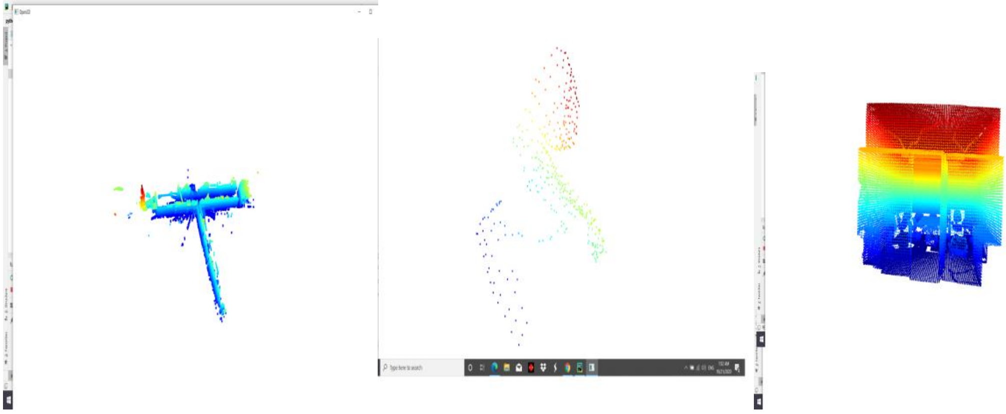
(c)chair



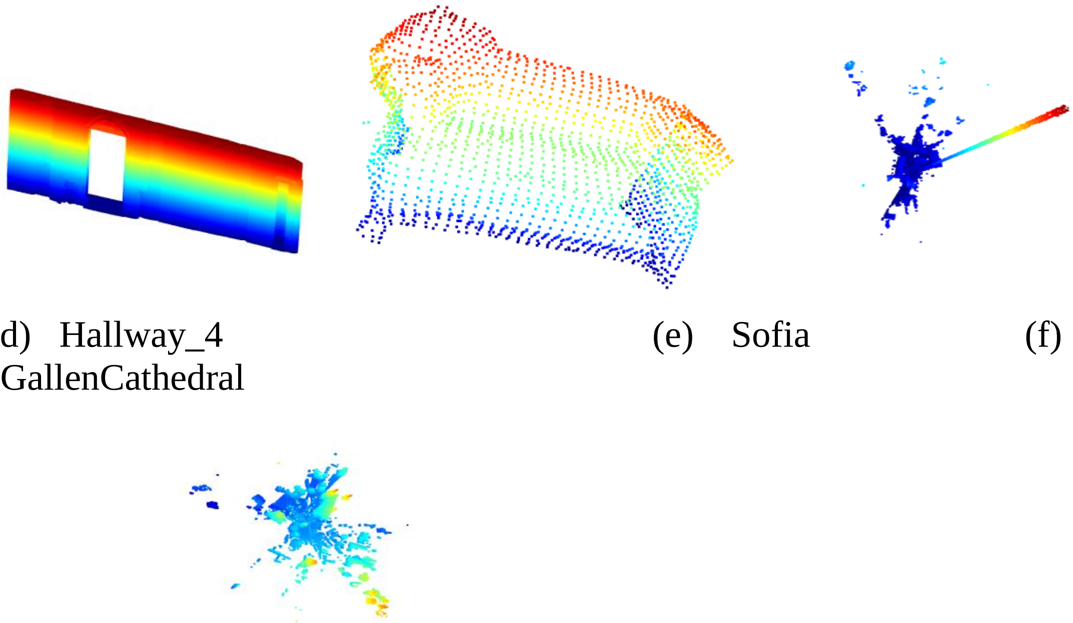
(d) Conference room (e) Hallway_4
(f) Sofia

(g) MarketplaceFeldkirch

Figure (2) The result after DBscan operation



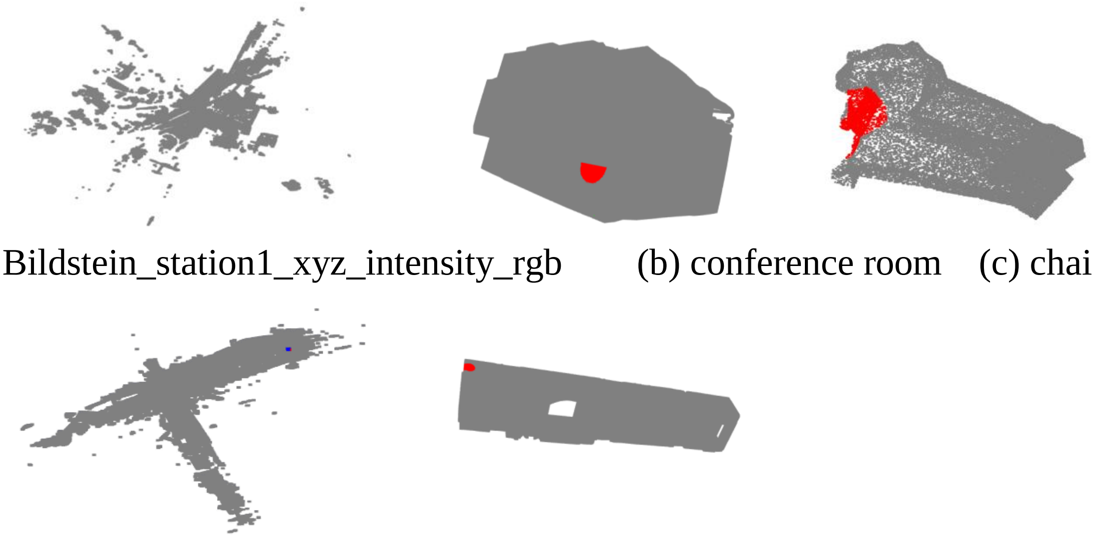
(a) MarketplaceFeldkirch_station4 (b) chair
(c)conference room



(d) Hallway_4 (e) Sofia (f)
StGallenCathedral

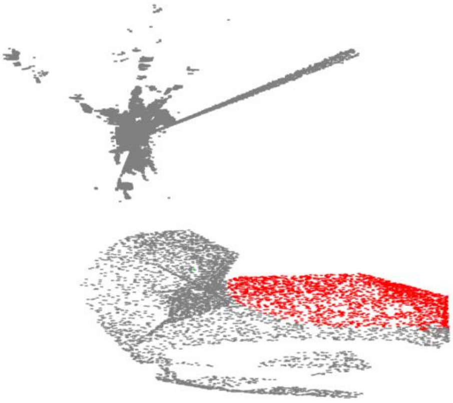
(g) Bildstein_station1

Figure (3) The result of voxel down sample operation



(a) Bildstein_station1_xyz_intensity_rgb (b) conference room (c) chair

(d) MarketplaceFeldkirch_station4 (e) Hallway_4



(h)StGallenCathedral_station6_rgb
(g) chair

Figure (4) The result of KDTree operation

9- The Conclusions

This paper introduces three operations for preprocessing large amount of point cloud outdoor and indoor dataset benchmark: voxel downsample, DBScane cluster, KDTree. These operations implemented an efficient data structure for 3D, the software process 29 million points while still enabling the user to visualize and navigation smoothly through the point cloud, reduction the points and structure and represent efficiency for easy used in many applications like automatically deriving semantic information.

References

[1] Haoming Lu, Humphrey Shi, “Deep Learning for 3D Point Cloud Understanding: A Survey” 18 Sep 2020, arXiv: 2009.08920v1 [cs.CV**]**

[2] Saifullahi Aminu Bello, Shangshu Yu1, Cheng Wang1, senior member, IEEE“ Review: deep learning on 3D point clouds “, arXiv: 2001.06280v1 [cs.CV] 17 Jan 2020.

[4] EMAN AHMED,ALEXANDRE SAINT, ABDELRAHMAN SHABAYEK, KSENIYA CHERENKOVA, RIG DAS, GLEB GUSEV, DJAMILA AOUADA, “A survey on Deep Learning Advances on 3 Data Representations”, arXiv:1808.01462v2 [cs.CV] 6 Apr 2019.

[5] Chao Cao, Marius Preda, Titus Zaharia, “3D Point Cloud Compression: A Survey ”, Web3D '19: The 24th International Conference on 3D Web Technology, July 2019, https://doi.org/10.1145/3329714.3338130.

[6] X. D. Zhang, C.K. Sun, C. Wang and S.H. Ye, “study on preprocessing Methods for color 3D point Cloud”, December 2004, Materials Science Forum (Volumes 471-472),
https://doi.org/10.4028/www.scientific.net/MSF.471-472.716.

[7] Zunran Wang, Chenguang Yang, Zhaojie Ju, Zhijun Li, Chun-Yi Su, “Preprocessing and Transmission for 3D Point Cloud Data”, International Conference on Intelligent Robotics and Applications, ICIRA 2017: Intelligent Robotics and Applications pp 438-449.

[8] Tian Qing-guo; Li Jin-tong,”Pre-processing of 3D scanning line point cloud data”
Date Added to IEEE Xplore: 04 November 2010, China Publisher: IEEE, DOI: 10.1109/ICCASM.2010.5622877

[9] Kazuo Sugimoto, R. Cohen, author A. Vetro , “Trends in efficient representation of 3D point clouds”, , Published 2017 Computer Science 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), DOI:10.1109/APSIPA.2017.8282059Corpus ID: 44235788.

[10] Siheng Chen, Baoan Liu,Chen Feng, Carlos Vallespi-Gonzalez, Carl Wellington,“3D point Cloud Processing and Learning for Autonomous Driving”, arXiv:2003.00601v1 [cs.CV] 1 Mar 2020.

[11] Built with Sphinx , “The PCD (Point Cloud Data) file format”, https://pointclouds.org/documentation/tutorials/pcd_file_format.html, 2020.

[12] “3D point cloud and mesh processing software Open Source Project”, http://www.cloudcompare.org/, 2020.

[13] FILE I/O, supported file format,
https://www.cloudcompare.org/doc/wiki/index.php?title=FILE_I/O, 9 February 2020.

[14] Yuxing Xie, Jiaojiao Tian, Member, IEEE and Xiao Xiang Zhu, Senior Member, IEEE, “ A Review of Point Cloud Semantic Segmentation” arXiv:1908.08854v1 [cs.CV] 23 Aug 2019.

[15] Hongjoo Park, Samsung Lim, John C. Trinder ,R Turner, “Voxel-Based Volume Modelling of INDIVIDUAL Trees Using TERRESTRIAL Laser

Scanners", Conference: The 15th Australasian Remote Sensing & Photogrammetry Conference (ARSPC), January 2010.

[16] Jan Elseberg, Dorit Borrmann, Andreas N¨uchter, "Efficient Processing of Large 3D Point Clouds" , October 2011, Conference: Proceedings of the XXIII International Symposium on Information, Communication and Automation Technologies (ICAT '11), DOI: 10.1109/ICAT.2011.6102102.

[17] Yasen Hu, "KD-Tree", https://yasenh.github.io/post/kd-tree/, Jul 18, 2019.

[18] JEREMY DAY, "Using k-d trees to efficiently calculate nearest neighbors in 3D vector space", FEBRUARY 09, 2018.

[19] Florent Poux, "Fundamentals to clustering high-dimensional data (3D point clouds)", 9 Jun 2020.

[20] Chunxiao Wang ,Min Ji , Jian Wang , Wei Wen , Ting Li and Yong Sun ,"An Improved DBSCAN Method for LiDAR Data Segmentation with Automatic Eps Estimation" , Sensors 2019, 19(1), 172; https://doi.org/10.3390/s19010172, 5 January 2019

[21] Qian-Yi Zhou, Jaesik Park, V. Koltun "A Modern Library for 3 D Data Processing", Published 2018, Corpus ID: 46797310.

[22] Charles Ruizhongtai Q i, "Deep Learning on Point Sets for 3D Classification and Segmentation", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); Honolulu, HI, USA. 21–26 July 2017; pp. 77–85.

اعادة معالجه البيانات الثلاثية الابعاد لقواعد البيانات الخارجية والداخلية

أماني يوسف نوري
الجامعة المستنصرية- العراق

مستخلص البحث:

هذا البحث يهتم بالبيانات ثلاثية الابعاد حيث يوفر شكل النقاط السحابية وصفا واقعيا وقويا لمشاهد كبيرة مختلفة ذات ابعاد ثلاث. شكل النقاط السحابية ذات الابعاد الثلاث تعاني من الضوضاء ودقه البيانات الغير عاليه ووجود بيانات غير مرغوب بها نتيجه الاجهزة المستخدمه في توليدها كاجهزة المسح بالليزر وبالتالي توثر على دقه النماذج التي تدخل اليها. حلا للمشاكل السابقه يتم عمل معالجه مسبقه وتحليل لهذة البيانات تهذف لتحسين البيانات وتخزينها ونقلها وتحسين جودتها باستخدام خوارزميات حسابيه ورياضيه. خوارزميه الضغط تهدذ لتقليل المساحه المستخدمه وخوارزميه التجميع لتقليل الضوضاء وخوارزميه الاختزال لتقليل النقاط الموجودة وخوارزميه اقرب جار لأجل العثور على أقرب نقاط الجيران. البرنامج المقترح يعالج 29 مليون نقطه سحابيه تم التعامل مع قاعدة البيانات الثلاثية الابعاد للمناظر الخارجيه والداخليه semantic3D outdoor dataset and Stanfored3D indoor وتم البرمجه باستخدام لغه البايثون python 3.7. مواصفات الحاسوب NAVIDI RTX 2060 و RAM 16 CORE i7MP وبرنامج free cloud compare software v2.6.3 .

الكلمات المفتاحية : النقاط السحابيه الثلاثية الابعاد ، خوارزميه اخفاء النقاط ، مكتبه open3d ، خوارزميه تجميع النقاط scan, عمليه downsample وخوارزميه KDtree .

ملاحظه : البحث مستخلص من اطروحه دكتوراه.