

A New Approach to 3-D Terrain Mapping

Daniel F. Huber and Martial Hebert

[dhuber, hebert]@ri.cmu.edu

The Robotics Institute, Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA 15213

Abstract

We discuss the problem of building large, high-resolution three-dimensional representations of unstructured terrain using terrestrial range sensors, which operate at the scale of meters to hundreds of meters. Issues specific to this sensing modality include widely varying resolution, absence of reliably detectable features, and very large data sets. We have developed a map building algorithm that registers and integrates sequences of range images, and we demonstrate its capabilities by building large terrain maps (260 x 166 meters) using ground-based and low-altitude terrestrial range sensors.

1 Introduction

In this paper, we discuss the problem of building large, high-resolution three-dimensional (3-D) representations of unstructured terrain using terrestrial range sensors. Terrestrial sensors operate over a range of meters to hundreds of meters and are distinguished from object modeling sensors, which operate in a range of millimeters to meters, and remote sensors, which operate at kilometer range and higher. Terrestrial range sensors can be ground-based or airborne and characteristically produce data with a large depth of field and variable resolution.

When modeling unstructured environments, it is necessary to address the issues unique to terrestrial range sensing, including widely varying resolution, absence of reliable features, and very large data sets. We have developed a map building algorithm that illustrates the importance of these terrain-specific issues, and we demonstrate the effectiveness of our algorithm at addressing these problems by building large (260 x 166 meters) terrain models from ground-based and low-altitude range sensor data.

Three dimensional terrain modeling supports two applications relevant to robotic systems: map building and localization. With map building, a series of local maps obtained from different viewpoints is combined into a consistent global map. This typically involves two steps: *map registration*, in which the rigid-body transforms between overlapping pairs of maps are determined, followed by *integration*, in which the registered maps are merged. With localization, position is determined by constructing a map from locally sensed terrain and identifying the position of this local map relative to a pre-existing global map.

In this paper, we concentrate on the map building application, but we note that our algorithms apply to the localization application as well.

The registration component of our map building algorithm is built upon an earlier surface matching system [6], which provides three significant capabilities not found together in any previous terrain modeling algorithm. First, we have no requirement for an initial approximation of the six degree of freedom (6-DOF) transform between views or knowledge of the sensor's orientation. Second, there is no need to detect explicit features in the environment because we rely on local shape signatures over the entire sensed surface. Finally, it is unnecessary to reduce the sensed data to the more limited elevation map representation.

Several other researchers have focused on the terrain mapping problem; see [13] for an extensive survey. Gennery computed the 3-DOF (x, y, and z) registration between ground-based and aerial terrain data using a coarse-to-fine search to minimize elevation differences [3]. Horn and Harris developed the elevation rate constraint equation, based on spatio-temporal derivatives, for registering elevation maps with 3-DOF (x, y, and z) assuming small motion between data sets [5]. Kweon and Kanade combined feature-based matching of high curvature points and lines with a coarse-to-fine pixel-based gradient descent search to register elevation maps [10]. Kamgar-Parsi et. al. used feature-based matching of contours of constant range to register ocean floor range images with 3-DOF (x, y, and θ) [9]. Zhang used the iterative closest point (ICP) algorithm to register views of a rock scene given an initial estimate of the transform [13]. Olson and Matthies used a coarse-to-fine search based on the Hausdorff distance to register elevation maps with 3-DOF (x, y, and z) [12].

The remainder of the paper is organized as follows: Section 2 provides an overview of the map building algorithm and provides a brief registration example; Section 3 discusses the issues specific to terrain mapping using terrestrial sensors; Section 4 describes face-based spin images, an extension which allows resolution independent registration; Section 5 explains our range shadow removal algorithm and the mesh generation procedure; and Section 6 presents our experimental setup and mapping results. Throughout the paper, we illustrate our ideas by showing aspects of our algorithm operating on two data sets: the mesa data set, a ground-based sequence of views along a

road passing down a curving valley between two hills, and the heli data set, aerial data along the border of a cliff.¹

2 Overview of the algorithm

Our map building algorithm follows the two-step process described in the introduction: map registration followed by integration. The registration component receives as input a pair of arbitrary polygonal surface meshes, possibly containing holes and disconnected patches, and outputs the 6-DOF rigid-body transform that best aligns the two meshes. In our experiments, we create meshes from range data obtained from two sensors, one ground-based and one aerial. The Ben Franklin 2 (BF2), a scanning laser range finder mounted on Navlab 2 and Navlab 5, two of Carnegie Mellon University's (CMU's) autonomous ground vehicles, produces 360 degree by 30 degree field of view range and reflectance images in a radius of 52 meters (Figure 1) [4]. The second sensor, a single line laser range finder mounted on one of CMU's autonomous helicopters, produces lines of range data in a sweeping pattern rather than as an image [11]. The different sensors require different processing to convert the data into surface meshes, but the algorithm operates identically otherwise.

To build a global map from a sequence of unregistered range images, we perform map registration on each sequential pair of maps as follows:

1. Convert the two range images into triangular surface meshes.
2. Identify corresponding points in the two meshes.
3. Calculate the rigid-body transform that best aligns the corresponding points.
4. Refine and verify the estimate using the iterative closest point (ICP) algorithm [1][13].

Figure 2 shows the result of the registration of a pair of views obtained with the BF2. Finally, the maps are transformed into a common coordinate system and merged using a voxel-based surface integration algorithm [7].

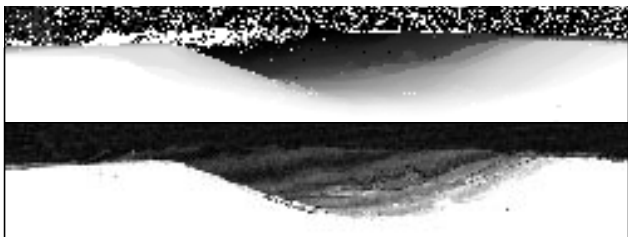


Figure 1: Example range (top) and reflectance (bottom) images acquired with the BF2 laser range finder.

1. Color versions of the figures in this paper can be viewed at www.cs.cmu.edu/~3dvision/mapping/iros99/.

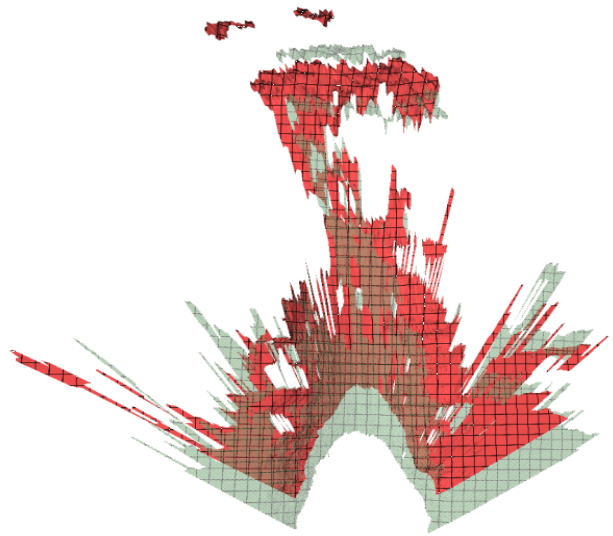


Figure 2: Top view of the successful registration of the first two data sets from the mesa data sequence, between which the vehicle travelled 3 meters. The lower mesh corresponds to the view in Figure 1.

With surface matching algorithms, finding corresponding points is the most difficult step due to the combinatorics of the problem. Our approach uses the spin-image representation introduced by Johnson [7]. Spin-images represent the local mesh shape by projecting surface points into a two dimensional coordinate system relative to a given oriented point² (Figure 3). Spin-images can then be rapidly compared using a similarity measure defined as a monotonically increasing function of spin-image correlation [7].

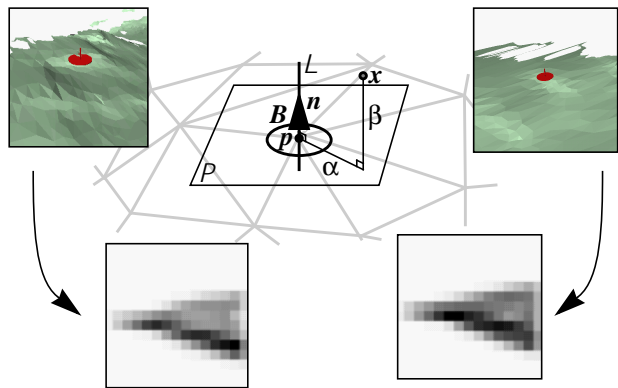


Figure 3: The spin-image for point p is found by recording the distance of all nearby points x from the surface normal n (α) and the distance from x to p along n (β). Corresponding points from different views (top left and right) have similar spin-images (bottom left and right).

2. An oriented point is specified by a 3-D location and an orientation and typically corresponds to a surface point and its normal.

3 Issues specific to terrain modeling with terrestrial sensors

We identify five issues that must be addressed when modeling terrain with terrestrial sensors: lack of features, arbitrary unknown transforms between views, large data sets, varying resolution, and range shadows. Most of these arise from the fact that terrestrial sensing platforms are situated within the environment rather than positioned to view it from afar. *Lack of features* – At the scale of terrestrial sensors, most natural terrain is nondescript and contains few, if any, reliably detectable shape-based features. *Unknown transform* – The general case registration problem of determining the 6-DOF transform between views arises frequently in terrain modeling in situations where localization aids such as GPS are unavailable or unreliable. *Large data sets* – The high sampling rate of laser scanners, the need to represent 3-D terrain with high accuracy, and the integration of many individual maps all ensure that large data sets are required at many stages of processing. Varying resolution and range shadows are discussed in detail in Sections 4 and 5 respectively.

Although there has been much work in the remote sensing community on building topographic terrain maps, the design of remote sensing systems allows the terrestrial sensing issues to be avoided. The location of the sensor ensures that the data will have approximately uniform resolution, and the predictable dynamics of the platform (satellite or airplane) provide a good estimate of the transform between views. Initial correspondences are chosen manually, and range shadows are not addressed.

Object modeling systems fare better at addressing terrestrial sensor problems but still fall short. Many systems use a calibrated turntable to measure the view transform. Resolution is fairly uniform due to the relatively large distance from which objects are viewed. The objects being modeled tend to have many distinctive surface features and can be represented with a small number of data sets. Range shadows frequently occur and are dealt with explicitly.

The surface matching engine upon which our mapping algorithm is based already addresses some of the issues specific to terrain modeling, with the most notable exception being the inability to handle varying resolution surface meshes. Our algorithm solves this problem through the use of face-based spin-images.

4 Face-based spin-images

The problem of varying resolution in range images is most apparent with ground-based, forward-looking sensors, where the terrain is viewed from a shallow angle. In such cases, variation on the order of 100 to 1 is not uncommon.

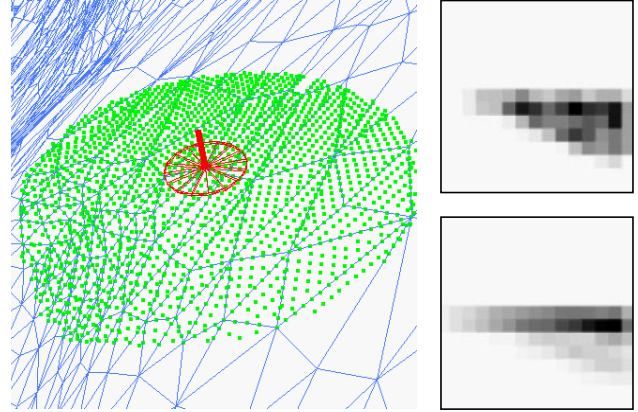


Figure 4: Points sampled in the creation of a face-based spin-image (left). The vertex-based spin-image for this point (top right) is degraded when compared with the corresponding face-based spin-image (bottom right).

Consequently, the registration process frequently compares surfaces of different resolutions.

In our algorithm, the effects of resolution variation appear during the formation of spin-images. When computing the spin-image for an oriented point, the projections of all *vertices* within a fixed region of influence are accumulated in a 2-D histogram which forms the pixels of the spin-image. If the mesh resolution of a region is too low for the chosen spin-image size, each histogram bin will accumulate too few data points, and coincidental location of vertices will degrade the spin-image (Figure 4).

One answer to this problem is to equalize the resolution using a mesh decimation algorithm [8], but such an approach is not practical for terrain map registration for two reasons. First, the loss of detail incurred in equalizing the resolution is so great that reliable registration is no longer possible. Second, uniform edge length decimation is computationally expensive when compared to unconstrained mesh decimation, and our implementation was unable to process large terrain meshes at a reasonable speed.

Our solution is to compute spin-images for the entire surface rather than just vertices. To compute these “face-based” spin-images, we sub-sample each mesh face by linearly interpolating in a raster-scan pattern. The newly sampled points are projected into spin-image coordinates, and, for each point, a constant factor is added to the corresponding spin-image pixel. In practice, we obtain a more uniform surface coverage by interpolating along the edges as well. Before scanning each mesh face, we use a quick geometric test to determine whether any part of the face could project into the spin-image.

The resulting spin-images represent the actual local surface shape more accurately than vertex-based spin-images,

subject to the limitations of the sampling frequency and the original mesh resolution. To verify this claim, we conducted an experiment comparing the similarity measure for spin-images using each method. We define $I_{i,M,m}$ to be the spin-image generated for vertex i using the vertices of mesh M and sampling method m , where $m=v$ for vertex-based spin-images and $m=f$ for face-based spin-images. Also, $C(P, Q)$ is the similarity measure between spin-images P and Q [7]. Using two registered meshes ($M1, M2$), we compute four spin-images ($I_{i,M1,v}, I_{i,M2,v}, I_{i,M1,f}$ and $I_{i,M2,f}$) for each vertex in $M1$. We then compute the similarity measure at each vertex for each sampling method:

$$C_{i,v,match} = C(I_{i,M1,v}, I_{i,M2,v}), i \in M1 \quad (1)$$

$$C_{i,f,match} = C(I_{i,M1,f}, I_{i,M2,f}), i \in M1 \quad (2)$$

We also compute the similarity measures for pairs of non-matching points:

$$C_{i,v,non-match} = C(I_{i,M1,v}, I_{j,M2,v}), i \neq j, i, j \in M1 \quad (3)$$

$$C_{i,f,non-match} = C(I_{i,M1,f}, I_{j,M2,f}), i \neq j, i, j \in M1 \quad (4)$$

Finally, we calculate the change in similarity for matching and non-matching points:

$$\Delta_{i,match} = C_{i,f,match} - C_{i,v,match}, i \in M1 \quad (5)$$

$$\Delta_{i,non-match} = C_{i,f,non-match} - C_{i,v,non-match}, i \in M1 \quad (6)$$

In Figure 5, the histograms of these quantities show that the similarity of matching points increases for the face-

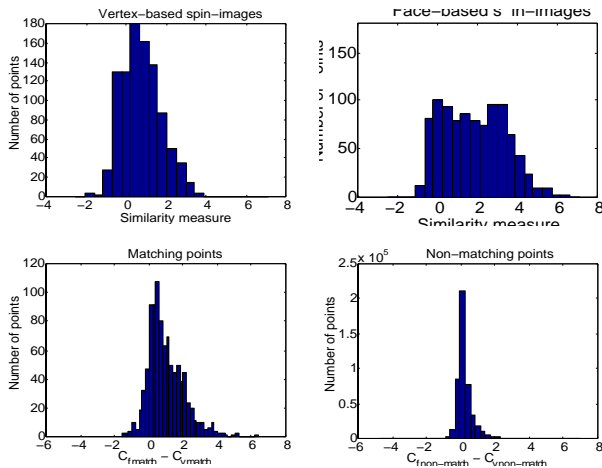


Figure 5: Histograms of the similarity measure for corresponding points using vertex-based spin-images (top left) and face-based spin-images (top right). The bottom row shows the pointwise change in similarity for matching (bottom left) and non-matching points (bottom right). Matching points increase more than non-matching points (mean difference 1.0 vs. 0.23, median difference 0.78 vs. 0.10)

based spin-images while remaining relatively unchanged for non-matching points. These results indicate that face-based spin-images indeed lead to improved matching of corresponding points.

Face-based spin-images have an additional advantage because spin-images are now defined in a resolution independent manner. This advancement has a number of implications for all spin-image matching systems. First, we have removed a constraint on the types of surface meshes for which spin-images can be computed, thereby generalizing spin-image representation. We can now replace the existing edge length equalization algorithm with a fast, shape-preserving mesh simplification algorithm [2]. Such an algorithm automatically adjusts the mesh resolution to fit the complexity of the local shape, representing flat areas at a much lower resolution than highly detailed areas. Second, *any* surfaces of widely different resolutions can now be compared with one another (not just terrain maps). Finally, surfaces can be stored and compared at the resolution that minimally represents them, reducing the required storage space and computation time for matching.

5 Creating surface meshes from range data

The conversion of range data into a representation suitable for registration is an essential step for ground-based sensors. Most previous systems create elevation maps, implicitly assuming the terrain is a strict function of elevation [3][5][10][12]. These systems introduce two limitations. First, elevation maps cannot represent certain classes of 3-D surfaces, including overhangs and vertical edges. Second, the sensor must provide a measurement of the vertical direction. Our approach is to translate range data into a triangular surface mesh, a representation which does not suffer from these drawbacks.

The basic range image conversion process is straightforward: pixels in the image are transformed to xyz coordinates and triangulated by introducing edges between each pixel and its 4-connected neighbors and one of its diagonal neighbors. However, the data must be processed to remove invalid data points, sensor noise, and range shadows. Otherwise, the resulting mesh will be a poor representation of the terrain, and registration will not be possible. To this end, we have implemented a series of filters. The most interesting of these is the range shadow filter. Other filters include thresholds on the range and reflectance, a median filter that removes points significantly different from the local median range to filter noise, isolated point removal and isolated hole filling.

Range shadow removal is a key step in producing accurate terrain maps. These artifacts are introduced into a mesh at

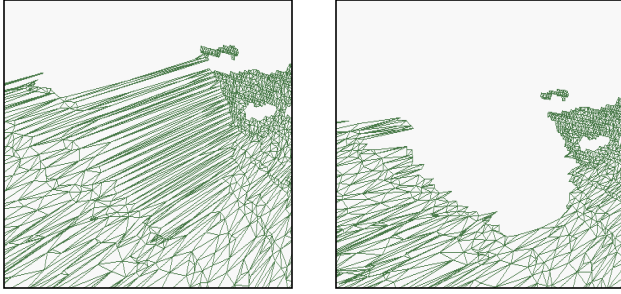


Figure 6: Range shadows occur at occluding boundaries in the surface mesh (left). The same surface with the range shadow removed by our algorithm.

the borders of occluding surfaces (Figure 6). Without their removal, range shadows generate large phantom surfaces in the final map that have no physical meaning. At the other extreme, overzealous range shadow removal may also eliminate significant regions of valid data.

Although our registration algorithm is robust to phantom surfaces created by range shadows, the surfaces can cause problems in the mesh integration stage. On the other hand, range shadow removal can create many small holes in the mesh, reducing the effectiveness of the mesh simplification algorithm, since it must retain many small triangles to accurately represent the boundary shape of these holes. Therefore, we use conservative range shadow removal for creating low-resolution meshes for registration and liberal removal for the high-resolution meshes used in the integration stage.

Range shadows appear as step edges in the range image, but conventional gradient-based edge detection is not applicable due to the varying resolution over the image. Our range shadow removal algorithm is a one-dimensional operator applied independently to each adjacent pair of points in each row and column of the range image (Figure 7).

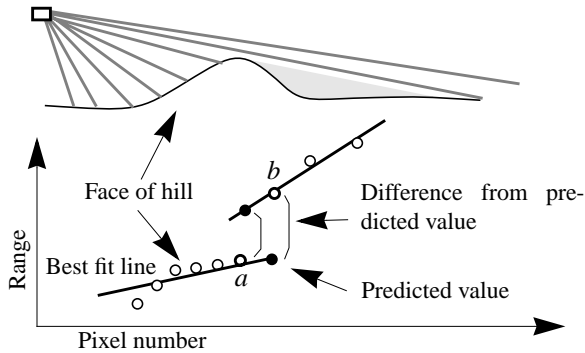


Figure 7: A typical range shadow is generated by a small hill. (top) Illustration of our range shadow detection operator (see text) (bottom).

1. Given adjacent points a and b , fit a line to point a and n points to its left using least squares.
2. Extrapolate the line to predict the range of point b , assuming the surface to the left of b is planar. (This is usually a reasonable approximation for natural terrain over very short distances.)
3. Repeat this approximation and extrapolation with point b and n points to its right.
4. A step edge is detected if the extrapolated values both differ from the measured ranges by more than a given threshold and if the difference changes sign.

This approach correctly detects step edges but ignores other types of edges (e.g., concave and convex edges). It overcomes the changing resolution problem by considering points in a very localized area. On the down side, the algorithm is susceptible to errors in the range values used for the least squares fitting. These errors are a function of the slope of the terrain, making it harder to detect range shadows in surfaces viewed at a shallow angle.

6 Experiments

We have applied our map building algorithm to range data from the BF2 and from the autonomous helicopter. The BF2 was set to generate range and reflectance images of size 6000 x 300, corresponding to angular resolutions of 0.06° horizontally and 0.1° vertically. The initial data is subsampled by a factor of 5 horizontally and 3 vertically and then converted to a mesh using the process described in Section 5 with parameters shown in Table 1.

The helicopter range data was preprocessed by the CMU Autonomous Helicopter group, but we will briefly describe the steps used. The combination of the line scanning sensor, helicopter motion, and position uncertainty causes sequential scan lines to overlap unpredictably. Therefore one cannot simply connect the data points as with the image based scanner. Instead, the points from a series of scans are projected orthographically into a grid, and the range is calculated using the closest point in each bin. Then the bins are treated as a range image and converted to a mesh using the same method as with the BF2.

parameter	mesa set	heli set
range shadow threshold (conservative/liberal)	3m/1m	n/a
low res. mesh size (faces)	2500	5000
high res. mesh size (faces)	15000	11000
spin-image size	5 x 5m	10 x 10m
grid size (face interpolation)	0.3m	1m

Table 1. Parameters used in our map building algorithm for the mesa and heli data sequences.

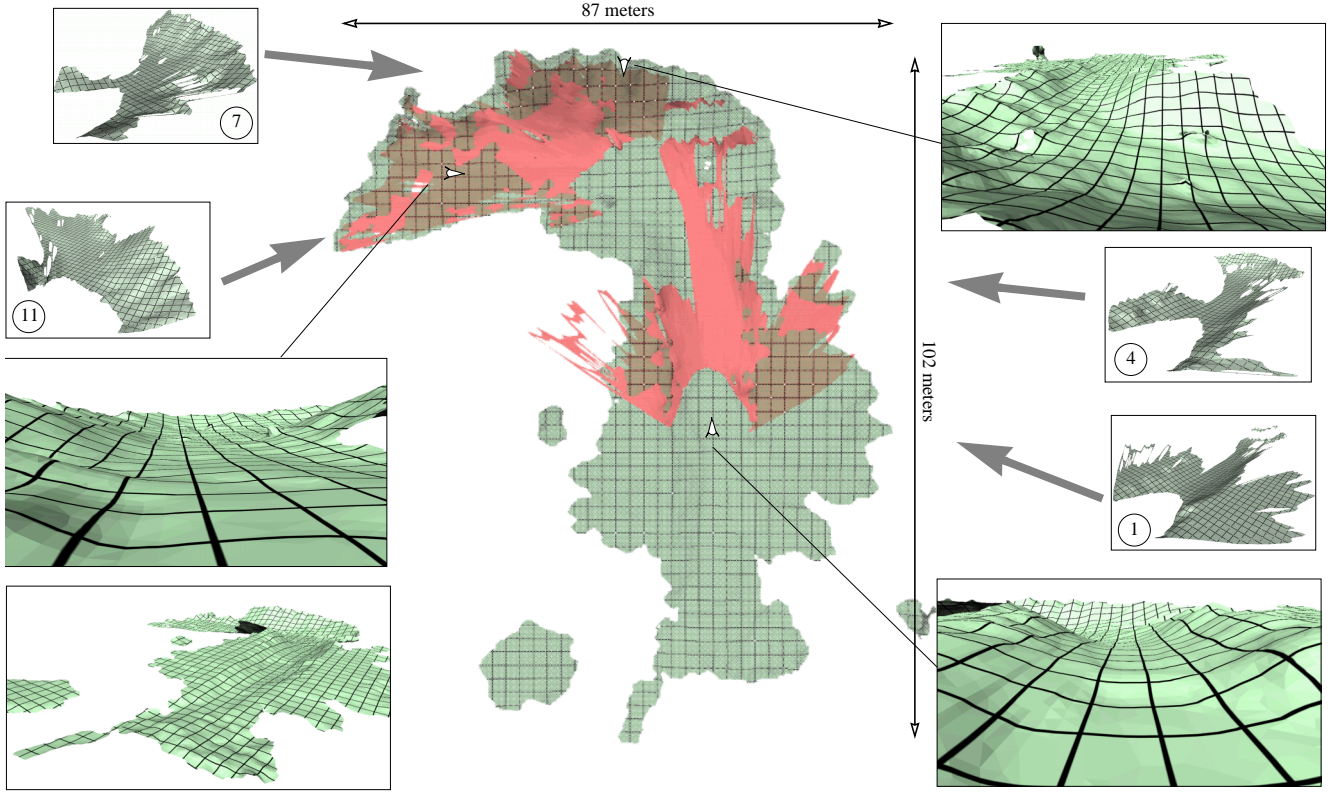


Figure 8: Top view of the integrated terrain map for the eleven data sets in the mesa sequence (center). The numbered insets illustrate individual data sets, and two sets (1 and 11) are overlaid on the top-view. The larger insets show various perspective views of the map, and the white arrows indicate the location and direction of the viewpoint. Grid lines are two-meters apart on the combined map and one meter apart on the individual data sets.

Once a mesh is formed, it is simplified to a predetermined number of faces using Garland's simplification algorithm [2]. A pair of simplified meshes is then presented to the registration engine [6]. Spin-images are generated for each point in one mesh and for a fixed percentage of points in the other. Candidate correspondences are found using the correlation-based similarity metric, and the best correspondences are grouped based on geometric consistency. We then determine the rigid-body transform that best aligns the correspondences from each group, and the candidate transforms are verified and ranked. We select the top match as our registration result. Finally, we use high-resolution versions of the surface meshes to refine our transform estimate using a modified version of the ICP algorithm. After all pairs of meshes in a sequence have been registered, they are integrated into a single global map using a voxel-based method described in [7].

This algorithm has been tested on several large data sets, two of which are shown here. The first data set was acquired with the BF2 mounted on the roof of Navlab 5 at a slag heap near CMU. The data is a sequence of 13 range images obtained at 3-5 meter intervals along a road leading between two hills. Near the 7th data set, the road

curves to the left. The registered data sets were merged into a single, global map shown in Figure 8. Table 2 summarizes the computational requirements for creating this terrain map.

The second data set was collected by the Autonomous Helicopter group during field tests at Haughton crater in the Canadian arctic as part of NASA's Haughton-Mars project. This remote location was chosen due to its similarity to Martian landscape. The integrated terrain map in Figure 9 was formed from three passes of the helicopter along the boundary of a 20-meter cliff and covers a 260 x 166-meter area.

processing stage	time (sec/ data set)	data set size (triangles)
preprocessing	97	3,600,000
create spin images	91	2,500
register	91	2,500
ICP refinement	195	15,000
integrate	152	15,000
final model	n/a	36,000

Table 2. Processing time for creation of the terrain model in Figure 8 using an SGI O2 R5000 computer.

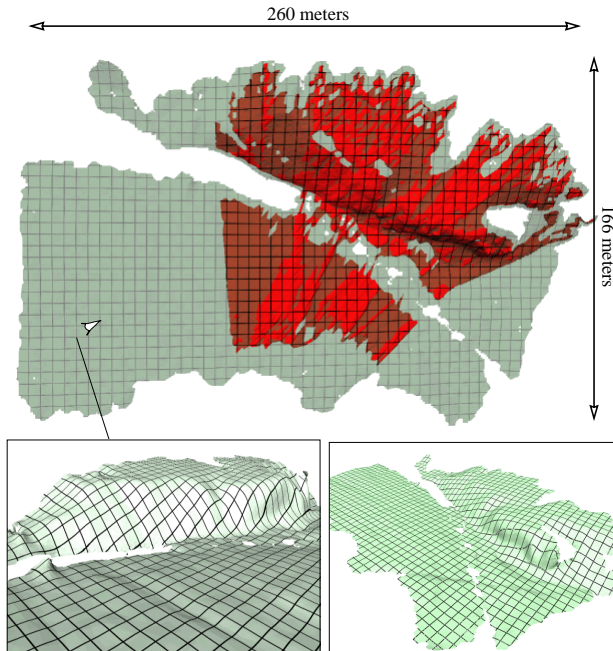


Figure 9: Top-view of the integrated map built using data from the autonomous helicopter (top). The scene shows a cliff (est. 20m high) with a river running along its base. Two perspective views (bottom). Grid lines are 5 meters apart.

7 Future work

Now that we have a reliable algorithm for building maps, we are beginning to analyze the limitations of the approach. Our next step is to investigate how terrain shape affects the algorithm's performance. We are pursuing two areas of investigation: intelligent selection of points and estimation of required overlap.

Intelligent selection of points refers to the method by which mesh points are selected for comparison in the correspondence matching phase of the registration algorithm. Currently, points are selected at random with the goal of evenly distributing the samples over the surface. Although this naive approach works well for object modeling and recognition, where the surfaces abound with interesting features. Unstructured terrain typically contains large, featureless regions, which do not contribute to registration. Thus, another reasonable criterion for point selection would be an estimate of the degree a point will contribute to the registration. By incorporating this additional criterion into the point selection process, we intend to improve the robustness of the current algorithm.

We are also studying the estimation of required overlap between views. We would like to determine, based on a given data set, the minimum overlap necessary to ensure successful registration. By looking at the way the surface points constrain registration for a particular area of over-

lap, we can estimate in advance the probability of successful registration from that view. This capability is a first step to the long-term goal of an integrated planning and terrain modeling system.

Acknowledgments

This research was supported in part by NSF Grant IRI-9711853 and by ONR Grant N00014-95-1-0591. We thank Ryan Miller and John Hancock for their valuable assistance in collecting and preprocessing the laser range finder data.

References

- [1] P. Besl and N. McKay. A method of registration of 3-D shapes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239-256, February, 1992.
- [2] M. Garland and P. Heckbert. Surface simplification using quadric error metrics, in *Proc. SIGGRAPH 97*, pp. 209-16, 1997.
- [3] D. Gennery. Visual terrain matching for a Mars rover, in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR '89)*, pp. 483-91, 1989.
- [4] J. Hancock, D. Langer, M. Hebert, R. Sullivan, D. Ingimarson, E. Hoffman, M. Mettenleitner, C. Froehlich. Active Laser Radar for High Performance Measurements, in *Proc. IEEE International Conference on Robotics and Automation*, pp. 1465-70, 1998.
- [5] B. K. P. Horn and J. G. Harris. Rigid body motion from range image sequences, *CVGIP: Image Understanding*, vol.53, no.1, pp. 1-13, January, 1991.
- [6] A. Johnson and M. Hebert. Surface matching for object recognition in complex three-dimensional scenes, *Image and Vision Computing*, vol.16, no.9-10, p. 635-51
- [7] A. Johnson. Spin-Images: A Representation for 3-D Surface Matching. Ph.D. Thesis, The Robotics Institute, Carnegie Mellon University, August 13th, 1997.
- [8] A. Johnson and M. Hebert. *Control of Polygonal Mesh Resolution for 3-D Computer Vision*, Carnegie Mellon Robotics Institute Technical Report CMU-RI-TR-96-20, February, 1997.
- [9] B. Kamgar-Parsi, J. L. Jones, and A. Rosenfeld. Registration of Multiple Overlapping Range Images: Scenes without Distinctive Features, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no.9, pp. 857-871, September, 1991.
- [10] I. S. Kweon and T. Kanade. High-resolution terrain map from multiple sensor data, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 278-292, February, 1992.
- [11] R. Miller and O. Amidi. 3-D site mapping with the CMU autonomous helicopter, in *Proc. Intl. Conf. on Intelligent Autonomous Systems (IAS-5)*, June, 1998.
- [12] C. Olson and L. Matthies. Maximum likelihood rover localization by matching range maps. In *Proc. IEEE International Conference on Robotics and Automation*, pp. 272-277, 1998.
- [13] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *Int'l Jour. Computer Vision*, vol. 13, no. 2, pp. 119-152, 1994.