Vol 7 No 1 (2025) *e*-ISSN: 2682-8383

Journal of Engineering Technology and Applied Physics

Independently Identifying Noise Clusters in 2D LiDAR Scanning with Clustering Algorithms

Chiew Wei Wen* and Chuan Hsian Pu
Department of Electrical and Electronic Engineering, University of Nottingham Malaysia, Semenyih, Selangor, Malaysia.
**Corresponding author*: nagel9888@gmail.com, *ORCiD*: 0000-0003-4121-733X
https://doi.org/10.33093/jetap.2025.7.1.7
Manuscript Received: 1 August 2024, Accepted: 11 October 2024, Published: 15 March 2025

***Abstract* — Light Detection and Ranging (LiDAR) refers to a range imaging method for distance objects based on the principle of laser ranging. LiDAR environmental mapping technology is often highly praised for its precise mapping information with intricate features for various detection or tracking based applications. The research proposes a novel method for independently identifying and filtering noise clusters in 2-Dimensional (2D) LiDAR scans based on 2 distinct clustering algorithms of K-Means and Density-Based Spatial Clustering of Applications with Noise (DBSCAN). Results show DBSCAN to be the better choice as it is more robust and resistance to noise and outliers in the dataset and is capable of identifying clusters of any shape making it more versatile. Furthermore, to address the issue of dead zones present in LiDAR scanning, an innovative solution based on interpolating the discontinuous spatial results of the LiDAR scanning result to further reconstruct a 3-Dimensional (3D) viewing model by stacking multiple copies of 2D LiDAR scanning results with varying elevation is demonstrated by the results of the study to be a viable economical alternative for 3D LiDAR mapping.**

Keywords—2D LiDAR Scanning, K-means, Density-Based Spatial Clustering of Applications with Noise (DBSCAN).

I. INTRODUCTION

In recent years, Light Detection and Ranging (LiDAR) technology has become essential across numerous industries, being highly valued for its versatility and accuracy in efficiently capturing detailed spatial data. The technology works by constantly emitting a pulsed infrared laser from its sensor, while at the same time measuring the time it takes for the same exact beam to be received by the receiving sensor after reflecting off a surface, such as a wall or occlusion in the scanning environment [1]. LiDAR sensors are capable of making highly precise measurements for example, an independent research found that when studying terrain mapping using LiDAR technology on a location with a difference of elevation of 15 cm, the LiDAR sensor was capable of attaining a result with localisation error as low as 8.14 cm with a mapping error of only 8.43 cm at a 4 cm map resolution [2].

LiDAR technology is shaping up to become an essential tool in various industries owing to its ability to accurately capture detailed spatial data. One prominent example is the natural resource management industry, LiDAR technology is ideal for accurately measure the terrain, vegetation density and canopy structure [3]. Besides that, LiDAR technology is also inseparable from high precision industries, such as the construction and engineering industry. The ability to recreate accurate and detailed topological maps is indispensable for engineers to assess slope stability and also detect hidden geological features [4]. Following that, LiDAR technology can also be used to carry out vital topological surveys like floodplain mapping to help risk management agencies monitor and evaluate the risk of a flood occurring during monsoon season [5]. Lastly, one of the more obscure industries that benefit from the advancement of LiDAR technology is the mining industry. With the help of LiDAR sensors to provide reliable elevation data and assist in performing infrastructure surveys engineers can now use the various information collected to optimise the tedious resource extraction process [6].

Published by MMU PRESS. URL: https://journals.mmupress.com/index.php/jetap/index

Although LiDAR mapping techniques are widely considered to be among the best options for environmental ranging and mapping, the technology is still immature and faces the challenge of optimization to deliver accurate, detailed, and precise spatial representations [7]. While it is undeniable that current modern day LiDAR technology is capable of providing a rich point cloud data with high spatial resolution, its performance can be easily hindered by external factors such as environmental occlusions, sensor noise, and data inconsistencies. Addressing these challenges is paramount to ensure that LiDAR technology mapping can consistently deliver a high level of reliability without compromising its accuracy under varying environmental circumstances. Ultimately resulting in LiDAR based solutions having improved flexibility in detection, further broadening the potential applications of LiDAR technology.

The primary goal of this study is to push the boundaries of 2-Dimensional (2D) LiDAR mapping by tackling the identified challenges posed by environmental factors and exploring innovative alternatives for a more accessible and economical approach to 3-Dimensional (3D) environmental mapping. This study introduces a novel approach to noise/outlier detection and filtering based on iterative clustering algorithms to reduce false detection and noise in the scanning dataset caused by unpredictable environmental factors. Additionally, the study also proposes an ingenious idea to portray multiple 2D LiDAR scanning results in a sequential manner where the 2D LiDAR scan results are stacked atop one another to form a layered 3D view model of the scanning environment, with an emphasis on scalability, precision, and consistency.

The following sections of the paper will discuss in greater detail in regards to the effectiveness of different clustering algorithms at independently identifying the noise clusters in the 2D scan result using K-means and Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithms, the viability of interpolating discontinuous spatial results in LiDAR scanning and to compare the practicality of a 3D view model compiled from multiple 2D LiDAR scans recorded with varying elevation (z-axis) positions.

II. METHODOLOGY

A. 2D LiDAR Scanning Dataset Collection

RPLiDAR A2M12 is used to collect the scanning results of indoor environment to produce the dataset for this research. The operational concept of 2-D LiDAR mapping is largely based on the principle of laser triangulation ranging. During the LiDAR scanning cycle, the RPLiDAR optical transmitter sensor will be intermittently emitting a modulated infrared laser beam through the optical window of the LiDAR sensor in whatever direction the sensor is presently facing. When the laser beam is reflected off a surface or object, the reflected signal is then subsequently captured by the receiver module of the sensor. The sensor will record the time interval between transmitting and receiving the signal to compute the distance between the sensor and the object.

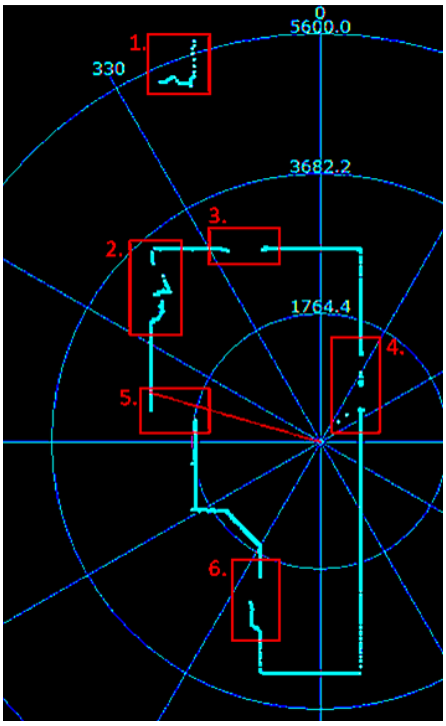

Fig. 1. 2D LiDAR scanning result of the indoor test environment.

Table I. Table detailing the scanning issues for each area.

Area	Scanning Issues
Area 1	False detection of dead zone corner, Area 5 which is visible in the reflection of the mirror in Area 3. Sensor incorrectly identifies the distance for the points.
Area 2	Undefined and jagged edges, caused by the clothes hanger affixed to the door. LiDAR is unable to trace the outline of the clothes hanger from its position.
Area 3	The position of the mirror, the sensor only plots the points after detecting the reflected laser beam from its receiver lens, the mirror cannot be mapped.
Area 4	Has sporadically detected points along the flat wall due to occlusions.
Area 5	The corner is located in the scanning dead zone of the current LiDAR position, blocked due to the position of the closet edge obstructing the laser.
Area 6	Half of the door is unidentified as the protruding door frame is blocking the LiDAR laser.

Following that, each reflected point is logged as an individual entry in a text file, along with various details such as the corresponding distance, angle, and scan quality of the point. All the recorded parameters are used to visualise the LiDAR scanning result on a Polar plot. According to the specification document, the hardware system of the LiDAR sensor is capable of performing high-speed sampling operations, being able to capture data at a maximum rate of 16,000 times per second at the maximum rotational speed of 15 Hertz (Hz). Figure 1 shows the raw 2D LiDAR mapping result captured under nighttime conditions with a rotational speed of 10 Hz. Observing the image, 6 different LiDAR scanning issues are identified in the indoor test scanning environment selected (in Table I) for this research.

The research proposes a series of methodical processes to help improve the 2D LiDAR scanning result. Figure 2 shows the proposed flowchart of LiDAR signal processing for the 2D interpolation and 3D reconstruction process.

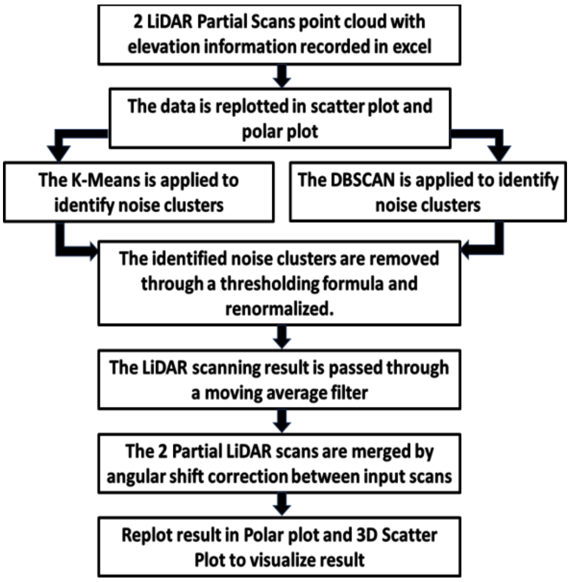

Fig. 2. Flowchart of LiDAR signal interpolation and reconstruction.

B. *Comparing Effectiveness of Clustering Algorithm*

The 2D LiDAR scanning dataset is used to evaluate the performance of the K-means and DBSCAN algorithms. Before the process can begin, the 2D LiDAR scanning result has to be transferred from a Polar coordinate plot format to a Cartesian coordinate plot format, where the y-axis corresponds to the distance measurement and the x-axis corresponds to the angular measurement in degrees.

Converting the point cloud data from Polar coordinates to Cartesian coordinates makes it easier to observe the trends in the LiDAR dataset. As now the Polar plot diagram unfolds in the form of a Cartesian plot, with the data trend signifying the spatial information of the scanning environment. This is a fundamental first step of the proposed solution before any data processing method can be carried out. The reason being that in Cartesian coordinates, the spatial information will be easier to extrapolate; furthermore, most of the mathematical function used in the research operates on the basis of Cartesian coordinate system. The aim of the signal processing solution is to assist in interpolating the LiDAR echo signals while at the same time excluding the noise and outliers from the scanning result, making cleaner and more spatially accurate reconstruction of LiDAR signals.

The two different clustering algorithm which was chosen by the research are the K-Means clustering and DBSCAN clustering. These two iterative clustering methods were chosen as they have a highly contrasting theoretical basis for clustering data. K-Means clustering functions by grouping data points together based on the point's relative distance from its nearest cluster centroid. Contrary DBSCAN operates by grouping nearby neighbouring points into dense clusters while excluding points in sparse regions (noise) by setting a distance threshold (ε) for which the algorithm will search for neighbouring point [8].

K-Means algorithm is a common clustering algorithm used in machine learning and data mining for identifying patterns within large datasets [9]. It works by first randomly selecting a specified number of centroids in a dataset. Each point is then assigned to their nearest centroid, by measuring the distance in Euclidean. Once all points are assigned a cluster, all centroids are recalculated with the new centroid being computed as the mean of all points in the cluster [10]. The process is then carried out iteratively until there are no more significant changes to the centroid. Convergence is then achieved with each data point being grouped in a specific cluster and final locations of the centroids being from the last averaging iteration.

The DBSCAN machine learning algorithm works by grouping together data points that are closely packed and identifies points that lie alone in low-density regions as outliers [8]. The algorithm starts with identifying a core point by the minimum number of neighbour points surrounding it. It then forms a cluster starting from the core point, constantly expanding by adding neighbouring core points. Points without sufficient neighbouring points are determined as the border points, which are located at the edge of clusters. While any points not reachable from any core point with the distance threshold will be considered as noise points or outliers in the dataset. This process will be repeated until all points in the dataset are either assigned to a cluster or marked as noise [11].

C. *Resolving Dead Zones in LiDAR Scans*

Dead zone areas are present in the 2D LiDAR results caused by the limited FOV and position of the LiDAR sensor. The main reason causing this particular issue to arise in scanning result is due to the mechanical design of the 2D LiDAR sensor.

A simple 2D LiDAR sensor works by having a pair of optical emitter and receiver constantly rotating 360° on the sensor's central fixed axis. Theatrically, if the sensor is placed in a room without any objects blocking the path of the laser, the sensor will capture the shape of the surroundings without the issue of discontinuous spatial information. However, if an object or occlusion happens to be located in between the position of the sensor and the wall, the path of the laser will be obstructed by the edges of the object. Since the position of the LiDAR sensor is fixed during operation, the area unreachable by the laser will be represented as a discontinuous line in the final result, where the spatial information of that dead zone is ambiguous and left up for debate.

To improve the overall comprehensiveness of the LiDAR scanning results, this research proposes a novel solution by combining the mapping information of first scan with a secondary LiDAR scan taken from a more advantageous position in the same environment to fill in the lack of spatial information in the primary scan which was blocked due to limited FOV caused by the mechanical design of the 2D LiDAR sensor.

D. Compiling Multiple 2D Scans into a 3D View Model

This part aims to examine the viability and effectiveness of a 3D environment mapping view model made by stacking multiple copies of 2D LiDAR scanning results. Each individual 2D LiDAR scans are sampled at varying elevation levels with the LiDAR sensor fixed on a laboratory scissor, to accurately measure and record the elevation information.

When the LiDAR datasets from the processor of the sensor is passed on to the MATLAB script, each individual LiDAR scan is recorded sequentially in a matrix and processed by the script to remove any noise clusters present in the LiDAR scanning dataset. Then all the LiDAR scans are plotted on their respective z-axis positions to construct a 3D digital representation of the scanning environment.

III. RESULTS & DISCUSSIONS

A. Results of Comparing Clustering Algorithms

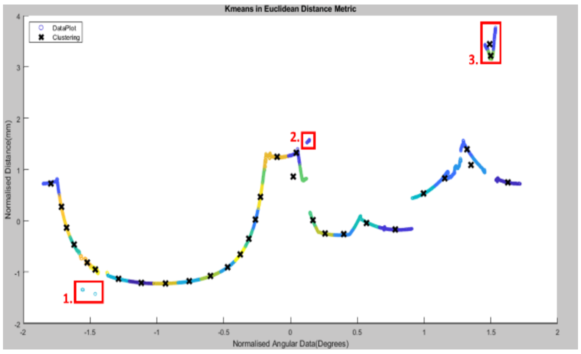

Fig. 3. K-Means clustering result for 2D LiDAR scanning result.

From Fig. 3, the K-Means clustering algorithm is shown to be capable of successfully independently identifying and separating 2 out of the 3 noise clusters from the rest of the 2D LiDAR scanning dataset. The condition of success is defined as the ability of the clustering algorithm to independently classify and group the noise points into their own respective cluster separating the noise points from the rest of the LiDAR scanning dataset.

The result visualised in Fig. 3, shows the K-Means clustering algorithm capable of sorting the points in Noise Cluster 2 and Noise Cluster 3 into their own individual distinct clusters with their respective centroids as these two noise clusters are determined by the algorithm to be located far away from any other cluster centroids of the data graph of the 2D LiDAR scanning dataset.

Conversely, for the case of Noise Cluster 1 the noise points are determined by the K-Means clustering algorithm to be grouped together with the nearest cluster centroid from the graph of the 2D LiDAR scanning dataset; Implying that the clustering algorithm failed to independently segregate the noise points in Noise Cluster 1 apart for the rest of the dataset by forming a separate cluster centroid for the noise points in Noise Cluster 1. Thus, signifying that the K-Means algorithm is confusing the Noise Cluster 1 to be a part of the LiDAR scanning dataset, while it is actually a by-product of the noise in the LiDAR scan.

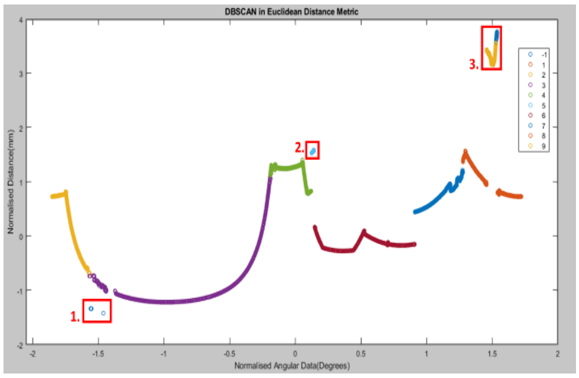

Fig. 4. DBSCAN clustering result for 2D LiDAR scanning result.

From Fig. 4, the DBSCAN clustering algorithm is shown to be capable of successfully independently identifying all 3 noise clusters in the 2D LiDAR scanning dataset and classifying them into distinct, separate clusters apart from the spatial information of the scanning environment. Through observation, it is concluded that the algorithm forms groups based on the proximity of the points from each other; forming several densely populated clusters in the final clustering result while also being flexible at filtering out outlier points from the dataset as noise.

In Fig. 4, each of the 3 noise clusters under observation are capable of being sorted by the DBSCAN algorithm into their own separate clusters which will be referred to as “Group” in the following section, which each individual LiDAR point referred to as “observations”. One of the unique characteristics of the DBSCAN algorithm is to be able to accurately group and label outlier points make it the optimal solution for handling noise.

The following Figs. 5, 6 and 7 will provide more clarity with detailed magnified views of the 3 noise cluster locations to better visually contextualise the minute differences between the two clustering results. The varying outcome observed from the two clustering algorithms is caused by their contrasting theoretical operation principle.

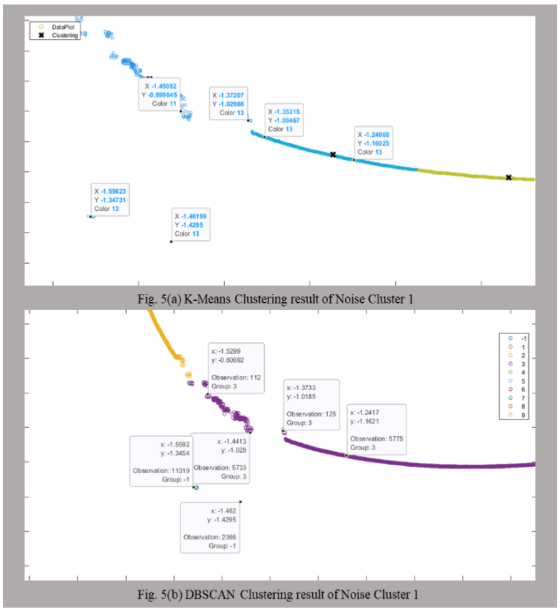

Fig. 5(a) K-Means Clustering result of Noise Cluster 1

Fig. 5(b) DBSCAN Clustering result of Noise Cluster 1

Fig. 5. Comparing the clustering results from the two clustering algorithms on Noise Cluster 1.

In Fig. 5, the data points in Noise Cluster 1 and their clustering information can be better observed. From Fig. 5(a) the result of the K-Means algorithm the sporadic outlier points of Noise Cluster 1 is grouped by the algorithm to be a member of nearest cluster centroid “Colour 13”. Thereby, failing to form its own separate cluster centroid in the centre of the outlier points, causing the algorithm to falsely identify it as a part of the LiDAR scanning data point. Conversely, in Fig. 5(b) the clustering result of the DBSCAN algorithm which forms groups based on the basis of density of data points can successfully determine that Noise Cluster 1 is spare and far apart enough to be identified as noise points. Therefore, DBSCAN is able to clearly distinguish and separate the Noise Cluster 1 apart from the rest of the LiDAR scanning data points.

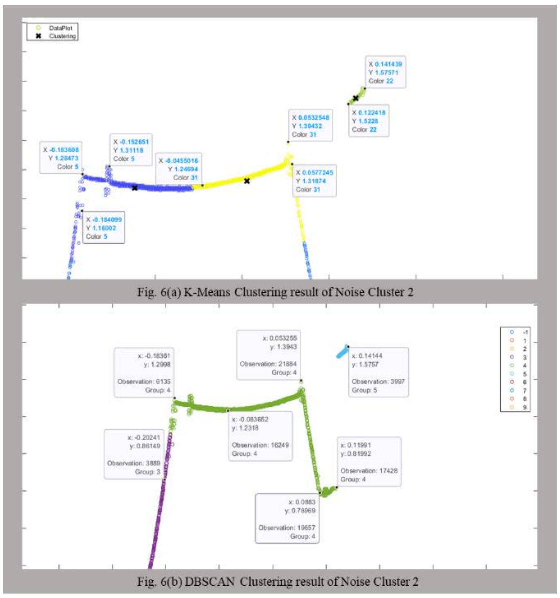

Fig. 6(a) K-Means Clustering result of Noise Cluster 2

Fig. 6(b) DBSCAN Clustering result of Noise Cluster 2

Fig. 6. Comparing the clustering results from the two clustering algorithms on Noise Cluster 2.

Figure 6 is a more detailed visual comparison of the K-Means and DBSCAN clustering results for Noise Cluster 2. In Fig. 6(a) after the K-Means algorithm achieved convergence and terminated the iterations, Noise Cluster 2 is determined to have sufficient outlier points and located far enough from the other centroids to qualify for its very own cluster centre. This is the ideal clustering result, as now all the noise points in Noise Cluster 2 are grouped together under a single classification of “Colour 22”, while also separating it apart from the rest of the LiDAR scanning data points. Contrarily, in Fig. 6(b) the DBSCAN algorithm also recognises Noise Cluster 2 as “Group 5”, a cluster independent from the rest of the LiDAR scanning data points. Hence, achieving the desired result of grouping together outlier points and segregating it from the rest of the data points. This desired clustering outcome makes it simpler to filter out the Noise Clusters present in the dataset by eliminating the clusters containing the outlier points, choosing to retain only the data points with the spatial information from the 2D LiDAR scan.

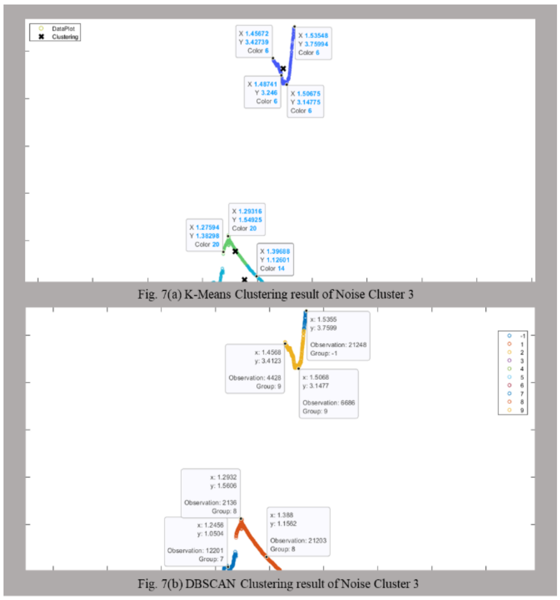

Fig. 7(a) K-Means Clustering result of Noise Cluster 3

Fig. 7(b) DBSCAN Clustering result of Noise Cluster 3

Fig. 7. Comparing the clustering results from the two clustering algorithms on Noise Cluster 3.

Lastly, Fig. 7 compares the K-Means and DBSCAN clustering result images of Noise Cluster 3. The clustering result for Nosie Cluster 3 is similar to the result presented in Noise Cluster 2. With both clustering algorithms able to independently identify and separate the outlier points of Noise Cluster 3 apart from the rest of the LiDAR scanning data points, satisfying the design conditions for this process.

In Fig. 7(a), the K-Means algorithm identifies Noise Cluster 3 has enough population and is far away enough to justify having its own cluster centroid. This essentially groups together all the outlier points in Noise Cluster 3 under the classification of a new group, “Colour 6”. While in Fig. 7(b), the DBSCAN algorithm is capable of segmenting the outlier points in Noise Cluster 3 from the rest of the LiDAR scanning data points. Interestingly, the outlier points in the upper part Noise Cluster 3 has been classified by the algorithm as “Group 1”; while the lower part of Noise Cluster 3 is categorised as “Group 9”. The differentiates the Noise Cluster 3 as a cluster independent from the rest of the LiDAR scanning data points which contain the spatial information of the scanning environment.

In conclusion, the results for this section has proven that the DBSCAN clustering algorithm is a more effective solution than the K-Means clustering algorithm for independently identifying noise points as separate clusters apart from the other points with spatial information about the environment. The reason being that the more flexible DBSCAN algorithm has the advantage of being more proficient at handling noise and locating clusters of arbitrary shapes since it doesn't assume any specific shape. Compared to K-Means algorithm which requires to specify the number of centroids beforehand as it groups points based on the proximity to the centroid, essentially forming rigid spherical clusters.

B. Supplementing Primary LiDAR Scan with Another LiDAR Scan

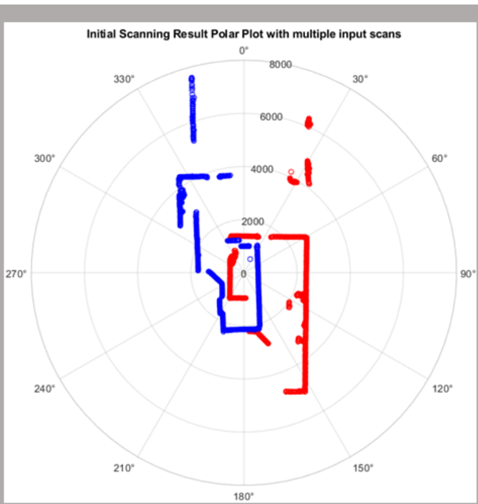

Fig. 8(a) LiDAR Scanning result before the MATLAB signal processing

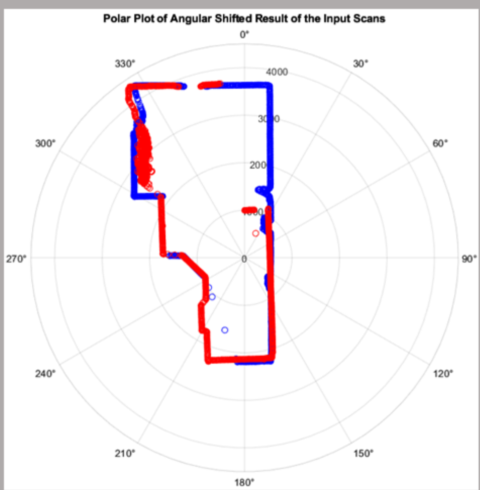

Fig. 8(b) LiDAR Scanning result after the MATLAB signal processing

Fig. 8. Comparison of the LiDAR scan before and after the MATLAB signal processing.

Figure 8 illustrates the LiDAR scanning results before and after the dead zone reconstruction process. Figure 8(a) shows the 2 LiDAR scanning result before the signal processing, the LiDAR scans are visualised on a Polar plot with the primary LiDAR scan result in blue and the second supplementary LiDAR scan result in red. Results show the two LiDAR scan results are not centred, as the position of the LiDAR sensor is shifted from its original position to record the secondary LiDAR scan result in red.

Therefore, the MATLAB solution aims to compute the gap between the 2 LiDAR scans in terms of the x and y dimensions of the Cartesian coordinates to translate the 2 LiDAR scans onto a shared centre. The solution also includes the use of the Structural Similarity Index (SSIM) function to correct any angular shift error between the two LiDAR scans in an attempt to align both LiDAR scans before combining the two LiDAR point cloud inputs into one complete singular result similar to the result shown in Fig. 8(b).

C. 3D View Model from Multiple 2D LiDAR Scans

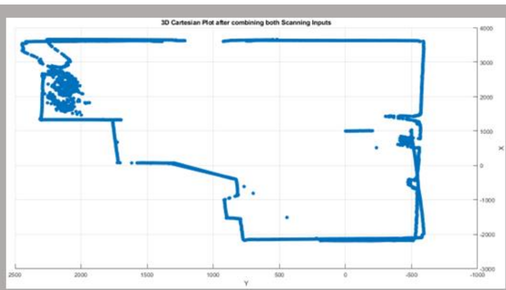

Fig. 9(a) Top view of the 3D view model of the combined multiple 2D LiDAR scan result

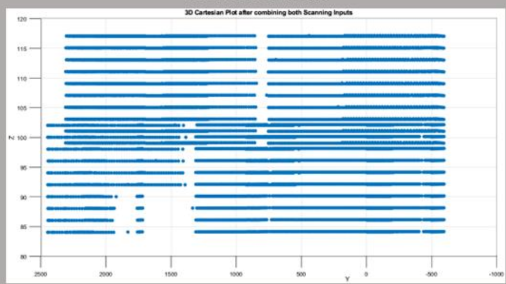

Fig. 9(b) Front view of the 3D view model of the combined multiple 2D LiDAR scan result

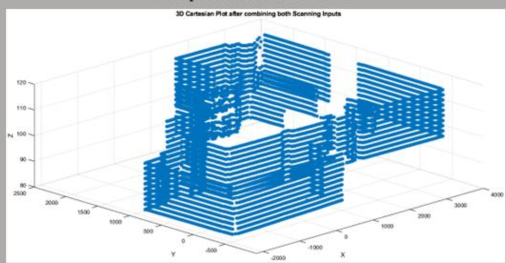

Fig. 9(c) Two-point view of the 3D view model of the combined multiple 2D LiDAR scan result

Fig. 9. 3D view model of the combined multiple 2D LiDAR scans plotted using a 3D scatter plot function.

Figure 9 is a visualisation of the 3D view model of the scanning environment plotted using the LiDAR point cloud from each individual layer of 2D LiDAR scan. The 3D reconstruction of the environmental mapping information is recorded in the form of a matrix and is visualised using the 3D scatter plot function in MATLAB.

Figure 9(a) is a simple top-down view of the completed 3D LiDAR view model in the x and y dimensions of the Cartesian coordinate plane. The top-down view shown in the image is in accordance with the previous Polar plot results after combining the 2 LiDAR scanning inputs, therefore the 3D reconstructed model of the scanning environment is inferred to be an accurate visual representation of the spatial information. Following that, Fig. 9(b) is the front view of the 3D view model of the scanning environment. The result of the front view of the 3D model is in line with expectations as the 3D view model is formed by recording multiple 2D LiDAR scanning results at varying elevation levels with the sensor at a fixed origin point, giving it a stacked layering appearance. Lastly, Fig. 9(c) shows a two-point view of the 3D view model, the figure visualising the scanning environment with a high degree of accuracy in the third dimensional space and also representing the scanning environment in an easy to understand and simplistic manner.

The 3D view model helps to present the scanned environment in a more dimensional and sized based perspective with more user-friendly viewing options instead of visualising the spatial information of the scanning environment in a vague and ambiguous 2D plane. Furthermore, the constructed 3D view model information could be stored in the form of a matrix for a variety of different future uses which requires a digital 3D reconstruction of the environment.

IV. CONCLUSION

The research proposes a more readily accessible alternative to 3D LiDAR mapping technology by stacking multiple 2D LiDAR scans forming a view model focused on dimensionality and scalability. To improve the accuracy of each individual 2D LiDAR scan result, the DBSCAN clustering algorithm was applied to the scanning dataset to independently identify and isolate the noise clusters in the scanning dataset. Furthermore, to address the discontinuous spatial issue of the LiDAR scan caused by limited FOV, the primary scan result is supplemented with a secondary LiDAR scan of the environment from a position with a better view of the dead zone. Lastly, the research aims to represent the recorded spatial data of the scanning environment in the form of a 3D view model, this helps to improve visualising the scanning result by presenting them in perspective relative to the scale and dimension of the objects in the environment.

ACKNOWLEDGEMENT

This research is fully funded and supported by the Department of Electrical and Electronics Engineering of the University of Nottingham Malaysia.

REFERENCES

[1] L. Giuffrida, G. Masera, and M. Martina, “A Survey of Automotive Radar and Lidar Signal Processing and Architectures,” *Chips*, vol. 2, no. 4, pp. 243–261, 2023.

[2] N. Van Toan, P. B. Khoi and S. Y. Yi, "A Single 2-D LiDAR Extrinsic Calibration for Autonomous Mobile Robots," *IEEE Trans. Instrument. and Measure.*, vol. 72, no. 8507109, pp. 1-9, 2023.

[3] A. E. Akay, H. Oğuz, I. R. Karas and K. Aruga, “Using LiDAR Technology in Forestry Activities,” *Environ. Monit. Assess.*, vol. 151, pp. 117–125, 2009.

[4] S. A. Gargoum and K. El Basyouny, “A Literature Synthesis of Lidar Applications in Transportation: Feature Extraction and Geometric Assessments of Highways,” *GIsci. Remote Sens.*, vol. 56, no. 6, pp. 864-893, 2019.

[5] G. K. Wedajo, “LiDAR DEM Data for Flood Mapping and Assessment; Opportunities and Challenges: A Review,” *J. Remote Sens. & GIS*, vol. 6, no. 4, pp. 1000211, 2017.

[6] H. Hu, T. M. Fernandez-Steeger, M. Dong, T. Nguyen and R. Azzam, “3D Modeling Using LiDAR Data and Its Geological and Geotechnical Applications,” in *18th Int. Conf. Geoinformatics*, Beijing, pp. 1-6, 2010.

[7] Y. Tian, F. Liu, H. Liu, Y. Liu, H. Suwoyo, T. Jin, L. Li and J. Wang, “A Real-Time and Fast LiDAR–IMU–GNSS SLAM System with Point Cloud Semantic Graph Descriptor Loop-Closure Detection,” *Adv. Intellig. Syst.*, vol. 5, no. 10, pp. 2300138, 2023.

[8] D. Deng, "DBSCAN Clustering Algorithm Based on Density," in *2020 7th Int. Forum on Electr. Eng. and Automat.*, Hefei, pp. 949-953, 2020.

[9] K. P. Sinaga and M. S. Yang, “Unsupervised K-means Clustering Algorithm,” *IEEE Access*, vol. 8, pp. 80716–80727, 2020.

[10] L. Morissette and S. Chartier, “The K-means Clustering Technique: General Considerations and Implementation in Mathematica,” *Tutor Quant Methods Psychol.*, vol. 9, no. 1, pp. 15–24, 2013.

[11] M. Hahsler, M. Piekenbrock and D. Doran, “Dbscan: Fast Density-based Clustering with R,” *J. Stat. Softw.*, vol. 91, no. jss.v091.i01, 2019.