SMUGGLING TUNNEL MAPPING USING SLIDE IMAGE REGISTRATION

By

Brian E. Okorn

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

COMPUTER SCIENCE

May, 2011

Nashville, Tennessee

Approved:

Professor Julie A. Adams

Professor Gautam Biswas

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Julie Adams for her help and mentoring during my time at Vanderbilt. She took a chance on me as a sophomore when she accepted me a summer research assistant and has guided me through both my Undergraduate and Masters program. Without her help, I do not believe I could have achieved nearly as much in my short time here. Thank you for all the doors you open and the support you gave me.

I would like to thank Dr. Gautam Biswas for his advice and time that he took to be on my thesis committee. Thank you to all the members of the Electrical Engineering and Computer Science Program and all the professors and students who all helped me to get to where I am. I thank the ASEE Foundation for the SMART Scholarship that, not only made my stay at Vanderbilt possible, but open the door to a future of possibilities. Thank you to US Navy SPAWAR Pacific, specifically Michael Bruch and Ryan Halterman, for their support. I look forward to continuing my work with them at SPAWAR Pacific. Thank you as well to the Centre for Applied Autonomous Sensor Systems at Orebro University for the hall data.

I would also like to thank my friends and family whose love and friendship kept me sane. My mother who always pushed me to never settle for less than my best and my father who taught me so much more than I could ever learn in a class room.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

θ	Rotation about the Natural Axis
θ,ψ,ϕ	Euler Angles
d	Distance along the Natural Axis
x,y,z	Cartesian Offsets
ICP	Iterative Closest Point
Loop Closure	The process of determining that a robot has mapped its present location at a previous time
LRF	Laser Range Finder
Natural Axis	The curve defining the center of a tunnel
Point Cloud	A set of 3D points
Range Image	Range data obtained from a Laser Range Finder
Scan	A set of points derived from a Laser Range Finder
Scan Matching	The process of determining the translation and rotation between the coordinate frames scans
Scan Registration	See Scan Matching
SLAM	Simultaneous Localization and Mapping
Slide Image	A frame invariant point descriptor used for localization in tunnel like environments

CHAPTER I

Introduction

There exist a large number of unmapped tunnels across the US-Mexican border used primarily for smuggling, which the US Government desires to map using robots. These tunnels are used to smuggle illegal drugs and weapons into the United States and pose a serious threat to the United States national security. Once a smuggling tunnel is detected from the surface using SONAR (Quivira et al., 2010) or electromagnetic field sensors (Stolarczyk et al., 2005) a bore hole is drilled into the tunnel and a small robot can be deployed to map the tunnel. With accurate maps of these smuggling tunnels, the US Government can obtain warrants for the exit points thus preventing dangerous drugs, weapons and terrorist from entering the country.

The generation a map of any environment requires a robot to be able to extract the relevant data from its sensor systems and determine how this new data fits with the data it previously collected. This process is called scan registration. This thesis uses a 3D laser scanner as the primary sensor system. The laser scanner produces a collection of 3D points corresponding to the geometric shapes of objects in the robots scan path. These collections are called point clouds. Given two point clouds, the mapping system must find a transformation matrix contacting the rotations and translations to bring one point cloud into the geometric reference frame of the other point cloud. Without accurate scan registration, it is very difficult to generate an accurate map any environment, especial one as structurally simple as smuggling tunnels. This thesis examines two novel approaches to the registration of 3D laser scan point clouds designed to handle the problems associated with the registration of smuggling tunnels. The presented algorithms leverage the unique structure inherent in smuggling tunnels to generate more accurate scan registrations.

Until recently most tunnel mapping has focused on mine systems. Smuggling tunnels, unlike mines, have a simple, highly linear structure. The tunnels are single, relatively straight hallway-like structures with few, if any intersections or off shoots. The lack of intersections leaves little opportunity for loop closure, the primary method of error correction in most autonomous mapping systems. A more accurate scan registration algorithm must be used to remove error from these environments. The walls of these tunnels are normally cut stone with few distinguishing features beyond power and ventilation lines. The lack of unique features makes registration difficult for standard scan registration algorithms. The two algorithms presented are specifically designed to handle the low feature environments of smuggling tunnels.

The presented algorithms use a variation on the slide image, a point descriptor originally designed for the mapping of underwater caverns using SONAR. This method is adapted to use larger, more complex laser scans to register tunnel scans. The most unique features of the walls are the texture of the cut rock. The

presented algorithms account for these features, allowing for better scan registration than the standard ICP algorithm.

Chapter II discusses the recent developments in the state of the art of mapping technologies, with a focus on localization and mapping. Chapter III presents two novel algorithms for matching 3D point clouds for tunnel like environments. Chapter IV describes several experiments conducted to validate the accuracy and flexibility of the presented algorithms. Finally, Chapter V provides the detailed contributions and conclusions, while also outlining future work.

CHAPTER II

Literature Review

The use of 3D autonomous mapping is new to the field of boarder control, but past applications of these mapping and registration methods show promise. The primary method of mapping environments has been the Simultaneous Localization and Mapping (SLAM) algorithms. These algorithms began by mapping a 2D plane, but have been applied to 2.5D and recently 3D environments. The 3D nature of any underground system of tunnels requires a 3D SLAM approach largely because of the uneven terrain. The uneven terrain causes 2D approaches to scan different portions of the environment during different scans. Most SLAM algorithms use stereovision cameras and a feature detecting approach, laser range finders (LRFs) and an iterative closest point (ICP) approach, or a combination of the two. Due to the uniform nature of smuggling tunnels, it is believed that feature based approaches will be ineffective; therefore, this thesis focuses on the ICP approach. A great deal of work mapping underground mines has been completed. Underground mines share many characteristics with smuggling tunnels, such as the environmental conditions and sensor requirements. This literature review examines some of the previous work on autonomous mapping to identify possible scan matching algorithms and sensor configurations, as well as the limitations posed by the subterranean environment.

II.1 SLAM

SLAM algorithms (Smith et al., 1990) are designed to compensate for the growth of error accumulated as a robot traverses an environment. The reduction of these errors is achieved by combining consecutive scans using a scan matching algorithm. After the scans are matched, the algorithm checks for a potential loop closure. A loop closure occurs when the robot revisits a location that has already been mapped. Once a loop is identified, the global positions of the scans along the currently mapped path are modified to match the position of the earlier, less erroneous mapping. Without this process, the accuracy of a map relies solely on the accuracy of the scan matching. Unfortunately, the use of only current scan matching algorithms tend to lead to numerous mapping errors.

II.1.1 2D SLAM

The standard sensor suite for 2D SLAM is a single forward facing LRF that generates a series of 2D environmental scans or a vision system that detects features in the environment. This thesis focuses on the methods that use laser point clouds due to constraints imposed by the tunnel environment. A common algorithm is

the FastSLAM Algorithm (Montemerlo et al., 2002). This algorithm is computationally fast and robust and is ideal for online processing. The FastSLAM algorithm utilizes the Rao-Blackwellized (Doucet et al., 2000) particle filter to estimate the probable location of each each scan in a global map. This filtering technique improves on previous algorithms (Thrun et al., 2000). Prior Algorithms represent the global environment as a single map; however, FastSLAM stores all the laser scans as odometry measurements, thus allowing the FastSLAM algorithm to deal with larger environments. A weakness of the FastSLAM algorithm is its use of a greedy maximization step to decrease the computational complexity of the algorithm. The use of this greedy maximization step can create large errors in scan registration when handling large loop closures. This problem is alleviated by only maximizing the measurement likelihood after a loop closure is detected; thus preventing the greedy maximization from processing scans with a low likelihood of matching previously mapped locations and thus lowering the potential for error.

II.1.2 2.5D SLAM

2.5D SLAM tends to be a simple improvement on the 2D SLAM algorithm, applying extra 3D data to the already complete 2D map. 2.5D SLAM algorithms can be used to generate 3D maps, while using 2D scans to create a globally consistent map. Thrun, Hahnel and Ferguson et al. provide an example of such an algorithm for abandoned mine mapping (Thrun et al., 2003). 2.5D SLAM algorithms do not utilize the data from the full 3D environment and do not capture the true three dimensional nature of the mapped environment. The lost data can be used to generate more accurate maps, thus compensating for the lack of loop closure in smuggling tunnel environments.

II.1.3 3D SLAM

3D SLAM requires the acquisition of 3D data, normally in the form of point clouds and range images generated from 3D LRFs. True 3D LRFs are expensive, but custom systems (Ohno and Tadokoro, 2005) can be built by actuating a 2D LRF and positioning the points accordingly. This actuation typically requires a nodding motion created by servo motors. Some cases, those normally pertaining to mine mapping, can be accommodated using a LRF that is rotated about its center (Bosse and Zlot, 2009), as seen at the top of Figure II.1. These methods generate different point densities. The nodding scan technique produces a relatively normal distribution of points that increase in density along the axis of the 2D sweep as the object nears the scanner. The rotated system produces a higher density image in the center of the scan with a point density that decreases radially away from the center of the scan image.

The real-time algorithm developed by Sagawa, Osawa and Echigo et al. (Sagawa et al., 2005) uses range image sequences in a very different 3D SLAM algorithm. Their algorithm uses a new weighting technique

Figure II.1: 2D Laser scanner in spinning format (Bosse and Zlot, 2009)

within the algorithm's ICP method to increase the accuracy of the alignments. The algorithm weighs the points based on their intensity value, using the observation that as the intensity value of a point decreases, so does the accuracy of the corresponding range data. The algorithm lacks a loop closure method and relies solely on the registration of the range images.

Feature Based SLAM algorithms localize using environmental features extracted from the sensor data, mostly lines (Choi et al., 2008), planes (Viejo and Cazorla, 2007) (Weingarten and Siegwart, 2006), and cylinders (Bosse and Zlot, 2009). The features found in each scan are compared to previous scans. The local position of the robot can be extrapolated from the comparisons and the global position is made more accurate though loop closure.

Viejo and Cazorla (Viejo and Cazorla, 2007) presented a 3D plane-based SLAM method for semi-structured environments. Their 3D SLAM algorithm uses plane matching to achieve scan registration. The use of the plane-based approach requires the environment to contain large flat surfaces that are matched or registered. This method greatly reduces the complexity of the data being matched, converting the point clouds into sets of planar patches. This method may work well in finished tunnels, but is likely to be highly ineffective in smaller crudely dug tunnels that lack large planer regions.

Bosse and Zlot's (Bosse and Zlot, 2009) algorithm uses a 3D laser range finder to detect the shape con-

straints of each voxel in the scan. The voxel spaces are compared using an ICP algorithm, taking into account the predominate shape of each voxel. This approach is the most applicable feature-based approach for smuggling tunnel mapping due to its fluid feature structure. However, the approach requires the environment to have a large number of unique, detectable features, which may not occur in all smuggling tunnels.

Feature based approaches are not appropriate for the presented research due to their reliance on highly structured environments with distinct repeatable features to support the localization process. Smuggling tunnels vary widely in structure; more finished tunnels may lend themselves to the feature-based approach due to their rigid structure and large number of regular geometric features. However, many smuggling tunnels lack such regularity and feature based approaches will register few detected features making localization using such algorithms impractical. For this reason, feature based approaches are not applicable to the problem of mapping smuggling tunnels studied in this thesis.

II.2 Iterative Closest Point

The primary scan matching class of algorithms are the Iterative Closest Point (ICP) algorithms. Beginning with an estimated start orientation, the algorithm iterates through possible new orientations in order to register one scan to another, until the distance between the closest points in the scans is below a given error threshold. The full algorithm is provided in Algorithm 1 (Zhang, 1994). This algorithm is susceptible to becoming stuck in a local minimum and missing the true translation and rotation between the scan coordinate frames. This algorithm was originally designed for 2D scans (Zhang, 1994), but was extended to deal with 3D point clouds (Rusinkiewicz and Levoy, 2001).

Most ICP algorithms (Cole and Newman, 2006) use only location data to register two scans, but several other types of data have been suggested to improve the scan registration accuracy, such as color or texture (Jost et al., 1998). Color has been the most successful feature integrated into the ICP algorithm (Godin et al., 2001). One major problem with color data is the acquisition and mapping of color data to the point cloud. No current LRFs can detect both color and position, thus a camera system detects the color data from a scene and then the point cloud is matched with the color images. The process of aligning a video image to a 3D laser scan can be computationally complex. The process of aligning streaming video with 3D laser scans can be simplified by using the intensity data available from many of the 3D LRFs (Godin et al., 1994) to generate an attributed range image.

Rasmussen (Rasmussen, 2002) investigated improvements to sensor fusion for autonomous road following by combining LRF, color and texture cues. Though the data fusion was used to improve road following, as opposed to SLAM, the presented approach demonstrates the improvements data fusion has for detecting similarities in environments. Rasmussen compared different trials of a learning algorithm using LRF, color

Algorithm 1 Original ICP Algorithm (Zhang, 1994)

ICP[$\{p_1\}$,$\{p_2\}$]

1: $\{p_1\}$= base point cloud
2: $\{p_2\}$= new point cloud
3: T = optimal transform
4: Estimate initial transform T_0
5: $T = T_0$
6: $\{p_2\} = T\{p_2\}$
7: $k = 1$
8: **while** $k < loop_{max}$&$error_T > error_{threshold}$ **do**
9: Find N_k closest point pairs $\{m_1\}\varepsilon\{p_1\},\{m_2\}\varepsilon\{p_2\}$
10: Estimate Transform T_k that minimizes the distance between pairs $\{m_1\},\{m_2\}$
11: $\{m_T\} = T_k\{m_2\}$
12: $\{p_2\} = T_k\{p_2\}$
13: $error_T =$ average distance between each transformed point pairs $\{m_1\},\{m_T\}$
14: $T = T_k T$
15: $k = k+1$
16: **end while**
17: **return** T

and texture data to determine which combination derived the best results. A combination of all three provided the best results. Rasmussen's algorithm provides insight into the possibility of using texture data, in addition to color data, when registering SLAM algorithm scans.

II.3 Data Fusion

A genetic ICP algorithm that fuses laser and image data was proposed by Muhlbauer, Kuhnlenz and Buss (Muhlbauer et al., 2008). This approach aligns the laser scan data using the standard ICP algorithm and then employs a genetic algorithm to fuse color stereovision data with the corresponding laser point clouds. This approach proved to be very accurate for fusing the data, preventing errors caused by false transforms from creating global discoloration in the point clouds. This algorithm was designed to color already aligned point clouds, but can be modified to color local point clouds. The genetic algorithm; however, is computationally expensive and may be impractical for real time scan matching.

The Iterative Closest Compatible Point (ICCP) algorithm (Godin et al., 2001) uses attributed range images to register 3D scans. Attributed range images consist of an array of three dimensional points, each with a list of additional attributes such as color, intensity or curvature. The algorithm functions similarly to the original ICP algorithm, but instead of finding the closest point using only the point's geometry, ICCP uses the additional attribute data as well as the geometric data. As a result, the registration accuracy is improved. This algorithm allows geometrically symmetric surfaces to be properly registered, even in the presence of nonsymmetrical attribute patterns. This algorithm may be very useful when registering tunnel scans, since smuggling tunnel geometry tends to be quite uniform and the addition of attributes such as color and texture

may improve the accuracy of the scan registration.

An interesting approach to colorized ICP was proposed by Druon, Aldon and Crosnier (Druon et al., 2006). This algorithm handles large colorized point clouds with insufficient geometric features in conjunction with general ICP. The algorithm a) detects the optimal color set to match the scans, b) removes all but the point with the selected hue, and c) runs the ICP algorithm. The target color set is selected based on the percentage of points with that hue; the ideal color has between 5% and 15% of the total points associated with it. This range prevents the selection of colors without enough points to accurately match the scans, but with few enough points to sufficiently reduce the computation time. This algorithm shows promise for smuggling tunnel mapping, but requires the scans to contain reliable rich color information.

Ohno and Tadokoro (Ohno and Tadokoro, 2005) described a system to generate a colorized environment using the fusion of laser point clouds and camera images for the search and rescue domain. A 3D scanner was custom built, comprised of a 2D Leuze RS4-4 LRF and a AXIS 205 webcam mounted on a Robotis DX-116 Servomotor with a TOKIN MDP-A3U7 motion sensor to sweep the environment. The SLAM algorithm used an ICP algorithm designed specifically for free form curves and speed up the correspondence search using a K-d tree. A voxel approach was employed for mapping, with a check for color consistency to remove phantom data form the generated 3D map. The SLAM algorithm does not take advantage of the available color data, which is only used to generate a 3D colorized environment. This SLAM algorithm may not be applicable to the smuggling tunnel domain because the alignment is completed prior to data fusion. However, the sensor configuration demonstrates that a simple custom sensor can easily generate 3D point clouds with mapped vision data.

II.4 Slide Images

ICP may not be an ideal scan registration algorithm for tunnel environments, but the slide image, a point descriptor designed for tunnel environments, may solve that problem. Slide images use the tunnel's natural structure to generate point descriptors. The slide image was originally developed to register sonar scans of flooded tunnel environments (Bradley et al., 2004). Tunnel like environments have a unique characteristic that allows for a simplification of the scan matching process. Tunnels contain a "natural axis" that can be detected and used as a reference point for scan matching. The tunnel's natural axis is a curve down the center of the tunnel. At any given tunnel segment, this axis can be calculated independent of the local coordinate frame or view perspective. This process was first discussed as a new variant of the Spin Image (Johnson and Hebert, 1999), a frame invariant point descriptor used for surface matching. The slide image is also a frame invariant point descriptor, meaning two slide images may be compared without determining the geometric relationship between their frames. Using the natural axis as a reference axis, the data can be stored as local polar models

comprised of a series of two dimensional histograms. These histograms are binned with respect to distance from the natural axis and rotation about the natural axis relative to a gravity vector. This model reduces the six dimensional search space, *x, y, z,* θ*,* ϕ and ψ, of the point cloud matching to a two dimensional search space, the rotation about the natural axis, θ, and the distance along the natural axis, d. Since its original development, slide images have been mentioned as future work (Silver et al., 2006), but further research has yet to be published.

II.5 3D Normal Distributions Transform

A recently developed method for point cloud registration is 3D Normal Distributions Transform (3D-NDT) (Magnusson et al., 2007). 3D-NDT uses a 3D variant of the Normal Distributions Transform, originally used for the registration of 2D range images (Biber and Strasser, 2003). Instead of using the 3D point for registration, 3D-NDT uses a voxel structure with each voxel containing a normal distribution describing the probability of finding a surface in that region. This approach gives a smooth model of the data with continuous first and second derivatives allowing for the use of standard numeric optimization methods to solve the registration problem. 3D-NDT used Newton's Method to find minima with respect to matching probability in the transform function. 3D-NDT, like ICP, is susceptible to local minima and relies on a good initial transform to generate an accurate matching. This method has been tested in subterranean environments and may be applicable to smuggling tunnel mapping.

II.6 Mine Mapping

The underground environment of mines is very similar to that of smuggling tunnels. The structures of the walls tend to vary from mine to mine, but each individual mine's walls tend to be nondescript with occasional support beams. The nondescript nature of mines, and in the case of smuggling tunnels, makes visual SLAM a poor choice for mapping, thus most mine mapping systems use LRFs (Thrun et al., 2003). These underground environments have six degrees of variability, and are truly a three dimensional environment. Both mines and smuggling tunnels have the potential for 3D obstructions, but these obstructions are less likely to be encountered in smuggling tunnels due to their recent and continuous use.

Carnegie Mellon University (CMU) has conducted a majority of the mine mapping research to date (Thrun et al., 2003). CMU conducts most of their experiments on the abandoned mines in Pennsylvania using a custom built robot called the Groundhog (Thrun et al., 2004). The Groundhog robot's chassis is comprised of the front halves of two all-terrain vehicles united with their steering in opposing direction in order to decrease the vehicle's turning radius, which is essential for navigating the tight turns of the mine systems, see Figure II.2. Both ends of the robot have a tiltable SICK laser range finder capable of capturing accurate

Figure II.2: The Groundhog, CMU's autonomous mine mapping robot.

3D scans. Prior experiments fixed two LFRs such that one faced forward, while the second faced directly upwards (Thrun et al., 2003). The forward facing LRF is used for mapping and obstacle avoidance, while the upward facing LRF generates a 3D map of the mine after the SLAM algorithm localizes the forward facing laser scans.

The system described by Ferguson et al. (Ferguson et al., 2003) runs in two phases. The motion phase uses the 2D LRF to guide motion, while a stationary phase allows the robot to generate a 3D scan of the environment and plan a new path. This algorithm generates a 2D map of the mine using Markov Random Fields and uses an A* search and C-space maps, generated using the 3D data, to plan motion.

Nuchter et al. (Nuchter et al., 2004) discussed a 3D SLAM algorithm designed to handle the six degrees of freedom inherent in a subterranean environment, three translational degrees, x, y and z and three rotational, θ, ϕ and ψ. The 3D scans are reduced and processed with a fast ICP variant in order to locally register the point clouds. The global registration uses the neighborhood of each scan, comprised of all scans with significant overlap with the current scan. This overlap is the portion of the known scans that will potentially correspond with a portion of the new scans. The overlap is calculated using the estimated location of each scan. K-d trees permit rapid access to the point data. This algorithm does not rely on loop closure, thus it is applicable to smuggling tunnel mapping.

Most of the CMU experiments are completed at the Bruceton Research mine near Pittsburgh (Huber et al., 2003). More advanced experiments have been conducted at the Florence Mine (Thrun et al., 2004), an

abandoned mine near Pittsburgh that has been partially flooded and deemed inaccessible to humans. Some experiments were also conducted at the Mathie Mines near Courtney, PA, (Nuchter et al., 2004) where the Groundhog robot mapped an unknown mine until it reached a broken support beam blocking the path. The Groundhog robot determined that there was no navigable path and returned to the surface.

Mines and smuggling tunnels share many structural attributes that allow mine mapping algorithms to be applicable to smuggling tunnel mapping. Both environments have truly 3D structures and the surface, shape and texture of the walls tend to vary widely, as seen in Figure II.3. This variability and irregularity prevent many current algorithms, such as feature detection, from being applicable to either domain. The variable depth component of each environment tends to require a three dimensional approach. One major advantage mines have over smuggling tunnels is that the mine shafts tend to cross paths creating ideal locations for loop closure. Such locations typically do not exist in smuggling tunnels, since these tunnels tend to be linear. Thus any improvement in mapping accuracy must come from an improvement in the scan matching or by forcing loop closures by retracing a path.

II.7 Summary

The 3D nature of the smuggling tunnels prevent the application of 2D SLAM approaches. Data fusion can be useful in these environments, but its use will be saved for future work. ICP and 3D-NDT are powerful algorithms that may be useful in this environment and can be used for testing. Slide Images show the most promise for this application, because they are specifically designed for tunnel environments. ICP was selected over 3D-NDT because ICP is slightly less likely to fall into local minima when not using an initial transform.

(a) (b)

(c)

Figure II.3: Types of Smuggling Tunnels: (a) rough, (b) semifinished and (c) finished

CHAPTER III

Slide Image Algorithms

III.1 Introduction

This chapter presents two new algorithms used to determine the registration between two 3D laser scans. The first algorithm uses slide images to determine the transformation matrix between the scans. The Slide Image algorithm is described in Chapter III.2. The second algorithm described is a fusion between the Slide Image algorithm and the ICP algorithm. The Fusion algorithm is described in Chapter III.3.

III.2 Slide Images

Tunnel like environments have a unique characteristic that allows for a simplification of the scan matching process. These environments contain a ”natural axis” (Bradley et al., 2004) that can be detected and used as a reference point in scan matching. The tunnel's natural axis is a curve down the center of the tunnel, shown in Figure III.1 as the red line through a blue point cloud.

At any given segment of the tunnel, this axis can be calculated independently of the local coordinate frame or view. This process was first discussed by Bradley, Silver and Thayer (Bradley et al., 2004) as a new variant of the Spin Image (Johnson and Hebert, 1999). The slide image is a frame invariant point descriptor for tunnel cross section matching. Two slide images may be compared without determining a geometric relationship between their coordinate frames. Using the natural axis as a reference axis, the data can be stored as local polar models in the form of a series of two dimensional histograms binned with respect to distance from the natural axis and rotation about the natural axis relative to a gravity vector. Using this method, the six dimensional search space of point cloud matching, defined by $x, y, z, \theta, \phi, \psi$, is reduced to a two dimensional search space defined by a distance along the natural axis, d, and a rotation about the natural axis, θ. This two dimensional search space is shown in Figure III.2 as the blue plane's distance along the red axis and the blue plane's rotation about the red axis, respectively.

The Slide Image algorithm is described in Algorithm 2. The Slide Image algorithm first detects an natural axis in the tunnel data. The algorithm, using this natural axis, partitions the data and slide images are generated for each partition, described in Chapter III.2.3. The partitions are used to generate slide images using the algorithm described in Chapter III.2.4. A comparison of the slide images for each scan is provided by the algorithm described in Chapter III.2.5. Finally, the comparison data is used to determine a registration transformation matrix using the algorithm described in Chapter III.2.6.

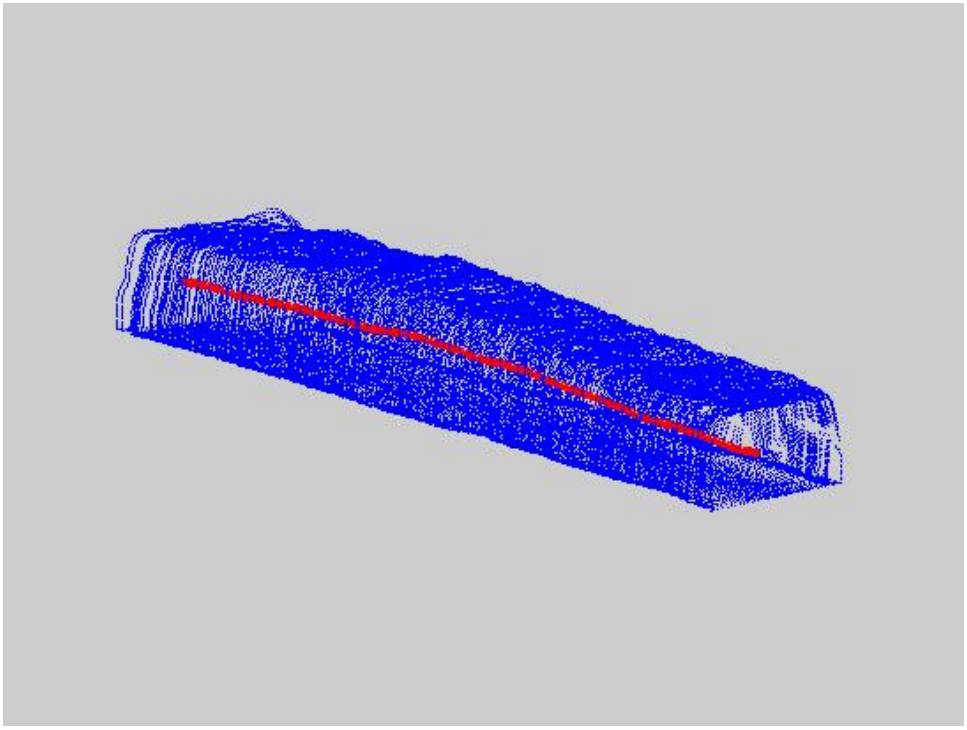


Figure III.1: Natural Axis

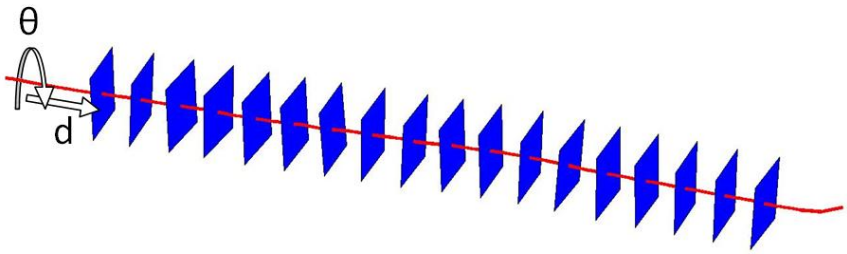


Figure III.2: Slide Image Search Space

Algorithm 2 Slide Image Algorithm.

SlideImageRegistration[$\{X\}$,Y]

X= new scan points
Y= base scan points
$natAxis_X$ = new scan natural axis
$natAxis_Y$ = base scan natural axis
R = rotational transformation matrix
T = translational transformation matrix
$\{X_{axis}\}$ = partitioned base scan along natural axis
$\{Y_{axis}\}$ = partitioned new scan along natural axis
$\{image_X\}$ = new scan slide images
$\{image_Y\}$ = base scan slide images
$\{image_X\} = SlideImageGeneration[\{X_{axis}\}, natAxis_X]$
$\{image_Y\} = SlideImageGeneration[\{Y_{axis}\}, natAxis_Y]$
$[d, \Theta] = SlideImageComparison[\{image_X\}, \{image_Y\}]$
$X_{matched}$= matched new scan points using translational offset d
$Y_{matched}$= matched base scan points using translational offset d
$[R,T] = SVDTransExtract[X_{matched}, Y_{matched}, \Theta]$
return $[R,T]$

III.2.1 Input

The input to the Slide Image Registration algorithm is a series of 3D point clouds derived from 3D laser scans of a smuggling tunnel like environment. The specification of the environment is discussed in Chapter III.2.

III.2.2 Output

The algorithm outputs a fully register 3D point cloud of the entire scanned area. This point cloud can be used in its raw form as a 3D map or projected to the ground plane to create an accurate 2D map. Each individual scan has a 3D transform associated with it to transform the local frame to the global coordinate frame.

III.2.3 Axis Detection

The detection of the natural axis should be view independent; however, the scan needs to be large and complex enough to accurately model the tunnel structure. The slide image registration algorithm requires that the majority of the scan must include an unbroken section of the tunnel's hull, for example, 10m (Bradley et al., 2004). The detection of the axis cannot occur locally in regions were portions of the hull are missing due to gaps in the data. The surrounding natural axis values can be used to define the natural axis to prevent gaps from causing large errors in the natural axis detection. Bradley, Silver and Thayer estimated the natural axis using a series of radial 2D sonar scans (Bradley et al., 2004). The center points of 10m worth of scans were used to generate a single slide image with a single linear axis. A covariance matrix decomposition method was used to calculate the axis from these center points.

The Slide Image Registration Algorithm, see III.2, requires a different approach due to the increased size

and complexity of each 3D scan. A partitioning of the data set is required such that the centers of each of the partitions are segments of the natural axis. The axis is estimated using two orthogonal 2D projections of the data. A shared major axis for the projections is selected based on the point cloud data. These axes can be determined by analyzing the covariance matrix of the 3D points. The Eigenvector with the largest Eigenvalue will be the major projection axis. The two minor axes will be the orthogonal projection of the second largest Eigenvector with respect to the major axis and the cross product of that vector and the major axis. The point cloud coordinates are projected to the two planes formed by the shared major axis and each individual minor axis. The coordinates of the projected data is binned according the major axis and the bin averages are calculated. The averaged coordinates from each of the orthogonal projections defines the projected axes. The projected axes are sorted by their major axis coordinates and iteratively combined. The missing second minor axis coordinates of each projected point are linearly extrapolated using the surrounding minor axis coordinates of the opposite projected axis. The coordinates of this axis are smoothed using a low pass filter along the major axis. A basis transform is required to return the axis to the Cartesian basis. A rough natural axis is estimated for each projection using the axis with the greatest range of values by binning the data along that axis and finding the center points of the bins, as seen in Figures III.3 and III.4 as red lines through blue projections. These projection axes are fused and smoothed to form an estimate of the full natural axis, as shown in Figure III.5 as a red line through a blue point cloud. Unlike the original method, the Slide Image Registration algorithm's detected axis consists of a curve created by the axes of the individual scan segments. This method is more prone to variance due to noise; however, the increased complexity of the axis allows the Slide Image Registration algorithm to use the structure of the axis to better register scans and estimate the transforms.

III.2.4 Slide Image Generation

Once the natural axis of the tunnel is estimated, the point cloud is partitioned using a number of rectangular segments, as seen in Figure III.6 as green boxes along a red natural axis segmenting a blue point cloud. Each box is spaced such that each center is equidistant from the center of the previous box segment. The center of each box lies on the natural axis curve and the largest face perpendicular to the natural axis at that point. These subsets may intersect and can be selected such that there is considerable intersection with no ill effect on the algorithm's outcome.

A rectangular range search is used to find the points within each partitioned region. The 3D range search can be approximated by a 2D oriented rectangular range search (Mehlhorn and Näher, 1999) to improve the speed of the search. The 2D search is performed on the 3D point cloud, while neglecting the vertical component with a post processing check to remove the extra points found by this approximation. This process

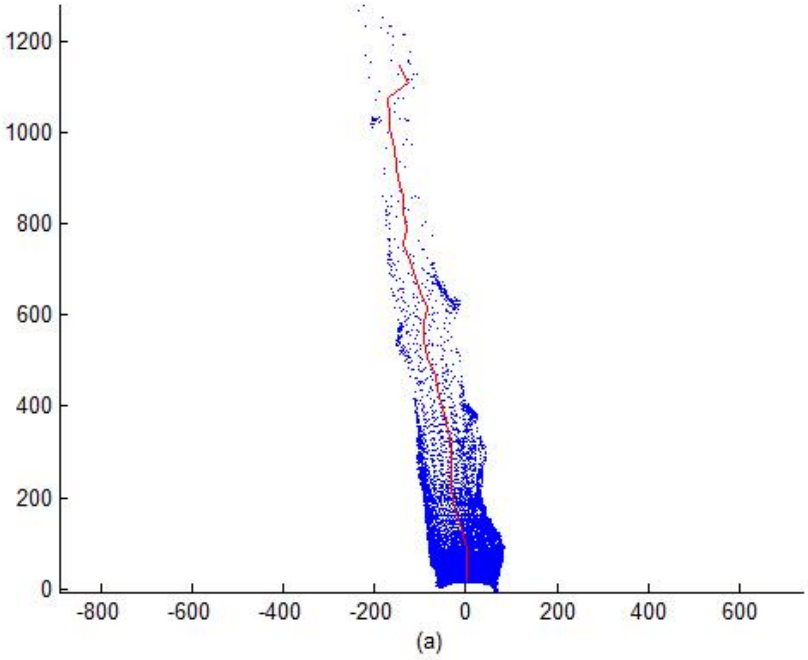


Figure III.3: XY Projection and Rough Axis. YZ projection in blue and projected natural axis in red.

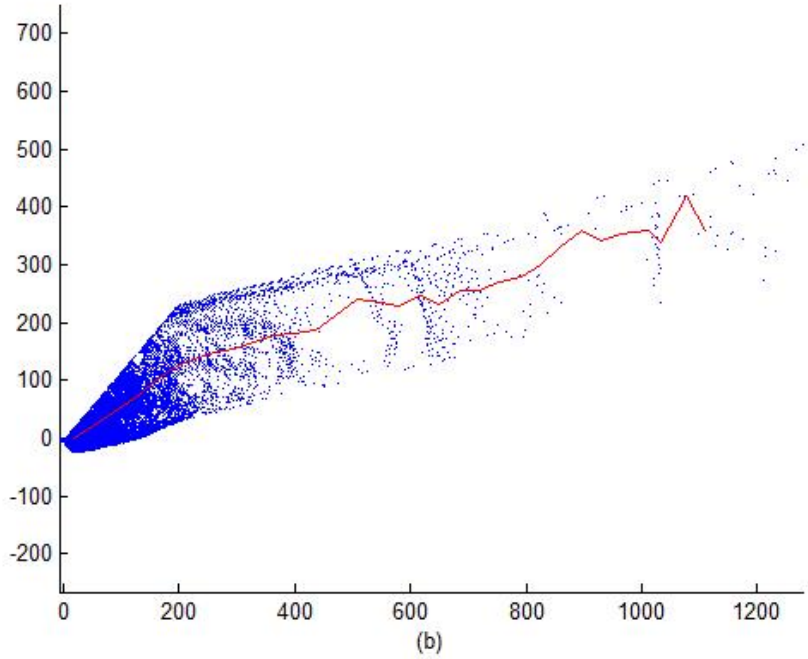


Figure III.4: YZ Projection and Rough Axis. YZ projection in blue and projected natural axis in red.

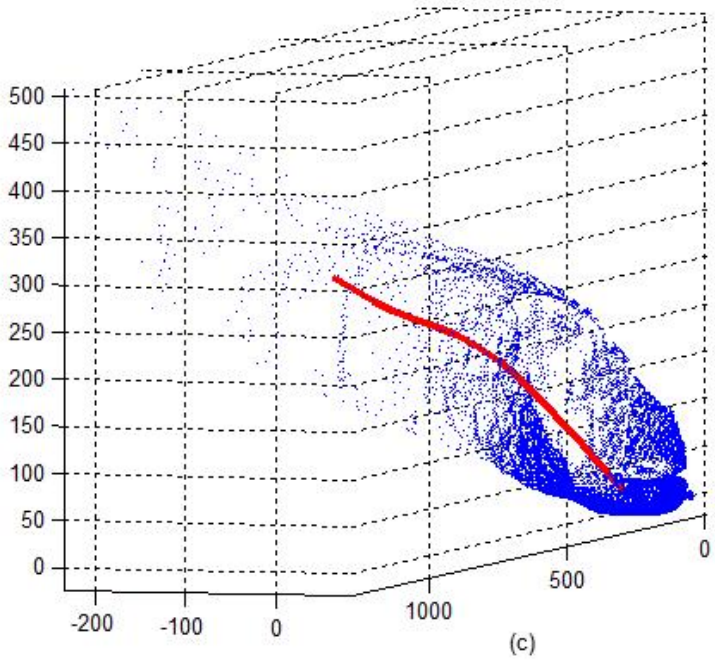


Figure III.5: Fused Natural Axis. Laser point cloud in blue and detected natural axis in red.

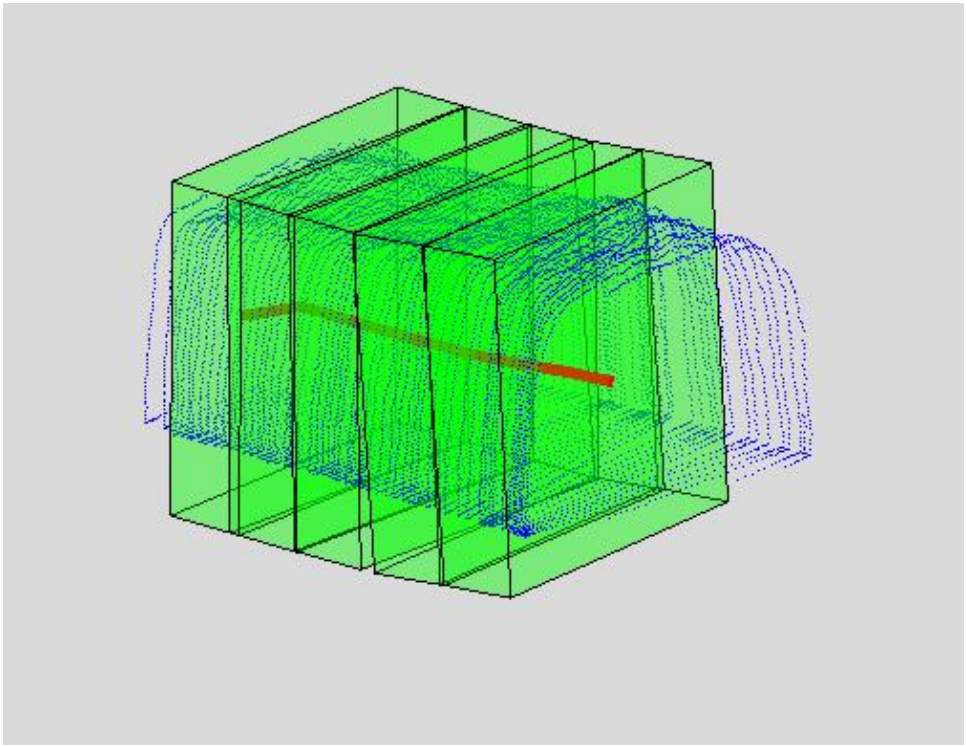


Figure III.6: Scan Segmentation for Slide Image Generation

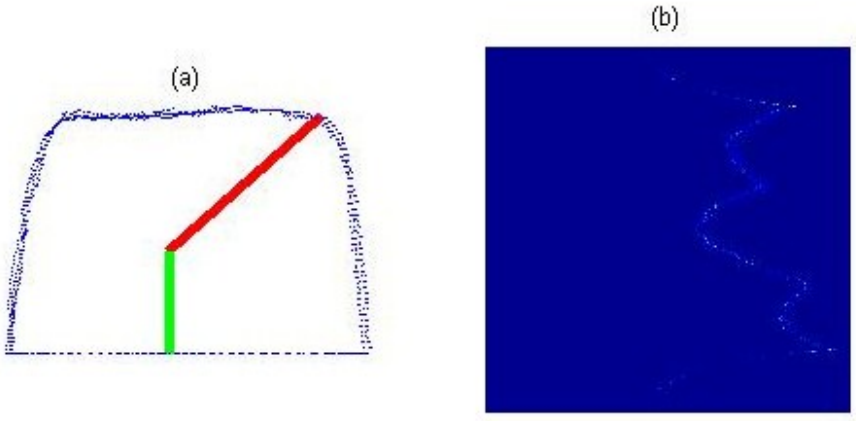


Figure III.7: Slide Image Histogram Generation: (a) Tunnel segment with Point and Gravity vector, (b) Standard 2D Histogram

is provided in Algorithm 3.

The distance from each point in a given partitioning box to the closest point on the natural axis segment associated with its partitioning box is calculated as well as the point's rotation about the natural axis segment using the projection of the negative vertical axis as a zero point, as shown by Figure III.7a. The values are then binned to generate the slide image histogram, as seen in Figure III.7b. The point densities of the laser scans are not uniform. The point density decreases as objects move farther from the laser range finder. This imbalance in corrected using a Gaussian blur with a σ proportional to the image's distance from the laser range finder. The Gaussian is plotted on the histogram centered at the coordinates given by d and θ, as shown in Figure III.8.

III.2.5 Slide Image Comparison

The slide images vary along two variables, d and θ. The rotational variance between scans is relatively low, thus the search over the domain of d can be decoupled from the domain of θ. The decoupling of the search for d and θ allows the best d value to be determined and used to search for the best θ. The full algorithm is provided in Algorithm 4.

The best value for d is determined by minimizing the absolute difference between the set of 2D histograms, and the matching histograms of the reference slide images at position d as d varies. Once this position is determined, θ can be calculated using the matched slide images. This calculation is completed individually due to the potential curved nature of the axis at any point in the tunnel. This curvature can cause

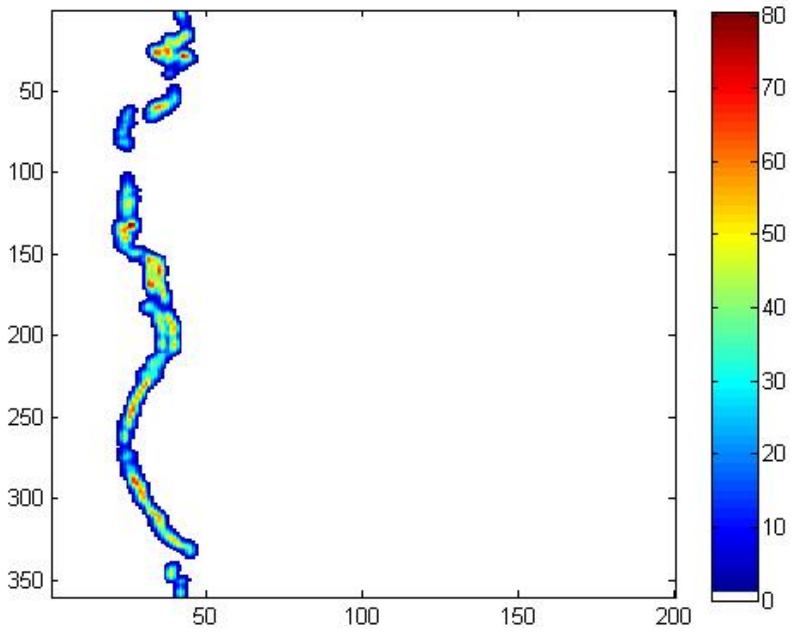


Figure III.8: Gaussian Burred Slide Image Histogram

Algorithm 3 Slide Image Generation.

SlideImageGeneration[$\{X\}$,Y]

Y= natural axis segments
rotBins = number of rotational bins
radBins = number of radial bins
for all $y_i \in Y$ **do**
  $image_i = rotBins \times radBins$
  X_i= points within the range of y_i
  y_{start}= first point of vector y_i
  $v_{gravity} = y_i \times [0,0,-1] \times y_i$
  for all $x_j \in X_i$ **do**

$$x_{closest} = y_{start} - y_i \cdot [(y_{start} - x_j) \bullet y_i]/||y_i||$$

$$v_{point} = x_{closest} - x_j$$

$$r = ||v_{point}||$$

$$\theta = \text{sign}(v_{point} \times v_{gravity} \bullet y_i) \cdot \arctan(||v_{point} \times v_{gravity}||, v_{point} \bullet v_{gravity})$$

$$image_i(\text{ceil}(\theta/rotBins), \text{ceil}(r/radBins)) += 1$$

  end for
end for
return $\{image\}$

Algorithm 4 Slide Image Comparison

SlideImageComparison[X,Y]

X =matched new scan natural axis points $[x_1, x_2 \dots x_n]$
Y =matched base scan natural axis points $[y_1, y_2 \dots y_n]$
Θ =rotational offsets$[\theta_1, \theta_2 \dots \theta_n]$
d =translational offset
R_x =overlapping range of slide images
R_y =overlapping range of slide images$+j$
O_{size} =size of overlapping range
m =image compression ratio
$imageDiffd(j) = \sum_{x,y} ||X(R_x,x,y) - Y(R_y,x,y)||/O_{size}$
$d = argmin_j(imageDiffd(j))$
$imageDiff\theta(k,X,Y) = \sum_y ||X(1 \dots n, 1 \dots k, y) - Y(1 \dots n, end-k+1 \dots end, y)||+$
$||X(1 \dots n, k+1 \dots end, y) - Y(1 \dots n, 1 \dots end-k, y)||$
for $j = 1$ to n/m **do**
    $X_c = compressedX_{mj \dots mj+m}$
    $Y_c = compressedY_{mj+d \dots mj+m+d}$
    $[\theta_{mj}, \theta_{mj+1} \dots \theta_{mj+m}] = argmin_k(imageDiff\theta(k, X_c, Y_c))$
end for
return $[d, \Theta]$

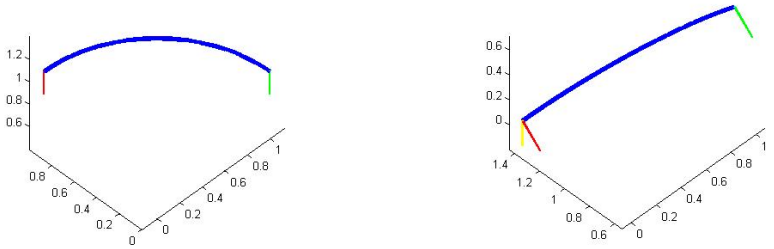


Figure III.9: Variance in θ within Single Scan

the rotational difference between the sets of images to be different for some images in the set, as seen in Figure III.9. The projected gravity vector for the left side of the curve, shown in red, is rotated away form the original gravity vector, shown in yellow, as the curve is rotated, but the right side projection, shown in cyan, remains unchanged.

The rotational offset, θ, is calculated by minimizing the difference between each of the matched slide images 2D histograms, as the position of the θ axis origin is shifted for the new images. An absolute difference is used to calculate the histogram difference, but several other metrics have been suggested by Bradley Silver and Thayer (Bradley et al., 2004). The local similarity in rotational differences allows for the compression of the set of slide images into a smaller set for the minimization of θ. This compression is completed by partitioning the full set of slide image histograms into n equal consecutive images and combining each partition

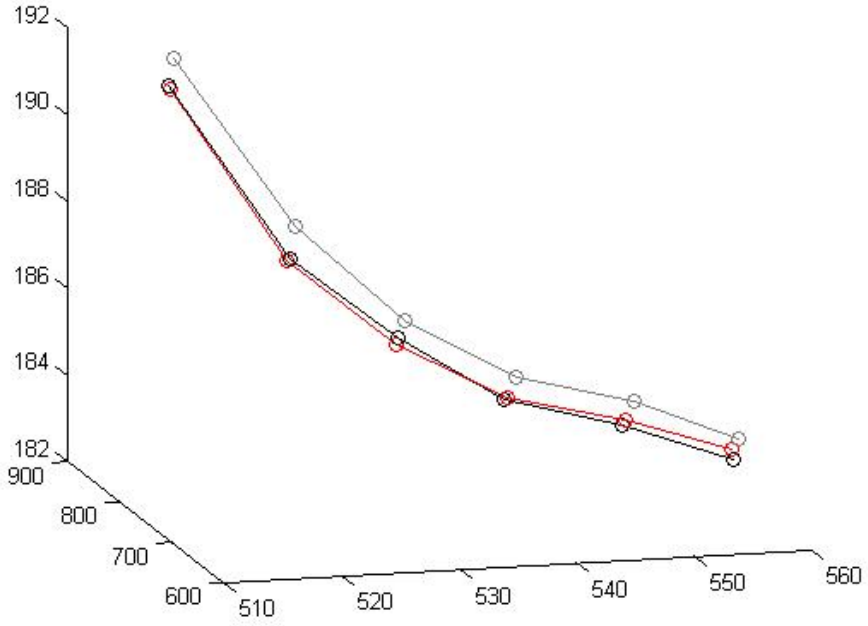


Figure III.10: Slide Image Transformation Extraction Reference Points

into a single image by summing the histograms.

III.2.6 Transform Extraction

The coordinate transform is extracted from the slide images using both the structure of the axis and the rotational mapping between the axes. Each slide image has two reference points, the center point of the natural axis and a point to compensate for the rotational offset. This rotational offset point is set to a defined distance from the center point of the natural axis, such that the vector formed by this point and the center point is perpendicular to the natural axis. The angle of this vector with respect to the gravity vector for the reference slide images is the rotational offset selected by the comparison function, as shown in Figure III.10 and zero for the new slide images. The base axis is shown as the black axis, the extracted transform axis is shown in red and the true transform axis in gray. These points represent a matched pair of points in two separate coordinate frames. The single value decomposition methodology used for ICP (Arun et al., 1987) can be used to generate the most accurate transform from one point set's coordinate frame to the other's, as described in Algorithm 5. Using this transform, the newly added slide images can be transformed to the proper coordinate frame and added to the reference map.

Algorithm 5 Transfrom Extraction

TransfromExtraction[X,Y,Θ]

X =matched new scan natural axis points $[x_1, x_2 \dots x_n]$
Y =matched base scan natural axis points $[y_1, y_2 \dots y_n]$
Θ =rotational offsets$[\theta_1, \theta_2 \dots \theta_n]$
r =radial offset
G_x =new scan natural axis gravity vector
N_x =new scan natural axis normal vector
G_y =base scan natural axis gravity vector
N_y =base scan natural axis normal vector
R =rotation
T =translation
for all matched pairs $(x_i \in X, y_i \in Y)$ **do**
    $N_x = (x_i - x_{i-1})/||x_i - x_{i-1}||$
    $G_x = N_x \times [0,0,-1]^T \times N_x$
    $G_x = G_x/||G_x||$
    $x_{rot} = r \cdot G_x + x_i$
    $X = X + x_{rot}$
    $N_y = (y_i - y_{i-1})/||y_i - y_{i-1}||$
    $G_y = N_y \times [0,0,-1]^T \times N_y$
    $G_y = G_y/||G_y||$
    $y_{rot} = r \cdot G_y + y_i$
    $Y = Y + y_{rot}$
end for
$A = X - \overline{X}$
$B = Y - \overline{Y}$
$[U,S,V^T] = SingleValueDecomposition(BA^T)$
$U(1 \dots n, n) = U(1 \dots n, n) \cdot det(UV^T)$
$R = UV^T T = \overline{Y} - R\overline{X}$
return $[R,T]$

III.2.7 Summary

The power of the Slide Image algorithm is its insensitivity to local minima. The reduction in search space allows the Slide Image algorithm to preform a global search over the transform variables, thus protecting it from local minima. A major weakness of the Slide Image algorithm is its heavy reliance on the accuracy of the estimated natural axis to generate the transforms. The transforms produced are only as accurate as the natural axis estimate. There may be inaccuracies in each estimate of the natural axis due to sensor noise, intersections or variation in the robot's position with respect to the scan. Chapter III.3 describes an algorithm that alleviates this error by using the full scan instead of the natural axis to extract the transform.

III.3 Slide Image ICP Fusion

The ICP algorithm, described in Chapter II.2, can be used to register the point clouds, but requires an accurate initial transform and information regarding which portions of the two scans overlap in order to provide accurate results and avoid local minima. The Slide Image algorithm can provide this information, Chapter

III.2. The Fusion algorithm combines the Slide Image algorithm's ability to generate an accurate transform without an initial transform with the ICP's algorithm's ability to refine the accuracy of an initial transform as well as the ICP algorithm's flexibility when mapping intersections.

The Fusion algorithm is described in Algorithm 6. The transform obtained from the slide image registration is used as an initial transform to seed the ICP algorithm. The point cloud is transformed using the initial transform and the overlapping region of each point cloud is isolated. The overlapping point clouds are registered using the ICP algorithm. The overlap is determined using the intersection of the ranges of transformed data. These ranges are determined by rotating the data such that the natural axis is parallel with the Cartesian axes. The range is equal to the means of the point locations along each axes, plus four times their standard deviations. The ICP algorithm is able to fine tune the registration of the point clouds beyond the matching of the natural axes. The fusion algorithm leverages the ability of the slide image algorithm to register point clouds without initial transform estimates and the ICP algorithm's ability to greatly improve the accuracy of initial transforms without relying odometry or a highly accurate natural axis.

Algorithm 6 Fusion Algorithm.

Fusion$[\{X\},Y]$

X= new scan points
Y= base scan points
$normAxis_Y$ = average base scan natural axis vector
R = rotational transformation matrix
T = translational transformation matrix
R_{SI} = slide image rotation
T_{SI} = slide image translation
$[R_{SI}, T_{SI}] = SlideImageRegistration[X,Y]$
$X_{trans} = R * X + T$: transformed new scan points
$\theta = atan2(normAxis_Y[2], normAxis_Y[1])$
$R_{norm} = rotmatrix(\theta)$: normalizing rotation matrix
$X_{norm} = R * X_{trans}$: axis normalized new scan points
$Y_{norm} = R * Y$: axis normalized base scan points
μ_X = average of X_{norm}
σ_X = standard deviation of X_{norm}
μ_Y = average of Y_{norm}
σ_Y = standard deviation of Y_{norm}
$X_{bounds} = \mu_X \pm 4\sigma_X$
$Y_{bounds} = \mu_Y \pm 4\sigma_Y$
$X_{overlap} = X_{norm} \cap X_{bounds}$
$Y_{overlap} = Y_{norm} \cap Y_{bounds}$
$X_{overlap} = R_{norm}^{-1} X_{overlap}$
$Y_{overlap} = R_{norm}^{-1} Y_{overlap}$
$[R_{ICP}, T_{ICP}] = ICP[X_{overlap}, Y_{overlap}]$
$[R,T] = [R_{ICP}, T_{ICP}] * [R_{SI}, T_{SI}]$
return $[R,T]$

III.3.1 Summary

The Fusion algorithm can accurately refine the transforms produced by the Slide Image algorithm and with the addition of the ICP algorithm can properly handle the registration of intersections. The detection of the overlapping regions helps prevent the ICP algorithm from finding drastically wrong registrations, but there is still potential that the ICP algorithm can over correct the initial transform and produce a worse registration than the initial Slide Image Algorithm. These scenarios should be rare in smuggling tunnel environments due to the environment's structural uniformity, but may occur more frequently in less tunnel like environments.

CHAPTER IV

Experiments

This chapter presents the experiments used to test the Slide Image Registration Algorithm described in Chapter III.2 and the Fusion Algorithm described in Chapter III.3. The results of the experiments are discussed and compared to that of the standard point cloud registration algorithm, Iterative Closest Point (ICP), described in Chapter II.2. The experiments were performed using multiple environments to test the flexibility of both algorithms. 3D-NDT, described in Chapter II.5 was a potential algorithm to test the Slide Image and Fusion algorithms against, but due to its heavy reliance on the initial transform and its linear programing approach to finding minima in the transform, the 3D-NDT algorithm found and terminated on the nearest local minima, which is almost never the true registration when not using an initial transform.

IV.1 Slide Image vs. Iterative Closest Point

The slide image algorithm was compared to an ICP algorithm using two distinct data sets from different environments to test the slide image algorithm's flexibility. One environment was a real boarder tunnel, the exact environment for which this algorithm was designed. The other environment was an office hallway from Orebro University in Sweden.

IV.1.1 Tunnel Data

The slide image and ICP algorithms were tested using scans of an actual boarder tunnel provided by SPAWAR Pacific. The features of this tunnel are characteristic of the features available in most smuggling tunnels. The tunnel contained a single straight passageway with a single intersection. Larger tunnels may contain more intersections, but results at this intersection will be indicative of the algorithms performance in similar scenarios. The tunnel's ground truth is shown in Figure IV.1. The ground truth transforms were obtained by hand aligning each scan to its neighboring scans using a custom Coin 3D model editor

IV.1.1.1 Data Collection

The data was collected using an iRobot PackBot with a custom built 3D laser scanner developed at SPAWAR, shown in Figure IV.3. The 3D laser scanner is comprised of a Hokuyo UTM-30LX 2D laser range finder mounted on a motorized nodder base, shown in Figure IV.4. The angle and speed of the nodding are measured using a rotary encoder. The laser scanner is a prototype model and the base was made using a rapid prototype, thus there exists some slippage in the scanner's vertical sweeps. This slippage causes a ghosting effect on the

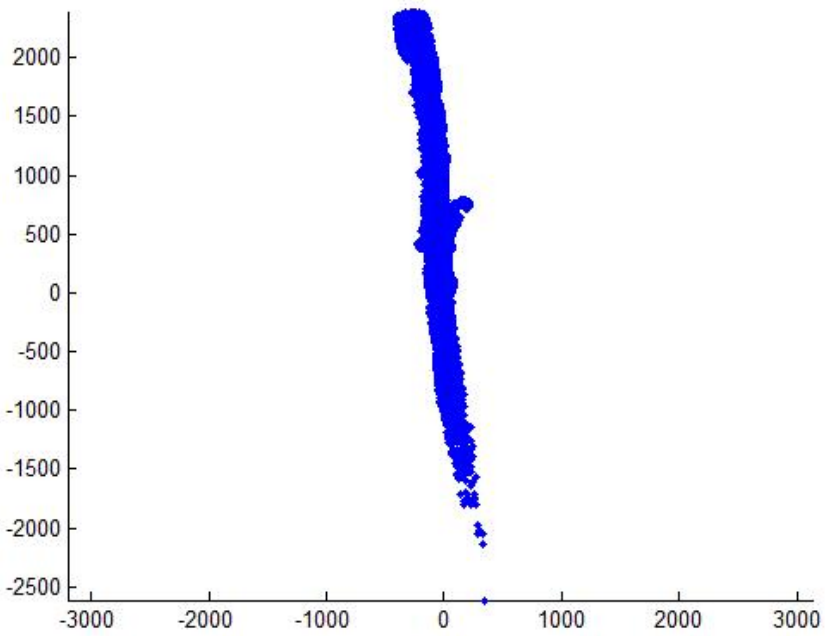


Figure IV.1: Boarder Tunnel Ground Truth

Figure IV.2: Tunnel Experiment Smuggling Tunnel

Figure IV.3: Border tunnel robot

ceiling and floor of the scans, which can be seen in Figure IV.5 as the two images of the ceiling marked by the red arrows. The final version of the 3D laser scanner will be made of more durable material to prevent such slippage. The robot was controlled by remote control, but future testing may use obstacle avoidance and path planning algorithms to control the robot.

During data collection, the robot traveled once up and down the entire length of the tunnel, stopping to capture 23 scans. Each scan created a point cloud containing an average of 250,000 points with a max range of 25 m. The sizes of these scans were reduced by voxelizing the space into 1 cm cube voxels and using only the center point from each occupied voxel. This voxelization process also compensated for some of the point density differences in the scans.

Both the slide image algorithm and a standard ICP algorithm were used to register consecutive scans. The registration of these scans is completed without the use of odometry or any other localization data, but this data can be used to generate more accurate maps. This additional localization data may be used to generate initial transforms to seed the registration process.

IV.1.1.2 Results

A sample registration is depicted in Figure IV.6. In this scan registration the ICP algorithm encountered a local minimum. The ICP algorithm encountered similar local minima in this environment in all but scan registrations 1, 3, 4 and 12. These early local minima are caused by the lack of large distinct features in the

Figure IV.4: Custom 3D laser scanner

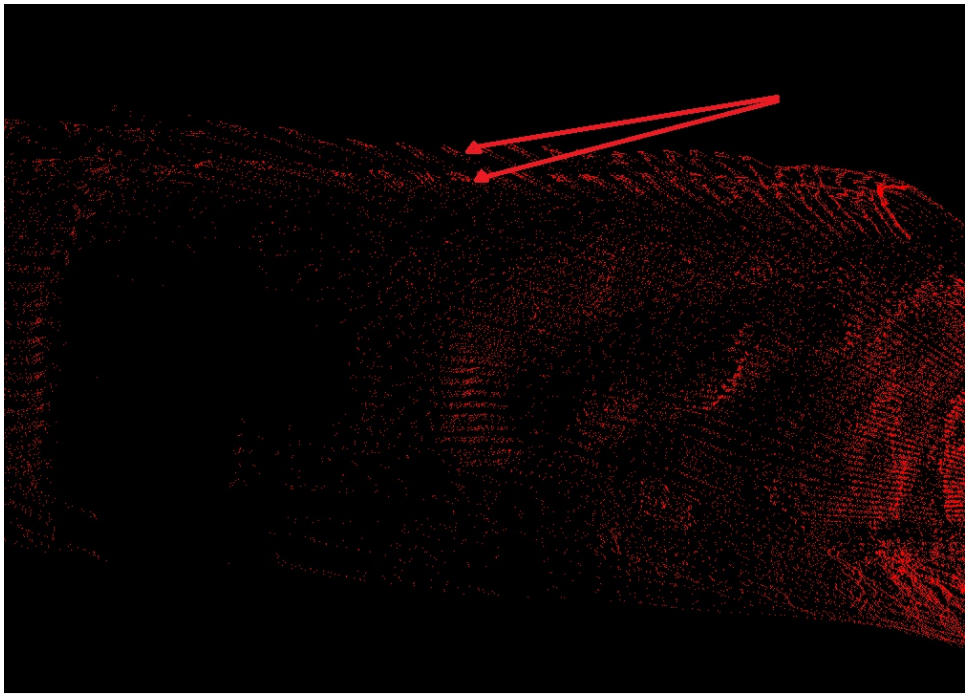


Figure IV.5: Tunnel laser scan ghosting

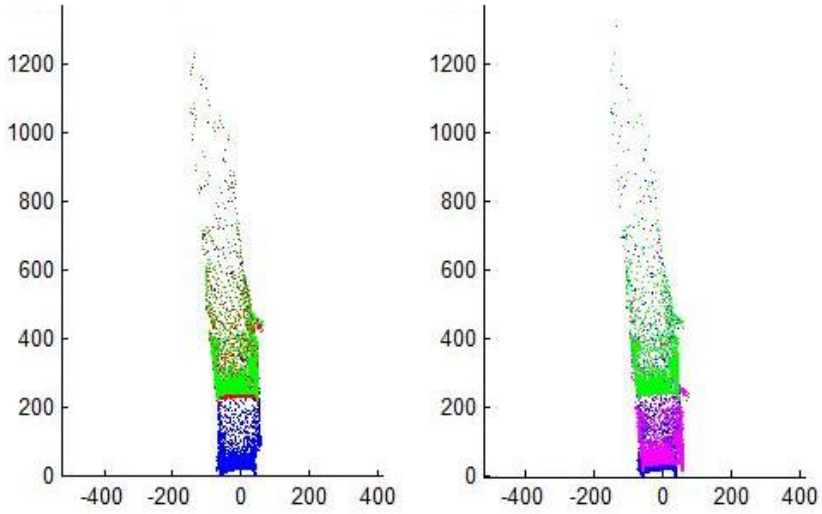


Figure IV.6: Registration of Tunnel Scans using Slide Images (left) and ICP (right). Base Scan in blue, Ground Truth in green, Slide Image Registration in red, and ICP Registration in magenta.

environment. The structure and surface of the walls are very similar at any given point, but slight differences can be detected when the overall structure of the tunnel is taken into account and focus is placed on the feature rich regions of the wall surfaces. The slide image algorithm excels at focusing on these feature rich regions by decoupling the surface of the walls from the overall tunnel structure.

The results of each algorithm were compared to the ground truth registration and the mean point error was computed for each scan registration, shown in Figure IV.7, using the distance between each transformed point and is corresponding ground truth position. The mean, standard deviation minimum and maximum of the point errors for each scan using both algorithms are provided in Table IV.1. The delta statistics were calculated using the differences between the slide image mean point error and the ICP mean point error for each individual scan. Negative delta results correspond to scans where the ICP algorithm out preformed the slide image algorithm. The slide image algorithm clearly outperformed the ICP algorithm in all but a single registration with an overall mean point error of 26.4 cm, a result that is over five times less than ICP's mean point error of 143.0 cm. The slide image registration had far more consistent results than the ICP algorithm, thus providing a more reliable registration algorithm in this environment. This result is seen by comparing the Slide Image registration standard deviation, 9.6 cm, to ICP's 58.6 cm standard deviation.

The only registration in which ICP outperforms the slide image registration, scan number 3, is provided in Figure IV.8. This pair of scans depicts an intersection in the tunnel, a feature that the slide image algorithm does not handle well. The slide image requires the detection of the natural axis to be accurate in order to

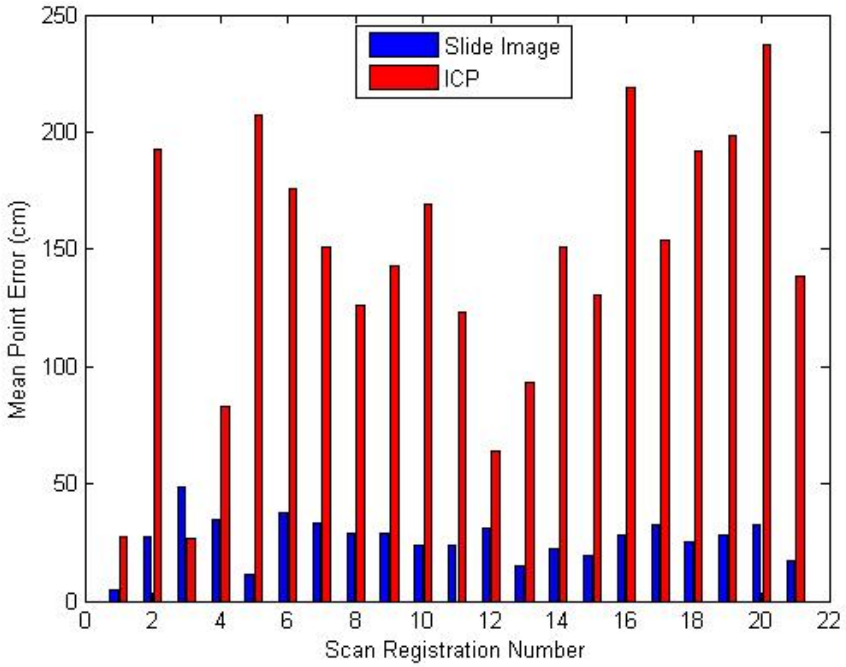


Figure IV.7: Individual Hall Scan Registration Point Error. Slide Image Registration in blue and ICP Registration in red.

Registration Algorithm	Mean (cm)	Std (cm)	Min (cm)	Max (cm)
Slide Image Registration	26.4	9.6	4.8	48.3
Iterative Closest Point	143.0	58.6	27.0	237.4
Delta (ICP - Slide Image)	116.6	59.3	-21.3	204.9

Table IV.1: Slide Image Tunnel Data

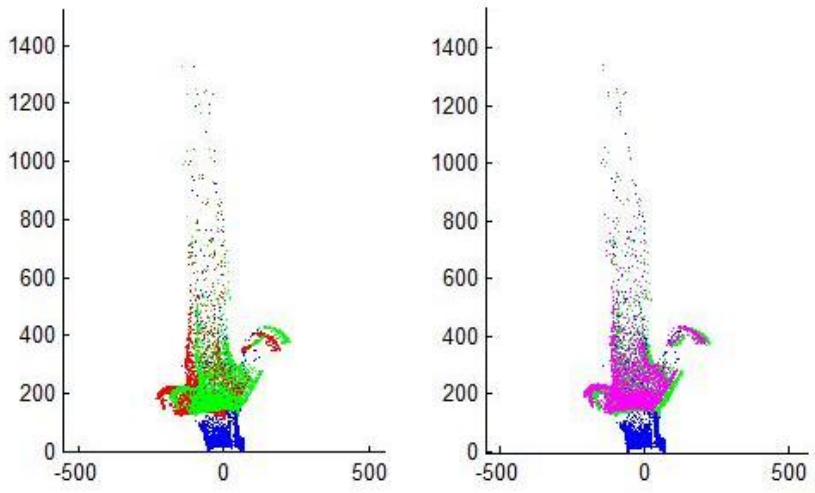


Figure IV.8: Registration of Tunnel Intersection Scans using Slide Images (left) and ICP (right). Base Scan in blue, Ground Truth in green, Slide Image Registration in red, and ICP Registration in magenta.

accurately match scans. In regions with intersections, there exists six potential axes, one for each potential path through the intersection and each with little data to define their path, seen as the six colored arrows shown in Figure IV.9. The assumption is that the robot will use the path straight through the intersection, but the data associated with the other paths are still included in the axis detection process. In spite of this issue, the slide image algorithm registered the scan with a mean error of 48.3 cm, well below the mean point error 143.0 cm for the ICP algorithm.

IV.1.2 Hall Data

The versatility of the slide image algorithm was tested by using scans from an office environment. The scans were generated using an omnidirectional laser scanner in a hallway at the AASS building of Orebro University, Sweden. This environmental dataset was chosen because of its structural similarities to subterranean tunnels. As well, the goal was to test the flexibility of the algorithm. The chosen hall has three major intersections that are not be properly registered by the slide image algorithm, but the remaining regions are registered correctly.

IV.1.2.1 Data Collection

The scans of this environment were generated using an omnidirectional 3D laser scanner, giving each scan a more broad view of its environment than the tunnel environment data, Chapter IV.1.1. A ground truth map was obtained using 3D-NDT and odometry with a total accumulated error of 63 cm (Magnusson, 2009), as

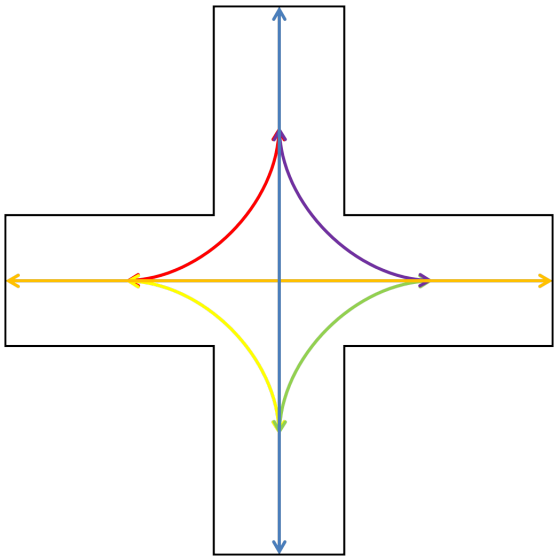


Figure IV.9: Potential axis paths at intersections

Registration Algorithm	Mean (cm)	Std (cm)	Min (cm)	Max (cm)
Slide Image Registration	220.0	331.7	32.7	926.9
Iterative Closest Point	116.0	122.9	4.8	334.6
Delta (ICP - Slide Image)	-104.0	368.1	-883.3	285.2

Table IV.2: Slide Image Hall Data

shown in Figure IV.10.

IV.1.2.2 Results

Similar to the experiment presented in Chapter IV.1.1, the slide image and the ICP algorithms were used to register consecutive scans. The individual mean point errors are shown in Figure IV.11 and an example registration is shown in Figure IV.12. Again, no odometry data was used by either algorithm. The slide image algorithm did not perform as accurately in this environment with a mean point error of 220.0 cm, twice that of the ICP algorithm, 116.0 cm. The results can be seen in Table IV.2. This decrease in performance is due to the intersections visible in the ground truth data. The registration of an intersection is shown in Figure IV.13. The mean point errors of the three registrations associated with these intersections, scan numbers 3, 4 and 9, were an order of magnitude higher than that of the other registrations. The influence of these scans can be clearly seen in the abnormally large standard deviation. The registrations are clearly better for the non-intersection scans. A second experiment using this data was conducted in order to understand the Slide Image algorithm's performance without the intersections.

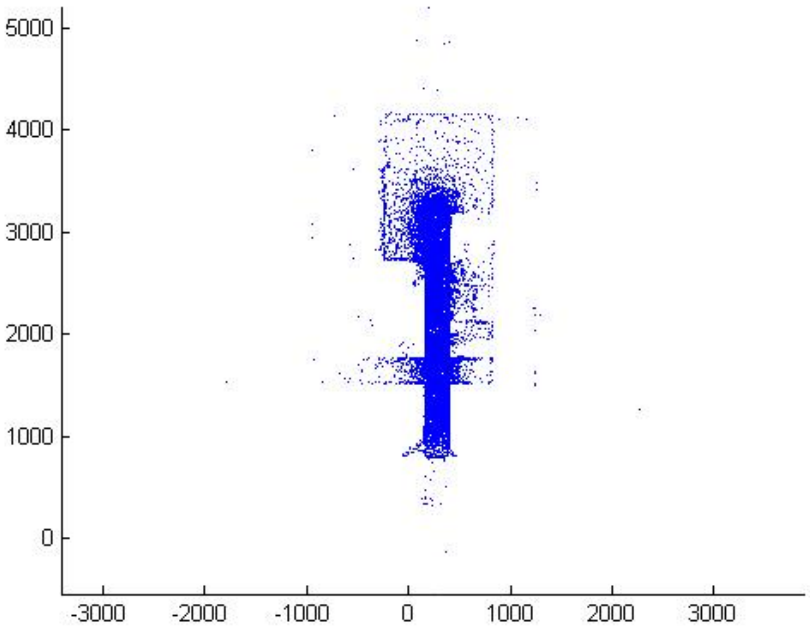


Figure IV.10: Hall Ground Truth

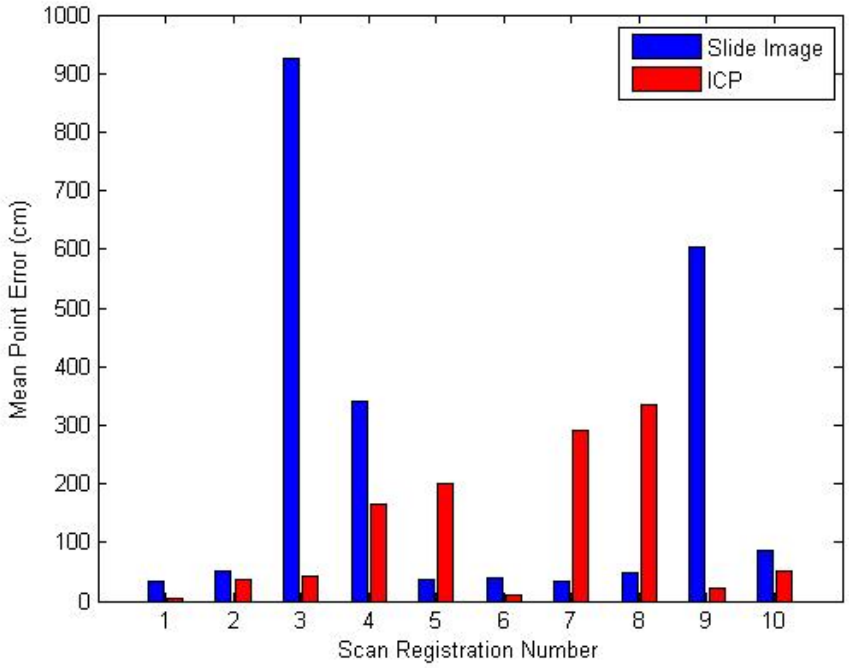


Figure IV.11: Individual Hall Scan Registration Point Error. Slide Image Registration in blue and ICP Registration in red.

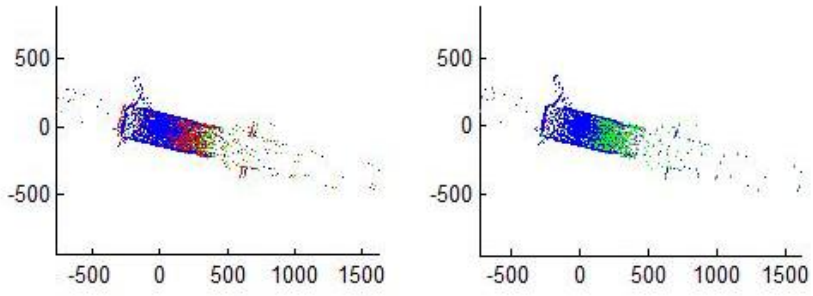


Figure IV.12: Registration of Hall Scans using Slide Images (left) and ICP (right). Base Scan in blue, Ground Truth in green, Slide Image Registration in red, and ICP Registration in magenta (fully occluded by ground truth).

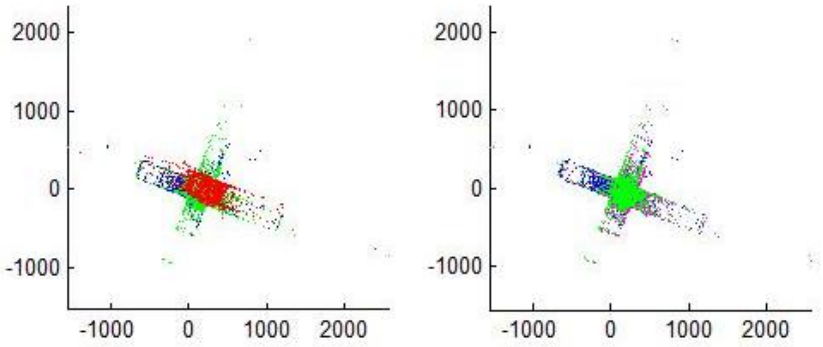


Figure IV.13: Registration of Hall Intersection Scans using Slide Images (left) and ICP (right). Base Scan in blue, Ground Truth in green, Slide Image Registration in red, and ICP Registration in magenta.

Registration Algorithm	Mean (cm)	Std (cm)	Min (cm)	Max (cm)
Slide Image Registration	46.9	19.0	32.7	86.8
Iterative Closest Point	132.7	140.1	4.8	334.6
Delta (ICP - Slide Image)	85.9	145.2	-34.8	285.2

Table IV.3: Slide Image Hall Data w/o Intersections

The second experiment removed the three scan registrations containing large intersections, scan numbers 3, 4 and 9, and new point error statistics were generated. When the scans without the intersections are used, the slide image algorithm out performed the ICP algorithm with a mean point error of 46.9 cm, almost three times lower than ICP's mean point error, 132.7 cm. The results can be seen in Table IV.3. The slide image registration also provides more consistent results than the ICP algorithm, with a standard deviation of 19 cm, almost 10 times less than that of the ICP algorithm, 140.1 cm.

The increase in mean point error associated with intersections in this data set over the tunnel data set is largely due to the difference in sensor configurations. The tunnel laser scanned directly forward, only catching a glimpse of the intersections as it passed, whereas the laser rangefinder used to collect the office environment data scanned in all directions, allowing it to obtain a full view down each intersection.

IV.1.3 Discussion

The Slide Image algorithm clearly outperformed the ICP algorithm in all non-intersection scan registrations. The Slide Image algorithm has been shown to be flexible enough to handle multiple tunnel environments but excels at registering environments with few intersections, such as smuggling tunnels. The Slide Image algorithm was able to avoid local minima inherent in tunnel environments, but was unable to properly handle intersections. The Slide Image algorithm's registration accuracy was limited by the natural axis detection accuracy, registering scans with poorly detected natural axes less accurately. The ICP algorithm falls into local minima quite easily, but the extra structural complexities of intersections allow ICP to register the scans containing intersections more accurately than the Slide Image algorithm. The fusion of these algorithms should be more accurate than either algorithm independently.

IV.2 Fusion Algorithm

The Fusion algorithm, described in Chapter III.3, was tested against the original Slide Image algorithm, described in Chapter III.2, and the ICP algorithm. The same environmental data sets from the prior experiments were used to validate the accuracy of the fusion algorithm. As with the previous experiments, no odometry data or initial transforms were used to register the point clouds.

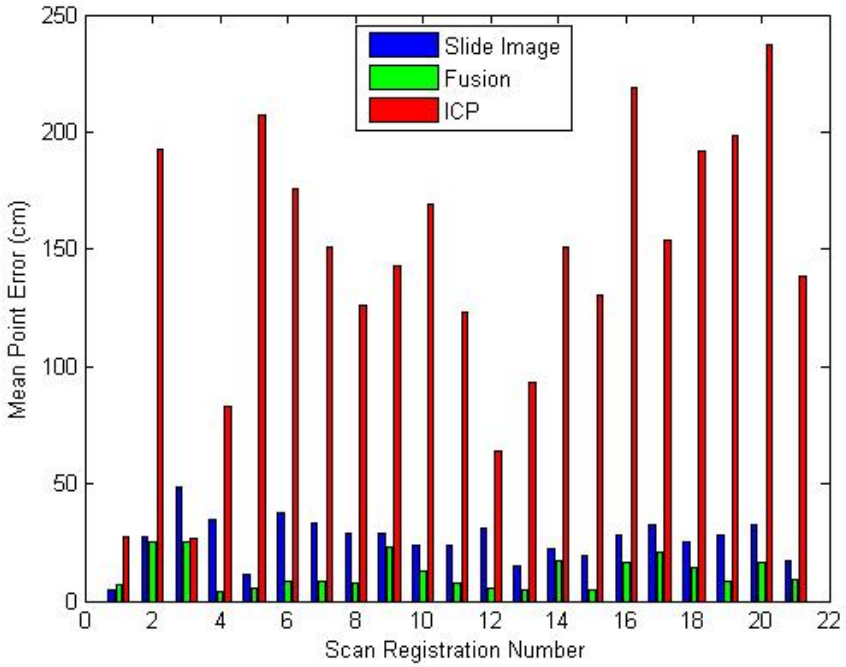


Figure IV.14: Individual Tunnel Scan Registration Point Error. Slide Image Registration in blue, Fusion Algorithm in green and ICP Registration in red.

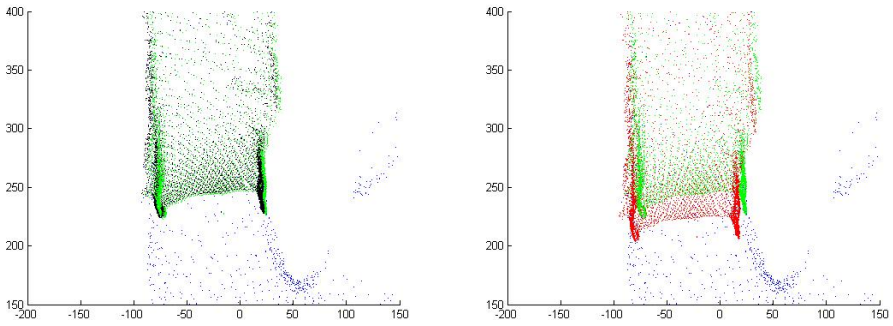
IV.2.1 Tunnel Results

The Fusion algorithm was tested using the tunnel data set and registered all but one scan more accurately than the Slide Image Algorithm. The difference between the Fusion algorithm's and the Slide Image algorithm's mean point error was 2.2 cm, a negligible difference. The Fusion algorithm was more accurate than the ICP algorithm in all but a single scan, which had a difference of 7.6 cm, a distance so small, with respect to the 12 m long tunnel scans being registered, that it may be due to slight inaccuracies in the manual registration of the ground truth. The results are shown in Figure IV.14 and the point error statistics can be seen in Table IV.4.

The Fusion algorithm successfully improved the registration of the Slide Image algorithm by fixing slight variances caused by inaccuracies in the natural axis, which can clearly be seen in the 14.4 point improvement in the Fusion algorithm over the Slide Image algorithm. One such occurrence can be seen in Figure IV.15, where the Slide Image algorithm was able to determine a rough alignment, but the deviations in the detection of the natural axis caused the transform extracted to be slightly shifted. The secondary ICP set in the Fusion algorithm was able to correct this error and generate a highly accurate registration.

Registration Algorithm	Mean (cm)	Std (cm)	Min (cm)	Max (cm)
Fusion Registration	12.1	7.1	4.2	25.5
Slide Image Registration	26.4	9.6	4.8	48.3
Iterative Closest Point	143.0	58.6	27.0	237.4
Delta (ICP - Fusion)	131.0	57.7	1.5	221.2
Delta (Slide Image - Fusion)	14.4	9.0	-2.2	30.3

Table IV.4: Fusion Tunnel Data



(a) (b)

Figure IV.15: Registration of Tunnel Scans using Fusion Agloritihm (a) and Slide Images (b). Base Scan in blue, Ground Truth in green, Fusion Algorithm in black and Slide Image Registration in red.

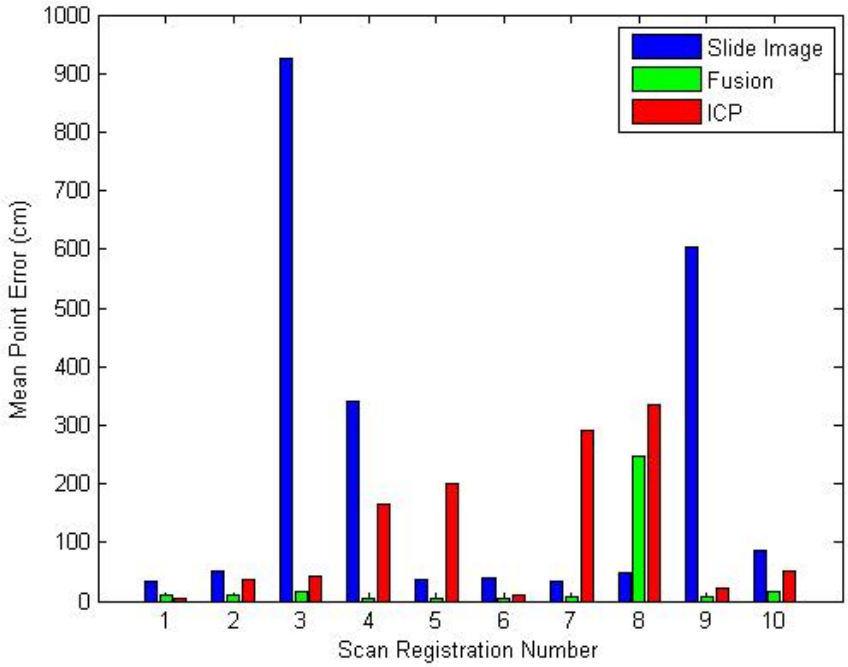


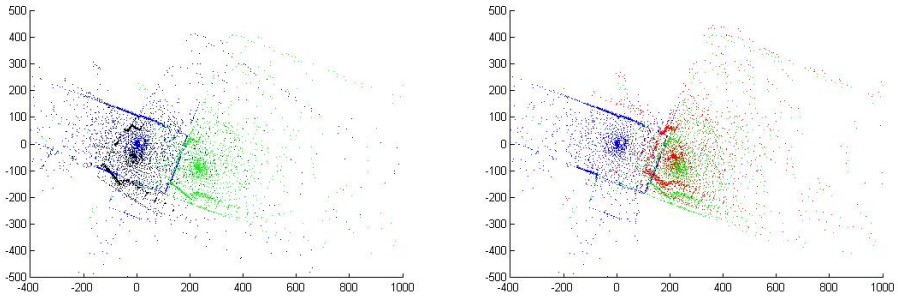
Figure IV.16: Individual Hall Scan Registration Point Error. Slide Image Registration in blue, Fusion Algorithm in green and ICP Registration in red.

IV.2.2 Hall Results

The results of the Fusion Algorithm on the Hall environment can be seen in Figure IV.16 and the point error statistics are available in Table IV.5. The Fusion Algorithm successfully improved on the Slide Image and ICP algorithms in all but one scan registration, scan registration 8. Scan registration 8, shown in Figure IV.17, is centered in a door frame leading into a large open room. The door frame and the open room caused the slide image portion of the fusion algorithm to match poorly. This position also created little overlap causing the ICP algorithm to be unable to find a proper matching and decreased the accuracy of the original slide image matching. In all other cases, the fusion algorithm improved the results of the scan registration and, even with the inclusion of the scan registration 8, boasted a 186.7 cm improvement in mean point error over the Slide Image algorithms and a 82.7 cm improvement over the ICP algorithm. A successful scan matching can be seen in Figure IV.18.

IV.2.3 Discussion

The results of the experiments show a clear improvement in scan registration accuracy for the Fusion algorithm over both the Slide Image algorithm and the ICP algorithm. The Fusion algorithm registered the scans containing intersections more accurately in both environments. The use of the slide image transform to seed

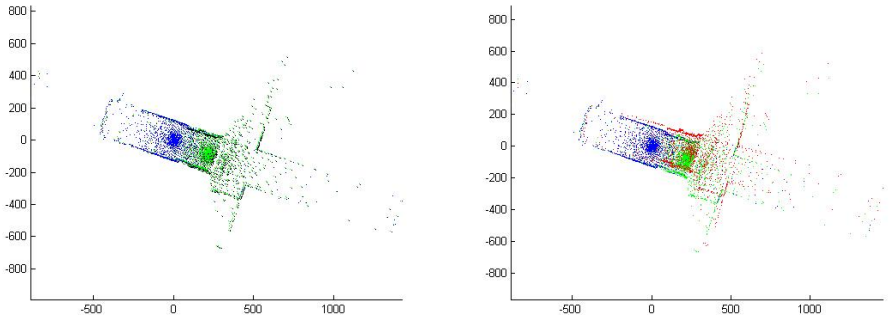


(a) (b)

Figure IV.17: Bad Registration of Hall Scans using Fusion Agloritihm (a) and Slide Images (b). Base Scan in blue, Ground Truth in green, Fusion Algorithm in black and Slide Image Registration in red.

Registration Algorithm	Mean (cm)	Std (cm)	Min (cm)	Max (cm)
Fusion Registration	33.3	75.4	3.1	247.5
Slide Image Registration	220.0	331.7	32.7	926.9
Iterative Closest Point	116.0	122.9	4.8	334.6
Delta (ICP - Fusion)	82.7	98.0	-6.9	283.0
Delta (Slide Image - Fusion)	186.7	333.1	-198.0	911.1

Table IV.5: Fusion Hall Data



(a) (b)

Figure IV.18: Registration of Hall Scans using Fusion Agloritihm (a) and Slide Images (b). Base Scan in blue, Ground Truth in green, Fusion Algorithm in black and Slide Image Registration in red.

the ICP algorithm improved the accuracy of all the scan registrations.

IV.3 Summary

The Slide Image Algorithm provided more accurate results when compared to the ICP algorithm, with the exception of scans containing intersections. The hall experiments provided evidence of the flexibility of the slide image algorithm to handle environments other than smuggling tunnels. The Fusion algorithm improved on the results of the Slide Image algorithm and the ICP algorithm by leveraging the strengths of both algorithms. These algorithms can be used, in concert with a SLAM system, to generate accurate maps of smuggling tunnels.

CHAPTER V

Conclusion

This thesis examined methods of autonomously mapping the smuggling tunnels that run across the US-Mexican border. The major contribution of this thesis is the development of two novel scan registration algorithms. The Slide Image algorithm was tailored to the simplified structure of tunnels and the Fusion algorithm was designed to correct potential registration errors caused by inaccurate natural axis detection and intersections by fusing the Slide Image algorithm with the Iterative Closest Point (ICP) algorithm.

The Slide Image algorithm accounts for the slight variations in the tunnel walls, as well as the overall geometric structure of the tunnel in order to provide a more sensitive and accurate registration when compared to the standard ICP algorithm. The accuracy of the Slide Image algorithm was validated using two distinct environments, an actual smuggling tunnel and an office hallway. The Slide Image Algorithm proved superior to the ICP algorithm in the tunnel and hall environment with the exception of intersections. The Slide Image algorithm does not require odometry and is not susceptible to local minima making it an effective means of mapping the nonintersecting regions of smuggling tunnels.

The Fusion algorithm is able to leverage the power of the Slide Image algorithm for the mapping nonintersecting regions of the smuggling tunnels, while providing the flexibility and fine grain accuracy of the ICP algorithm in the more structurally complex intersection regions. This combination of power and flexibility is achieved by using the Slide Image algorithm to generate an initial transform of the point cloud that is used to find the overlapping regions of the data. With this information, the ICP algorithm is able to accurately correct this initial transform and prevent intersections from causing large errors in the registration. The power and flexibility of the fusion algorithm was evaluated using the tunnel and hall data environments and proved to be more accurate than the Slide Image and the ICP algorithms, making it the ideal candidate for smuggling tunnel mapping.

V.1 Future work

A major weakness of both the Slide Image and the Fusion algorithms is the natural axis detection. Future work is required to improve the quality of the natural axis detection through more complex methods, such as probabilistic wall detection or Hough transforms.

A potential method for generating more accurate natural axes is to use features, such as wires or ventilation shafts, to define the location of a natural axis about which the data can be centered. These features will need to be regularly visible in all smuggling tunnels for this method to be viable. If these features are

not readily available in all environments, artificial features, such as reflective fiducial markers or paint can be used to define a natural axis. The fiducial markers or fiducial paint will be detectable in the laser scans as a high intensity return and the positions of these markers can be used to define the natural axis. A fiducial deployment system will require custom hardware to be place on the robot, which may not be ideal for this application.

The scan accuracy can potentially be improved by combining the laser scan point clouds with color data from a camera system or the intensity data already available from the laser scanner. For this method to be effective, the environment will need to contain several unique colors or reflective intensities to generate a large improvement. The addition of these systems requires an additional registration step to map the color data to the laser point cloud, which may add error to the registration if not done correctly.

Finally, the Fusion algorithm can be improved through the use of 3D-NDT, instead of ICP in the registration correction step. The 3D-NDT algorithm may be more effective in refining the initial transform produced in the Fusion algorithm.

BIBLIOGRAPHY

Arun, K. S., Huang, T. S., and Blostein, S. D. (1987). Least-squares fitting of two 3-d point sets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(5):698–700.

Biber, P. and Strasser, W. (2003). The normal distributions transform: a new approach to laser scan matching. In *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 3, pages 2743 – 2748 vol.3.

Bosse, M. and Zlot, R. (2009). Continuous 3d scan-matching with a spinning 2d laser. pages 4312 –4319.

Bradley, D., Silver, D., and Thayer, S. (2004). A regional point descriptor for global localization in subterranean environments. In *IEEE conference on Robotics Automation and Mechatronics (RAM 2005)*, volume 1, pages 440 – 445.

Choi, Y., Lee, T., and Oh, S. (2008). A line feature based slam with low grade range sensors using geometric constraints and active exploration for mobile robot. *Auton. Robots*, 24:13–27.

Cole, D. and Newman, P. (2006). Using laser range data for 3d slam in outdoor environments. pages 1556 –1563.

Doucet, A., Freitas, N., Murphy, K., and Russell, S. (2000). Rao-blackwellised particle filtering for dynamic bayesian networks. pages 176–183.

Druon, S., Aldon, M., and Crosnier, A. (2006). Color constrained icp for registration of large unstructured 3d color data sets. pages 249 –255.

Ferguson, D., Morris, A., Hahnel, D., Baker, C., Omohundro, Z., Reverte, C., Thayer, S., Whittaker, W., Whittaker, W., Burgard, W., and Thrun, S. (2003). An autonomous robotic system for mapping abandoned mines.

Godin, G., Laurendeau, D., and Bergevin, R. (2001). A method for the registration of attributed range images. *3D Digital Imaging and Modeling, International Conference on*, 0:179.

Godin, G., Rioux, M., and Baribeau, R. (1994). Three-dimensional registration using range and intensity information. *Videometrics III*, 2350(1):279–290.

Huber, D., and Vandapel, N. (2003). Automatic 3d underground mine mapping. *International Journal of Robotic Research*.

Johnson, A. and Hebert, M. (1999). Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(1):433 – 449.

Jost, T., Schutz, C., and Hugli, H. (1998). Modeling 3d textured objects by fusion of multiple views. *European Signal Processing Conference*.

Magnusson, M. (2009). *The Three-Dimensional Normal-Distributions Transform — an Efficient Representation for Registration, Surface Analysis, and Loop Detection*. PhD thesis, Orebro University. Orebro Studies in Technology 36.

Magnusson, M., Lilienthal, A., and Duckett, T. (2007). Scan registration for autonomous mining vehicles using 3d-ndt: Research articles. *J. Field Robot.*, 24:803–827.

Mehlhorn, K. and Näher, S. (1999). *LEDA: a platform for combinatorial and geometric computing*. Cambridge University Press, New York, NY, USA.

Montemerlo, M., Thrun, S., Koller, D., and Wegbreit, B. (2002). Fastslam: A factored solution to the simultaneous localization and mapping problem. pages 593–598.

Muhlbauer, Q., Kuhnlenz, K., and Buss, M. (2008). Fusing laser and vision data with a genetic icp algorithm. *Proceedings of the 10th International Conference on Control, Automation, Robotics and Vision*.

Nuchter, A., Surmann, H., Lingemann, K., Hertzberg, J., and Thrun, S. (2004). 6d slam with an application in autonomous mine mapping. 2:1998 – 2003 Vol.2.

Ohno, K. and Tadokoro, S. (2005). Dense 3d map building based on lrf data and color image fusion. pages 2792 – 2797.

Quivira, F., Fassbender, K., Martinez-Lorenzo, J., and Rappaport, C. (2010). Feasibility of tunnel detection under rough ground surfaces using underground focusing spotlight synthetic aperture radar. In *Technologies for Homeland Security (HST), 2010 IEEE International Conference on*, pages 357 –362.

Rasmussen, C. (2002). Combining laser range, color, and texture cues for autonomous road following. In *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, volume 4, pages 4320 – 4325 vol.4.

Rusinkiewicz, S. and Levoy, M. (2001). Efficient variants of the icp algorithm. In *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*, pages 145 –152.

Sagawa, R., Osawa, N., Echigo, T., and Yagi, Y. (2005). Real time 3d environment modeling for a mobile robot by aligning range image sequences.

Silver, D., Ferguson, D., Morris, A., and Thayer, S. (2006). Topological exploration of subterranean environments. 23(6):395–415.

Smith, R., Self, M., and Cheeseman, P. (1990). *Estimating uncertain spatial relationships in robotics*, pages 167–193. Springer-Verlag New York, Inc., New York, NY, USA.

Stolarczyk, L. G., Troublefield, R., and Battis, J. (2005). Detection of underground tunnels with a synchronized electromagnetic wave gradiometer. volume 5778, pages 994–1001. SPIE.

Thrun, S., Burgard, W., and Fox, D. (2000). A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, volume 1, pages 321 –328 vol.1.

Thrun, S., Hahnel, D., Ferguson, D., Montemerlo, M., Triebel, R., Burgard, W., Baker, C., Omohundro, Z., Thayer, S., and Whittaker, W. (2003). A system for volumetric robotic mapping of abandoned mines. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA*, pages 4270–4275.

Thrun, S., Thayer, S., Whittaker, S., Baker, C., Burgard, W., Ferguson, D., Hahnel, D., Montemerlo, M., Morris, A., Omohundro, Z., Reverte, C., and Whittaker, W. (2004). Autonomous exploration and mapping of abandoned mines. *IEEE ROBOTICS AND AUTOMATION MAGAZINE*, 11:2005.

Viejo, D. and Cazorla, M. (2007). 3d plane-based egomotion for slam on semi-structured environment. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*.

Weingarten, J. and Siegwart, R. (2006). 3d slam using planar segments. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 3062 –3067.

Zhang, Z. (1994). Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13:119–152. 10.1007/BF01427149.