2023 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
AUTONOMY, ARTIFICIAL INTELLIGENCE, ROBOTICS (AAIR) TECHNICAL SESSION
AUGUST 15-17, 2023 - NOVI, MICHIGAN

A SEMANTICALLY CLASSIFIED GEO-SPATIAL 3D OCTREE VOXEL BASED SYSTEM FOR GEOTECHNICAL SITE CHARACTERIZATION

Matthew E. Richards[1], Kevin F. Murphy[2], Israel Lopez Toledo[2], Ahmet Soylemezoglu, PhD[2]

[1]Information Technology Laboratory, Engineer Research and Development Center, Vicksburg, MS
[2]Construction Engineering Research Laboratory, Engineer Research and Development Center, Champaign, IL

ABSTRACT

Geotechnical site characterization is the process of collecting geophysical and geospatial characteristics about the surface and subsurface to create a 3-dimensional (3D) model. Current Robot Operating System (ROS) world models are designed primarily for navigation in unknown environments; however, they do not store the geotechnical characteristics requisite for environmental assessment, archaeology, construction engineering, or disaster response. The automotive industry is researching High Definition (HD) Maps, which contain more information and are currently being used by autonomous vehicles for ground truth localization, but they are static and primarily used for navigation in highly regulated infrastructure. Modern site characterization and HD mapping methods involve survey engineers working on-site followed by lengthy post processing. This research addresses the shortcomings for current world models and site characterization by introducing Site Model Geospatial System (SMGS). This site model leverages an octree spatial data model to store heterogeneous geotechnical information in a Volumetric Pixel (Voxel) grid, which allows for more efficient algorithms in data analysis and fusion. SMGS provides a real-time, dynamically updated, 3D data model with semantically derived costmaps for navigation and Engineer operations, ground truth localization without GPS, and produces standard Geographic Information System (GIS) maps.

Citation: M. Richards, K. Murphy, I. Lopez Toledo, A. Soylemezoglu, ”A Semantically Classified Geo-spatial 3D Octree Voxel Based System for Geotechnical Site Characterization,” In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 15-17, 2023.

1 INTRODUCTION

1.1 Problem Space

In many geotechnical or construction operations, a site needs to be investigated to make actionable decisions. When operating construction equipment autonomously or at a stand off distance in a dynamic environment, knowledge about the site becomes less known as the site is morphed. In some situations, it may be unsafe for personnel to be present to investigate a site. A conservative approach would be to operate autonomous or remotely controlled vehicles to perform an investigation. A robotic investigation can also improve the speed, efficacy, and efficiency of a sensor scan. Current Robot Operating System (ROS) world models [5] and HD maps [1] can be limited in scope, static, stateless or limited in state, require post processing, and location relative [2] without ground truth.

Site characterization is a process for collecting geophysical and geospatial characteristics about the surface and subsurface of an area of interest under investigation [3]. The information collected is compiled into a 3D model for storage and visualization. Collecting this information can be a challenge when the investigation site is hazardous, for example military or disaster sites. Additionally, each sensor modality for collecting and storing data can be diverse and use different data formats. This usually requires manual or specialized software to compile the data into a site characterization report or visualization product.

1.2 System Design Goals/Requirements

The goal is to start with as much a priori information available about a site from geospatial products or aerial surveys and characterize a site with real-time data collection. The system collects and processes sensor data to produce semantic information that is used to update a priori information stored in a geospatial database. Finally, the system will produce real-time knowledge that the Engineer, robotic navigation system, autonomous system, or remote operator can use to make actionable strategic and operational decisions.

1.3 System Overview

In architecture and construction, a *site model* sometimes refers to a physical three dimensional contour model of an engineering project. The site model described here is a georeferenced and geotechnical world model with the information requisite for conducting Engineer operations in unregulated environments. The data is stored in a modern geospatial, NoSQL, non-relational database. Because these datasets are extremely large, they are considered big data and non-relational databases have been analyzed to perform better than relational database with big data [8]. This data can then be retrieved with geospatial queries for generating maps, visualization, navigation, or end effector control.

The Site Model Geospatial System (SMGS) introduced here is a ROS package with nodes, libraries, and utilities for creating and updating the site model geospatial database. The package is intended to be included in a ROS stack on a robotic vehicle equipped with the sensor modalities necessary to conduct a geotechnical investigation. The data collected from the sensors is fused and stored in the SM database. The sensor data is processed real-time to produce geospatial information products that can be used to navigate, control Engineer equipment, or assess a site under investigation.

1.4 A Priori Information

Earth's entire surface has been mapped through satellite and aerial mapping; however, the quality, resolution, accuracy, and types of available geospatial data are inconsistent. The purpose of the SMGS database is to combine various inconsistent data to create a 3-Dimensional (3D) geospatial information model that is uniform and homogeneous with occupied space. This is accomplished by

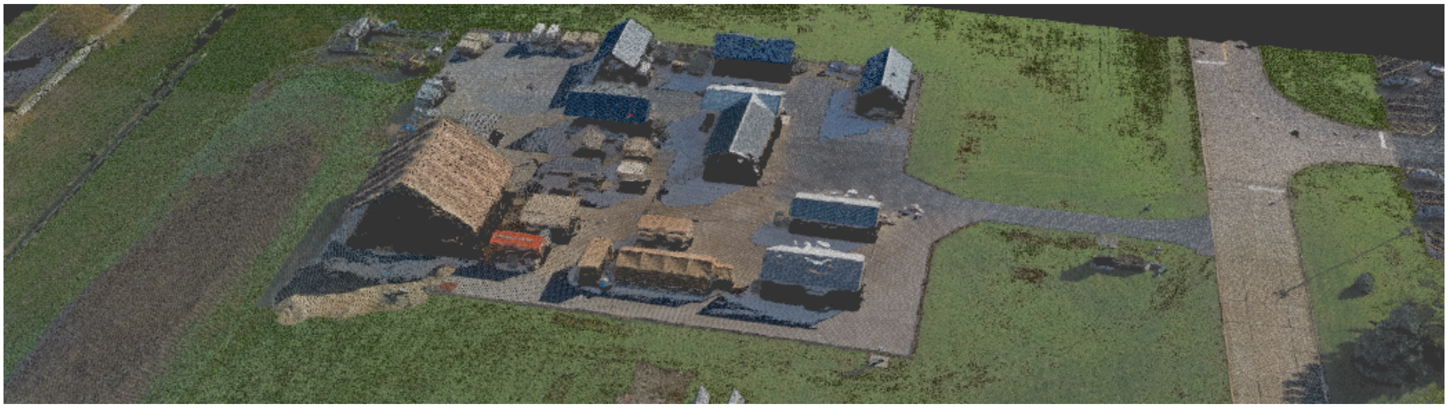

Figure 1: Voxel Site Model Color Visualization in ROS RViz.

initially populating the SMGS geospatial database with a priori data and information about the engineering site. This information can be imported from raster images, point clouds, or vector files from GIS, LiDAR, RaDAR, photogrametry, or other geospatial sources. The a priori information is converted to a point cloud, a set of points identified by cartisian coordinates in 3D space, that are inserted into the geospatial database as a GeoJSON point object with the additional attributes, such as color, soil type, etc.. Figure 1 is an example of a priori information loaded into the site model database as a Digital Surface Model (DSM) raster acquired from an Unmanned Aircraft System (UAS) and displayed in ROS RViz as a marker array topic.

1.5 Operational Overview

A robotic vehicle with sensors and ROS computers navigates a site to be characterized for an engineering operation. The SMGS collects real-time data from the sensors, proccesses the data to create useful information, and stores this information in a ROS Pointcloud2 datatype. Information can include color, material type, object classification, soil moisture, or any other information useful to the Engineer. The point clouds are imported real-time and localized to the stored a priori voxel grid data. Existing voxels are updated or new voxels are created.

As data is being collected, the SMGS publishes point clouds, voxel maps, georeferenced rasters, and navigation maps from the combined a priori and real-time information. The voxel maps can be used by the operators or Engineer for actionable decisions. The rasters are published with a tile server, such as GeoServer. Finally, the navigation maps are used by guidance and control to autonomously operate the equipment.

2 RELATED WORKS

The primary requirements for the site characterization software are the ability to input a priori geospatial information, localize sensor data to a priori information, collect multiple modalities of geotechnical data, process the geotechnical data into information, store information in a high performance spatial database, and produce geospatial or control products. There were many available ROS packages that conducted localization and mapping at the time of this research project; however, the main mapping products that were extensively evaluated as a solution to the problem space were RTABMap [9], Robotic Technology Kernel (RTK) World Model, and Octomap [10]. None of these software packages met the minimum requirements for site characterization and modification to add the required features would necessitate complete rewrites from the core structure.

3 CONCEPTS

3.1 Data, Information, Knowledge, Wisdom

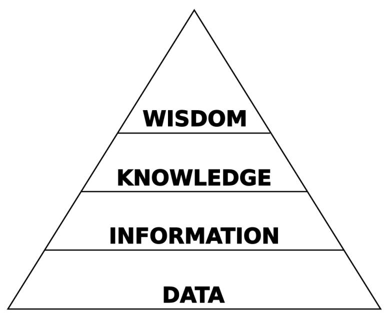

Figure 2: Ackoff's DIKW Hierarchy

The overarching theme with the SMGS is based on Ackoff's data-information-knowledge-wisdom (DIKW) hierarchy [7], as illustrated in figure 2. The idea is that raw data is collected from the environment and processed to produce information. Information is collected and presented to form knowledge. Knowledge is actionable, but only until the environment changes; therefore, knowledge is time-sensitive. Wisdom is a product of time and experience, which is represented by the spatial database in the SMGS. The SMGS hierarchy is illustrated in figure 3. The database can be analyzed later to understand and improve on the process.

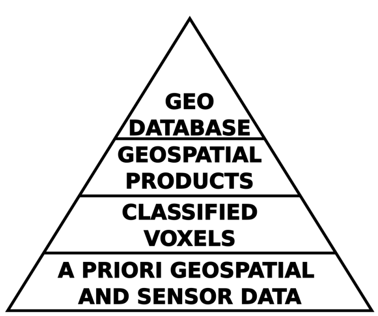

Figure 3: SMGS DIKW Hierarchy

3.2 Voxels

Volumetric Pixels (Voxels) are volumes of space, typically cubes, with uniform distances between centroids. The voxel, Figure 4, represents all points within its boundaries and are homoginized with the information associated with each point. Voxels can be represented as a point cloud by using a point within the bounds to represent the voxel, in this case, the centroid. By combining all points within a volume of space into a voxel, the point cloud is reduced in size and becomes more uniform in density. The reduced point cloud requires less processing and is visualized as solids without meshing, which can be difficult to accomplish accurately on point clouds.

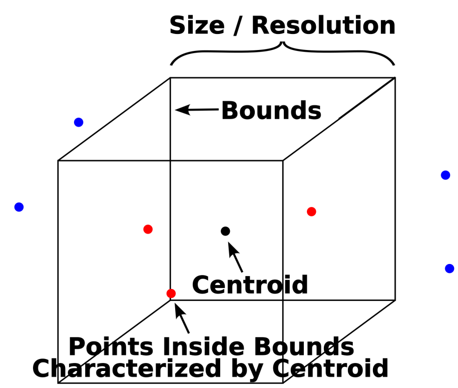

Figure 4: Voxel Representation of a Point Cloud

3.2.1 Minecraft

The initial inspiration for this site model came from a popular video game *Minecraft*, a Microsoft product, released in 2011 [6]. The game is entirely voxel based with semantically classified "blocks" that define the world. The world can be morphed by adding and removing voxel blocks. The voxels represent recognizable soil types, rock, minerals, vegetation, water, ice, snow, and other features using colors and textures. A key feature is that large world models are easily rendered and played on low performance computers.

Figure 5: Minecraft
[6]

3.2.2 Practical Applications of 3D Voxel Models

A geological survey of the Netherlands was used to create 3D geological voxel models to help visualize and understand subterranean features [15]. This was the initial idea with the SMGS, to create a 3D model of surface and subsurface features.

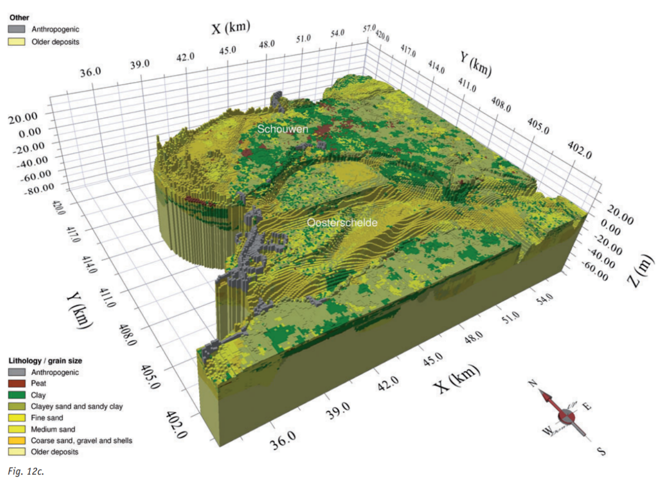

Figure 6: Voxel 3D Lithology Example of Zeeland Netherlands [15]

The idea of using voxels to visualize and store information about subterranean features was also explored in archaeology [14]. A voxel-based depiction of all trench IX units in Akroterion, Kythera (Figure 8) is a good example for this application.

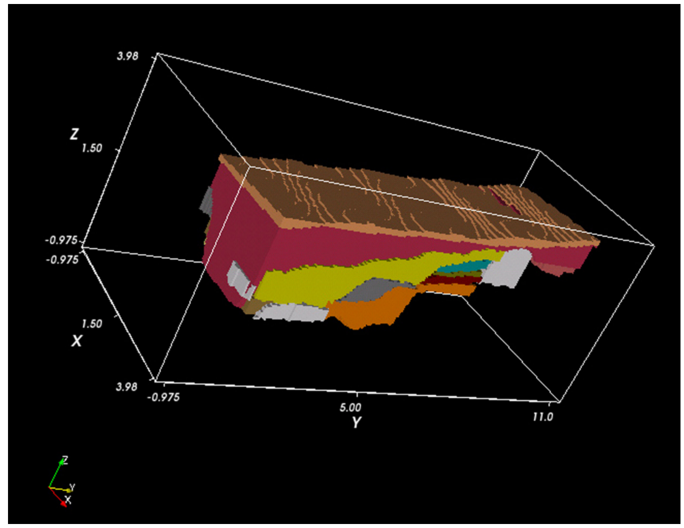

Figure 7: 3D voxel-based depiction of all trench IX units, Akroterion, Kythera, generated in GRASS GIS and visualised in ParaView [14]

3.3 Octree

In computer science, 1-dimensional data can be sorted and searched much faster than an iterative search (Figure 8) using a ”binary search” or ”logarithmic search”. A binary search (Figure 9) is performed on an ordered list by finding the midpoint of the list and comparing the indexed value to the search value. If the indexed value is less than the search value, the upper limit is reduced to the midpoint and the search is repeated. If the indexed value is greater than the search value, the lower limit is increased to the midpoint. The search is repeated until the search value equals the indexed value. If the value is never found, an error is thrown.

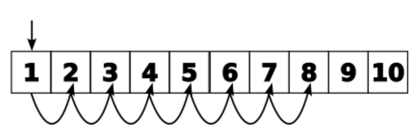

Figure 8: Iterative Search Steps

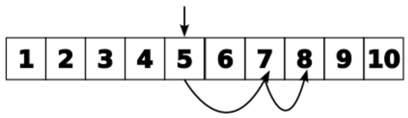

Figure 9: Binary Search Steps

Octrees are a 3-dimensional version of the binary search (Figure 10). The Octree is created using a recursive data structure where each node contains eight child nodes. By populating an octree with occupied space from a point cloud, the points are naturally voxelized. A search function finds the child nodes that bound the point being searched and recurses the tree until the search level is reached or the point is not found.

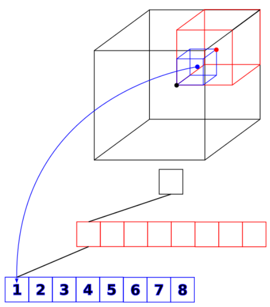

Figure 10: Three Level Octree Structure and Search Example

When a new point is added to or updated in the octree, the following steps are recursed until the octree level is reached:

- Step 1: Compute and update this voxel node with new values
- Step 2: If current node level == destination level then stop recursing
- Step 3: Find the child node octant that bounds the point
- Step 4: Recurse to the new or existing child node octant and start at Step 1

3.4 A Priori Data

When working with geospatial data, there are multiple data formats of a priori information available for nearly every square meter of the Earth. The site model should be preloaded with as much information as available about the engineering site prior to characterization by the vehicles. This includes rasters, vectors, and point clouds to name the three most common root data types. Rasters are two dimensional arrays of data that represent elevation, color, or other types of data for a geographic area of Earth. Vectors represent lines and polygons that define inclusive geographic boundaries or lines of data. Points in point clouds represent a single geographic coordinate with information about that location.

A priori data is initially imported into a non-relational, or sometimes referred to as a NoSQL (Not only SQL) database. NoSQL databases offer dynamic data schemas, faster spatial queries, object oriented data approach with BSON (Binary JavaScript Object Notation), and greater scalability over traditional relational databases [8]. As the data is imported, it is converted into an octree data structure and stored as a binary object in the database. To increase performance, each binary object contains a portion of the octree below a configurable level.

3.5 Point Cloud Registration

Point cloud registration is the process of aligning point clouds with overlapping features or along edges. The SMGS can use multiple methods of point cloud registration. One of the methods SMGS uses for point cloud registration is Iterative Closest Point (ICP). SMGS uses ICP to align incoming point clouds from sensors to the a priori point cloud. The base ICP implementation in the C++ Point Cloud Library (PCL) was tested as a proof of concept for registration. Other methods are currently being tested with improved results. Understanding how ICP works, helps with understanding how the point cloud from a sensor is aligned to the a priori point cloud data. The general concept with ICP is that it finds correspondences between the source point cloud and the target point cloud. A center of mass is computed between the correspondences and then the center of masses are aligned and rotated. This process is repeated (iterated) until a local minima condition is met.

As an example of ICP, Figure 11 illustrates two point clouds and a correspondence distance threshold. The correspondence distance is used to limit the distance for matching points.

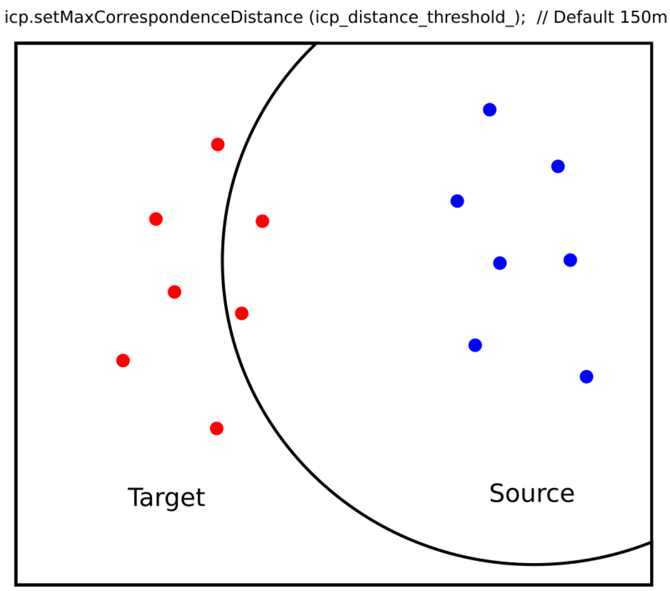

Figure 11: Example Source and Target Point Clouds

The first step is to match the corresponding points that are closest between the target and the source. A center of mass is determined for each group of corresponding points.

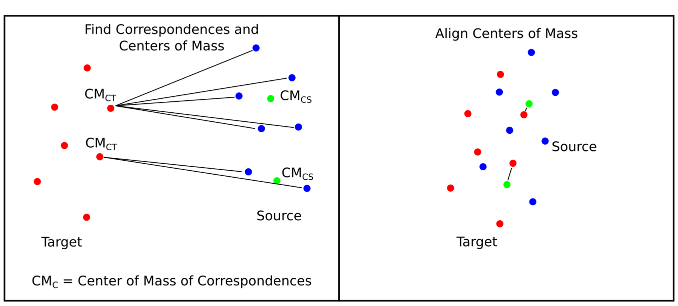

Figure 12: Example Iteration 1

The next step is to align the centers of mass as close as possible.

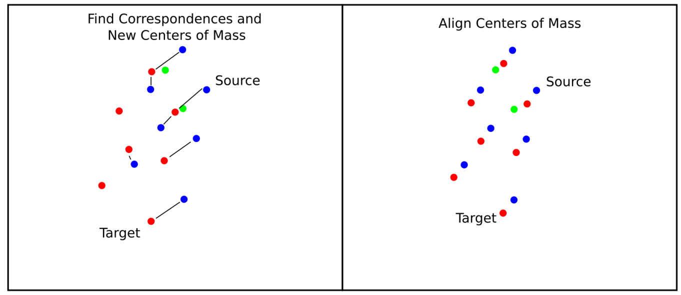

Figure 13: Example Iteration 2

The process is repeated, or iterated, until the criterion for ending the iterations are met. With the base ICP, there are three criterion. The first is the number of iterations, then the transformation epsilon, and finally the euclidean distance. If any of these criterion are met, ICP will stop and produce a final point cloud, a transform between the point clouds, and a fitness score based on the distance between the source and target points.

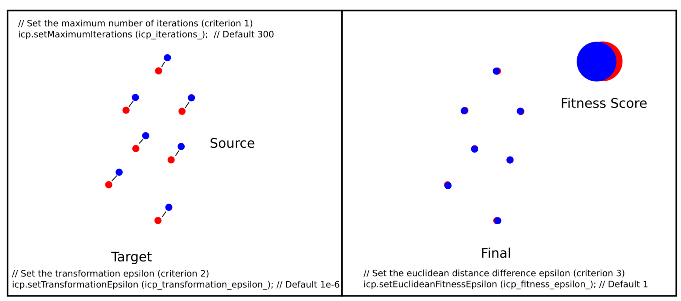

Figure 14: Example Iteration 3

4 SITE MODEL GEOSPATIAL SYSTEM (SMGS)

4.1 Technical Overview

The SMGS is a ROS package containing multiple nodes and utilities written in C++. The nodes can be configured to start with launch files, depending on the functionality required from the system. The nodes are used to collect data from the sensor topics, localize the vehicle to the a priori information, write information to the geospatial database, publish information to ROS topics, write geospatial maps to the file system or tile server, write geotagged images to the file system, broadcast services for interacting with the system, and writing raw data to a pose graph database.

4.2 Data Structure

The data schema is stored in memory in an octree structure. The data schema can be expanded as needed and since the binary schema object is stored directly to the database, it will automatically be updated in the database. The data schema includes information such as color, moisture content, soil classification, and soil California Bearing Ratio (CBR).

4.3 Non Relational Database

Non-relational database systems, commonly called Not only SQL (NoSQL) databases, are designed to manage *big data*. Big data is defined as extremely large and complex data sets. Non-relational databases were designed to be more efficient and have faster read and write times than traditional relational database systems with big data [8]. The size and complexity of the geospatial point clouds with additional attributes in the site model data structure classify it as big data.

Database normalization is a process of creating many data tables with relationships to reduce data redundancy and improve data conformity. Non-relational databases are de-normalized, using very few relationships between data collections. Collection is a term used to reference tables in a non-relational database. By reducing the number of relationships and trusting the software to maintain data conformity, non-relational databases have less processing overhead and fewer disk operations.

4.4 Ground Truth Localization

The SMGS can use Simultaneous Localization and Mapping (SLAM) or other localization methods though the ROS TransForm (TF) system; however, it is intended to be used with the included ground truth localization node that uses a priori GIS data. Most SLAM systems assume no information or map is available to localize from and sets the origin of the map to the starting location. The map is created based on relative locations and many SLAM algorithms, such as GraphSLAM, require returning to a location to perform loop closures [11]. If a loop closure cannot be performed, dead reckoning is used and drift is compounded over time (Figure 15). Additionally, SLAM can struggle to localize in dynamic environments [12]. The output from a SLAM system is typically used to create a world model and only contains navigation information, such as ground slope and obstacle information. SMGS improves upon SLAM by fusing a priori information with sensor data and localizing to ground truth features on a geospatial map.

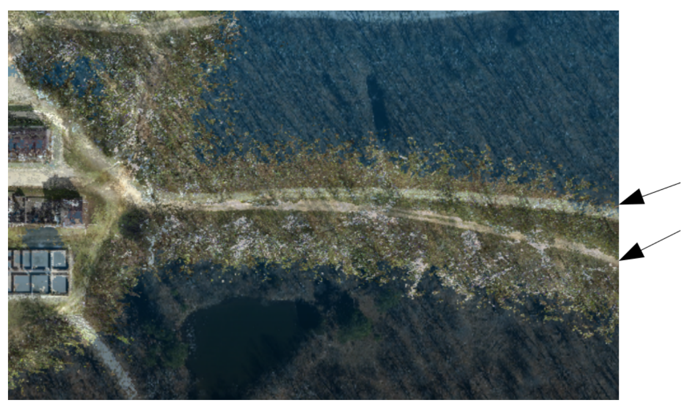

Figure 15: Road Diverges as Error Compounds without Loop Closures in RTABMap using GraphSLAM

Ground truth localization systems expect a priori geospatial map data for localization and subsequently perform corrections when a ground truth is found. Since the a priori information is geolocated, the system can localize to geographic locations. The a priori map is thus updated based on matched locations within the map.

High Definition (HD) maps are a form of ground truth localization. HD maps are created with a resource intensive combination of AI and human-in-the-loop processing. They are used by autonomous vehicles for ground truth localization and navigation in known and mostly static environments. Construction equipment also requires information for navigation, but additionally requires information about the soils, materials, and hazards that are within the environment where the equipment is being operated. HD maps are expensive, labor intensive, and require lengthy time to produce. The SMGS is designed to automate the process of creating something similar to an HD map for localization by rapidly updating and storing 3D geotechnical information in a geospatial database that can be used to create geospatial maps, navigation costmaps, and 3D representations of a construction site.

The SMGS node for ground truth localization is called *Orienteer*. Figures 16 through 18 were captured in ROS RViz from the Orienteer node topics. The red point cloud in Figure 16 is the point cloud from the LiDAR sensor, transformed to the UTM frame through odometry. The blue point cloud in Figure 17 is the point cloud after ICP aligned it to the white a priori point cloud. Using this transform, a new position estimate from the UTM frame is published. Figure 18 illustrates the rotation and translation correction from the ground view, where red is the initial estimate and blue is the final transform. A score produced by the registration algorithm is one variable that is used to compute a covariance matrix. The covariances can be used by an Extended Kalman Filter (EKF) in the localization system to fall back on other localization systems, such as odometry, when poor alignments due to sensor error or poor quality a priori data.

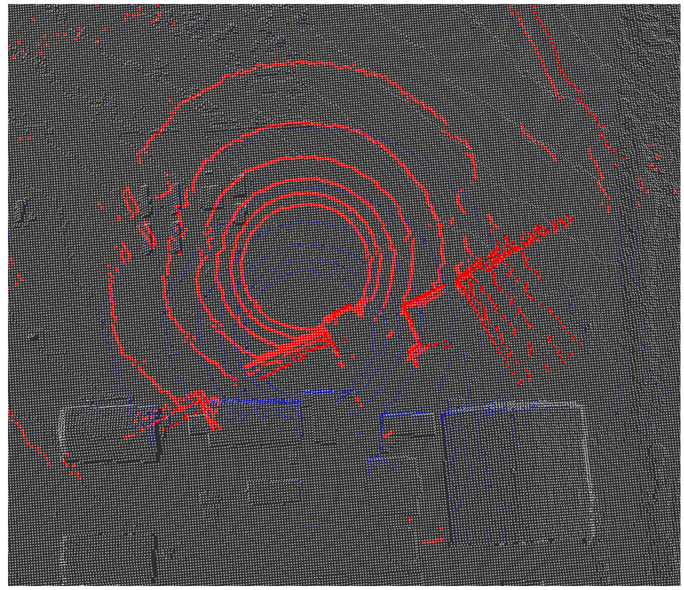

Figure 16: Position Estimate from Robot Localization (Red)

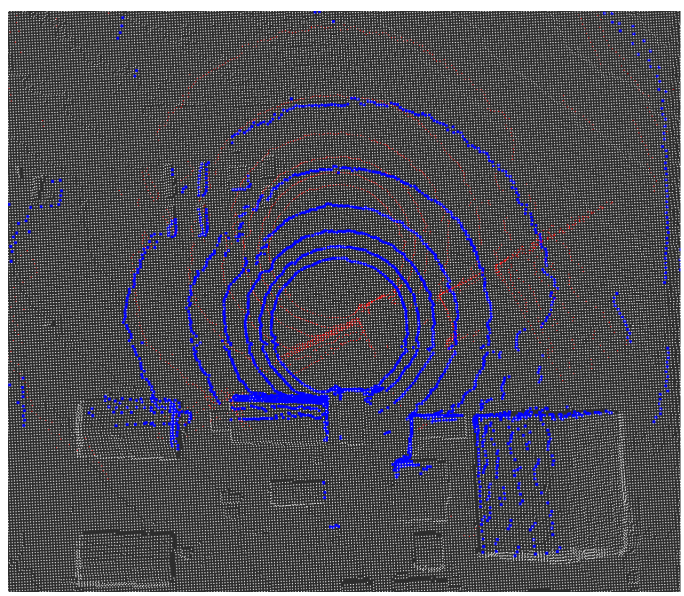

Figure 17: Final Transform from ICP (Blue)

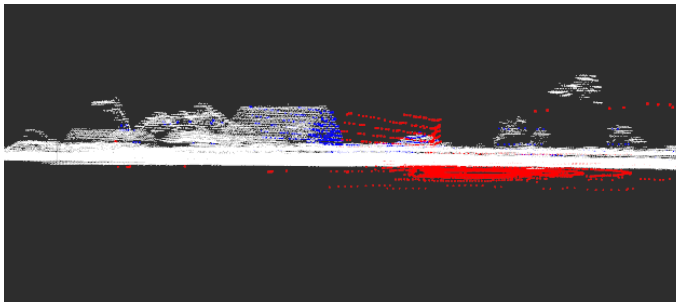

Figure 18: Ground View Showing ICP Transform

4.5 Data Collection

Most of the data is collected from the ROS sensor topics through a cloud fusion node. This node combines the color and classification information from images with the LiDAR point cloud to produce a ROS Pointcloud2 message. This specific ROS cloud fusion package is not a requirement, if using other pointcloud topics that produce a Pointcloud2 message. The Pointcloud2 topic is subscribed to by the SMGS Cartographer node. The SMGS Cartographer node updates the octree data structure in memory, updates the database, and offloads data based on distance as the vehicle moves to reduce memory usage.

SMGS inherently trusts the localization system that generates the transform (TF) tree; however, promising research by the authors is currently being conducted to improve the quality of the data by performing a second point cloud registration prior to inserting the information or discarding information based on a covariance matrix from localization.

4.6 Clearing

Database updates until this point have added new information. In contrast, clearing is the mechanism which removes outdated information from the database.

4.6.1 Voxel Clearing Overview

Previously added information in the site model database becomes outdated for several disparate reasons, including changing environment, inconsistent a-priori data, octree resolution, etc. Dynamic environments, such as other moving entities operating around the system using SMGS, will add voxels each time some perceived volume is occupied. An object moving through the robotic view will add occupied voxels through the object's entire path, creating a wall in the site model. Operations such as earth-moving, construction, demolition, or anything else that causes a difference between previously built environment models and current reality introduce spatial error through outdated information.

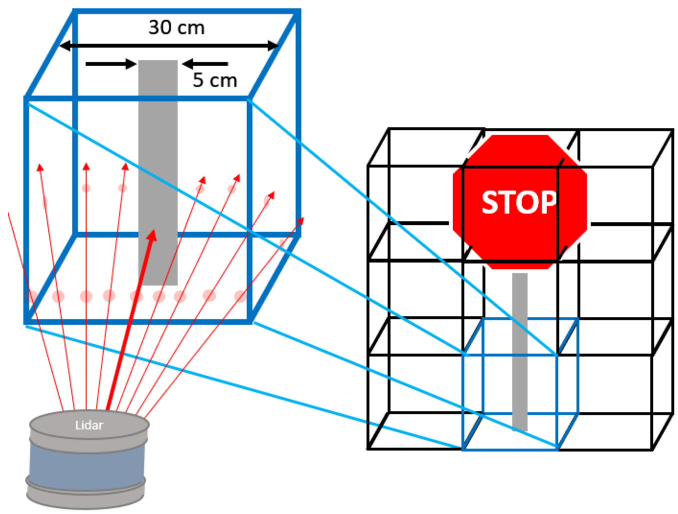

Figure 19: Many Lidar beams missing pole in an occupied voxel

Differences in a-priori data and reality can be viewed similarly to dynamic environments except on a larger temporal scale. An example of this and active research topic [16] is characterizing the same locations in different seasons and weather.

4.6.2 Challenges Identifying Incorrect Voxels

The challenge is identifying when information should be updated or the voxel removed. The obvious but naive approach is removing any voxel that is seen as unoccupied, such as a lidar passing a ray through an occupied voxel. The problem is a voxel's volume may be partially occupied by or contain objects smaller than a leaf level voxel's resolution, such as a pole (Figure 19). In these cases, many lidar beams will likely miss the obstacle but others may hit it, resulting in constantly shifting occupancy, wasting computational resources and undesirable jumps in cost-maps and planners. Additional challenges include sensor self occlusions, measurement range, and non-deterministic ray directions.

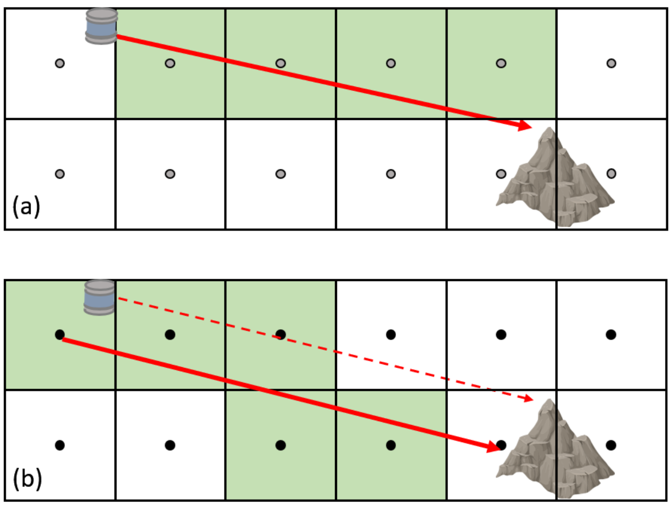

Figure 20: Lidar beam detecting an obstacle and highlighting the voxels to be cleared. (a) shows tracing with exact positions from the point cloud, while (b) shows the trace after voxelization, showing differences in what voxels being traversed with the same ray

Voxalization of a point cloud, commonly employed to significantly reduce computation, introduces problems associated with angle of incidence. As seen in Figure 20, a ray that is traced between exact points will enter different voxels than those of the ray traced between voxelized points. The less steep the angle, the greater these issues become.

4.6.3 Probabilistic Certainty

The Spacio-Temporal Voxel Layer [17] is a method that avoids ray tracing clearing challenges by decaying voxels with time and resetting the decay when a occupancy is detected. While effective for local maps, the temporal disposal of all information and high overhead cost of constant observation of each voxel make it unsuited for map building and site modeling.

This research's approach combines this decaying occupancy with probabilistic occupancy grids [18]. Here, occupancy is implemented probabilistically; however, voxel decay is initiated by a lidar ray traveling through an occupied voxel without hitting anything. When pierced, the probability of occupancy is reduced. This removes the overhead of updating every voxel temporally and only access the voxels which are traversed. The site model's octree data structure facilitates rapid and efficient and ray tracing between non-voxelized spatial points.

4.7 ROS Topics

The SMGS package has multiple nodes for publishing localization, cost maps, point clouds, images, and marker arrays that can be viewed in RVIZ. Figure 21 is an example of an updated site model viewed in RViz as a marker array. The blue color in the trees are LiDAR returns from the leaves and colored from the sky.

4.7.1 Geospatial Maps

The SMGS package has multiple nodes for writing geospatial raster maps to the file system. These maps can then be consumed by GIS software for visualization. There is also a CLI utility for exporting the maps in different formats, such as LAS, LAZ, mesh, and raster files. The maps can contain information from the voxels, such as color, classification, soil moisture, etc.. Figures 29 and 30 are examples from the SMGS.

4.7.2 Geotagged Images

The SMGS system can geotag and combine images from image topics. The images can be stored

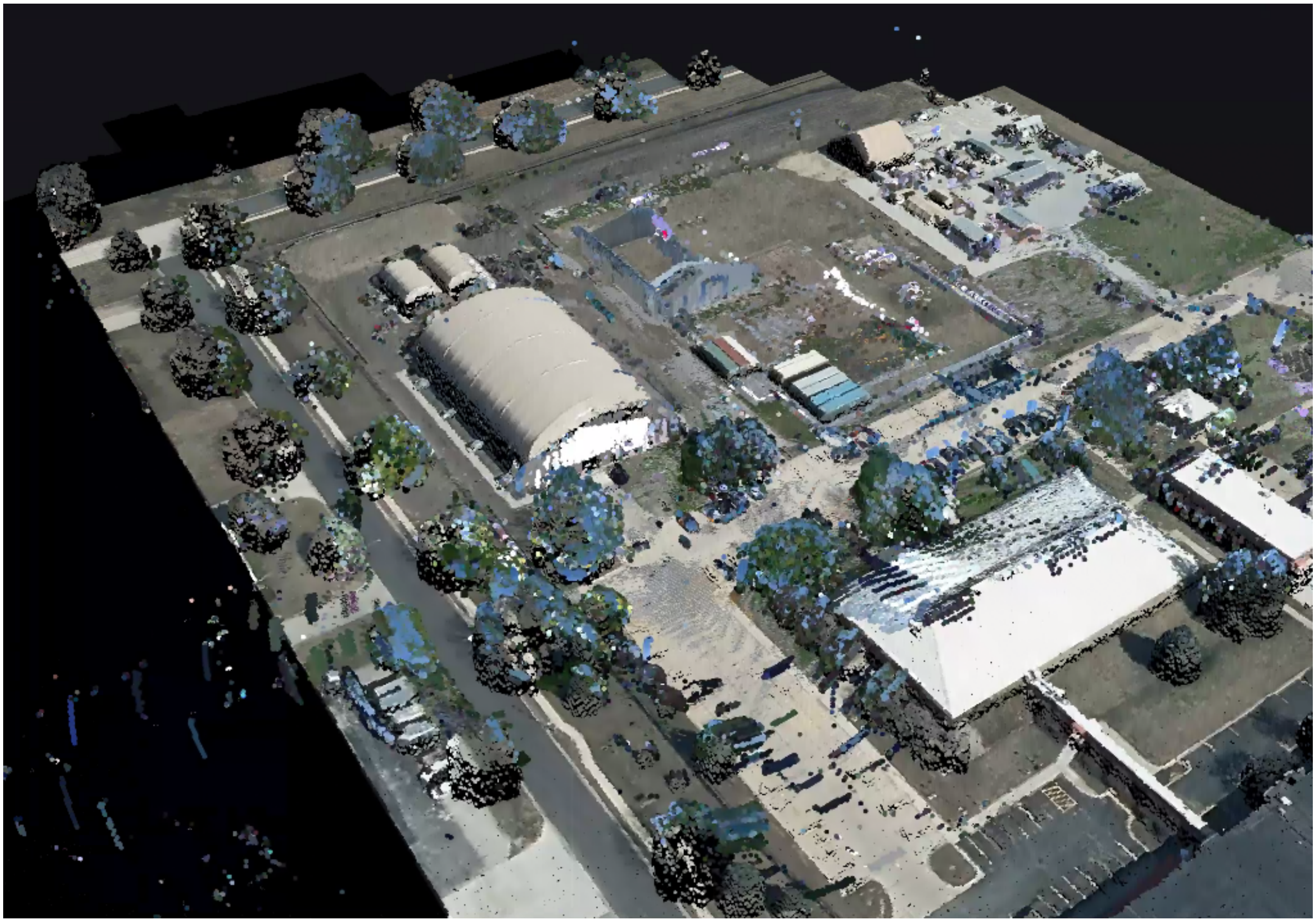

Figure 21: Updated Site Model Visualization in ROS RViz.

directly on the file system or in the database as a Binary Long Object (BLOB). Figures 22 and 29 are examples of a geotagged image file and displaying geotagged images in a GIS.

4.7.3 Map Tile Service

When running the geospatial map nodes, the raster images can be written to a map tile service, such as GeoServer. These images will be updated real-time and can be opened in GIS software such as ATAK, ArcGIS, QGIS or GRASS.

4.8 User Experience

The output map products described in the previous section are served to the user locally or over the network through various interfaces. The robotic platforms host the Site Model Database and Interface servers on their network. These allow any user interface to request map products from the platform or query the mapping information straight from the database. This means that users can utilize the output map products in three different ways. The user can view the maps directly on the platform (i.e. utilizing a remote desktop to the platform's computer), query the database and render the map

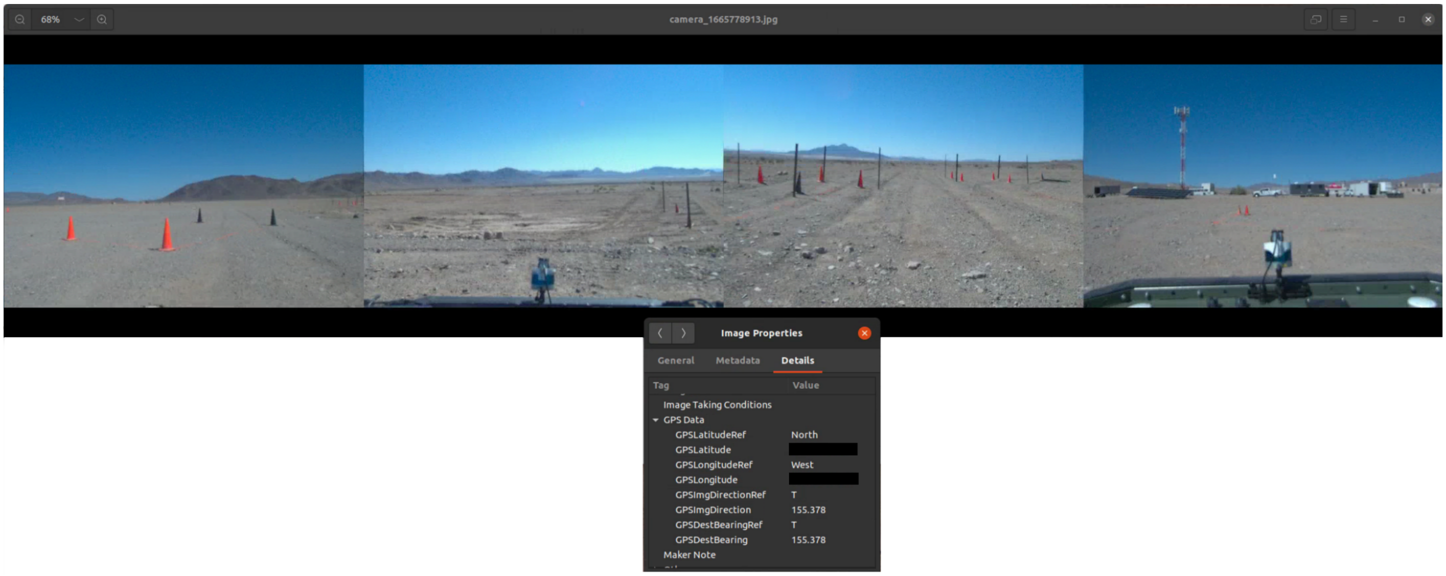

Figure 22: Geotagged and Merged Images from Four Cameras

in any desired interface, or request the finalized map products through the platforms' API servers and display it in any desired interface.

4.8.1 Interface Servers

The robotic platforms open up to outside devices through two Interface servers that are hosted on the platforms' network. These Interfaces are Geoserver and our own Platform API. GeoServer is an open source map tile server for sharing geospatial data. Through Geoserver users can request 2D map products utilizing GIS supported systems such as ATAK, ArcGIS and QGIS. The platform generates 2D map products real time and pushes them to Geoserver, to make them available over the network. The Platform API is a modular interface framework that allows any interface to interact with the robotic platform through http calls over the network. This API allows interfaces to execute commands on the robotic platform and retrieve Site Model output products. Thus in general, any interface that has access to the platform's network is able to retrieve Site Model's output products and display them to the user.

4.8.2 User Interfaces

Having the platform open to any device provided the opportunity to explore various technologies for user interfaces. Technologies such as Virtual Reality (VR), mobile devices and desktop workstations were used to develop user interfaces that consume Site Model products. Virtual Reality was utilized to provide an immersive view of the Site Model maps. With this interface users are able to move around Site Model 3D maps and have a closer look on any information they require. This VR application was developed using the Unity game engine; it queries all the necessary information to build maps from the Site Model database, over the network, and then it renders the high fidelity 3D maps to the user. To provide users with Site Model's 2D map products on the go, the mobile application Android Tactical Assault Kit (ATAK) was utilized. ATAK is a mission planner and a military situation awareness app. This application supports GIS maps and thus is able to retrieve Site Model's 2D maps through Geoserver. ATAK is also able to interact with the robotic platform through a plugin developed for this application, depicted in figure 23. This plugin allows users to execute commands on the vehicle and utilizes the Platform API to communicate with the robot. The plugin provides users with mission planning and execution capabilities needed to perform the site characterization. For desktop workstations, the ROS applications RVIZ and Foxglove were used. These open source visualization and debugging tools allow users to view Site Model's 2D and 3D map products straight from the robotic platform's computer. In case the robot is on the field, users can remote desktop and still have access to these tools.

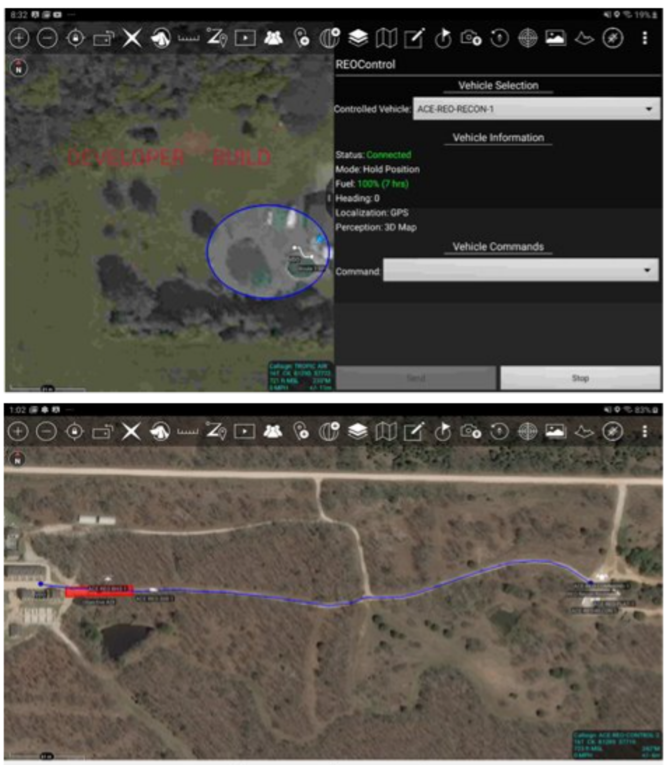

Figure 23: REO Plugin for ATAK

5 SYSTEM VALIDATION

The SMGS was designed to introduce new functionalities, capabilities, and features to modern mapping systems. To observe its performance, the system was operated and tested at different sites. During these experiments the vehicle would roam around an unknown area autonomously or via teleoperation, while performing persistent mapping and generating map products with the SMGS. In some cases Army Engineers operated the vehicles and monitored the map products in order to get feedback from future end-users.

5.1 Platform and Hardware

The SMGS was tested on a variety of hardware platforms from an Argo Atlas Xtreme Terrain Robot (XTR) to a Clearpath Jackal. The main requirements were a ROS Pointcloud2 topic and odometry localization hardware. Sensors tested were Velodyne VLP-16, Ouster OS1-64, Basler Cameras, and ZED Cameras to name a few. Figure 24 is one of the site characterization platforms used with the SMGS. It is an electric Argo XTR J8, operating with four onboard computers, active cooling, two VLP-16 LiDARS, one OS1-64 LiDAR, four Balser RBG cameras, and a ZED Stereo camera. It is optionally equipped with a dynamic cone penetrometer, thermal imaging sensors, and other geophysical sensors.

Figure 24: Argo J8 Site Characterization Robot

5.2 Field Tests

The SMGS has been tested in different environments and locations with a combination of researchers and warfighters. This section will go over the experiences and results observed from experimentation performed in three different environments. These environments were: a lake area, urban, desert, and forest.

5.2.1 MSSPIX

Maneuver Support, Sustainment, and Protection Integration Experiments (MSSPIX), is an annual event where scientists from the U.S. Army's Combat Capabilities Development Command (DEVCOM CBC) assesses emerging technologies. Instead of groups presenting and showing demonstrations of their technologies, MSSPIX applies them to military operations, puts them in the hands of soldiers, and conducts simulation missions using the newly introduced technologies.

The SMGS was part of MSSPIX 2020 and 2021, which took place at Fort Leonard Wood in Missouri. For this assessment, a group of combat engineers were tasked with performing recon in a Military Operation on Urban Terrain (MOUT) and a forest area. They performed site characterization in support of teleoperated construction equipment for obstacle removal and emplacement. The SMGS was the key component in the missions, as it allowed the engineers to have an idea of where they could operate the construction equipment. The results for this event were that engineers were able to map and monitor 1km of forest trail through a mix of autonomy and teleoperation. Utilizing the SMGS's map products and the platform's sensors, the engineers were able to understand the environment and operate the construction equipment with no direct line of sight. The engineers were trained on the SMGS, the ATAK plugin and the robotic platform in just a few days and complete the mission without intervention from the researchers and developers. Figures 25 and 26 are examples from RViz of the site characterization without GNSS at MSSPIX. The blue voxels in the trees are updated color from the sky passing through the leaves and the red path is the path of the robot on the road. Figure 27 is a view of the robot model in RViz at the same location. This feat highlights the effectiveness, practicality and usability of the overall system.

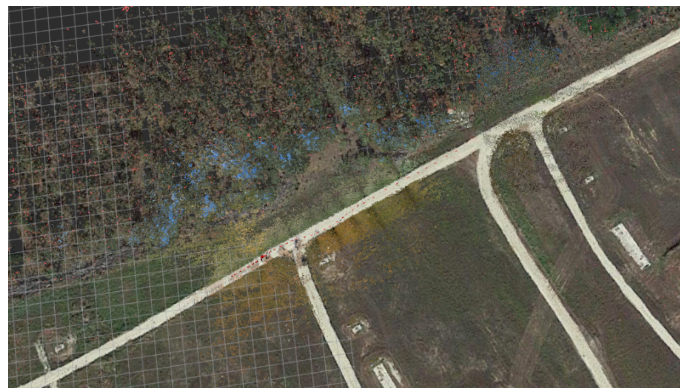

Figure 25: Updated 3D Map from SMGS

Figure 26: Updated 3D Map with Path from SMGS

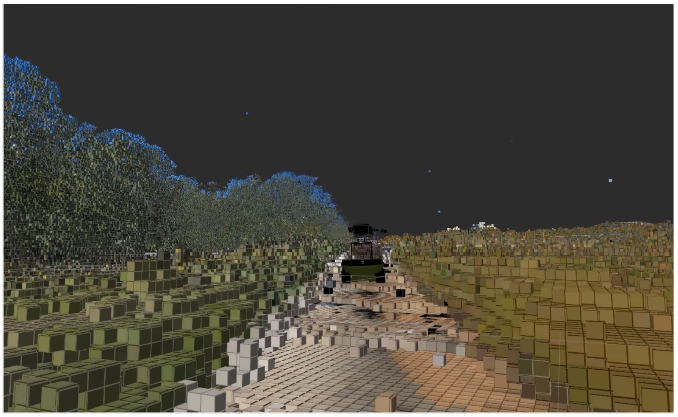

Figure 27: 3D View of Robot Model from SMGS in RViz

5.2.2 Project Convergence

Project Convergence (PC) is the Army's efforts to integrate itself into joint force operations. PC brings different military organizations together to conduct joint maneuvers responding to mock situations, while exploring the use of new technologies.

The SMGS was part of PC22, which took place at Fort Irwin in California. For this assessment, a group of combat engineers were tasked with performing recon in a desert area and teleoperating construction equipment, to create fighting positions. Once again the SMGS was the key component of the operation, since it provided the engineers with enough information to determine possible digging sites. The results for this event were that engineers were able to map, with no direct line of sight, over 2km of desert area through a mix of autonomy and teleoperation. This environment resulted to be very challenging for the SMGS, since it had little features to record and report back to the user. For this mission, the engineers utilized ATAK and ArcGIS to consume the SMGS map products. The soldiers were able to successfully dig several fighting positions with the help of the SMGS map products and the vehicle's sensors. Engineers were trained on the SMGS, the ATAK plugin and the robotic platform for a little less than a week. The SMGS was used continuously for almost a week from 6 to 8 hours a day and it proved to be a robust and reliable system. It accurately mapped over 2km of the barren desert and recorded all of the information in the Site Model database without the use of GNSS. Figure 28 is the map before site characterization. Figure 29 is a geospatial raster viewed in a GIS, with the green markers on the robot's path at the location of the stored camera image files. The color difference highlights the area that was mapped by the robot along the green markers in comparison to the a priori data. This raster and image shows a berm in the a priori information that was removed and was used by the combat engineers to plan fighting positions. Figure 30 is the same map, showing the relief to illustrate higher resolution 3D mapping and the berm that was removed.Once again the success of this assessment highlights the effectiveness, practicality and usability of the overall system.

Figure 28: A Priori Map in GIS Before Site Characterization

Figure 29: Updated Geospatial Raster from SMGS

Figure 30: Relief Map in GIS from SMGS

6 SUMMARY AND CONCLUSION

In this work, the Site Model Geo-spatial System (SMGS) is introduced to address shortcomings in modern site characterization and world mappings. The SMGS is a multi-functionality C++ ROS package utilizing a non-relational database. The SMGS' modular design is made to configure to any ROS-based platforms and interface with its instrumentation. The SMGS takes in a-priori Geographic Information System (GIS) maps and sensor data and produces updated GIS maps, 3D environmental models, geo-tagged images, and navigational cost maps. SMGS enriches its output by adding non-traditional geotechnical information, including soil type, moisture content, and material type. The GIS information is encapsulated within 3D pixels (voxels) organized into octrees, enabling more efficient and accelerated data algorithms. SMGS was validated in simulation, with offline recorded-data tests, and through several operational experiments conducted in diverse environments by SMGS developers and external operators. Provided sufficient localization or a feature rich environment, the SMGS is capable of meeting all of the goals of ingesting geospatial information, updating and improving the information, and producing new or updated geospatial products. While there are no constraints other than processing power for using this system on aerial vehicles, the advantage of mapping with ground vehicles is the ability to map under canopies and with geotechnical sensors in close proximity to the ground. The SMGS successfully combined open source software with different robotic research development platforms to demonstrate operations and conduct missions in GNSS denied environments under real time processing constraints.

Mapping with ground vehicles is challenging, due to the low angles of attack from the sensors, localization without Real-Time Kinematic (RTK), differential GPS, or other GNSS. Without robust GNSS denied localization capabilities as described in this paper, the accuracy of the mapping system is severely impacted. For example, as errors in the localization compound, the mapping system may start to produce artifacts such as double walls on buildings, etc. Additionally, most photogrammetry/LiDAR based localization techniques require salient features to be present within the environment to provide reliable ground truth localization. This finding was corroborated by the authors during the field experiments described in Sections 5.2.1 and 5.2.2.

The SMGS assumes reputable sources of a priori geosaptial information, for example recent aircraft system or National Geospatial-Intelligence Agency (NGA) data. Testing validated that geospatial data could be out of sync with terrain or structural changes to the site, for example in figures 28 and 29, as long as there are enough static elements to register the incoming point cloud to the a priori point cloud. With the case of the registration failing in Orienteer, the covariance matrix will reflect the error to the EKF and other localization systems, such as odometry or ATLIS [20], will be used for updating the map. Accuracy may be diminished from Orienteer until the system is able to find new features to register the point clouds, but mapping continues unless the covariance from localization is outside the threshold. Additional research is currently being conducted by the authors to use other registration algorithms, such as Normal Distributions Transform (NDT) [19], to improve performance, accuracy, and the covariance calculations.

The design of the SMGS incorporates an enterprise geospatial database system that is capable of mirroring and sharding the database across multiple platforms. Currently, the authors are conducting additional research and development to ensure the data is synchronized with overlapping pointclouds from multiple platforms mapping at the same time. Once tested and validated, this architecture will allow the site model information captured by a particular unmanned platform to be shared with others, which will serve as an essential component for enabling collaborative task execution, progress tracking, and change detection by multiple robots.

In the future, researchers plan to expand SMGS to include negative obstacle detection, graph-based SLAM corrections for updating voxel data when loop closures are available, add additional sensor modalities (radar, sonar, radiometric, magnetometer based, multi-spectral, cone penetrometer, etc), and implement SMGS on earth moving vehicles. Future research also needs to be completed to improve discarding bad sensor data, for example implementing a multi-hit policy where multiple sensors validate point cloud data. Additionally, the SMGS currently produces a global cost map based on slopes that can be used by navigation; however, future research shows promise with classification and geophysical information from the site model used to create better cost maps or provide information to the NATO Next Generation Reference Mobility Model [21].

7. REFERENCES

References

[1] G. Mattyus, S. Wang, S. Fidler, and R. Urtasun, “Hd Maps: Fine-Grained Road Segmentation by Parsing Ground and Aerial Images,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3611–3619.

[2] W.C. Ma, et al., ”Exploiting Sparse Semantic HD Maps for Self-Driving Vehicle Localization,” in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2019, pp. 5304-5311. Macau, China, November 4-8, 2019

[3] J.E. Loehr, A. Lutenegger, B. Rosenblad, A. Boeckmann, ”Geotechnical Site Characterization”, U.S. Department of Transportation, Publication No. FHWA NHI-16-072, 2017.

[4] D. Meagher, ”Geometric Modeling using Octree Encoding”, Computer Graphics and Image Processing, 1982, pp. 129–147

[5] M. Lomas, E.V. Cross II, J. Darvill, R.C. Garrett, M. Kopack, K. Whitebread, “A Robotic World Model Framework Designed to Facilitate Human-robot Communication”, in Proceedings of the Association for Computational Linguistics, SIGDIAL 2011:

the 12th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2011, pp. 301–306.

[6] Microsoft, ”Minecraft”, Computer Software, 2023

[7] Russell L. Ackoff, “From Data to Wisdom”, Journal of Applied Systems Analysis, Volume 16, 1989, pp. 3-9.

[8] Agarwal, Sarthak and Rajan, KS (2017) ”Analyzing the performance of NoSQL vs. SQL databases for Spatial and Aggregate queries,” Free and Open Source Software for Geospatial (FOSS4G) Conference Proceedings: Vol. 17 , Article 4

[9] Labbé, Mathieu & Michaud, François. (2018). RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation: LABBÉ and MICHAUD. Journal of Field Robotics. 36. 10.1002/rob.21831.

[10] Hornung, A., Wurm, K.M., Bennewitz, M. et al. OctoMap: an efficient probabilistic 3D mapping framework based on octrees. Auton Robot 34, 189–206 (2013). https://doi.org/10.1007/s10514-012-9321-0

[11] Sebastian Thrun, Michael Montemerlo, ”The GraphSLAM Algorithm with Applications to Large-Scale Mapping of Urban Structure,” The International Journal of Robotics Research, Vol. 25, No. 5–6, May–June 2006, pp. 403-429. DOI: 10.1177/0278364906065387

[12] M. F. Silva et al. (Eds.): ”Evaluation of SLAM Algorithms for Highly Dynamic Environments,” ROBOT 2019, AISC 1093, pp. 28–36, 2020.

[13] Grisetti, Giorgio; Stachniss, Cyrill; Burgard, Wolfram (2007). Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters. IEEE Transactions on Robotics, 23(1), 34–46. doi:10.1109/tro.2006.889486

[14] Lieberwirth, Undine. (2008). 3D GIS Voxel-Based Model Building in Archaeology. 10.

[15] Stafleu, Jan & Maljers, Denise & Gunnink, Jan & Menkovic, A. & Busschers, F.s. (2011). 3D modelling of the shallow subsurface of Zeeland, the Netherlands. Netherlands Journal of Geosciences. (submitted). 10.1017/S0016774600000597.

[16] F. Cao, F. Yan, S. Wang, Y. Zhuang, and W. Wang, “Season-Invariant and Viewpoint-Tolerant LiDAR Place Recognition in GPS-Denied Environments,” IEEE Transactions on Industrial Electronics, vol. 68, no. 1. Institute of Electrical and Electronics Engineers (IEEE), pp. 563–574, Jan. 2021. doi: 10.1109/tie.2019.2962416.

[17] Macenski S, Tsai D, Feinberg M. Spatio-temporal voxel layer: A view on robot perception for the dynamic world. International Journal of Advanced Robotic Systems. 2020;17(2). doi:10.1177/1729881420910530

[18] K. Doherty, J. Wang, and B. Englot, “Bayesian generalized kernel inference for occupancy map prediction,” 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, May 2017. doi: 10.1109/icra.2017.7989356.

[19] Biber, Peter; Straßer, Wolfgang (2003). ”The normal distributions transform: A new approach to laser scan matching”. Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453). Vol. 3.

[20] K. Niles, S. Bunkley, W. J. Wagner, I. Blankenau, A. Netchaev, A. Soylemezoglu, “Satellite Image Template Matching with

Covariance Estimation for Unmanned Ground Vehicle Localization: Active Terrain Localization Imaging System (ATLIS)," In Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium (GVSETS), NDIA, Novi, MI, Aug. 16-18, 2022.

[21] M. McCullough, P. Jayakumar, J. Dasch, and D. Gorsich, "The Next Generation NATO Reference mobility model development," Journal of Terramechanics, vol 73, 2017, pp. 49-60. doi: https://doi.org/10.1016/j.jterra.2017.06.002.