International Journal of Engineering and Geosciences
https://dergipark.org.tr/en/pub/ijeg
e-ISSN 2548-0960

Accuracy Evaluation of LiDAR-SLAM Based 2-Dimensional Modelling for Indoor Environment: A Case Study

Aleyna Başaran [1], Veli İlçi [*1]

[1] *Ondokuz Mayis University, Department of Geomatics Engineering, Türkiye, basarannaleyna@gmail.com, veli.ilci@omu.edu.tr*

Cite this study: Başaran, A., & İlçi, V. (2025). Accuracy Evaluation of LiDAR-SLAM Based 2-Dimensional Modelling for Indoor Environment: A Case Study. International Journal of Engineering and Geosciences, 10 (1), 74-83.

https://doi.org/10.26833/ijeg.1519533

Keywords
LiDAR
SLAM
Modelling
NDT

Research Article

Received:
Revised:
Accepted:
Published:

Abstract

The rapid development of sensor technologies has led to smaller sensor sizes and lower costs. Today, the easy-of-use purchasing of sensors such as cameras, Light Detection and Ranging (LiDAR), Radio Detection and Ranging (RADAR), Inertial Measurement Units (IMUs), and Global Navigation Satellite System (GNSS) receivers have led to significant developments in many applications such as robotics and unmanned vehicles. Sensor data is transformed into information or products thanks to the methods. Simultaneous Localization and Mapping (SLAM) is one of the critical methods in which the vehicle's location is determined, and the environment is modelled. This method can realize applications using detection sensors such as cameras, LiDAR, or RADAR. This study aimed to model an indoor area with a two-dimensional (2D) LiDAR sensor placed on an Unmanned Ground Vehicle (UGV) and to analyse the accuracy of the produced model. Normal Distribution Transform (NDT) - Particle Swarm Optimization (PSO) algorithm was used to generate the 2D model from the collected LiDAR data. The NDT-PSO algorithm was executed on the Robot Operating System (ROS) installed on the Jetson Nano Developer Kit, and a real-time 2D model of the working area was processed. The reference lengths of the 75 facades in the 232 m² indoor space were measured using a total station and calculated with CAD software. Percent error values were evaluated by comparing the reference and model lengths of the facades.

1. Introduction

With the development of technology, there have been significant advances in mobile robot technologies, and nowadays, robots have become usable for essential tasks. In order for a robot to reach its goal location safely, it must plan a path that can proceed without striking obstacles [1]. The ability of mobile robots to create maps and track their positions while moving in a specific environment is generally achieved by the Simultaneous Localization and Mapping (SLAM) method. With sensor data, SLAM allows the robot to understand its environment and update its position as it moves. In this way, the robot can adapt to changes in its environment and track its current location on the map in real-time [2].

With the advancement of the SLAM method, significant developments have been made in spatial mapping [3]. SLAM enables robots to move safely and effectively with real-time path planning, especially in tasks requiring complex manoeuvres [4]. Thanks to these capabilities, mobile robots can be used in areas such as rescue operations [5], industrial automation [6], agriculture [7], logistics [8], and cleaning [9], archeology [10], increasing work efficiency and helping people avoid tasks that may be dangerous. Mobile robots can generally be classified as outdoor (such as agricultural robots), or indoor robots (such as warehouse logistics robots). This classification helps to understand what jobs and environments a robot can be used in [11].

In outdoor areas, Global Navigation Satellite Systems (GNSS) are mainly used for positioning purposes [12–14]. While GNSS provides high-accuracy position information in outdoor environments, such as low signal strength, multipath, obstacles, and scattering, negatively affect signals and prevent them from reaching the desired positioning accuracy [15, 16]. GNSS and Inertial

Measurement Unit (IMU) sensors are used together to reduce GNSS-related errors, increase position accuracy, and ensure continuity in partially and entirely GNSS-denied environments [17]. IMU consists of an accelerometer, gyroscope, and magnetometer sensors that determine the robot's position, speed, and direction. However, IMU cannot be used alone for long-term applications because the accuracy provided by the IMU decreases over time due to drift error. GNSS and IMU sensors are used concurrently to eliminate each other's deficiencies and provide more accurate long-term and continuous position and orientation information [18]. Loosely-Coupled and Tightly Coupled models containing Kalman Filter (KF) are generally used in GNSS/IMU integration. In the Loosely-Coupled architecture, the speed and position information provided by GNSS is used to adjust the IMU, while in the Tightly-Coupled architecture, the raw GNSS measurements are directly integrated into the navigation filter. However, these integration models are known to be more successful in outdoor environments [19].

Depending on the sensor type, the SLAM method is divided into Light Detection and Ranging SLAM (LiDAR-SLAM) and Visual SLAM (V-SLAM) [20]. Monocular, stereo, and RGB-D cameras are used in V-SLAM. While the monocular cameras cannot determine the actual size of images, stereo, and RGB-D cameras have been developed to ensure metric accuracy. Stereo cameras estimate the depth of each pixel but require a complex structure and high processing power. RGB-D cameras obtain depth information using infrared light or the time-of-flight (ToF) principle, however, they have disadvantages such as narrow measurement range, limited field of view, and sensitivity to sunlight [21].

Recently, advances in sensing hardware have reduced the size and weight of LiDAR, enabling portable LiDAR systems to be more widely used in navigation [22]. LiDAR, an active sensor, has significant advantages such as providing more accurate information about the environment, being less affected by light changes, and requiring simple error models [23–25]. Therefore, point cloud information obtained with LiDAR contains direct geometric relationships, making the robot's path planning and navigation more precise.

The open-source Robot Operating System (ROS), developed in 2007, is frequently used to run SLAM algorithms [26]. It offers tools and libraries for running SLAM algorithms, processing data, and visualizing the results [27]. Robotic problems such as autonomous navigation and environmental interaction can be solved with ROS, and visualization can be realized with software such as RViz and Gazebo [28].

In this study, a remotely controlled Unmanned Ground Vehicle (UGV) was designed to produce a two-dimensional (2D) model of an indoor area. A 2D LiDAR sensor and a processor were mounted on the UGV, and the LiDAR data collected during the vehicle's movement and visualized in real-time with the ROS. The collected data was processed using the Normal Distribution Transform (NDT) Particle Swarm Optimization (PSO) (NDT-PSO) LiDAR-SLAM algorithm, and a 2D model of the indoor environment was produced. In order to determine the accuracy of the model, the facades in the study area were measured using terrestrial surveying methods, and the reference dimensions of the facades were calculated. The percent error values of the facades in the model were determined by comparing the reference dimensions.

2. LiDAR Sensors

Distance measurement was technically introduced by Middleton and Spilhaus in 1953, but the LiDAR principle as we know it today emerged after Dennis Gabor's holography method in 1945 and the development of new techniques such as laser in 1960 [29]. The term LiDAR initially emerged as a combination of the terms light and radar [30]. This technology is used today, especially in autonomous vehicles [31], mapping [32], environmental monitoring [33] and many other areas.

LiDAR systems are classified as one-dimensional (1D), two-dimensional (2D), and three-dimensional (3D). The working principle of these systems is the same, but there are differences, such as the use of a point-to-point measurement mechanism, the scanning mode system, and the number of laser beams used. In 1D LiDAR, the distance between an obstacle and an object is measured using a fixed laser beam [34]. 2D LiDAR sensors record X and Y parameters using only a single beam axis and determine the positions of objects on a surface in the horizontal plane. 3D LiDAR sensors work like 2D sensors, but additional measurements determine an object's 3D position and height [35] but more processing time is required due to the larger data output of 3D LiDAR [36]. 2D LiDAR systems are widely preferred for production at lower costs, higher accuracy, and ease of commercialization in more basic usage areas [37, 38]. The cost difference between 2D and 3D LiDAR sensors significantly impacts these sensor's applications [39].

LiDAR sensors work by scanning the Field of View (FOV), which indicates how many views the sensor can provide in a specific angular range in horizontal and vertical directions [40]. FOV and lateral resolution are determined by factors such as the receiver optics, photodetector size, and the divergence angle of the laser. While the FOV for 2D LiDAR is limited to the horizontal plane, 3D scanners include both horizontal and vertical planes [41]. Particularly in autonomous vehicles and drones, a wide FOV is vital in obtaining a 360-degree view of the environment [42].

LiDAR systems obtain distance information based on the ToF propagation of light [43, 44]. ToF is directly estimated from the time difference between transmitting and receiving signal pulses [45]. The distance calculation between the object and the sensor is shown in Equation 1.

$$d = \frac{c.T}{2} \quad (1)$$

Where d is the distance between the object and the sensor, c is the speed of light, and T is the period.

2.1. 2D LiDAR-SLAM

The general workflow of 2D LiDAR-SLAM can be given as follows:

- An initial map of the surroundings is created from LiDAR data to obtain environmental information,
- Based on the created map information, the starting position of the mobile robot and the decisive points around are calculated,
- The mapping and positioning process continue during the movement of the mobile robot. New maps are created, and the mobile robot's position information is updated,
- New landmarks are created depending on the new location of the mobile robot and the current map,
- The determined new points are compared with the previous points, and the detected new points are used to update the position of the mobile robot,
- The mobile robot navigates the environment based on current data and thus renews the route planning [46–49].

Many recent studies have been conducted on 2D LiDAR in indoor environments. In [50], the YDLIDAR X4 sensor and Raspberry Pi 3B processor were used together with a mobile robot that can move manually or automatically to create a 2D map of a room. The position of the robot was determined using the Hector SLAM algorithm. They performed wall tracking of the room using Proportional-Integral-Derivative (PID) control and created a 2D mapping of the room with an error rate of 5.32%. [51] aimed to segment room maps using 2D RPLIDAR A1 and Raspberry Pi. The study was carried out in eight different environments, and room segmentation was achieved at 96.5%. In [52], it was argued that GNSS and cameras are insufficient for Unmanned Aerial Vehicles (UAVs) to operate autonomously in closed and low-light environments. To create real-time estimates of the UAV pose, they used cheap and lightweight 2D LiDAR, which is not affected by light compared to other sensors and can work in closed areas such as historical buildings, and obtained a 3D model. [53] compared semantic information such as location, direction, and shape of the environment with metric information using 2D LiDAR in building areas. They verified the success of the location recognition system by comparing it with the Adaptive Monte Carlo Localization (AMCL) algorithm. Results indicate that as the number of points increases, the location accuracy also increases.

2.2. 2D LiDAR-SLAM algorithms

Nowadays, many algorithms are being developed for 2D LiDAR-SLAM, which is becoming more widespread day by day [67,68]. One of these algorithms, Hector-SLAM, is proposed by Kohlbrecher [54] which is based on the Gauss-Newton approach and scan matching technique that is used for environmental 2D grid mapping, and LiDAR with a high update rate performs the scan matching task to quickly and accurately determine the robot's position [55]. Unlike other SLAM algorithms, it relies not on odometry data but on laser scans and a pre-built map. Gmapping, which uses the Rao-Blackwellized Particle Filter (RBPF), is another 2D LiDAR-SLAM algorithm that aims to minimize the variance of particle weights when choosing the recommendation distribution. Unlike other SLAM algorithms, RBPF determines the robot's motion model using point cloud information, not as a recommendation distribution of particle sampling that helps reduce significant errors caused by motion noise [9]. On the other hand, Google developed Cartographer for real-time mapping of indoor spaces[22]. The primary purpose of this algorithm is to optimize efficiency by processing data from particle filters. Creating the map is executed by dividing the data into small pieces and then combining these pieces to create the road. At the same time, this algorithm also uses a mechanism to reduce errors from the robot's position. In this way, a more accurate and reliable map can be obtained [56]

In addition to these algorithms, there are also point cloud matching algorithms such as Iterative Closest Point (ICP) and Normal Distribution Transform (NDT) in 2D LiDAR-SLAM. The ICP algorithm finds the transformation parameters by minimizing the Euclidean distance between the nearest neighbour points. However, since the ICP must create exact matches between points, the quality of the raster registration may decrease when the point cloud is sparse [57]. ICP does not always provide a global solution, and the result often depends on the accuracy of the initial pose. If an incorrect starting pose is obtained, the algorithm produces either an incorrect or a local solution [58]. NDT offers a more compact spatial representation of a point cloud [59]. NDT exceeds the limitations of ICP, such as long operation time and initial point-related errors, by matching the maps using statistical characteristics [60]. On the other hand, the drawback of the NDT, a map-based scan matching algorithm, is the significant changes in the robot's surroundings and occlusion by moving objects [61].

2.2.1. NDT-PSO

NDT is an improved version of an occupancy table approach and is often used to process data such as a 2D laser scan [62]. NDT uses a set of local probability density functions of Gaussian normal distributions to transform a point cloud into a piecewise continuous function. In this method, each normal distribution describes the surface shape in an occupied space cell. In this way, NDT preserves each cell or region's surface features and distribution while creating a continuous representation from the point cloud. NDT can process the point cloud, record data, and define environmental features. For example, it is used in applications such as SLAM to determine location using the geometry and features of the environment [63]. The NDT scan matching algorithm calculates the average of cell points and its covariance in Equations 2 and 3, respectively.

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{2}$$

$$\sigma = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x}) * (x_i - \bar{x})^T \tag{3}$$

Here, n represents the number of points in each subsets x, $\bar{x}$ is the mean value of the points, and σ is the covariance of the x_i points.

Widely used SLAM methods are scan-matching approaches that can be computationally burdened, and sensitive to the position initialization error. The fact that PSO falls to the global minimum with a higher probability without position initialization provides an advantage in solving the SLAM problem [64].

PSO is an algorithm inspired by the movements of the swarm [65]. In the iterative PSO algorithm, each solution is called a *particle*, and the solution process starts with a population called a *swarm*. Swarm optimization is formulated as Equation 4:

$$v_{t+1}(p) = \sum_n F_N\,(p, w_N) \tag{4}$$

In Equation 4, where p is the given particle, v is the velocity vector, t is the iteration, F_N is the function of the different attractive forces affecting the particle motions, and w_N is the weighting factors that balance the importance of each force. The F_N is affected by three forces namely momentum, cognitive, and social behaviours. Momentum behaviour is expressed by the pulling forces to maintain its current motion.

$$F_M = w_M v_t\,(p) \tag{5}$$

In Equation 5, w_M refers to the inertial weight factor and v_t is the particle's current velocity. At each iteration, the position and velocity of each particle within the population are updated using *Pbest* and *Gbest* values. *Pbest* represents the state in which a particle has come closest to a solution, while *Gbest* represents the state in which it has come closest to a solution among all particles. In this process, each particle moves toward a new location [66] The second behaviour F_c, as shown in Equation 6, indicates the forces affecting the particle towards Pbest.

$$F_c = w_c\,|rand_1|\,\left(Pbest_t(p) - X_t\,(p)\right) \tag{6}$$

In this equation, w_c represents the cognitive weight factor, X is the position vector, X_t gives the current position vector of the particle, $rand_1$ represents the randomly generated value at each speed change, which varies between 0-1.

$$F_s = w_s\,|rand_2|\,\left(Gbest_t - X_t\,(p)\right) \tag{7}$$

The third behaviour F_s, shown in Equation 7, considers the social behavior of the particle and adjusts its movement according to $Gbest$. This value represents the best position the swarn has reached so far. w_s refers to the social weight factor, $rand_2$ refers to its randomly generated value at each speed change and varies in the range of 0-1. The position of each particle is updated according to Equation 8.

$$X_{t+1}\,(p) = X_t(p) + v_{t+1}\,(p) \tag{8}$$

In NDT-PSO, the position estimation problem is solved by encoding the geometric transformation (translation-Tx, Ty, and rotation - θ) between two scans on a particle (Equation 9).

$$X_{(p)} = T = \left(T_x, T_y, \theta\right) \tag{9}$$

More detailed information about the NDT-PSO algorithm is given in [64].

The performance of NDT-PSO in indoor and outdoor areas has been tested in both static and dynamic environments [64]. The map and estimated positions obtained with the NDT-PSO approach maintain accuracy under various conditions, including loop-closure situations and scenarios with moving objects. The algorithm offers high accuracy and reliability, including real-time applications against complex and variable environmental conditions. It has also been determined that the algorithm works quickly and efficiently, meeting the speed and response times required for real-time systems. This feature makes NDT-PSO an effective solution for systems with real-time application requirements like mobile robots.

3. Case study

In this study, an unmanned ground vehicle (UGV) was developed for 2D mapping of an indoor area. YDLiDAR TG-30 laser scanner with a range of 30 meters was used to collect detailed information about the environment. Jetson Nano processor was used to collect and process LiDAR data. Power was supplied to the processor using a 7.4V LiPo battery, and a regulator was also used to step down the power to the appropriate voltage. A schematic representation of the UGV and sensor settlement is given in Figure 1.

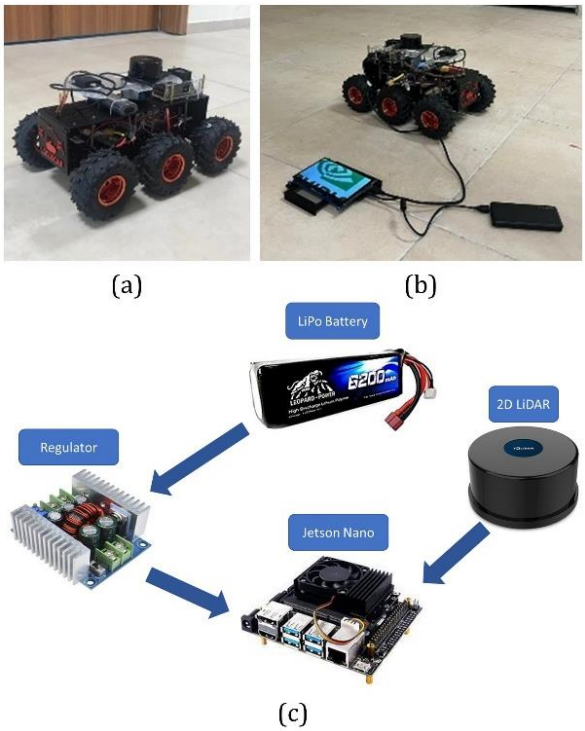

Figure 1. General view of the developed UGV (a and b) and the used devices and system configuration (c)

A shelter of 232 m^2 that is in the Ondokuz Mayis University, Geomatics Engineering Building was chosen as the study area. The angle and distance observations of the detail points from the facades were measured with the Topcon GM-50 Series total station (Figure 2a). This total station provides 5" angle accuracy and 2mm+2ppm non-prism distance accuracy. Then, the point coordinates of the detail points were calculated and converted into a 2D drawing in the CAD environment (Figure 2b). A 2D map of the study area was produced in a local coordinate system, and these values were accepted as reference values for the study area.

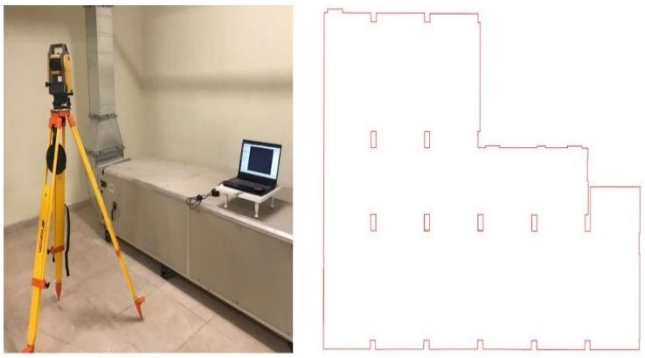

Figure 2. (a) 2D Mapping the working area with Total Station, (b) 2D reference map of the study area

The SLAM method maps the environment and accurately estimates the vehicle's pose. In this study, the NDT-PSO algorithm, one of the SLAM methods, was used. While NDT provides modelling of the environment, PSO provides pose estimation. The LiDAR data collected through the ROS system installed on the Jetson Nano processor was processed using the NDT-PSO algorithm, and real-time imaging was provided with Rviz. CloudCompare software was used to visualize the produced 2D model and measure the facade lengths on the model (Figure 3).

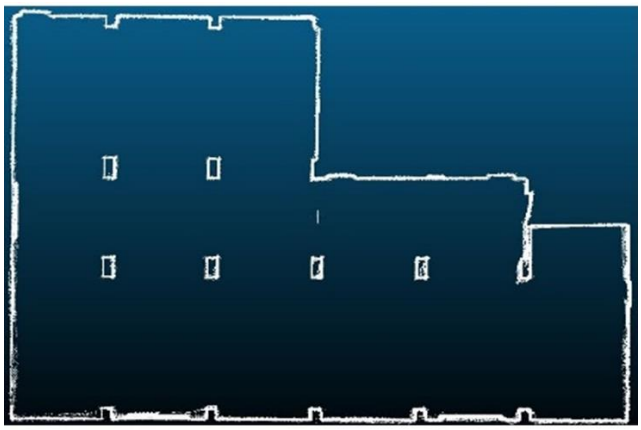

Figure 3. 2D model obtained with the NDT-PSO algorithm

4. Results

Seventy-five facades of the indoor area were determined to analyze the provided accuracies by the NDT-PSO algorithm (Figure 4). The reference lengths of the facades were measured on the CAD environment, while the facade lengths on the 2D model produced with the NDT-PSO algorithm were measured using CloudCompare software. Percent error values, which indicate the magnitude of the error, were calculated by comparing the facade lengths on the reference and model dimensions. The percent error was calculated according to Equation 10.

$$\varepsilon_p = \frac{|y_r - y_m|}{|y_r|} * 100 \quad (10)$$

In Equation 10, ε_p, y_r, and y_m define the percent error, reel value, and measured value, respectively. The detailed statistics are given in Table 1.

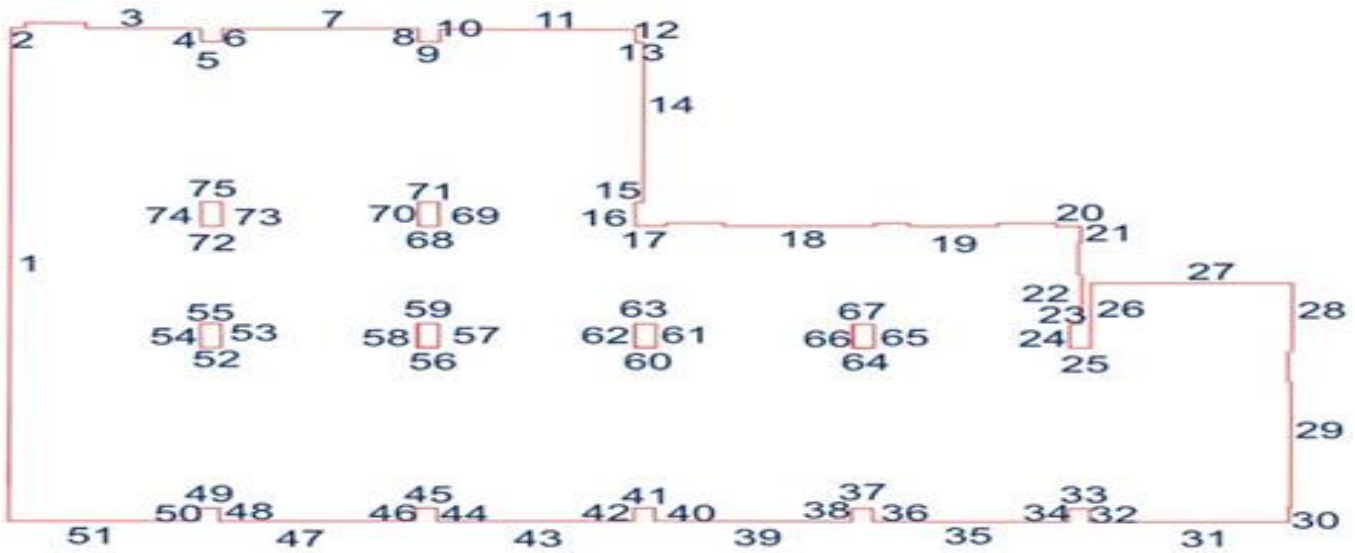

Figure 4. The given facade numbers on the CAD drawing

Table 1. The reference, model, difference, and percent error values related to the facades

Facade No	Reference (cm)	Model (cm)	Error (cm)	% Percent Error	Facade No	Reference (cm)	Model (cm)	Error (cm)	% Percent Error
1	1498.1	1500.0	1.9	0.1	**39**	308.0	309.0	1.0	0.3
2	21.2	20.7	0.5	2.4	**40**	40.0	40.2	0.2	0.5
3	179.3	180.5	1.3	0.7	**41**	33.3	33.1	0.1	0.4
4	39.5	39.2	0.3	0.7	**42**	38.8	38.7	0.1	0.3
5	32.6	32.7	0.1	0.2	**43**	307.4	306.7	0.7	0.2
6	39.0	38.7	0.3	0.9	**44**	40.0	40.8	0.8	2.1
7	305.3	305.5	0.2	0.1	**45**	33.0	32.4	0.6	1.9
8	40.0	39.9	0.2	0.5	**46**	39.1	39.1	0.1	0.2
9	33.6	33.1	0.6	1.7	**47**	306.4	310.3	3.9	1.3
10	38.0	38.9	0.9	2.3	**48**	40.0	38.9	1.1	2.8
11	305.5	308.6	3.1	1.0	**49**	33.0	32.4	0.6	1.8
12	39.0	39.9	0.9	2.3	**50**	39.2	38.2	0.9	2.3
13	13.5	13.9	0.4	2.7	**51**	296.7	295.7	1.0	0.3
14	486.0	490.8	4.8	1.0	**52**	32.4	32.6	0.2	0.7
15	12.5	12.6	0.1	0.5	**53**	73.0	71.6	1.4	1.9
16	71.3	70.3	1.0	1.4	**54**	73.0	71.0	2.0	2.8
17	48.0	48.1	0.1	0.2	**55**	32.4	32.9	0.5	1.6
18	236.0	237.3	1.3	0.5	**56**	33.0	33.5	0.5	1.4
19	139.0	141.8	2.8	2.0	**57**	73.1	71.0	2.1	2.9
20	39.0	38.0	1.0	2.6	**58**	73.1	71.0	2.1	2.9
21	51.0	52.4	1.4	2.7	**59**	33.0	32.9	0.1	0.3
22	149.0	149.9	0.9	0.6	**60**	33.0	32.5	0.5	1.6
23	18.7	18.6	0.1	0.5	**61**	72.2	71.4	0.8	1.1
24	72.6	71.0	1.6	2.2	**62**	72.2	70.8	1.3	1.9
25	33.0	32.4	0.6	1.9	**63**	33.0	32.7	0.4	1.2
26	197.0	196.6	0.4	0.2	**64**	32.9	32.5	0.4	1.1
27	315.2	317.8	2.7	0.8	**65**	72.0	71.2	0.8	1.1
28	209.0	210.9	1.9	0.9	**66**	72.0	70.8	1.2	1.6
29	383.2	383.5	0.3	0.1	**67**	32.9	32.0	0.8	2.6
30	42.3	41.2	1.1	2.6	**68**	32.0	32.3	0.4	1.1
31	311.5	316.8	5.3	1.7	**69**	73.0	71.1	1.9	2.6
32	40.0	39.5	0.5	1.3	**70**	73.0	71.6	1.4	1.9
33	33.0	33.6	0.6	1.7	**71**	32.0	31.7	0.3	0.9
34	40.0	39.3	0.7	1.7	**72**	32.4	32.3	0.1	0.4
35	305.4	306.7	1.3	0.4	**73**	73.0	71.5	1.5	2.0
36	39.7	39.9	0.1	0.3	**74**	73.0	72.4	0.6	0.9
37	33.0	32.3	0.7	2.2	**75**	32.4	31.6	0.8	2.5
38	39.1	39.8	0.8	1.9					

The table shows the data comprising 75 facades ranging from 0.13 to 15 meters. It was calculated that the minimum and maximum difference values between the two data sets were 0.1 cm and 5.3 cm, respectively, and the average of the difference values was 1.0 cm. Figure 5 shows that percent error values are in the range of 0-3%.

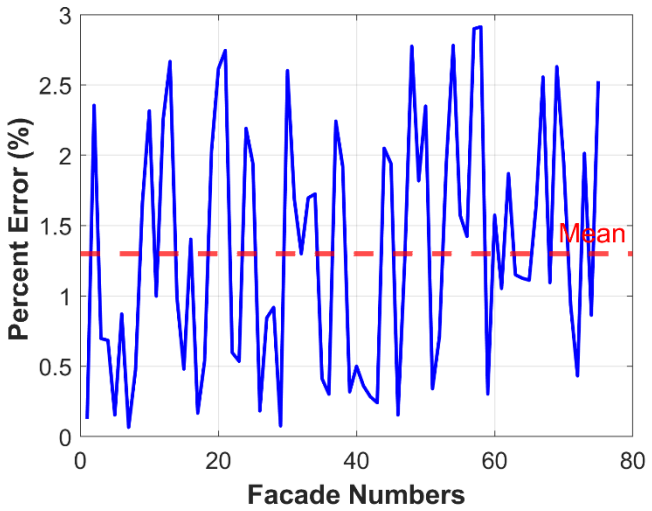

Figure 5. Percent error (%) of the façades calculated from the reference and model lengths

Current standards for large-scale maps in Türkiye are regulated by Large-Scale Map and Map Information Production Regulation 2018. According to this regulation, detailed measurements of facade control are given in Equation 11.

$$d = 0.05 + 0.001 * S \quad (11)$$

Where d is the error limit between the reference and model values in meter, and S is façade length in meter. According to this formula, it is seen that the error limits, depending on the length, vary between 5.0 and 6.5 cm. Considering that our model errors vary between 0.1 and 5.3 cm, it can be concluded that the model data can be used for facade length estimation.

There is no similar study conducted with the NDT-PSO algorithm that compares the model and the reference lengths. The closest study to ours is the study performed by [50] using the Hector SLAM method explained in Section 2.1. In the study, the actual wall lengths of the room were compared with the wall lengths measured with 2D LiDAR and the average error percentage was determined as 5.32%. When comparing the results with our study, it can be concluded that the obtained average percent error of 1.3% provides higher performance.

5. Conclusion

Nowadays, the types of applications using mobile robots are increasing day by day. One of the most critical applications is SLAM, in which mobile robots can calculate their positions and model their surroundings. This study aimed to produce a 2D model of an indoor area using the 2D LiDAR sensor data mounted on a mobile robot and to determine the accuracy of the produced model. NDT-PSO SLAM algorithm was used to generate the model from LiDAR data. In order to determine the accuracy of the model produced with the SLAM algorithm, the reference data of the study area was produced using a total station. When the reference map is compared with the model obtained with the NDT-PSO algorithm, it is seen that error values in the range of 0-3% are reached proportionally. Thanks to this model, which was produced in real-time with the NDT-PSO algorithm, it has been observed that the intense workloads of measurement, calculation, and drawing required in terrestrial techniques can be eliminated. These accuracy values show that the method can be used for most applications except for studies requiring mm-level accuracy. However, modelling with 2D LiDAR produces only a 2D model of the work area, which cannot provide sufficient detailed information for applications requiring 3D modelling. In future studies, modelling will be carried out with 3D LiDAR-SLAM algorithms.

Acknowledgement

This work has been supported by Ondokuz Mayıs University Scientific Research Projects Coordination Unit under project numbers PYO.MUH.1906.22.002, and PYO.MUH.1908.22.080. We also appreciate the LOCUS-TEAM members for their support during this study.

Author contributions

Aleyna Başaran: Software, Field study, Data curation, Visualization, Writing-Original draft preparation **Veli İlçi:** Conceptualization, Methodology, Investigation, Validation, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

1. Han, S., & Xi, Z. (2020). Dynamic Scene Semantics SLAM Based on Semantic Segmentation. IEEE Access, 8, 43563–43570. https://doi.org/10.1109/ACCESS.2020.2977684
2. Tee, Y. K., & Han, Y. C. (2021). Lidar-Based 2D SLAM for Mobile Robot in an Indoor Environment: A Review. In 2021 International Conference on Green Energy, Computing and Sustainable Technology, (GECOST) (pp. 1–7). Miri: IEEE. https://doi.org/10.1109/GECOST52368.2021.9538731
3. Han, X., Li, S., Wang, X., & Zhou, W. (2021). Semantic Mapping for Mobile Robots in Indoor Scenes: A Survey. Information, 12(2), 1–14. https://doi.org/10.3390/info12020092
4. Özbayrak, S., & İlçi, V. (2024). Visual-SLAM based 3-dimensional modelling of indoor environments. International Journal of Engineering and Geosciences, 9(3), 368–376. https://doi.org/10.26833/ijeg.1459216
5. Ghorpade, D., Thakare, A. D., & Doiphode, S. (2017). Obstacle Detection and Avoidance Algorithm for Autonomous Mobile Robot using 2D LiDAR. In 2017 Third International Conference on Computing, Communication, Control and Automation (ICCUBEA) (pp. 1–6). Pune, India: IEEE. https://doi.org/10.1109/ICCUBEA.2017.8463846
6. Gomes, D., Alvarez, M., Brancalião, L., Carneiro, J., Gonçalves, G., Costa, P., … Pinto, V. H. (2022). Data Analysis for Trajectory Generation for a Robot

Manipulator Using Data from a 2D Industrial Laser. Machines, 10(10), 907. https://doi.org/10.3390/machines10100907

7. Shang, Y., Wang, H., Qin, W., Wang, Q., Liu, H., Yin, Y., … Meng, Z. (2023). Design and Test of Obstacle Detection and Harvester Pre-Collision System Based on 2D Lidar. Agronomy, 13(2), 388. https://doi.org/10.3390/agronomy13020388
8. Afrisal, H., Nugraha, G. K., Nanda, A. A., Setiyadi, A. D., Toirov, O., Ismail, R., … Setiawan, I. (2022). Mobile Robotic-Arm Development for A Small-Scale Inter-Room Logistic Delivery using 2D LIDAR-guided Navigation. Teknik, 43(2), 158–167. https://doi.org/10.14710/teknik.v43i2.45642
9. Sui, L., & Lin, L. (2020). Design of Household Cleaning Robot Based on Low-cost 2D LIDAR SLAM. In 2020 International Symposium on Autonomous Systems, (ISAS) (pp. 223–227). IEEE. https://doi.org/10.1109/ISAS49493.2020.9378863
10. Kaderli, L. (2021). Documentation Methods from Tradition to the Present: Case Study Cappadocia. Advanced LiDAR, 1(1), 18–26.
11. Niloy, M. A. K., Shama, A., Chakrabortty, R. K., Ryan, M. J., Badal, F. R., Tasneem, Z., … Saha, D. K. (2021). Critical Design and Control Issues of Indoor Autonomous Mobile Robots: A Review. IEEE Access, 9, 35338–35370. https://doi.org/10.1109/ACCESS.2021.3062557
12. Koca, B., & Ceylan, A. (2018). Uydu Konum Belirleme Sistemlerindeki (GNSS) Güncel Durum ve Son Gelişmeler. Geomatik Dergisi, 3(1), 63–73.
13. İlçi, V., & Peker, A. U. (2022). The Kinematic Performance of Real-Time PPP Services in Challenging Environment. Measurement, 189, 110434. https://doi.org/10.1016/j.measurement.2021.110434
14. Li, Y., & Ibanez-Guzman, J. (2020). Lidar for Autonomous Driving: The Principles, Challenges, and Trends for Automotive Lidar and Perception Systems. IEEE Signal Processing Magazine, 37(4), 50–61. https://doi.org/10.1109/MSP.2020.2973615
15. İlçi, V., Gülal, E., & Alkan, R. M. (2018). An Investigation of Different Wi-Fi Signal Behaviours and Their Effects on Indoor Positioning Accuracy. Survey Review, 50(362), 404–411. https://doi.org/10.1080/00396265.2017.1292672
16. İlçi, V., Gülal, E., & Alkan, R. M. (2020). Performance Comparison of 2.4 and 5 GHz WiFi Signals and Proposing a New Method for Mobile Indoor Positioning. Wireless Personal Communications, 110, 1493–1511. https://doi.org/10.1007/s11277-019-06797-x
17. Onyekpe, U., Palade, V., & Kanarachos, S. (2021). Learning to Localise Automated Vehicles in Challenging Environments Using Inertial Navigation Systems (INS). Applied Sciences, 11(3), 1270. https://doi.org/10.3390/app11031270
18. Gürtürk, M., & İlçi, V. (2022). The Performance Evaluation of PPK and PPP-based Loosely Coupled Integration in Wooded and Urban Areas. Earth Sciences Research Journal, 26, 211–220. https://doi.org/10.15446/esrj.v26n3.100518
19. Jiang, W., Li, Y., Rizos, C., Cai, B., & Shangguan, W. (2017). Seamless Indoor-Outdoor Navigation based on GNSS, INS and Terrestrial Ranging Techniques. Journal of Navigation, 70, 1183–1204. https://doi.org/10.1017/S037346331700042X
20. Dai, Y., Wu, J., Wang, D., & Watanabe, K. (2023). A Review of Common Techniques for Visual Simultaneous Localization and Mapping. Journal of Robotics, 2023, 1–21. https://doi.org/10.1155/2023/8872822
21. Chen, Y., Zhou, Y., Lv, Q., & Deveerasetty, K. K. (2018). A Review of V-SLAM. In International Conference on Information and Automation (pp. 603–608). Wuyi Mountain, China: IEEE.
22. Huang, L. (2021). Review on LiDAR-based SLAM Techniques. In 2021 International Conference on Signal Processing and Machine Learning, (CONF-SPML) (pp. 163–168). Stanford, CA, USA: IEEE. https://doi.org/10.1109/CONF-SPML54095.2021.00040
23. Sarıtaş, B., & Kaplan, G. (2024). A Comprehensive Study on Enhanced Accuracy Analysis of LIDAR Data: The Example of Skopje. Advanced LiDAR, 4(1), 9–18.
24. Sevgen, S. C. (2019). Airborne LiDAR Data Classification in Complex Urban Area Using Random Forest: A Case Study of Bergama, Turkey. International Journal of Engineering and Geosciences, 4(1), 45–51. https://doi.org/10.26833/ijeg.440828
25. Demirel, Y., & Türk, T. (2024, December 31). Assessment of the location accuracy of points obtained with a low-cost Lidar scanning system and GNSS method. Mersin Photogrammetry Journal. Mersin University. https://doi.org/10.53093/mephoj.1540159
26. He, F., & Zhang, L. (2023). Design of Indoor Security Robot based on Robot Operating System. Journal of Computer and Communications, 11(5), 93–107. https://doi.org/10.4236/jcc.2023.115008
27. Olalekan, A. F., Sagor, J. A., Hasan, M. H., & Oluwatobi, A. S. (2021). Comparison of Two SLAM Algorithms Provided by ROS (Robot Operating System). In 2021 2nd International Conference for Emerging Technology (INCET) (pp. 1–5). Belagavi, India: IEEE. https://doi.org/10.1109/INCET51464.2021.9456164
28. Nagla, S. (2020). 2D Hector SLAM of Indoor Mobile Robot using 2D Lidar. In 2020 2nd International Conference on Power, Energy, Control and Transmission Systems (ICPECTS) (pp. 1–4). Chennai, India: IEEE. https://doi.org/10.1109/ICPECTS49113.2020.9336995
29. Roriz, R., Cabral, J., & Gomes, T. (2022). Automotive LiDAR Technology: A Survey. IEEE Transactions on Intelligent Transportation Systems, 23(7), 6282–6297. https://doi.org/10.1109/TITS.2021.3086804
30. Protopopov, V. V. (2009). Laser Heterodyne Radars and Lidars (Vol. 149). Berlin, Heidelberg: Springer Series in Optical Sciences.

31. Gao, B., Hu, G., Zhong, Y., & Zhu, X. (2021). Cubature Kalman Filter with Both Adaptability and Robustness for Tightly-Coupled GNSS/INS Integration. IEEE Sensors Journal, 21(13), 14997–15011. https://doi.org/10.1109/JSEN.2021.3073963
32. Bitenc, M., Lindenbergh, R., Khoshelham, K., & van Waarden, A. P. (2011). Evaluation of a LIDAR Land-Based Mobile Mapping System for Monitoring Sandy Coasts. Remote Sensing, 3(7), 1472–1491. https://doi.org/10.3390/rs3071472
33. Uthe, E. E. (1983). Application of Surface Based and Airborne Lidar Systems for Environmental Monitoring. Journal of the Air Pollution Control Association, 33(12), 1149–1155. https://doi.org/10.1080/00022470.1983.10465705
34. Mehendale, N. , & Neoge, S. ,. (2020). Review on LiDAR technology. SSRN, 1–9. https://doi.org/http://dx.doi.org/10.2139/ssrn.3604309
35. Ozdemir, S., Akbulut, Z., Karsli, F., & Acar, H. (2021). Automatic extraction of trees by using multiple return properties of the lidar point cloud. International Journal of Engineering and Geosciences, 6(1), 20–26. https://doi.org/10.26833/ijeg.668352
36. Peng, Y. , Qu, D. , Zhong, Y. , Xie, S. , Luo, J. , & Gu, J. (2015). The Obstacle Detection and Obstacle Avoidance Algorithm Based on 2-D Lidar. In 2015 IEEE International Conference on Information and Automation (pp. 1648–1653). Lijiang, China: IEEE.
37. Promneewat, K., & Taksavasu, T. (2024). Performance of Affordable 2D Cave Scanning Technique from LiDAR for Constructing 3D Cave Models. Advanced LiDAR, 4(1), 1–8.
38. Meng, J., Wan, L., Wang, S., Jiang, L., Li, G., Wu, L., & Xie, Y. (2021). Efficient and Reliable LiDAR-Based Global Localization of Mobile Robots Using Multiscale/Resolution Maps. IEEE Transactions on Instrumentation and Measurement, 70, 1–15. https://doi.org/10.1109/TIM.2021.3093933
39. Queralta, J. P., Yuhong, F., Salomaa, L., Qingqing, L., Gia, T. N., Zou, Z., & Westerlund, T. (2019). FPGA-based Architecture for a Low-Cost 3D Lidar Design and Implementation from Multiple Rotating 2D Lidars with ROS. In 2019 IEEE Sensors (pp. 1–4). Montreal, QC: IEEE. https://doi.org/10.1109/SENSORS43011.2019.8956928
40. Nguyen, X. T., Kim, H., & Lee, H. J. (2020). A Gradient-Aware Line Sampling Algorithm for LiDAR Scanners. IEEE Sensors Journal, 20(16), 9283–9292. https://doi.org/10.1109/JSEN.2020.2986819
41. Raj, T., Hashim, F. H., Huddin, A. B., Ibrahim, M. F., & Hussain, A. (2020). A survey on LiDAR scanning Mechanisms. Electronics, 9(5), 741. https://doi.org/10.3390/electronics9050741
42. Behroozpour, B., Sandborn, P. A. M., Wu, M. C., & Boser, B. E. (2017). Lidar System Architectures and Circuits. IEEE Communications Magazine, 55(10), 135–142. https://doi.org/10.1109/MCOM.2017.1700030
43. Hassan, T., Fath-Allah, T., Elhabiby, M., Awad, A., & El-Tokhey, M. (2022). Detection of GNSS no-line of Sight Signals Using LiDAR Sensors for Intelligent Transportation Systems. Survey Review, 54(385), 301–309. https://doi.org/10.1080/00396265.2021.1937458
44. Günen, M. A., Erkan, İ., Aliyazıcıoğlu, Ş., & Kumaş, C. (2023). Investigation of geometric object and indoor mapping capacity of Apple iPhone 12 Pro LiDAR. Mersin Photogrammetry Journal, 5(2), 82–89. https://doi.org/10.53093/mephoj.1354998
45. Dai, Z., Wolf, A., Ley, P. P., Glück, T., Sundermeier, M. C., & Lachmayer, R. (2022). Requirements for Automotive LiDAR Systems. Sensors, 22(19), 7532. https://doi.org/10.3390/s22197532
46. Z. J. Chong, B. Qin, T. Bandyopadhyay, M. H. Ang, E. Frazzoli, & D. Rus. (2013). Mapping with Synthetic 2D LIDAR in 3D Urban Environment. In 2013 IEEE/RSJ International Conference on Intelligent Robots and Syste (IROS) (pp. 4715–4720). Tokyo: IEEE.
47. Gul, F., Rahiman, W., & Nazli Alhady, S. S. (2019). A Comprehensive Study for Robot Navigation Techniques. Cogent Engineering, 6(1). https://doi.org/10.1080/23311916.2019.1632046
48. Habib, M. K. (2007). Real Time Mapping and Dynamic Navigation for Mobile Robots. International Journal of Advanced Robotic Systems, 4(3), 323–338.
49. Khan, M. U., Zaidi, S. A. A., Ishtiaq, A., Bukhari, S. U. R., Samer, S., & Farman, A. (2021). A Comparative Survey of LiDAR-SLAM and LiDAR based Sensor Technologies. In 2021 Mohammad Ali Jinnah University International Conference on Computing (MAJICC) (pp. 1–8). Karachi, Pakistan: IEEE. https://doi.org/10.1109/MAJICC53071.2021.9526266
50. Rivai, M., Hutabarat, D., & Jauhar Nafis, Z. M. (2020). 2D mapping using omni-directional mobile robot equipped with LiDAR. Telkomnika (Telecommunication Computing Electronics and Control), 18(3), 1467–1474. https://doi.org/10.12928/TELKOMNIKA.v18i3.14872
51. Zheng, T., Duan, Z., Wang, J., Lu, G., Li, S., & Yu, Z. (2021). Research on Distance Transform and Neural Network Lidar Information Sampling Classification-Based Semantic Segmentation of 2D Indoor Room Maps. Sensors, 21(4), 1–20. https://doi.org/10.3390/s21041365
52. Petrlik, M., Krajnik, T., & Saska, M. (2021). LIDAR-based Stabilization, Navigation and Localization for UAVs Operating in Dark Indoor Environments. In 2021 International Conference on Unmanned Aircraft Systems (ICUAS) (pp. 243–251). Athens, Greece: ICUAS. https://doi.org/10.1109/ICUAS51884.2021.9476837
53. Bae, S. H., Joo, S. H., Choi, J. H., Park, H. J., & Kuc, T. Y. (2022). Localization System Through 2D LiDAR based Semantic Feature For Indoor Robot. In 2022 19th International Conference on Ubiquitous Robots (UR) (pp. 338–342). Jeju, Korea: IEEE. https://doi.org/10.1109/UR55393.2022.9826250
54. P, T. S. , P, S. , Muppidi, A. J. , & Pagala, P. S. (2020). Analysis of Computational Need of 2D-SLAM

Algorithms for Unmanned Ground Vehicle. In Proceedings of the International Conference on Intelligent Computing and Control Systems (pp. 230–235). Madurai: IEEE.
55. Eliwa, M., Adham, A., Sami, I., & Eldeeb, M. (2017). A critical comparison Between Fast and Hector SLAM Algorithms. REST Journal on Emerging trends in Modelling and Manufacturing, 3(2), 44–49.
56. Trejos, K., Rincón, L., Bolaños, M., Fallas, J., & Marín, L. (2022). 2D SLAM Algorithms Characterization, Calibration, and Comparison Considering Pose Error, Map Accuracy as Well as CPU and Memory Usage. Sensors, 22(18), 6903. https://doi.org/10.3390/s22186903
57. Das, A., & Waslander, S. L. (2014). Scan Registration Using Segmented Region Growing NDT. International Journal of Robotics Research, 33(13), 1645–1663. https://doi.org/10.1177/0278364914539404
58. Prados Sesmero, C., Villanueva Lorente, S., & Di Castro, M. (2021). Graph SLAM Built Over Point Clouds Matching for Robot Localization in Tunnels. Sensors, 21(16), 5340. https://doi.org/10.3390/s21165340
59. Jang, K. W., Jeong, W. J., & Kang, Y. (2022). Development of a GPU-Accelerated NDT Localization Algorithm for GNSS-Denied Urban Areas. Sensors, 22(5), 1913. https://doi.org/10.3390/s22051913
60. Lee, J., Lee, K., Yoo, A., & Moon, C. (2020). Design and Implementation of Edge-Fog-Cloud System Through HD Map Generation from Lidar Data of Autonomous Vehicles. Electronics, 9(12), 1–15. https://doi.org/10.3390/electronics9122084
61. Kan, Y. C., Hsu, L. T., & Chung, E. (2021). Performance Evaluation on Map-Based NDT Scan Matching Localization Using Simulated Occlusion Datasets. IEEE Sensors Letters, 5(3), 1–4. https://doi.org/10.1109/LSENS.2021.3060097
62. Huang, H.-C., Xu, S. S.-D., Lin, H.-C., Xiao, Y.-S., & Chen, Y.-X. (2023). Design and Implementation of Intelligent LiDAR SLAM for Autonomous Mobile Robots Using Evolutionary Normal Distributions Transform. Soft Computing, 5321–5337. https://doi.org/10.1007/s00500-023-09219-0
63. Chen, S., Ma, H., Jiang, C., Zhou, B., Xue, W., Xiao, Z., & Li, Q. (2022). NDT-LOAM: A Real-Time Lidar Odometry and Mapping With Weighted NDT and LFA. IEEE Sensors Journal, 22(4), 3660–3671. https://doi.org/10.1109/JSEN.2021.3135055
64. Bouraine, S. , Bougouffa, A. , & Azouaoui, O. (2020). NDT-PSO, a New NDT based SLAM Approach using Particle Swarm Optimization. In 2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV) (pp. 321–326). Shenzhen, China: IEEE. https://doi.org/doi: 10.1109/ICARCV50220.2020.9305519.
65. Feng, H. M. (2006). Self-Generation RBFNs using Evolutional PSO Learning. Neurocomputing, 70(1–3), 241–251. https://doi.org/10.1016/j.neucom.2006.03.007
66. Jain, M., Saihjpal, V., Singh, N., & Singh, S. B. (2022). An Overview of Variants and Advancements of PSO Algorithm. Applied Sciences, 12(17), 8392. https://doi.org/10.3390/app12178392
67. Yakar, M.; Yılmaz, H.M.; Mutluoğlu, Ö. Comparative evaluation of excavation volume by TLS and total topographic station based methods. Lasers Eng. 2010, 19, 331–345.
68. Yakar, M., Yilmaz, H. M., & Mutluoglu, O. (2014). Performance of photogrammetric and terrestrial laser scanning methods in volume computing of excavtion and filling areas. Arabian Journal for Science and Engineering, 39, 387-394.