

Autonomous 3D Scanning of Construction Sites Using Terrestrial Laser Scanner and UGV

Pranav Shevkar¹  and Jan Luca Fahrendholz-Heiermann¹ 

¹Chair of Individualized Production in Architecture, RWTH Aachen University, Aachen, Germany

E-mail(s): shevkar@ip.rwth-aachen.de, fahrendholz@ip.rwth-aachen.de

Abstract: Accurate representation of construction sites for Artificial Intelligence (AI) processing and process monitoring demands complete 3D point clouds. Conventionally, a tripod-mounted terrestrial laser scanner (TLS) is employed for generating 3D point clouds at multiple locations, followed by post-processing, which is time-consuming. This paper proposes an autonomous scanning system integrating TLS with an Unmanned Ground Vehicle (UGV) to generate 3D point clouds. The approach employs frontier-based exploration to autonomously navigate the robot to optimal scan waypoints within predefined GPS-marked areas on an outdoor construction site. Subsequently, the high accuracy and long-range TLS scanner record a point cloud, which is registered, and a new optimal waypoint for the robot is computed. This iterative exploration and scanning process continues until the point cloud is complete and no new scanning frontier is identified within the scan area. The effectiveness of the newly developed autonomous scanning technique is evaluated through a comparative analysis of number of scan positions, point cloud completeness, and the time needed for manual scanning of outdoor construction sites. The proposed pipeline enables efficient collection of high-quality point clouds in outdoor environments, laying the groundwork for further artificial intelligence processing.

Keywords: 3D point cloud generation, ROS, autonomous exploration, terrestrial laser scanner, wave-front frontier detection



Erschienen in Tagungsband 35. Forum Bauinformatik 2024, Hamburg, Deutschland, DOI: 10.15480/882.13544

© 2024 Das Copyright für diesen Beitrag liegt bei den Autoren. Verwendung erlaubt unter Creative Commons Lizenz Namensnennung 4.0 International.

1 Introduction

Point cloud data technology has become a vital component of the construction industry, offering numerous applications that significantly enhance various aspects of construction projects. A primary application of point cloud data is in 3D model reconstruction, which includes both geometric and semantic model reconstruction. Additionally, point cloud data is crucial for quality inspection tasks, such as dimensional quality inspection, surface quality inspection, and displacement inspection [1]. Furthermore, 3D point clouds are instrumental in generating 3D data [2] and are utilized on construction sites for progress monitoring as well as raw material monitoring. This enables automated comparisons between the as-built and the as-planned state, allowing early detection of deviations during the construction process [3], [4].

This research is focused on the generation of point clouds on construction sites, specifically in outdoor environments. Flying laser scanners such as the 'Leica BLK2FLY', tower crane mounted laser scanners such as 'Blickfeld cube 1' as well as Terrestrial Laser Scanners (TLS) are particularly suited for outdoor scanning. This study focusses on TLS scanners due to their high dimensional accuracy and capability to generate dense point clouds. These scanners typically incorporate complex optics, motors, single-board computers, MEMS sensors, GPS antennas, and batteries for capturing and processing point clouds. However, manually scanning outdoor environments using a tripod is labor-intensive and time-consuming, highlighting the need for automated laser scanning solutions. Although tower crane-mounted laser scanners enable faster scanning, they are prone to increased blind spots caused by shadowing. On the other hand, flying laser scanners provide lower detail, are highly dependent on the weather conditions, and strict aviation regulations which can limit their usage in certain areas [5].

In this study, we utilize a Riegl VZ-400i terrestrial laser scanner mounted on an INNOK Heros 224 mobile robot. This setup allows for remote navigation to the desired scan positions. On larger construction sites, such as the one in this study, approximately 10 scans from various positions are required to construct a comprehensive and detailed point cloud, which remains a time-consuming process. To address these challenges, we propose an autonomous scanning system that integrates TLS and an UGV for optimal and fully autonomous 3D point cloud creation in outdoor environments. This system leverages a streamlined pipeline to enhance efficiency and reduce the time and labor associated with point cloud generation on construction sites.

The paper is organized as follows: Section 2 introduces related works; Section 3 describes the proposed system; Section 4 presents the experimental analysis; discussions about the current implementation are mentioned in Section 5; and finally, our conclusions are presented in Section 6.

2 Related Work

Autonomous 3D scanning of built environments was first developed by Surmann et al. [6] where they used a UGV equipped with a 3D laser scanner for navigation. The next best view (NBV) was selected based on the highest information gain value. In [7], Strand et al. proposed a 2D grid-based next best view strategy for 3D scanning of indoor environments. In this approach, navigation-specific information of the 3D world was projected onto a 2D grid. A weighting function was computed by simulating laser beams in 360° to select the next best viewpoint. Kim et al. introduced the Ground Robot Mapping Infrastructure (GRoMI) for dynamic SLAM-based navigation for 3D scanning using multiple 2D scanners [8]. Researchers in [9] developed a two-stage strategy for pose planning in outdoor environments. First, a 2D map was used as a reference to create a rough initial 3D model of the environment. This model was then used to plan the next best view in the second stage. Liang et al. used a TLS-based laser scanner on a UGV for autonomous scanning and implemented a UFOMap-based next best view algorithm for optimal scanner placement [10].

While there have been several studies on the development of autonomous 3D scanning using TLS and UGV, most research has focused on indoor scanning. This study specifically aims to automate the 3D scanning process in outdoor environments with minimal prior information.

3 Methodology

3.1 Hardware

For scanning the construction sites, a Riegl VZ-400i terrestrial laser scanner is used which provides predefined scanning patterns. Pattern 'Panorama40' is chosen for this application as it provides 34mm resolution at a distance of 50 m. This pattern records 22.5 Mio points at a frequency of 1200 kHz in 45 seconds.

To conduct the scanning autonomously, we employ the HEROS 224 robot from Innok Robotics as the UGV. This mobile robot has a compact footprint (920 x 730 x 440 mm), which enables it to navigate in tight spaces. With its sturdy construction and high ground clearance of 141 mm, it can easily traverse rough terrain. The UGV can carry a maximum payload of 200 kg. It is equipped with a SICK safety laser scanner for obstacle detection and an Xsens MTI-30 IMU, which provides absolute orientation through the fusion of accelerometer, gyroscope, and magnetometer data. Additionally, an Emlid Reach RS2+ multi-band RTK GNSS receiver is retrofitted on the rear side of the UGV to facilitate GPS-based navigation.

To securely mount the Riegl scanner to the Innok UGV, a custom fixture is installed on the front side of the robot. This fixture holds the Riegl scanner securely while allowing for easy mounting and unmounting. The UGV with the Riegl TLS is shown in Figure 1.



Figure 1: Innok UGV mounted with the Riegl TLS

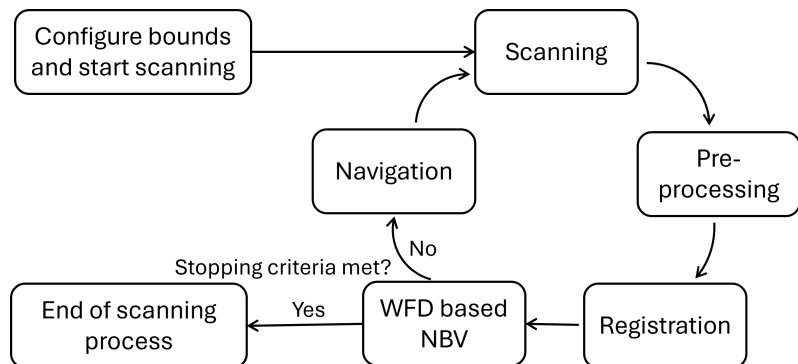


Figure 2: Flowchart of the implemented autonomous scanning pipeline

3.2 Software

Figure 2 provides an overview of the scanning pipeline of the developed system. The prior information for starting the algorithm includes a list of GPS coordinates that acts as a boundary for the scanning which is defined in a YAML file. The pipeline comprises five distinct modules that operate in a loop to capture the point cloud of the construction site which is similar to the work presented in [10]. These

modules are scan acquisition, pre-processing, frontier detection, navigation, and registration. Each module is further divided into sub-modules, which are discussed in detail in the following sections.

3.2.1 Scan acquisition module

The ROS2 package `ros-riegl-vz`, provided by RIEGL Laser Measurement Systems GmbH, was used for the development of scanner control and point-cloud data acquisition nodes. This package enables the writing of ROS2 nodes for configuring scan parameters, scanner control, and point cloud acquisition. The scanner, UGV computer, and processing computer are all connected to the same WLAN network, allowing the topics to be discoverable over the network.

Scanning is triggered automatically using a ROS2 service when the robot reaches the defined scanning location. Predefined parameters such as scan pattern, level of detail, fast scan registration mode, and GPS bounds are used. These parameters significantly affect the scan duration; hence, selecting the optimal pattern and level of detail that meets the requirements is crucial. Subsequently, the acquired scan is published over a topic as a `sensor_msgs/PointCloud2` message, which can take between 2 to 3 minutes with the selected configuration depending on the size of the point cloud.

3.2.2 Pre-processing

In the pre-processing module, every scan undergoes initial processing before further operations on the point cloud. During scanning, a large number of points outside the area of interest are captured, resulting in unnecessary accumulation of points in the registered point cloud. This not only increases the size of the point cloud but also extends processing times. To mitigate this issue, each scan is initially cropped using GPS bounds to remove extraneous points and reduce the size of the 3D scan. The first scan is established as the global reference scan, with its origin serving as the reference system origin.

Next, the cropped scan undergoes down-sampling to further reduce the number of points, utilizing a voxel size of 0.5 meters. Subsequently, Fast Point Feature Histograms (FPFH) are computed for both the source and target point clouds. These FPFH values play a crucial role in computing the initial global RANSAC transformation.

3.2.3 Registration

In the registration module, a transformation matrix is calculated and applied to the newly pre-processed point cloud to align it with the reference point cloud. Initially, a coarse principal axis alignment is performed using a transform obtained from the robot pose. Fine alignment is then done using point-to-plane ICP algorithm using the Open3D library which is available for Python [11]. Point-to-plane ICP requires normal estimation which is done using the hybrid KD-tree approach. In this research, the search radius was set to 0.1m and maximum number of nearest neighbors for this search to 30. This returns a registration result, which includes the refined transformation matrix that best aligns the new point cloud to the previous registered point cloud.

3.2.4 Next Best View Waypoint Computation

Each time a new scan is acquired, it undergoes transformation using the registration-derived transformation matrix. The point cloud is subsequently filtered along the z-axis based on parameters minZ and maxZ. The minZ parameter is configured to remove ground undulations, while maxZ ensures the robot navigates around structures such as shelters at the construction site, not considering them as obstacles. This filtered point cloud is then down-projected to generate an Occupancy Grid Map (OGM). The robot's current absolute pose is calculated by fusing GPS, IMU, and magnetometer sensor data. The Wavefront Frontier Detection (WFD) algorithm, implemented as a ROS2 node, identifies all frontiers in the OGM larger than twice the robot's size. The median pose of each frontier is computed and sorted based on Euclidean distance from the robot's current pose. Using the Nav2 package in ROS2, the robot navigates towards the closest frontier median. If the path planner fails to compute a path to the nearest frontier, the next closest frontier is selected iteratively [12]. Upon reaching the median pose, a new scan is acquired, and the process repeats. Offline mapping from the SLAM toolbox integrates the down-projected submap with the global map [13].

4 Experimental Analysis

To evaluate the effectiveness of the developed pipeline, we conducted an experiment on a reference construction site on the Melaten Campus of RWTH Aachen University covering an area of approximately 4000 m². The chosen site featured a diverse array of elements, including tower cranes, excavators, containers, shelters, and various other construction materials.

First, the GPS coordinates of the scan area were recorded in the configuration file of the planning and processing node. The area is visualized on the map in Figure 3 below.



Figure 3: Visualization of the scan area on a map

Next, a scan was captured following the process outlined in Figure 2. The robot then moved to the next scan location determined by the WFD algorithm. After each scan, the OGM was updated with

the occupancy information computed by down-projecting the transformed 3D point cloud, and new frontiers were explored. During navigation, the robot avoided obstacles detected by the Sick laser scanner and re-planned the route to the goal location. During experimentation, we optimized the frontier area to avoid navigation to smaller frontiers and carefully adjusted the maximum height to allow the robot to pass through shelters without entering other open doors.

In this case study, the robot computed 12 frontiers to cover the complete construction site within the given GPS bounds. Every point cloud captured by the TLS scanner was successfully registered. However, some gaps were found in the registered point cloud at locations where the robot could not successfully plan a path. The scan locations are shown in the top view of the registered point cloud in Figure 4.

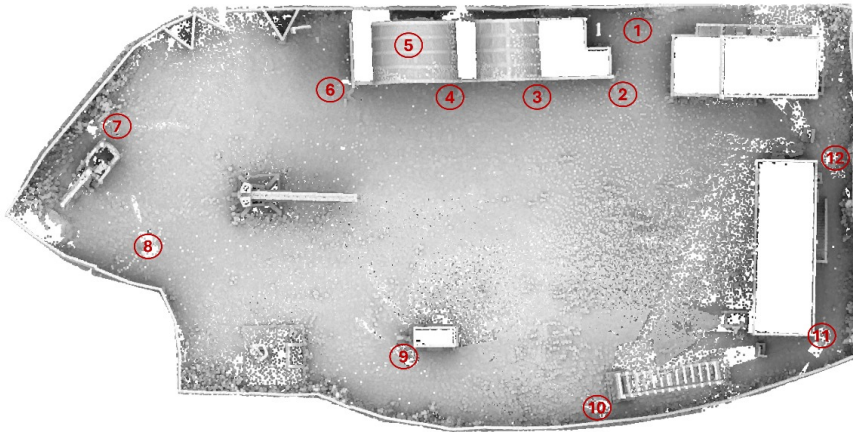


Figure 4: Sequence of scans acquired using the WFD algorithm for getting the scanner position

To compare the efficacy of the proposed pipeline, an experienced colleague performed a manual scanning process of the same construction site. Using the manual method, 16 scans were required to cover the entire site. An experimental comparison between the manual and autonomous scanning methods is presented in Table 1. The comparison shows that the automated method required fewer scans and less time to cover the construction site. The automated method results in fewer points compared to the manual scanning process due to reduced redundancy in scanning the same areas.

Table 1: Comparison of Automated and Manual Methods

Method	#Scans	#Points	Time	Pre-processing time
Automated	12	6,545,862	1 h	0
Manual	16	9,542,703	2 h	2 h

The pre-processing time is zero in case of the proposed automated method as the pre-processing is conducted on-the-fly and the point-cloud is registered. In case of manual method, the point-clouds must be downloaded from the TLS and then processed in a software such as RiSCAN PRO to get the registered point-cloud.

5 Discussion

The proposed solution follows the "stop-scan-plan-go" scanning pattern, contrasting with the "scanning-while-navigating" pattern that requires continuous planning and exploration using the SLAM technique's lifelong mapping algorithm. Although "scanning-while-navigating" approaches, such as those in [8] and [14], are efficient, they often generate noisy point clouds, rendering them unsuitable for capturing architectural details. In comparison, automated Terrestrial Laser Scanning (TLS) methods, as discussed in [10], are more precise. Additionally, some researchers have suggested computing the NBV using unknown voxels [15]. However, our research found that this method is most effective in closed environments.

Implementing an automated TLS system mounted on an UGV can lead to a high initial cost but the reduction in labor costs alone can be significant, as UGVs eliminate the need for skilled operators and can work continuously without breaks or fatigue. Considering an initial UGV cost of €60,000, amortized over 30 projects and maintenance cost of €200 per project, the total operational cost would be €2200 per project. Labour cost with a skilled worker taking €20 per hour for a 20 day project would be €3200 per project. While there is an upfront cost associated with implementing an automated TLS system on a UGV platform, the long-term benefits often outweigh these initial expenses through substantial reductions in labor costs, increased operational efficiency, and enhanced safety measures.

6 Conclusion

This research presents an autonomous UGV-based scanning method tailored for outdoor environments, requiring minimal prior information and human intervention. The method achieves scanning accuracy comparable to that of an experienced human operator, with a reduced number of scans. This was validated through testing on a reference construction site.

Future work will focus on optimizing the algorithm to analyze gaps in the point clouds and compute a NBV pose to address them. Additionally, this research can be expanded to include the identification and analysis of specific construction elements, such as sand piles, for volumetric data extraction.

Funding

This work is part of the KI for BauChain research project funded by Zukunftbau, Bundesinstitut für Bau-, Stadt- und Raumforschung (BBSR), and Bundesministerium für Wohnen, Stadtentwicklung und Bauwesen (BMWSB) (project number 10.08.18.7-22.15).

References

- [1] Q. Wang and M.-K. Kim, "Applications of 3d point cloud data in the construction industry: A fifteen-year review from 2004 to 2018", *Advanced Engineering Informatics*, 2019. DOI: 10.1016/j.aei.2019.02.007.
- [2] C. Thomson and J. Boehm, "Automatic Geometry Generation from Point Clouds for BIM", *Remote Sensing*, vol. 7, no. 9, pp. 11 753–11 775, 2015. DOI: 10.3390/rs70911753.

- [3] A. Braun, A. Borrmann, S. Tuttas, and U. Stilla, “BIM-Basead Progress Monitoring”, in *Building Information Modeling*, 2018, pp. 527–544.
- [4] J. Fahrenholz, L. Kirner, and S. Brell-Cokcan, “Entwicklung und umsetzung einer automatisierten baufortschrittsüberwachung mittels deep learning basierend auf punktwolken und bauinformationsmodellen”, in *IoC - Internet of Construction*, S. Brell-Cokcan and R. Schmitt, editors, Wiesbaden: Springer Vieweg, 2024. DOI: 10.1007/978-3-658-42544-9_22.
- [5] A. Rao, M. Radanović, Y. Liu, *et al.*, “Real-time monitoring of construction sites: Sensors, methods, and applications”, *Automation in Construction*, vol. 136, p. 104 099, Apr. 2022. DOI: 10.1016/j.autcon.2021.104099.
- [6] “An autonomous mobile robot with a 3d laser range finder for 3d exploration and digitalization of indoor environments”, *Robotics and Autonomous Systems*, vol. 45, no. 3, pp. 181–198, 2003. DOI: <https://doi.org/10.1016/j.robot.2003.09.004>.
- [7] M. Strand and R. Dillmann, “Using an attributed 2d-grid for next-best-view planning on 3d environment data for an autonomous robot”, Jul. 2008, pp. 314–319. DOI: 10.1109/ICINFA.2008.4608017.
- [8] P. Kim, J. Chen, and Y. Cho, “Slam-driven robotic mapping and registration of 3d point clouds”, *Automation in Construction*, vol. 89, pp. 38–48, May 2018. DOI: 10.1016/j.autcon.2018.01.009.
- [9] P. S. Blaer and P. K. Allen, “Data acquisition and view planning for 3-d modeling tasks”, in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 417–422. DOI: 10.1109/IROS.2007.4399581.
- [10] N. Liang, Y. P. Ang, K. Yeo, X. Wu, Y. Xie, and Y. Cai, “Bimbot for autonomous laser scanning in built environments”, *Robotics*, vol. 13, p. 22, Jan. 2024. DOI: 10.3390/robotics13020022.
- [11] Q.-Y. Zhou, J. Park, and V. Koltun, “Open3D: A modern library for 3D data processing”, *arXiv:1801.09847*, 2018.
- [12] A. Topiwala, P. Inani, and A. Kathpal, “Frontier based exploration for autonomous robot”, *arXiv:1806.03581*, 2018.
- [13] S. Macenski and I. Jambrecic, “Slam toolbox: Slam for the dynamic world”, *Journal of Open Source Software*, vol. 6, p. 2783, May 2021. DOI: 10.21105/joss.02783.
- [14] J. Zhang and S. Singh, “Loam : Lidar odometry and mapping in real-time”, *Robotics: Science and Systems Conference (RSS)*, pp. 109–111, Jan. 2014.
- [15] D. Duberg and P. Jensfelt, “Ufomap: An efficient probabilistic 3d mapping framework that embraces the unknown”, *IEEE Robotics and Automation Letters*, Aug. 2020. DOI: 10.1109/LRA.2020.3013861.