MAPPING BUILDING INTERIORS WITH LIDAR: CLASSIFYING THE POINT CLOUD WITH ARCGIS

Jason R. Parent[a]*, Chandi Witharana[b], Michael Bradley[a]

[a]Dept. of Nat. Res. Sci., University of Rhode Island, Kingston, USA - jason_parent@uri.edu, michael_bradley@uri.edu
[b]Dept. of Nat. Res. and the Env., University of Connecticut, Storrs, CT, USA - chandi.witharana@uconn.edu

KEY WORDS: Lidar, indoor, interior, building, point cloud, classification

ABSTRACT:

Accurate maps of building interiors are needed to support location-based services, plan for emergencies, and manage facilities. However, suitable maps to meet these needs are not available for many buildings. Handheld LiDAR scanners provide an effective tool to collect data for indoor mapping but there are no well-established methods for classifying features in indoor point clouds. The goal of this research was to develop an efficient manual procedure for classifying indoor point clouds to represent features-of-interest.

We used Paracosm's PX-80 handheld LiDAR scanner to collect point cloud and image data for 11 buildings, which encompassed a variety of architectures. ESRI's ArcGIS Desktop was used to digitize features that were easily identified in the point cloud and Paracosm's Retrace was used to digitize features for which imagery was needed for efficient identification. We developed several tools in Python to facilitate the process. We focused on classifying 29 features-of-interest to public safety personnel including walls, doors, windows, fire alarms, smoke detectors, and sprinklers.

The method we developed was efficient, accurate, and allowed successful mapping of features as small as a sprinkler head. Point cloud classification for a 14,000 m^2 building took 20-40 hours, depending on building characteristics. Although the method is based on manual digitization, it provides a practical solution for indoor mapping using LiDAR. The methods can be applied in mapping a wide variety of features in indoor or outdoor environments.

1. INTRODUCTION

1.1 Introduction

Maps of building interiors provide critical information that allows public safety personnel to pre-plan responses to disasters, building managers to efficiently manage their facilities, and visitors to find their destinations. Accurate building maps will be key for indoor navigation and tracking systems that improve the safety and effectiveness of first responders as well as improve convenience of visitors to public buildings (Kunhoth et al. 2020). However, accurate "as-built" floor plans that are suitable for current and future needs are not available for many buildings.

Efforts to map indoor environments face some unique challenges caused by limited sight lines, rapid changes in elevation (i.e. stairs), inability to use global positioning systems (GPS), and inability to use mechanized platforms for carrying remote sensing instruments. Indoor mapping systems must be highly mobile to avoid blind spots and compact and lightweight enough to be carried comfortably by a human. Light detection and ranging (LiDAR) scanners create point clouds which can represent features in 3D space. The recent emergence of handheld LiDAR scanners provides a practical solution to indoor mapping by combining LiDAR with high mobility. These scanners use inertial mapping units (IMU) with simultaneous localization and mapping (SLAM) algorithms (Droeschel and Behnk 2018; Diosi and Kleeman 2005; Thrun et al. 2004; Castellanos et al. 1998) to generate point cloud data as the unit is carried through the building. These scanners also typically integrate a wide angle or spherical RGB camera to collect images simultaneously with the LiDAR data and which can be used to colorize the point cloud.

LiDAR-based systems are rapidly advancing to improve data collection capabilities; however, there are no established procedures for classifying point cloud data or extracting features-of-interest for indoor environments. Software that is commonly used in GIS and remote sensing focuses on classification of data that represent outdoor environments. These software tend to be poorly suited to the massive datasets and unique characteristics of point clouds created for indoor environments.

This research aims to develop techniques for classifying features-of-interest for indoor LiDAR point clouds. We present an efficient manual process based on ESRI's[1] ArcGIS Desktop and Paracosm's[2] Retrace software along with custom tools developed in Python. The process combines the advantages of using imagery for feature identification with the precise positioning provided by the point cloud. We focus on mapping features of interest to first responders including walls, doors, windows, fire alarms, sprinkler heads, etc. The procedure and tools we developed will help establish effective methods of classifying point clouds and mapping indoor environments. The tools and products developed in this research will be made available for download at https://www.edc.uri.edu/indoor-mapping-project.

[1] Environmental Systems Research Institute, www.esri.com

[2] Paracosm is an Occipital Company (https://occipital.com).

This contribution has been peer-reviewed.

2. METHODS

2.1 Study Area Description

The study area consisted of 11 buildings located in Enfield and Storrs, Connecticut. The buildings consisted of 6 schools, 1 office building, 1 industrial building, and 3 university buildings. The buildings had a combined floor space of approximately 130,000 m^2. Nine of the buildings had multiple stories. The buildings were constructed between 1960 and 2017 and represented a variety of architectures and features including high atrium ceilings, gymnasiums, classrooms, offices, and garages.

2.2 Equipment and Data Collection

LiDAR and spectral image data were collected with Paracosm's PX-80 scanner (Figure 1). The scanner collects LiDAR and image data simultaneously using a Velodyne VLP-16 LiDAR puck integrated with a 250º RGB camera (Paracosm 2018). The LiDAR sensor collects up to 300,000 pts/sec with a maximum range of 80 m and a reported relative point accuracy of ±2 cm. The Lidar sensor has a 360º horizontal field-of-view (FOV) and a ±15º vertical FOV. The camera captures RGB images at a 3.2 MP resolution and 50 frames per second with a 360º horizontal FOV and a 250º vertical FOV. The user interface for the PX-80 is an iPad running Paracosm's Capture software. The software provided a real-time preview of the point cloud so that scanning problems could be found and addressed while on site. The outputs of the PX-80 included a colorized point cloud in an .las v1.2 format, a point cloud in a .ply format, and photos from the camera in .jpeg formats.

Data were collected by carrying the PX-80 through the data collection area at a medium walking speed (e.g. 2.5 kph). Scan durations were limited to 30 minutes or less, per Paracosm's recommendations, to minimize progressive error accumulation (i.e. drift) in the unit's inertial mapping unit (IMU). Whenever possible, we closed scan loops by ending the scan where we started to allow the SLAM software to compensate for drift.

We followed several practices, recommended by Paracosm, to minimize the occurrence of "tracking loss" in which the SLAM algorithm fails and causes major misalignments of features in the scan. Given the scanner's 360º horizontal field-of-view, the goal of these practices was to help ensure the SLAM algorithm was able to use previously mapped areas to maintain orientation when entering new areas. These practices included moving smoothly with no sudden changes in motion, limiting motion to 2 axes at a time, pausing in doorways for 3-5 seconds, and walking down the center of hallways especially when going around corners. The scanner was particularly prone to tracking loss after exiting small, cluttered rooms (e.g. closets and utility rooms) and when scanning narrow stairwells. Thus, we avoided entering small rooms by scanning them from the door and tipping the scanner into the room to collect points along the wall and ceiling just inside the doorway. This technique provided good point cloud representation but did not consistently provide good image representation of the room. We scanned stairwells as separate scans so that the data could be quickly recollected if a problem was observed in the scan preview. Stairs were scanned starting at the top and working downward. At the top of each flight of stairs, the scanner was tipped forward to scan the stairs below before proceeding down. When moving down the stairs, the scanner was held vertically. Following these practices helped to reduce occurrences of major misalignments but did not eliminate them entirely.

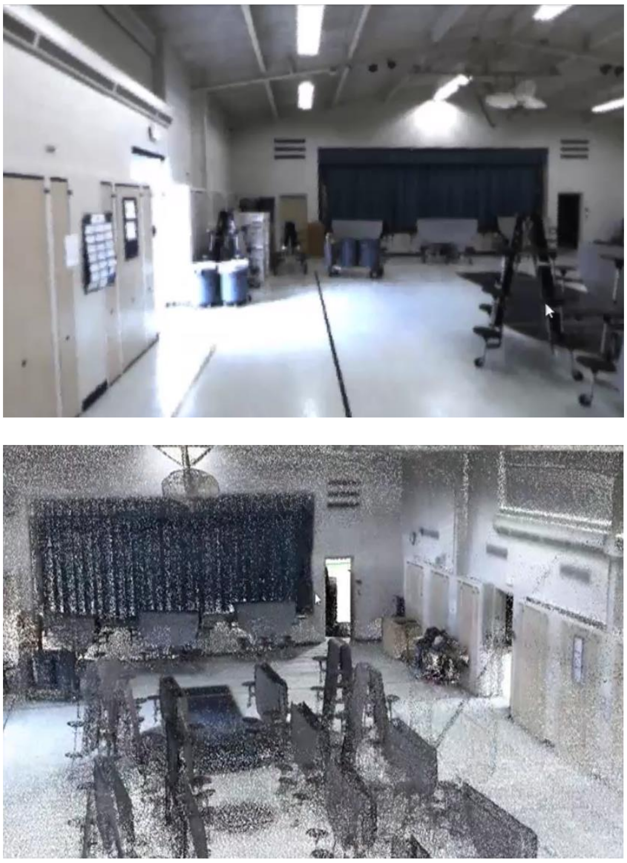

Figure 1: Spherical imagery (top) and colorized point cloud (bottom) viewed in Paracosm's Retrace and Rapidlasso's LasTools software, respectively.

2.3 Software

The PX-80 used Paracosm's Capture v2.0 at the time of data collection. We used ESRI's ArcGIS Desktop v10.7 software to digitize features from the .las point cloud data and quality check results. We used Paracosm's Retrace v1.3 software to map features based on the RGB imagery and a heavily thinned .ply version of the point cloud. Retrace provided an immersive view of the spherical imagery and point cloud. Our procedure required the use of several custom scripts that we developed in Python v2.7. We used the liblas module for Python[3] to process the .las point clouds in scripts.

2.4 Pre-processing for Point Cloud Classification

After data collection, the PX-80 processed the data using its on-board processer to "reconstruct" RGB colorized point clouds and images tied to the point cloud's local reference system. The reconstruction process was automatic once it was initiated. The point clouds downloaded from the scanner were colorized and ready for classification.

We pre-processed the point cloud data output from the PX-80 to improve the efficiency of mapping with the point cloud data. The full resolution point clouds for a typical 20-minute scan contained well over 100 M points and could not be rendered efficiently in ArcGIS even when displayed with the lowest

[3] https://liblas.org/

resolution options available. To improve rendering time and mapping efficiency, we used custom Python scripts to create low- and medium-resolutions version by sampling 1 point in 500 and 100, respectively. In addition to improved rendering, thinning the point cloud had the added benefit of making walls stand out more prominently because it greatly reduced the point density of ceilings and floors but had little noticeable effect on the point densities for walls when viewed from directly overhead. The low-resolution point cloud (i.e. 1 point in 500) was adequate for mapping walls, doors, and windows whereas the medium-resolution point cloud was typically adequate for mapping all other features-of-interest (Figure 2).

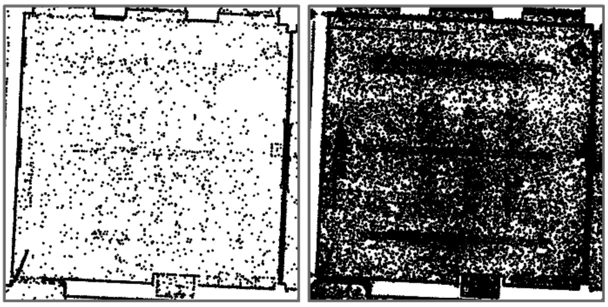

Figure 2: Point cloud thinned to 1 point in 500 (left) of original density (right) emphasizes walls, doors, windows while greatly improving rendering speed.

2.5 Point Cloud Classification

For the classification process, we aimed to map 29 types of features that were relevant to public safety including walls, doors, windows, fire detection and suppression system components, utility shut-offs, first aid, and hazardous materials (Table 1). Large features (i.e. walls, doors, and windows) could typically be digitized easily from the point cloud using ArcGIS. Smaller features (i.e. smoke detectors, fire alarms, sprinkler heads, etc.) were most efficiently mapped using the Retrace software.

Feature types	Mapping method
Wall, railing	Digitize as lines in ArcGIS
Exit door, non-exit door, fire door, window, high window (i.e. >1.5m above ground), elevator, roof access	Digitize as lines, primarily in ArcGIS; digitize in Retrace only if imagery needed for identification.
Stairs	Digitize as polygons in ArcGIS
Smoke detector, fire alarm, fire alarm pull station	Digitize as points in Retrace; correct positions in ArcGIS.
Sprinkler heads, HVAC vent	Digitize as points in Retrace; no correction possible
Fire extinguisher, standpipe, sprinkler riser, electrical panel, AED, first aid equipment, hazardous materials, large furniture, fire alarm control panel, exit sign	Digitize as diagonal lines in Retrace from lower corner to opposite upper corner; correct positions in ArcGIS.

Table 1: Target feature types and mapping methods.

We manually interpreted and digitized walls, doors, and windows as line features in ArcGIS and used Retrace to view the imagery to assist interpretation where features could not be clearly identified. On-screen digitizing in ArcGIS was done in a 2D view from an overhead perspective. The walls were digitized as closed loops by snapping the loop start and end points, whenever possible. Doors and windows were mapped as lines spanning the full opening of the door or window and snapped to the walls. The types for digitized features were recorded in the attribute table of the shapefile. The elevations of the bottom and tops of doors and windows were measured using ArcGIS's LiDAR Profile Tool. The minimum and maximum elevations were recorded for each feature in the attribute table. Doors and windows had standard heights and elevations for a given scan so it was only necessary to measure elevations for a few representative features. Representative features were chosen at opposite ends of the scan in case there was any significant vertical drift in point cloud elevations. If no significant drift was observed (i.e. < 30 cm), the elevations for representative features were copied to the attributes for all features of the same type. Elevations did not need to be measured for walls because they were assumed to extend from floor to ceiling.

We used Retrace to map small objects and other features that could not be easily located and identified with the point cloud alone (Figure 3). Retrace's Measure Tool uses the point cloud to determine the 3D coordinates of the locations clicked with the cursor. These coordinates can be labelled and exported as an .obj file, read in Python, and converted into a GIS vector format. The tool allowed us to identify the locations of features-of-interest using an immersive view of the imagery and obtain relatively accurate coordinates of the features based on the point cloud. We used the Measure Tool to obtain point coordinates corresponding to the centers of smoke detectors, fire alarms, fire alarm pull stations, sprinkler heads, heating and air conditioning (HVAC) vents. Exit signs, windows, fire extinguishers, standpipes, first aid items, utility shut-offs, and furniture were mapped by drawing a line diagonally from a lower corner to the opposite upper corner. The line represented the horizontal span and the upper and lower elevations of the feature. Doors were mapped by drawing a line across the top of the door jamb which was always visible in the imagery. The line represented the horizontal span and the uppermost elevation of the door; the lowest elevation was assumed to be at floor level. We only mapped windows and doors in Retrace if they were difficult to identify from the point cloud in ArcGIS. This included windows that were covered by closed blinds and doors that were closed and flush to the wall. We developed a script to convert the sets of coordinates, exported from Retrace to an .obj file, to either point or line shapefiles depending on how the feature was represented by the Measure Tool. The features were checked in ArcGIS to ensure accurate horizontal and vertical placement.

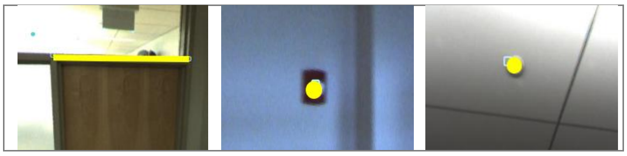

Figure 3: Representation of example feature types as lines or points (yellow) in Retrace including door (left), fire alarm (middle), and smoke detector (right).

We developed another script to create polygons based on the point and line features obtained through digitization in ArcGIS or Retrace (Figure 4). The polygons represented the 3D spaces occupied by each feature. For point features, 3D cylindrical buffers were created based on our estimates of the horizontal and vertical dimensions of each type of feature. The vertical dimension of the buffer was represented by adding a minimum and maximum elevation to the attributes for each feature. The vertical component of buffers of ceiling-mounted features were exaggerated (e.g. +/- 40 cm) to account for minor elevation drift across the scan. A given scan never included multiple floors, except in the case of stairwells, so there was little concern of the vertical buffer encompassing points not associated with the feature. For line features, the script snapped the features to the nearest walls to ensure they were parallel to their associated wall. Then rectangular buffers were created around each line out to a perpendicular distance that we estimated to correspond to the depth of each type of feature. For line features (i.e. doors, windows, etc.), the vertical extents were represented by the minimum and maximum elevations measured with ArcGIS's Profile Tool and recorded in the attribute table. We avoided relying on elevations obtained from Retrace as they often had errors >30 cm which were large enough to cause failures to classify corresponding points in the point cloud. The polygons were checked in ArcGIS to ensure accurate placements.

We used the polygons to classify points in the point cloud that were contained within the 3D space represented by each polygon. The classification was done using a script that we developed which created a new .las file with the updated point classification. Each point in the point cloud was classified as the feature type associated with the polygon in which it was contained. To reduce the size of the classified .las file while preserving features-of-interest, we selectively thinned the point cloud based on the physical dimensions associated with the different types of features. Point densities for large features, such as walls, were thinned heavily (e.g. 1 point in 50) whereas point densities for small features like sprinkler heads were not thinned at all.

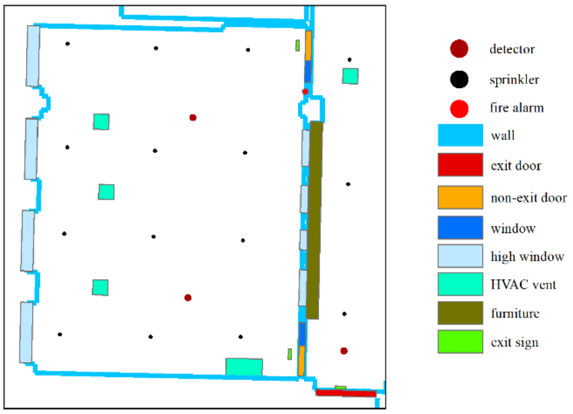

Figure 4: 3D polygons representing features-of-interest.

RESULTS AND DISCUSSION

We were able to successfully map features-of-interest for most spaces in the study buildings (Figure 5). However, we were unable to consistently map small ceiling-mounted features (e.g. sprinklers, smoke detectors) in gymnasiums and other areas with high ceilings or excessive ceiling clutter. Our strategy of scanning small rooms from the doorway occasionally prevented us from using Retrace to map features in those rooms. Retrace only provides image viewpoints for locations directly on the track of the scanner so the available viewpoints often did not provide useful views for rooms that had not been entered. Thus, we usually could not map sprinkler risers or electrical panels which tended to be in small or cluttered rooms for which entry with the scanner was avoided. This problem did not prevent us from mapping large features (i.e. walls, windows, doors).

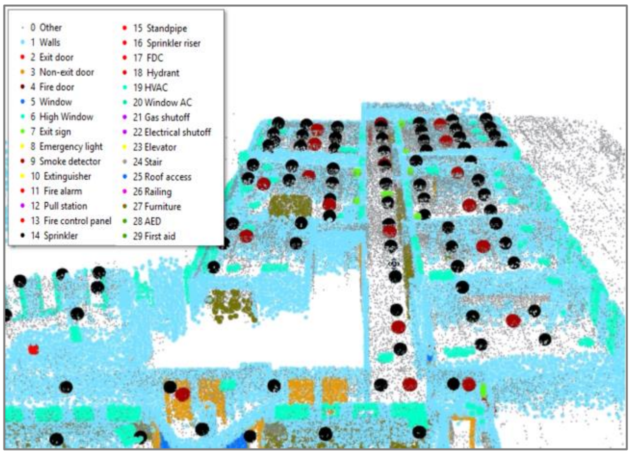

Figure 5: Example of classified point cloud.

Data collection was done by our fire department collaborators who had no related technical experience and minimal training. The firefighters were able to independently operate the PX-80 after only 1-2 hours of supervised training. Scanning a large multi-story building of 14,000 m^2 took an estimated 3-5 hours depending on the characteristics (e.g. number of rooms and stairs) of the building.

The data generated for a 14,000 m^2 building required roughly 200 GB of storage space. Point densities were approximately 1 pt/cm^3 for most areas which were collected at a moderate walking pace of roughly 2.5 kph. These densities represented objects as small as fire alarms and smoke detectors with enough detail to make visual detection possible from the point cloud when viewed at very large scales. Sprinkler heads were impossible to see in the point cloud especially when they were recessed into the ceiling but they were easily visible in the imagery.

We explored development of algorithms for automatically extracting smoke detectors based on their shape and 3D structure. Although we were able to reliably detect smoke detectors in our test dataset, we abandoned the effort because of time constraints and difficulties in differentiating smoke detectors from the many other features (e.g. cameras) with a similar shape, size, and color. Automated extraction of wall- and ceiling-mounted features is certainly possible but automated identification of those features is likely to be a major challenge especially considering the structural and spectral variability of these features across different buildings.

Retrace was a very useful tool for mapping small features including sprinklers, smoke detectors, and fire alarms which were impractical to map directly from the point cloud. The locations exported from Retrace were usually within 0.5 m of the corresponding feature in the point cloud both in terms of horizontal and vertical distances. Vendors of handheld and other types of LiDAR scanners likely provide software that is equivalent to Retrace and may be able to provide a similar function in terms of mapping features.

We estimate that it takes 20-40 hours to complete the point cloud classification of a 14,000 m^2 building when mapping all the target features (Table 1). This time is reduced by about 30-50% if only walls, doors, and windows are mapped because the Retrace phase of the process can be eliminated. Although the process requires considerable manual effort, it can be performed by personnel with relatively little experience in GIS and it likely provides higher quality results than can be achieved with automated methods. The scripts that we developed were key tools for the classification process and will be made available along with the data on our project website (https://www.edc.uri.edu/indoor-mapping-project).

CONCLUSIONS

This study developed an efficient manual procedure for classifying features-of-interest in indoor LiDAR point clouds. The procedure used ESRI's ArcGIS and Paracosm's Retrace software to digitize features using either the point cloud and image data, respectively. We successfully mapped features ranging from doors and windows to sprinkler heads and fire alarms. The data collection process required little technical skill and with the aid of custom scripts we were able to classify the point clouds efficiently. We estimate that collecting data and classifying the point cloud for a 14,000 m^2 building took 20-40 hours depending on building characteristics. The script tools and products developed in this research will be made available for download at https://www.edc.uri.edu/indoor-mapping-project.

ACKNOWLEDGEMENTS

Funding for this project was provided by the National Institute of Standards and Technology (grant 70NANB18H250). We are grateful for the support of the Enfield Fire Department, especially Paul Januszewski, Mathew Mucci, and Michael Carifa. We also thank James Hurd, Patrik Sokolowski, and Christopher Ramm for their work in classifying the point clouds to create the products of this project.

REFERENCES

Castellanos, J. A., Martinez, J. M., Neira, J. and Tardos, J. D., "Simultaneous map building and localization for mobile robots: a multisensor fusion approach," in Proceedings. 1998 IEEE Int. Conf. Robot. Autom. (Cat. No.98CH36146), 1244 –1249.

Diosi, A. and Kleeman, L., "Laser scan matching in polar coordinates with application to SLAM," in 2005 IEEE/RSJ Int. Conf. Intell. Robot. Syst. IROS, 1439 –1444 (2005).

Droeschel, D. and Behnke, S., "Efficient continuous-time SLAM for 3D lidar-based online mapping," in Proc. - IEEE Int. Conf. Robot. Autom., 5000 –5007 (2018).

Kunhoth, J., A. Karkar, S. Al-Maadeed, and A. Al-Ali. 2020. Indoor positioning and wayfinding systems: a survey. *Hum. Cent. Comput. Inf. Sci*. 10:18. 41p.

Paracosm. 2018. PX-80 specifications. Occipital Inc. 5p. Accessed at https://intec.com.my/paracosm-px-80/ (April 2021).

Thrun, S., Montemerlo, M., Koller, D., Wegbreit, B., Nieto, J. and Nebot, E., "Fastslam: An efficient solution to the simultaneous localization and mapping problem with unknown data association," J. Mach. Learn. Res., 4 (3), 380 –407 (2004).