Unmanned Ground Vehicle Navigation Using Aerial Ladar Data

Nicolas Vandapel , Raghavendra Rao Donamukkala and Martial Hebert
Carnegie Mellon University
5000 Forbes avenue
Pittsburgh, PA 15213, USA
vandapel@ri.cmu.edu

***Abstract*— In this paper, we investigate the use of overhead high-resolution three-dimensional (3-D) data for enhancing the performances of an Unmanned Ground Vehicle (UGV) in vegetated terrains. Data were collected using an airborne laser and provided prior to the robot mission. Through extensive and exhaustive field testing, we demonstrate the significance of such data in two areas: robot localization and global path planning. Absolute localization is achieved by registering 3-D local ground ladar data with the global 3-D aerial data. The same data is used to compute traversability maps that are used by the path planner. Vegetation is filtered both in the ground data and in the aerial data in order to recover the load bearing surface.**

***Index Terms*— Unmanned Ground Vehicle, terrain registration, localization, path planning, vegetation filtering, overhead data, autonomous navigation**

I. Introduction

Autonomous mobility for Unmanned Ground Vehicles (UGV) in unstructured environment over long distances is still a daunting challenge in many application including space and agriculture. We consider here terrain traversal: safe driving from an origin location to a destination location specified in some coordinates system. We do not consider terrain mapping or exploration where the robot has to build a map of a given area or has to look for some specific area of interest. A robot using on-board sensors only is very unlikely to select an optimal path to traverse a given terrain, facing dead-ends or missing less dangerous areas, out of reach of its sensors. The use of a priori overhead data help to cope with such a problem by providing information to perform global path planning. The robot, using its on-board high resolution sensors, is then left with the detection and avoidance of local obstacles which have eluded the lower resolution aerial sensor. In addition, local information captured by the UGV can be registered with information in the aerial data, providing a way to position absolutely the vehicle. This scenario is relevant for desertic, polar or planetary navigation. In less harsh climates, vegetation will introduce a new set of challenges. Indeed vegetated areas are made of unstable elements, sensitive to seasons, difficult to sense and to model with a good fidelity. Vegetation can obscure hazards such as trenches or rocks, it prevents also the detection of the terrain surface, which is used in all terrain-based localization or to compute a traversability measure of the terrain for path planning.

In this paper, we investigate the use of overhead high-resolution three-dimensional (3-D) data for enhancing the performances of an UGV in vegetated terrains. Data were collected using an airborne laser and provided prior to the robot mission. Through extensive and exhaustive field testing, we demonstrate the significance of such data in two areas: robot localization and global path planning. Absolute localization is achieved by registering 3-D local ground laser data with the global 3-D aerial data. The same data is used to compute traversability maps that are used by the path planner.

Our approach differs from the one presented in [1], [2] where air-ground vehicle collaboration is demonstrated. An unmanned autonomous helicopter ([3]) maps at high resolution the terrain ahead of the ground vehicle path. This overhead perspective allows the detection of negative obstacles and down-slopes, terrain features otherwise extremely difficult to perceive from a ground level point of view.

This article aims at 1) providing a complete and synthetic view of the work we published already in several articles, see [4], [5], [6], 2) enhancing the findings with additional results, and 3) introducing new results.

The article is divided into six sections. Section II presents in details the field tests conducted in 2001-2002, the sensors used and the data collected. Section III focuses on terrain surface recovery. Section IV and V presents two applications of the use of aerial lidar data: 3-D terrain registration for robot localization and traversability map computation for path planning. Finally, in Section VI we conclude our work.

II. Field experimentation

In this section we present the initial terrains and sensors we used to validate our approach during the first phase of the program. Then we present successively the scenario of the field experiments we conducted during the second phase, the different terrains in which we operated, and finally the sensors we used.

A. Phase-I: initial evaluation

During the initial phase of our research program, we tested our terrain registration method and traversability map computation using data from various ground and aerial mapping systems, in several environment setting. The most noticeable experiments were conducted using the high resolution Zoller-Fröhlich (Z+F) LARA 21400 laser [7] and the CMU autonomous helicopter [3]. Two test sites were primarily used: the MARS site and the coal mine site. The MARS site was engineered in order to include repetitive and symmetric structures and it was in addition instrumented and geo-surveyed in order to allow a performance assessment of the registration method. Additional tests were performed in an open coal mine containing larger scale terrain features, over ten meters in height. With this data we performed controlled tests, investigating the influence of the terrain surface resolution, the method's parameters, the rotation and translation constraints to perform (*Air-Ground, Air-Air, Ground-Ground*) terrain registration. Details can be found in [4]. The rest of this section deals with the field experiments we conducted during the phase-II, in 2002.

B. Phase-II: field experiments

The main set of data used for this article was collected on four different sites characteristic of very different types of terrain. Each test site was divided into courses having its own set of characteristic terrain features, natural and occasionally man-made obstacles. The tests were performed during the second phase of the project: in February, in a wooded areas with large concentrations of tree canopies; in March in a desertic rocky area with scattered bushes, trees and cacti, sculpted with gullies and ledges; in August in an area located at more than 3,000 meters in altitude this alpine terrain contains large slope terrains, cluttered by rocks and covered by pines trees; and in November, in a mostly flat terrain, covered by tall grass and woods, traversed by water streams and containing man-made obstacles. Each field test will be referenced respectively as *Wood test*, *Desert test*, *Alpine test* and *Meadow test*.

Each field test was instrumented and geo-surveyed. Several antennas were installed in order to relay the robot status and navigation data to the base camp. After each course test the terrain was scouted and various pieces of information were collected such as DGPS position of salient terrain features or trees, measurement of vegetation height and so forth. Access to such data makes definitively the test performed exhaustive and intensive

Table I contains, for each field experiment, the number of courses, of runs, of waypoints and the total distance traversed.

TABLE I
FIELD EXPERIMENTS STATISTICS

	Courses	Runs	Waypoint	Distance (km)
Desert test	5	38	25	8.3
Alpine test	5	16	47	3.2
Meadow test	5	23	23	10.5

C. Airborne sensor

The aerial laser data was collected in the weeks before the experiment by an outside contractor using a manned helicopter equipped with a Saab TopEye mapping system [8].

The helicopter was operated at 400 meter above the ground. The laser beam is deflected by an oscillating mirror which produces a Z-shaped laser track along the flight path. The range resolution is 1 cm and the point position accuracy varies between 10 and 30 cm, depending on the altitude of the aircraft. The laser sensor records two echoes per pulse (first and last), with a power of separation of 1.8 m. Porous or partial illuminated objects less than 1.8 m in height will produce one echo only per pulse. For each field experiment, the helicopter flew over the test area along several directions to produce higher point density, 1 to 52 points per square meter, 5 to 13 in average. Table II shows the area covered (bounding box) and the number of points collected for each field test [1].

TABLE II
AERIAL LIDAR DATA

	Wood test	*Desert test*	*Meadow test*
Area covered (km^2)	2.9×2.4	3.8×2.9	2.4×2.9
Nbr. points (millions)	43.5	38.2	41

On the first day of the field test, the aerial data was split into 200 m x 200 m submaps with a 20 m overlap typically. Each submap was then processed (vegetation removal, local signature construction, see the next two sections). Each day a new set of waypoints was provided, the corresponding submaps were then used to complete the mission.

D. The unmanned ground vehicle

1) Mobile platform and sensors: The vehicle is based on the chassis of an ATV with a 1.63 m × 2.49 m footprint and 17 cm ground clearance. It is equipped with state of the art inertial navigation system complemented by a military GPS. The exteroceptive sensors include a ladar, FLIR and stereo color cameras and are mounted on a turret head located at the front of the vehicle. This head can pan and tilt. The ground ladar is a medium-range, high-speed rugged laser developed by GDRS for mobile robot navigation

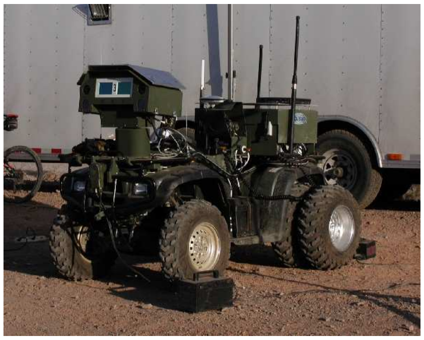


Fig. 1. Unmanned ground vehicle used during the phase-II. This vehicle is based on an ATV chassis. The laser radar, mounted on a pan-tilt head, is visible at the front of the vehicle. The rear box contains the different computers necessary to control the autonomous behavior of the robot.

[1] Wild fire in California in the summer 2002 prevented the collection of the aerial data for the *Alpine test*

2) Software architecture: The software architecture is made of three components for aerial data processing, UGV control and terrain-based localization. Aerial data was processed off-board the ground vehicle on a laptop running Linux. The paths planned were feed to the robot prior to the mission. The ground vehicle computer architecture is the NIST 4D-RCS [9] architecture, implemented on power PC boards running VxWorks. The localization software was implemented on a laptop running linux and installed on-board of the robot. Ground ladar data was read from a Neutral Message Language (NML) buffer and processed on the laptop.

III. Terrain surface recovery

Both our applications, terrain registration and path planning, require the recovery of the terrain surface. In this section we address the problem of removing the vegetation to uncover the terrain surface.

A. Vegetation filtering

We implemented two methods to filter the vegetation. The first one takes advantage of the aerial lidar capability to detect multiple echoes returned per laser pulse emitted. The ground ladar does not have this capability, so a second filtering method had to be implemented.

1) State of the art: Filtering lidar data has been mainly studied in the remote sensing community with three objectives: producing surface terrain models [10] (in urban or natural environment), studying forest biomass [11], and inventoring forest resources [12]. To filter lidar data authors used linear prediction [10], mathematical morphology (grey opening) [13], dual rank filter [14], texture [15], and adaptive window filtering [16]. All these methods are sensitive to the terrain slope. In the computer vision community Mumford [17] pioneered the work for ground range images. Macedo [18] and Castano [19] focused on obstacle detection among grass for outdoor ground robot navigation. Lacaze in [20], proposes to detect vegetation by looking at the permeability of the scene to laser range measurement. Another method ([21]) is to look at the local point distribution in space and uses a Bayes classifier to produce the probability of belonging to 3 classes - vegetation, solid surface and linear structure.

2) Methods implemented:

a) Multi-echoes based filtering: The lidar scans from multiple flights are gathered and the terrain is divided into 1 m $\times$ 1 m cells. Each point falling within a given cell is classified as ground or vegetation by k-mean clustering on the elevation. Laser pulses with multiple echoes (first and last) are used to seed the two clusters (vegetation and ground respectively). Single echo pulses are assigned initially to the ground cluster. After convergence, if the difference between the mean value of the two clusters is less than a threshold, both clusters are merged into the ground cluster. The clustering is performed in groups of 5x5 cells centered at every cell in the grid. As we sweep the space, each point is classified 25 times and a majority vote defines the cluster to which the point is assigned.

b) Cone based filtering: We had to implement a new method for filtering the vegetation from ground data. Our approach is inspired by [22] and is based on a simple fact: the volume below a ground point will be free of any lidar return. For each lidar point, we estimate the density of data points falling into a cone oriented downward and centered at the point of interest, as shown in Figure 2.

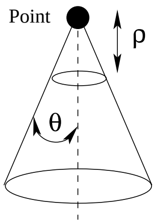


Fig. 2. Cone based filtering. Point: point to be clustered as ground or vegetation. θ opening of the cone. ρ minimum elevation before taking into account points.

While the robot traverses a part of the terrain, ladar frames are registered using the Inertial Navigation System (INS). The INS is sensitive to shocks (e.g., a wheel hitting a rock), which causes misalignment of consecutive scans. In order to deal with slightly misaligned frames, we introduce a blind area defined by the parameter ρ (typically 15 cm). The opening of the cone (typically 10-20^o) depends on the expected maximum slope in the terrain and the distribution of the points. Figure 3 illustrates an example of points distribution for a tree as seen by an aerial and a ground sensor. In the latter case, the trunk will occlude the terrain surface on top of which the ladar can perceive the tree canopy. If the cone opening is too narrow, the vegetation point will be misclassified. This case also illustrates the fact that by nearly taking the minimum elevation in each cell of a “gridded” terrain, we cannot recover the ground surface.

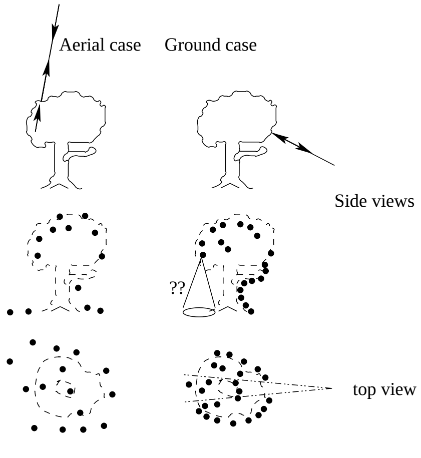
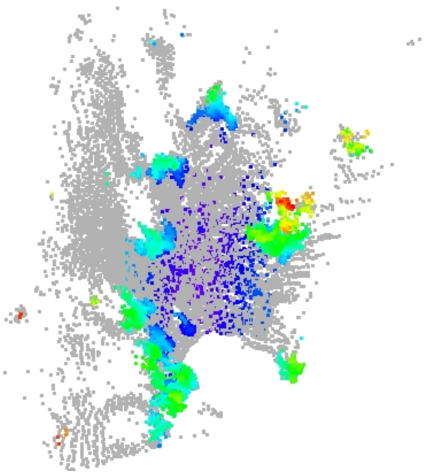
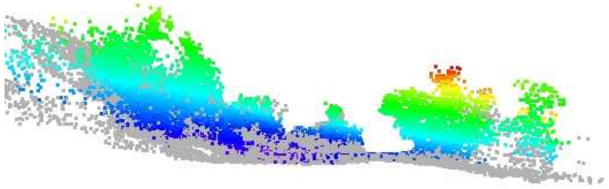


Fig. 3. Points distribution in a tree for an aerial lidar sensor (left) and a ground ladar sensor (right).

3) Example: Figure 4 shows an example of terrain surface recovery in ground ladar data using the cone based filtering method. This approach has been used to produce the results presented section IV. Our current implementation filters 67,000 points spread over 100 m $\times$ 100 m, in 25 s on a Pentium III, 1.2 GHz.



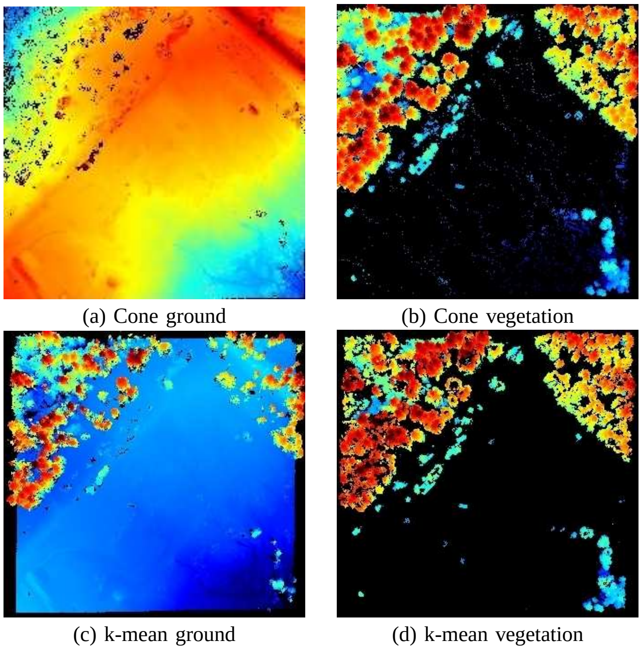
(a)



(b)

Fig. 4. Example of ground ladar data filtering. (a) Top view of a flat terrain with a cliff containing 3 m high trees and smaller bushes. (b) Side view of the same scene. In grey, the load bearing surface and in color the vegetation.

Figure 5 shows a comparison of the two methods implemented for load bearing surface recovering using aerial lidar data. The performance of the cone-based method over the k-mean method is clearly visible.



(a) Cone ground

(b) Cone vegetation

(c) k-mean ground

(d) k-mean vegetation

Fig. 5. *Meadow test*. Comparison of the cone based (a)-(b) and k-mean (c)-(d) method for terrain surface recovery. Data is shown as top views of 3-D points cloud with the elevation color-coded. The scene is 200×200 m^2 in size.

B. Performance evaluation

The fidelity of the terrain surface recovered depends on two main factors: the raw 3-D data collected by the lidar and the vegetation filtering technique. We assess both here.

1) Lidar data quality: The lidar absolute error was estimated using ground features recognizable in the lidar data and for which we collected DGPS points. We choose three of them: the center of the flat root of an isolated building, a large flat concrete bed and the center of a crater. The results of the first case are presented in Table III.

TABLE III

LIDAR ABSOLUTE ERROR, ROOF DATA SET

	DGPS	lidar	Error (m)/STD
Easting	485,262.94	485,263.02	0.08
Northing	3,437,988.76	3,437,989.01	0.25
Elevation	107.00	106.90	0.1/0.0532

For the second feature, the large concrete bed laying at the ground level, the mean elevation error is 4 cm with a standard deviation of 5.36 cm. For the third feature, the bottom of a large conical hole, the elevation error is 11 cm and and the horizontal error position is 7 cm. Those results are much better than the accuracy reported in the literature for such a mapping system [8]. But in all cases, the measured elevation of a flat surface fits in a bounding box of 22 cm height within which we cannot classify points as ground or vegetation. Overall we were able to detect trenches, terrain depression with water, tall grass areas, trails below tree canopy. But we failed to detect concertina wires, poles and fallen tree trunks below the canopy like the one illustrated in Figure 6.

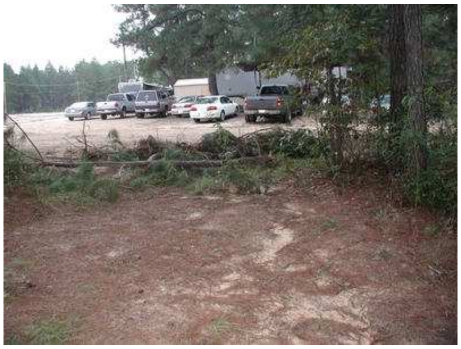


Fig. 6. *Meadow test*, example of missed features: a follen tree truck cluttered by vegetation.

2) Vegetation filtering: In addition we took a closer look at the quality of the vegetation filtering. We took a worst case scenario with low point density (one over-fly of the area) with tall brush and thin trees. 928 (84) points have been classified as ground (vegetation), 66 vegetation points have been misclassified as ground (7% of the total), no ground point has been misclassified outside the 22 cm envelop. It is not possible to determine the nature of a 3-D point inside that envelop because of the limits on the sensor resolution. This is a good example of the current limitation of the terrain recovery method.

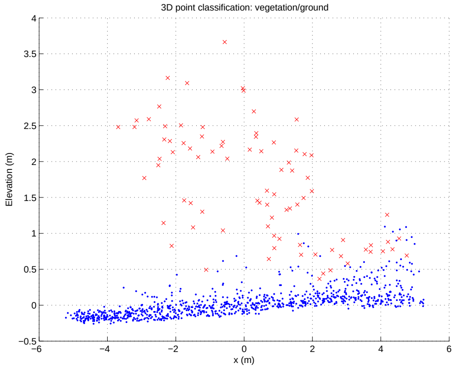


Fig. 7. Vegetation filtering error. The red crosses are used for the points classified as vegetation, and the blue dots are used for ground points.

IV. Air-Ground terrain registration

A. Overview

In the previous section, we presented a method to recover the load bearing surface of terrains with a vegetal cover, a preliminary step to perform terrain-based robot localization. Our approach is to align ground laser data with aerial laser data in order to find the position and the attitude of the ground mobile robot. By establishing 3-D feature points correspondence between the two terrain maps and by extracting a geometrically consistent set of correspondences, a 3-D rigid transformation can be estimated, see Figure 8.

In this section, we compare our approach to other Air-Ground terrain registration methods for robot localization. We present the surface matching engine used, and we then focus on the extensions that are needed for practical Air-Ground registration, including the incorporation of positional constraints and strategies for saliency points selection. We present the system architecture and data structures used in practical operation of the registration system with full-scale maps. We finally present results obtained during field experiments.

B. Air-Ground terrain registration for robot localization

Ground localization can be achieved in mountainous terrain by using the horizon skyline acquired by a panoramic camera and a topographic map. Several authors followed this idea, their approach differ by the nature of the feature extracted from the skyline. Talluri ([23]) proposes to compute for each position in the map the height of the skyline and to compare it with it counterpart in the topographic map. Sutherland ([24]) extracts elevation peak as features and focuses on the selection of the best set to be used to minimized the localization error by triangulation. Stein ([25]) proposes a two steps approach: first, he reconstructs the skyline for some position in the map as a linear piecewise function and index them in order to map them efficiently with the skyline extracted from a camera, second, a verification is performed by matching the skyline. Cozman ([26]) follows the same path but in a probabilistic framework. Such method cannot be applied to our problem because they rely on too specific skyline typically found only in mountainous area. In addition their reported accuracy, hundred of meter, is too poor.

Yacoob ([27], [28]) proposes to acquire random range measurements from a static position and to match them in a digital elevation map. The orientation and altitude of the vehicle is supposed known. This approach was envisioned in the context of a planetary terrain. Single range measurements do not provide enough information to disambiguate between returns from vegetation and the terrain surface. This approach cannot not be pursued.

Li ([29]) proposes to localize a planetary rover through bundle adjustment of descent images from the lander and rover stereo-images. Our primary ground sensor is a ladar and we have access to higher density and resolution aerial data.

C. Registration by matching local signatures

We present here the approach to 3-D surface matching we decided to follow. Given two 3-D surfaces, the basic approach is to compute signatures at selected points on both surfaces, that are invariant by changes in pose. Correspondences are established between points with similar signatures. After clustering and filtering of the correspondences, the registration transformation that aligns the two surfaces is computed from the correspondences. The key to this class of approaches is the design of signatures that are invariant and that can be computed efficiently.

We chose to use as the signatures introduced by Andrew Johnson in [30] – the "spin-images" – and used since then in a number of registration and recognition applications. A spin-image is a local and compact representation of a 3-D surface that is invariant by rotation and translation. This representation enables simple and efficient computation of the similarity of two surfaces patch by comparing the signatures computed at selected points on the patches. Interpolation between the input data points [31] is used to make the representation independent of the surface mesh resolution.

The signatures are computed at selected data points – the basis points – at which surface normal and tangent plane are computed. For each point in the vicinity of a basis point, two coordinates can be defined: α the distance to the normal and β the distance to the tangent plane. The spin-image surface signature is a 2-D histogram in α and β computed over a support region centered at the basis point. Parameters include the height and width of the support region, and the number of cells used for constructing the histogram. Proper selection of these parameters is crucial for ensuring good performance of the matching algorithm. We will return to that point in the experiments section.

D. Improvements

1) Using a priori positional information: In its most general form, the matching algorithm attempts to find correspondences by comparing signatures directly, without restrictions on the amount of motion between the two data sets. Although this is appropriate for some unconstrained surface matching problems, additional constraints can be used in the terrain matching scenario. Specifically, constraints on the orientation and position of the robot from the navigation sensors can be used to limit the search for correspondences.

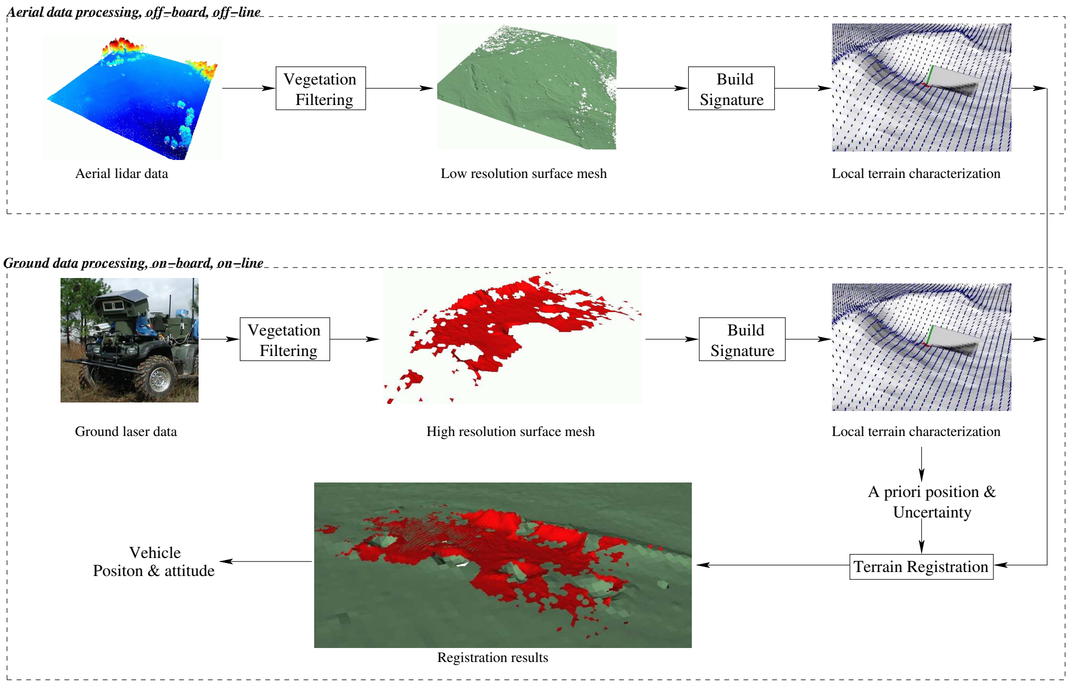
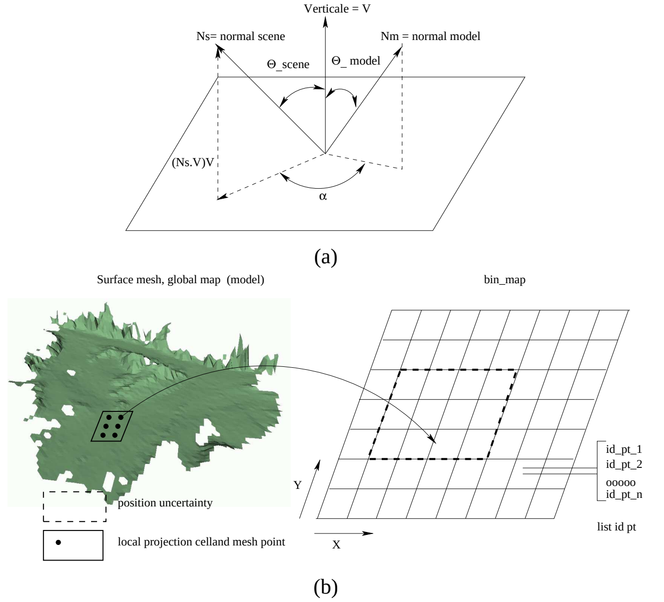

Aerial data processing, off–board, off–line
Aerial lidar data
Vegetation Filtering
Low resolution surface mesh
Build Signature
Local terrain characterization
Ground data processing, on–board, on–line
Ground laser data
Vegetation Filtering
High resolution surface mesh
Build Signature
Local terrain characterization
A priori position & Uncertainty
Terrain Registration
Registration results
Vehicle Positon & attitude

Fig. 8. Overview of the terrain-based localization method. The top section is concerned with the off-line processing of the aerial ladar data. The bottom section deals with on-line ground ladar data processing and comparison with a priori data.

Positional constraints are used as follows. To generate a correspondence between two oriented points we check the relative orientation of the normals, see Figure 9-(a). If the difference in heading (projection in the horizontal plane) and in elevation (projection in the vertical plane) are below limit angles spin-images are compared and a correspondence is eventually produced. The a priori position of the robot and a distance threshold is used to constraint in translation the search of potential correspondences, see Figure 9-(b). For each oriented point in the scene mesh (the ground data), only oriented points in the model mesh (the aerial data) which satisfy the translation constraints are considered to produce eventually a correspondence.

2) Point selection strategy: In realistic registration applications, the terrain data may contain millions of data points. In such situations, it is not practical to use all the data points as candidate basis points for matching. In previous registration systems [30], [31], a fraction of the mesh vertices, chosen randomly, was used to build the correspondences. Although it reduces the number of basis points to a manageable level, there are several practical problems with direct uniform sampling. First, a substantial percentage of the points may be non-informative for registration – as an extreme example, consider those points for which the terrain is flat inside the support region – and uniform sampling would not reduce that percentage. Second, a more subtle problem is that a high density of basis points is acceptable on the reference aerial map, because the signatures are computed off-line, but a lower density is preferable in the ground data since it is processed on-line.



(a)

(b)

Fig. 9. Using a priori positional information: (a) in orientation and (b) in position.

In [32], Johnson proposed a method for point selection based on the surface signatures. In that approach, the signatures and the position of each oriented point is concatenated to produce a vector. This high dimensional space is reduced using PCA and clusters are extracted to isolate unique landmarks. Although very effective, this approach has two major drawbacks for our application. First, the algorithm is computationally expensive because the signatures must be computed at all the input data points prior to point selection and because of the cost of performing compression and clustering in signature space. Second, this approach was used for registering multiple terrain maps taken from similar aerial viewpoints. As a result, the input data sets would exhibit similar occlusion geometry. In our case, ground and aerial data may have radically different configurations of occluded areas, which needs to be taken into account explicitly in the point selection. Also, that approach did not take into account the need for asymmetric sampling mentioned above – high density of points on the reference map and low density on the on-line terrain map from the ground robot.

We developed a different point selection strategy in which the basis points from the input data set are progressively filtered out by using a series of simple tests, most of which do not require the computation of the signatures. Eventually, a fraction of points uniformly sampled from the remaining set is retained. The final sampling implements the asymmetric point selection strategy with a high density of points in the model (the aerial data) and a low density in the scene (the ground data). Details of the various criteria used for point selection are described below.

a) Flat areas: Points in flat area or area of constant slope do carry any information for registration. In such regions the signatures will have high mutual similarity. As a result, the registration procedure will create a large number of correspondences that are hard to filter out. To discard such areas we compute local statistics of the surface normals in the support area of the basis points. Even though the surface normals are noisy, smoothing the mesh and using a region based criteria allow us to filter correctly the points. This method does not require the computation of signatures.

b) Range shadows: Self-occlusions in the ground data in the support region centered at a given basis point may corrupt the signature computed at that point. Because of the extremely different sensor geometry used for acquiring the aerial data – viewing direction nearly orthogonal to the terrain surface – and the ground data – viewing direction at low grazing angle, terrain occlusions have a radically different effect on the signatures computed from the aerial and ground data sets, even at the same point and with the same support region, see Figure 10-(a)-(b). It is therefore imperative that the occlusion geometry be explicitly taken into account so that basis points at which the signatures are moderately corrupted by occlusions can be selected.

We developed a method to reject oriented points too close from map borders or from occluded areas. Figure 10 illustrates our method. To detect the occlusion situations, we compare the surface of the spin-image support area, the circle, with the surface really intersected, the surface inside the circle. We compute the area ratio between both surface patches. We filter oriented points if this ratio is below a threshold. This approach eliminates those signatures which contain a large occlusion that would not appear in the aerial data. This method does not require the computation of the spin-images.

c) Information Content: To retain a basis point as a feature for matching, we also require a minimum level of information content, defined as the number of lines occupied in the signature. Figure 11 presents one rejected signature and one retained signature.

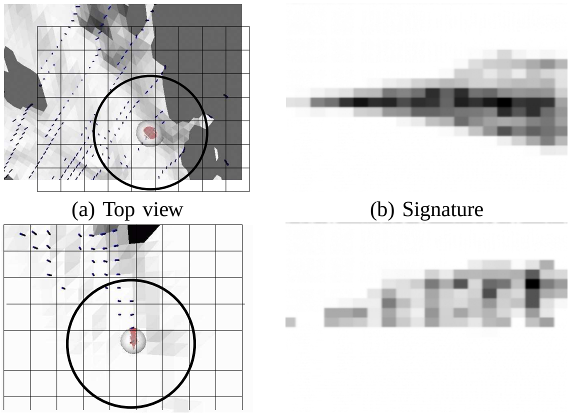

(a) Top view
(b) Signature
(c) Top view
(d) Signature

Fig. 10. Range shadow filtering. Mesh and signature for the same physical point using the ground and the aerial data. (a/b) Ground data. (c/d) Aerial data. The circle represents the swept area by the spin-image.

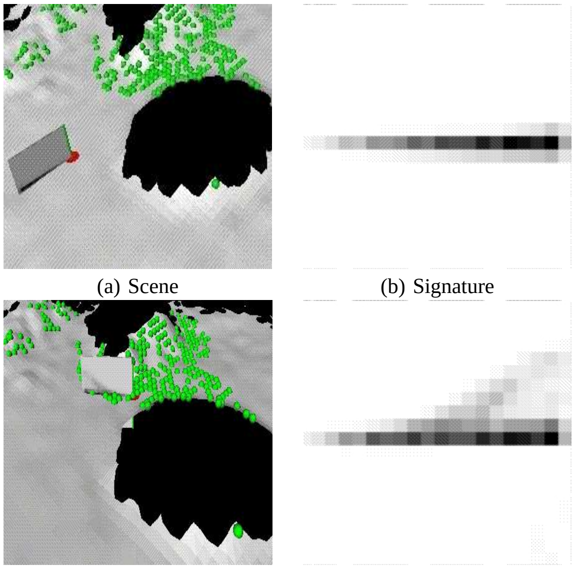

(a) Scene
(b) Signature
(c) Scene
(d) Signature

Fig. 11. Information content in signature. (a)/(c) Oriented points. In red the current point with the signature displayed on the right. In Green the points with high level of information content. In grey the surface mesh. (a/b) Retained signature. (c/d) Rejected signature.

E. Operational system

In practical applications, we deal with maps of several kilometers on the side with millions of data points. In such cases, even after point selection, it is simply not practical to store and index into the entire reference data set. In practice, the reference map is divided into sub-maps that are large enough to include a typical scan from the ground sensor, but small enough to fit in memory. In the system described below, we use 300 m × 300 m as the default patch size. For each patch, the list of selected points and the corresponding signatures are generated and stored.

At run-time, the robot's dead reckoning system provides a position estimate that is used to retrieve the appropriate 300 m × 300 m patch. The dead reckoning system is also used to provide positional constraints derived from the uncertainty model.

TABLE IV

Desert test, LEDGE 6/7 GROUND DATA STATISTICS

	Ledge 6	Ledge 7
Frames	14	33
3-D points	37 K	100 K
Distance traversed	19	41.53 m
Signatures	543	859

TABLE V

Desert test, LEDGE 9 REGISTRATION STATISTICS

Step	Timing (ms)	Notes
Build correspondences	4043.63	465 correspondences created
Votes	2.88	451 correspondences left
Correlation	2.24	188 correspondences left
Geometric consistency	784.63	52 correspondences left
Creates matches	846.85	one created
Cluster matches	10.23	
Match verification	1400.97	
Best match	0.27	

In addition to the storage issue, there is an indexing issue in which the signatures at points that are within the limits imposed by the positional constraints are efficiently retrieved from the set of points. Even within a 300 m × 300 m patch, efficient retrieval is a concern. In the current implementation, a data structure is added to the basic grid for efficient retrieval of basis points that are within translational tolerance of a query point, and whose surface normals is within angular tolerance of the normal at that query point.

F. Results

In [4] we presented an metric evaluation of the registration performances of the method proposed. In this section we present registration results (Air-Ground and Ground-Ground) obtained during the field experiments.

1) Air-Ground registration: We showed three sets of Air-Ground registration from the *Desert test* and one from the *Meadow test*. Figure 12 shows two examples of terrain registration along a small hill. The robot drove along the hill on a flat terrain with sparse dry vegetation. Figure 12-(a) shows the hill and Figures 12-(b)/(c) show the registration results. Ground data information (number of ladar frames integrated, distance traversed, number of 3-D points, and number of signatures used for registration) are presented in Table IV. The registration method parameters used are:

- Spin-image parameters: 0-3 × ±2 m in size, and 20 × 20 pixels in resolution.
- Area of interest: ±20 m in position, 20^o in heading, and 20^o in orientation to the vertical.

The second set of results are obtained from the wash area during the *Desert test*. The wash area is a riverbed like terrain made of the lower flat terrain bordered on each side by higher ground. Vegetation is denser than in the ledge area with the presence of large trees and numerous bushes as it can be seen in Figure 13-(a)/(b). The registration results are shown in Figure 13-(c)/(d). Table VI and Table V provide timing information and details on the registration process.

TABLE VI

Desert test, WASH 9 REGISTRATION STATISTICS

Step	Timing (ms)	Notes
Build correspondences	1263.15	177 correspondences created
Votes	27.05	148 correspondences left
Correlation	1.74	40 correspondences left
Geometric consistency	35.62	14 correspondences left
Creates matches	38.98	one created
Cluster matches	0.05	
Match verification	0.01	
Best match	0.01	

Finally the third set of results come from the *Meadow test*. The terrain is made of two parallel piles of soil, two meters in elevation, separated by a narrow corridor. Both piles are covered with meter tall grass and a tree. The scene can be shown in Figure 14-(b). Table VII contains details on the data used (number of frames, distances traversed, and signatures used).

TABLE VII

Meadow test, DATA STATISTICS

	Ground	Aerial
Frames	51	*
Distance traversed	30.77	*
Signatures before/after filtering	2371 / 2001	39186 / 2446

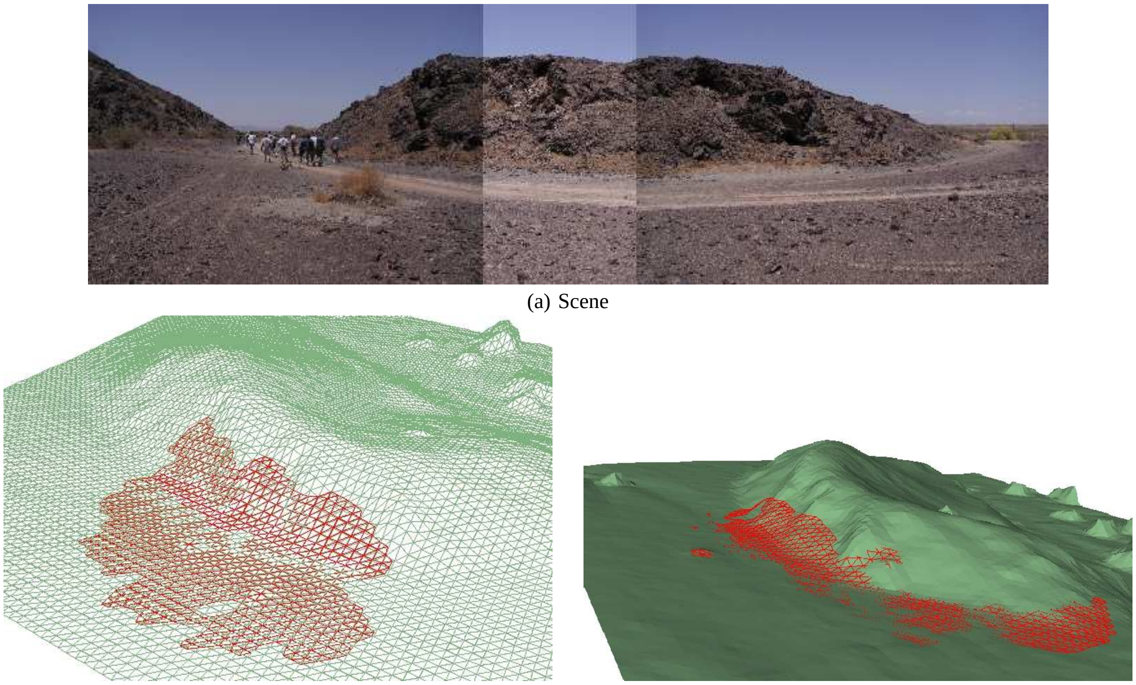
2) Temporal registration: The last two examples presented here come from the *Desert test* and the *Meadow test*. In the first case we registered two data sets collected as the robot navigate along the ledge of the hill shown previously. In the second case the data were collected in two static positions with the ladar turret doing a pan-tilt coverage of the terrain.

In this section we presented our approach for robot localization by aligning ground ladar data with aerial ladar data. We presented the improvement of the method we made, the integration on-board a mobile robot and results from different field experiments. In the next section we will focus on path planning.

V. TRAVERSABILITY MAP FOR PATH PLANNING

A. Overview

In this section, we describe the use of high-resolution aerial data for planning paths for autonomous navigation of an UGV. These paths computed prior to the missions are global, in that aerial data available for the entire region of interest is used in computing the paths. In contrast to the trajectories planned using on-board sensors which only provide a local picture, the a priori paths take into account the global picture of the surrounding area thereby preventing the robot from falling into local traps. The raw data is specified as points in the 3-D space whereas path planners typically take cost maps as input. We process the raw data and compute cost maps that reflect the robot's difficulty of traversing a piece of terrain. This is done using the reconstructed 3-D ground surface and a static vehicle model. These cost maps are used to generate viable mission paths between a start and a distant goal point. All the points are specified in the 3-D coordinate system. The objective of a robot's mission is to autonomously navigate from the start to end position passing through a set of given intermediate waypoints along the least risky path. The paths computed a

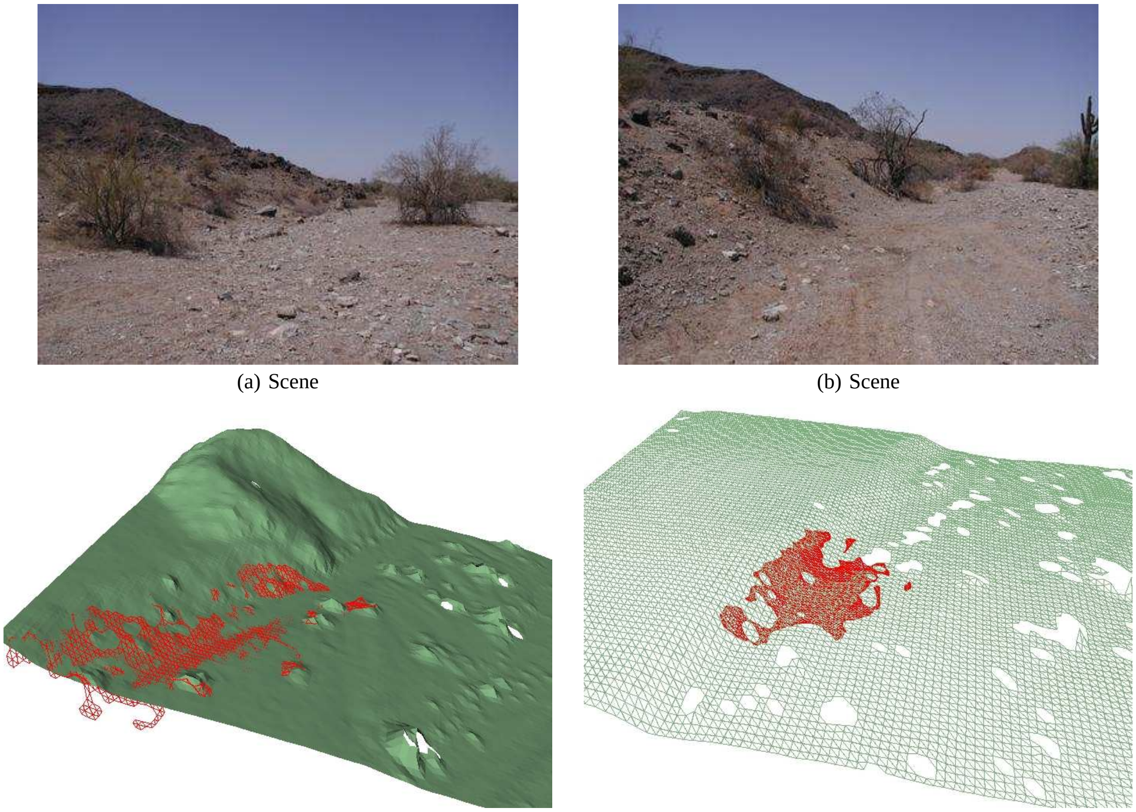


(a) Scene

(b) Air-Ground registration

(c) Air-Ground registration

Fig. 12. *Desert test*, Air-Ground registration. (a) Panoramic view of the scene. (b)/(c) Ledge 6/7 with in green the aerial data and in red the ground data. See Table IV for more information on the data collected.



(a) Scene

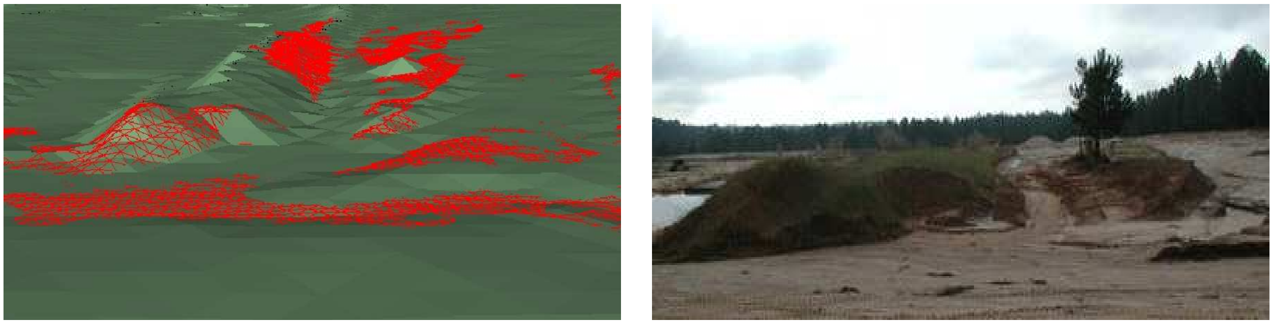
(b) Scene

(c) Air-Ground registration

(d) Air-Ground registration

Fig. 13. *Desert test*, Air-Ground registration. (a)/(b) scenes and (c)/(d) registration results with in green the aerial data and in red the ground data. (a)/(c) Ledge 9, see Table V for registration result statistics. (b)/(d) Wash 9, see see Table VI for registration result statistics.

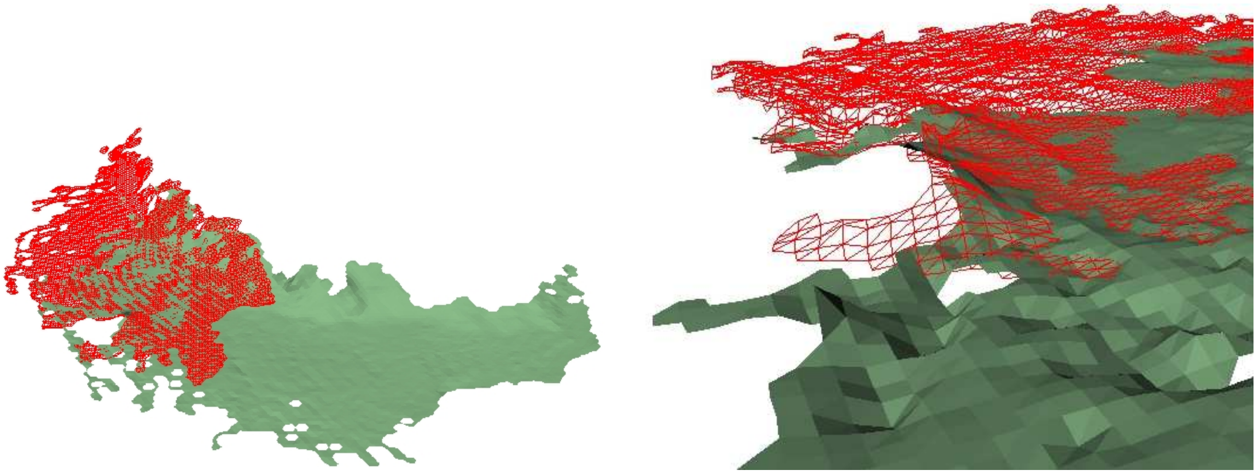
priori (using the cost maps) are the least cost paths touching all the specified points. The paths were successfully used in a variety of test missions, some as long as one kilometer. We examine the effect of vegetation filtering on the performance of the path planner. We also study the influence of the vehicle model on traversability cost evaluation. The quality of the cost



(a) Air-Ground registration

(b) Scene

Fig. 14. *Meadow test*. (a) Ground (red) and aerial (green) aerial data aligned. (b) Scene picture. See table VII for information on the data used.



(a) Correct Ground-Ground registration

(b) Incorrect Ground-Ground registration

Fig. 15. *Desert test* and *Meadow test*, Ground-Ground registration. (a) Correct match from the *Desert test*. (b) Incorrect match from the *Meadow test*.

map generated is assessed using various ground truth data.

The layout of this section is as follows. We mention some of the prior work in Section V-B. Section V-C describes the various steps in going from the raw aerial data to the mission path generation. We follow this up by results of our algorithm in Section V-D. The evaluation of the results is described at the end of the Section V-E.

B. Path planning for mobile robot

While the usage of aerial range data for outdoor robot navigation is relatively new, computing cost of robot traversal for mobile robot navigation in rough terrain has been studied before. These maps are typically constructed from the environment perceived by the on-board sensors and are used mainly to compute robot trajectories (instantaneous direction of travel). The common denominator in most of the methods is using some or all of the following in analyzing the risk of traversing a particular region of the terrain: *vehicle models (dynamic/static), terrain parameters, kinematic constraints of the robot, interaction between wheels and the terrain.*

In [33], Iagnemma *et al* study reactive behavior control of the robot and show that detailed models of vehicle and terrain can accurately predict the dynamics of the mobile robot in rough terrain. Cherif and his team address motion planning of a robot in 3-D [34] [35]. Besides using the dynamic model of the robot, their planning method takes into account the robot-terrain interactions and the kinematic constraints of the robot. The dynamic modeling and vehicle-terrain interactions require instantaneous state (position, attitude and velocity,.. etc) and rich description of the terrain surface. We are interested in planning global paths, rather than local trajectories for the robot. The latter is a more complex problem as they are computed online and take into account the instantaneous state of the robot and the data from on-board sensors with an aim to estimate the best direction to move forward at any time instant. On the other hand, we compute cost maps off-line and hence do not have access to the state of the robot and feedback data from the sensors. Also, the vehicle is expected to travel at relatively slower speeds (around 2-5 m.p.s) and a static model is sufficient to estimate the risk in traversing a particular cell.

[36] [37] discuss an online terrain parameter estimation, which will be very useful in predicting the traversability of the rover (as a function of the terrain parameters). However, this requires training the terrain models with large amounts of data and availability of robot's state, which are not available in our scenario.

[38] generate a *goodness map*, where the goodness in a cell (cell size same as dimension of the robot) is a function of the roll, pitch of the supporting plane and roughness of the terrain. [39] compute traversability probability along eight directions. For cost, they use horizontal/vertical energy functions.

Our work is tested on real data and actual robot missions, unlike some of the earlier work which is tested only on simulations[39], [40], [41], [42]. In the following subsection we discuss the algorithm used in computing the cost maps from the aerial data.

C. Method

This subsection discusses, in detail, the steps in aerial data processing and computation of cost maps that are input to a planner. We also mention a grid-based planner that we used to test the cost maps generated by our algorithm.

1) Traversability map: Our goal is to provide a *safe* path between way points based on aerial lidar data. Figure 16 contains a flow chart of our approach. Starting with the raw data we first divide the space into 1 m × 1 m cells. Ladar data points are then segmented into two clusters: vegetation and ground using the vegetation filtering algorithm described in section III. This segmentation is done in order to recover the ground surface underneath the canopy. For each cell, we calculate the risk of traversing that cell in form two kinds of maps. In the first map, we evaluate the ratio between the number of vegetation points (the points classified using algorithm in section III) and the total number of points. This measure defines a confidence criterion on the terrain reconstructed below the canopy. We call it the *vegetationness* of the cell. This criterion is used by the path planner and allows the planner to consider trajectories under canopy that would not have been considered if the vegetation filtering was not performed.

In the second map, the traversability at each location in the map is evaluated using the standard approach of convolving a vehicle model with the elevation map. More precisely, using the ground points (after vegetation filtering) we compute traversability cost maps (one every 45^o in heading) as follows:

- Interpolate the position of each tire on top of the ground surface,
- Fit a plane and extract the current roll, pitch, and remaining ground clearance of the vehicle,
- These values are remapped between 0 (non-traversable) and 1 (traversable) and then thresholded, using a sigmoid function and the static performance model of the vehicle (maximal frontal and side slope, ground clearance of the vehicle).

The static vehicle parameters are specified as absolute limits of angle and distance (roll cannot be greater than 15^o for example). Any angle greater than the limit should be declared hazardous for traversal. This usually results in a binary cost map (traversable or non-traversable), which is not very useful in path planning. Also, an angle slightly less than the threshold is also very hazardous, if not catastrophic. To incorporate this, we use a sigmoid function (which are smooth and non-linear) to map the roll, pitch and clearance values of a cell to a value between 0 and 1. It gives high cost (close to non-traversable) to cells where the angle is close to the threshold. This continuous range ($[0, 1]$) will be useful in choosing one path against another as compared to a binary range (0/1). The final traversability value assigned to the cell is the least favorable of the three criteria. If one of the three criteria exceeds the robot's limits, the cell is marked as non-traversable.

2) Planner: To test our approach we used a grid-based path planner [43] to determine the path of least cost in our "gridded" map. The cost at each node is computed as follows :

$$C_{comb.}(\theta) = \frac{1}{(1 - C_{trav.}(\theta))^2} + \frac{1}{(1 - C_{veg.})^2} \qquad (1)$$

With $C_{trav.}(\theta)$ the directional traversability of the cell with θ the heading of the robot when it entered that cell; $C_{veg.}$ is the *vegetationess* value of the cell; finally $C_{comb.}(\theta)$ is the cell cost used to determine the path in this specific cell along that direction.

D. Results

The algorithm was used on real data from missions in challenging environments. The various test sites included natural terrain features including, vegetation (dense/sparse, trees with thin/thick trucks), ledges, meadows (with tall grass), water stream, barb wire fence, etc. In this article we show results from courses which reflect typical mission settings. The first result shows our success/failure in detecting various kinds of obstacles present in a typical mission while the second one shows the importance of filtering vegetation as a preprocessing step. We picked two settings which clearly depict these two situations.

1) Example from the Meadow test*:* Figure 17 shows the result of our approach. This course in the test site had one intermediate way point, besides the start and finish points. The environment contained sparse vegetation (of thin pine trees) and tall grass (dense) and a wide trail, a small ledge, a fort whose entrance was narrow and the pathway to the entrance guarded by barbed fence on either sides. This mission consists of obstacles various kinds. The costs maps computed in this region reflect the obstacles in the region. For example the isolated pine trees in the region can be mapped to non-traversable dots in the vegetationness map. The narrow pathway was declared 'traversable' while the fort walls were declared non-traversable. In this mission, we successfully detected large obstacles such as fort walls and medium obstacles such as thin pine trees. We failed to recover thin obstacles (barbed fence wire). However it should be noted that this wire is very thin to have many aerial lidar hits on it.

Figure 17-a gives an aerial picture of the region. One can see the amount of vegetation in green. Figure 17-b gives a snapshot of 3-D data of the area surrounding this course. (In white is the ground that is recovered by the algorithm and in color is the vegetation data encoded by height of the vegetation). One can notice the trail, fort in the course and that the region sparsely populated with thin pine trees. The three points are marked in red. This is a display of the aerial range data provided to us. It also shows the path planned using the aerial data. Figure 17-c shows the traversability map computed using the static vehicle model and the terrain from (a). This is one of the eight maps computed. This map reflects that the mud walls of the fort are non-traversable, so is the ledge by the trail. The dots

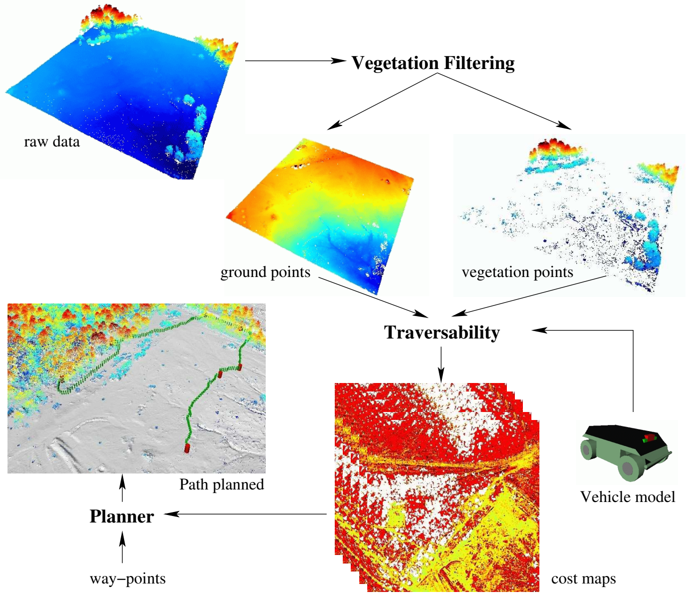
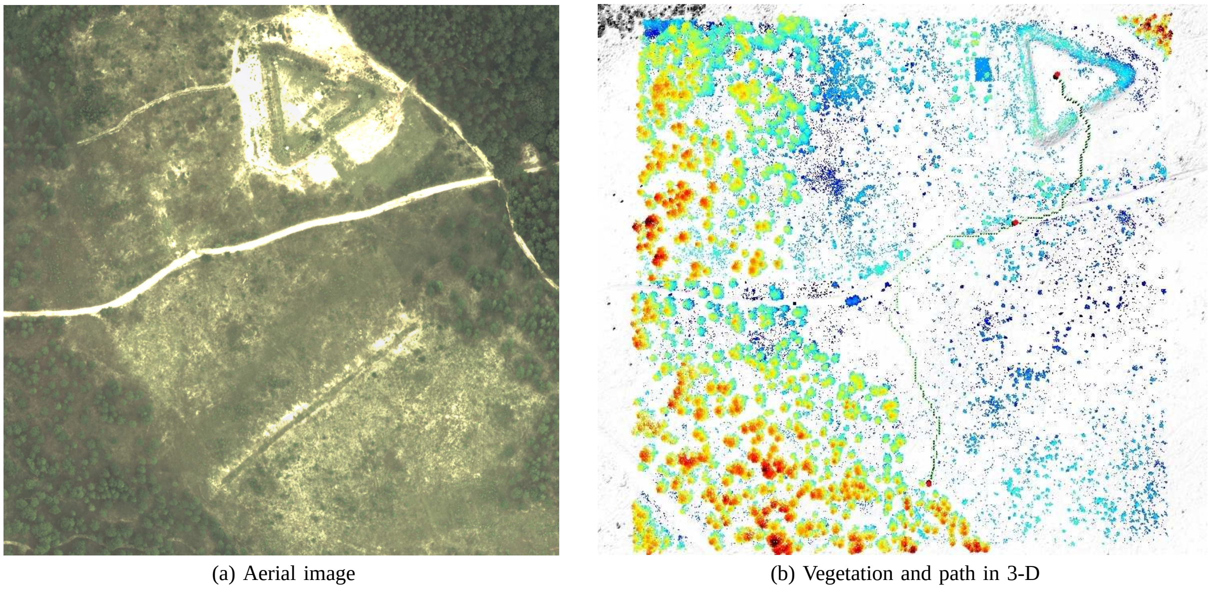


Fig. 16. Overview of the path planning approach using traversability maps from aerial ladar data. All steps are performed off-line. The path is then loaded into the robot. It will track the path while doing local obstacle avoidance.


(a) Aerial image

(b) Vegetation and path in 3-D

Fig. 17. Path planning and vegetation filtering. Yankee course, *Meadow test*. (a) Aerial image of test area with the way points as black dots. (b) Top view of a 3-D rendering of the course the ground (in white) surface represented with a mesh, and displayed in color (elevation: blue (low) to red (high)) the 3-D points classified as vegetation. In black is the path computed

in green is the path planned. Notice that the path does not cut any non-traversable regions.

2) Example from the Wood test*:* Here we show the importance of vegetation filtering in a mission, especially under dense canopy. Figure 18 presents an example of path obtained with our planner using the *Wood test* data set. The area is densely covered with tall trees producing a continuous canopy over the ground. The density of foliage is such that sufficient ground points were sensed to reconstruct the terrain below the canopy. Figure 18-(a) is an aerial view of the scene with the initial 15 way points plotted in black. The total cumulative distance between each way points is 152 meters. Notice the density of the vegetation in the area. In this mission, we detected various trees and deep pits in the area. Figure 18-(b) represents the vegetation map. The path computed by the path planner is overlaid in black. As expected the vegetationness values are very high in the region, as the large fraction of the laser hits are on the foliage. Figure 18-(c)/(d) are the traversability maps computed with and without vegetation filtering respectively. The green areas are traversable and the dark red terrain is non-traversable. Points in blue are part of a water pond. The interest of the method is explicit on this example. Without filtering the canopy points, the area is completely non-traversable. Notice that a viable path is produced on the reconstructed surface. During the field test a different path planner was used to produce a similar path passing through the same way points [43]. The robot actually navigated autonomously along this path, avoiding local obstacles (small obstacle, overhanging branches) not perceived in the aerial lidar data. This shows that vegetation filtering was successful in recovering the terrain under canopy. However, one should notice that this success depends on the number of hits on the ground (as compared to the ones on the foliage), which is affected by the density of foliage. This factor is incorporated in the vegetationness values.

E. Evaluation

The results of various steps in the method are evaluated in this section using ground truth data wherever possible. Quality of the final paths generated depends on various things that happen in each phase of the algorithm. We attempt to assess the quality of performance of various phases of the algorithm to understand them better. Among other things, we evaluate the relative influence of the vehicle model on the cost maps, measure the performance of the planned paths, and evaluate the ground recovery algorithm.

1) Cost maps fusion: We tested the relative influence of the traversability and *vegetationess* map on the results produced by the path planner with the *Desert test* data set. We performed 3 different tests using 47 pairs of starting/ending points, selected randomly in the scene. We computed a path for each of them using three different sets of maps: 1) the 8 directional traversability maps and the *vegetationess* map, 2) one directional map and the *vegetationess map*, 3) the 8 directional traversability maps only. Each path produced, 141 in total, has been evaluated visually using a high resolution aerial image (17 cm/pixel) as ground truth. The Table VIII summarized the results obtained. In all cases, a valid path is known to exist and a failure is recorded whenever a path cannot be generated from the cost map. The table contains in addition to the failure rate, the average length of the paths. This table concurs with the intuitive notion that it is indeed advantageous to consider maps in multiple directions rather than just one. In this particular case not considering the vegetationness cost at each cell did not have much effect on the success rate because the area was desertic and did not have any tall overhanging trees. But from Figure 18-(c)/(d) it is very clear that it might be crucial in some situations.

TABLE VIII
STATISTICS ON THE RELATIVE INFLUENCE OF THE COST MAP

Evaluation	Trav.	Vegetation	Failure rate	Average length
#1	8 dir.	Yes	4.2%	528.9 m
#2	1 dir.	Yes	19.2 %	534.7 m
#3	8 dir.	No	4.2%	521.1 m

2) Influence of the vehicle model: We used a static vehicle model composed of three parameters: the ground clearance, the transversal and the longitudinal maximum slope angle (roll, pitch). We are interested in determining their influence on the path planned (and hence the cost maps computed). The actual vehicle ground clearance is 17 cm, smaller than the error envelop of the vegetation filtering. Figure 19 illustrates the influence of this parameter. The example is from the Area No. 2. The terrain is made of tall grass cluttered with trees. A dirt road traverses the terrain. Figure 19-(a) shows the path planned with the actual ground clearance: the trajectory follows the edge of the wood, reach and follow the road and enter the fort. Figure 19-(b) shows a more aggressive path where the ground clearance is set at 50 cm. This is the actual trajectory followed by the robot using the mobility ladar for obstacle detection. The ground clearance criterion is more sensitive to terrain recovery errors than the maximum slope angles criteria. Such analysis can be used to fine tune the static model of the robot.

3) Path evaluation using the actual robot trajectory: In this section we compare the actual paths executed by the robot with the initial path computed using the cost maps. Figure 20 shows such an example for Area No. 1. When we have the paths executed by the robot, we can be assured of the fact that the cells where the robot visited are actually ’traversable’. This information can be used to go back and adjust certain facts about the robot/terrain parameters.

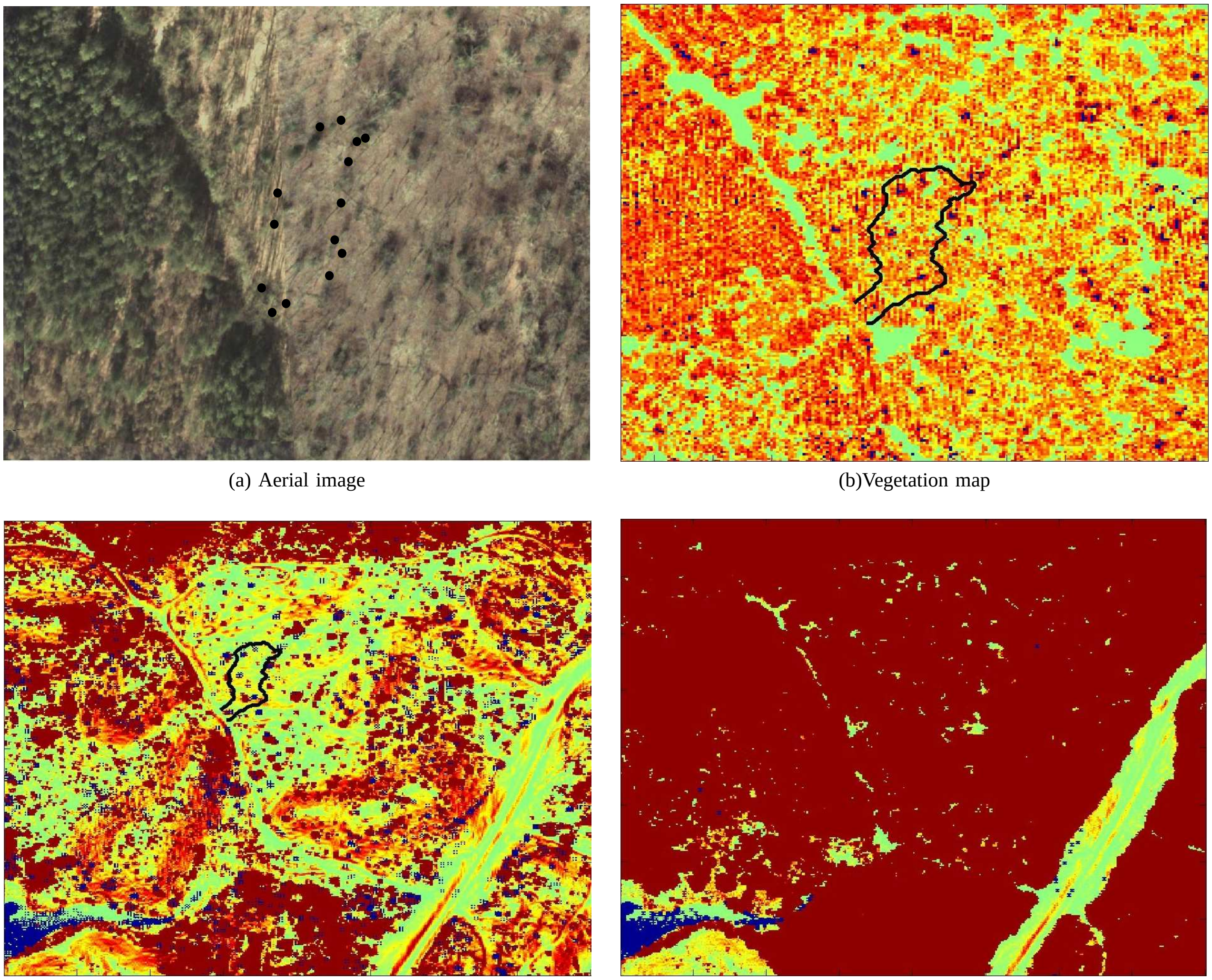
Our criteria would be the path length of the robot in the unknown[2] and in the non-traversable areas respectively. We evaluated the two criteria for 29 paths performed on the 5 test areas. Table IX shows the summary of statistics for each of the 5 test sites. Please refer to [5] for a detailed performance results on individual runs. The first criterion is an indicator of the amount of information missed in areas actually traversed by the robot. This information could be used, given this type of terrain and the neighboring area, to predict the traversability cost in similar regions. The second criterion indicates how conservative we have been or how poor is the terrain surface recovery. This is a partial result dealing only with false-negative cells. The evaluation of false-positive traversable cells requires scouting the site with the robot and collecting ground measurements. Time constraints did not permit doing it. From Table IX, we see that in wooded terrains, area No. 3 and area No. 4, our traversability map generation did not perform well as one can expect because of sparse laser beams hitting the ground below the canopy. The summary of path lengths has been generated by summing up corresponding lengths for all the runs in an area.

TABLE IX
TRAVERSES STATISTICS. SUMMARY OF PATH LENGTHS FROM 29 EXPERIMENTS IN 5 DIFFERENT AREAS. LENGTHS FROM MULTIPLE RUNS IN AN AREA HAVE BEEN ADDED TOGETHER TO SUMMARIZE THE RESULTS.

Course	length (m)	non-trav. (m)	unknown (m)	%
Area # 1	1806.97	80.01	144.32	14.86
Area # 2	3686.03	242.42	17.98	7.13
Area # 3	2652.39	473.43	818.83	48.81
Area # 4	3216.29	480.05	378.55	30.06
Area # 5	948.34	102.21	61.86	18.20

4) Path evaluation using the robot navigation system: In this section we propose to compare the vehicle attitude collected during one of the missions with the attitude for the same position computed using the aerial lidar data and the ground ladar data. Results are presented in Figure 21. The

[2] unknown areas are those where we do not have sufficient ground hits



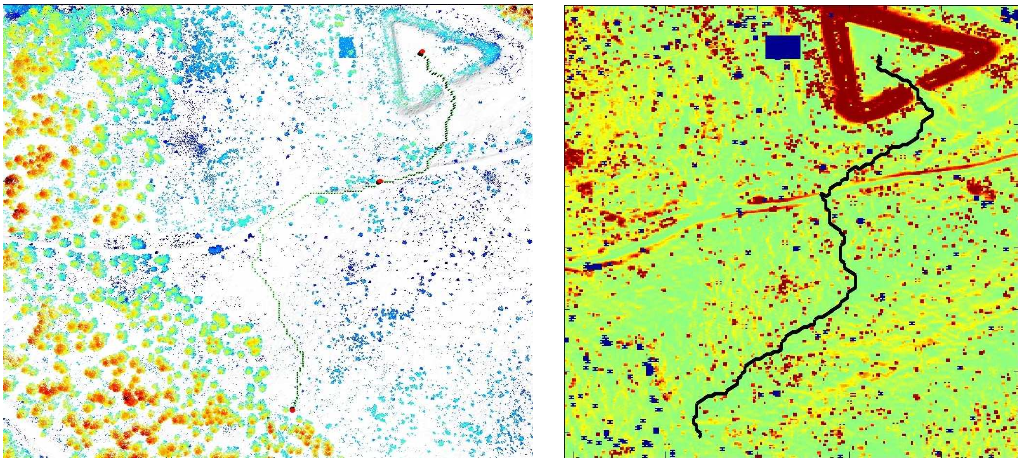
(a) Aerial image

(b)Vegetation map

(c)Traversability map

(d)Traversability map

Fig. 18. Path planning and vegetation filtering. *Wood test*.(a) Aerial image of test area with the way points as black dots. (b) Vegetation map (color scale from green to red: no vegetation to highly vegetated. In black the path computed. (c) Traversability map with the path overlay. Vegetation has been filtered. Color scale, from green to red: traversable to non-traversable. (d) Traversability map. Vegetation has not been filtered. No path has been found. Color scale, from green to red: traversable to non-traversable. The area is 200×200 m^2.



(a) Conservative path (3-D rendering)

(b) Aggressive path (cost map)

Fig. 19. Ground clearance influence, terrain No. 2 from the *Meadow test*. Figure (a) is a top view of a 3-D rendering of the scene with in white the ground terrain surface represented with a mesh, in color (elevation: blue (low) to red (high)) the 3-D points classified as vegetation. The red poles represents the starting and ending point. The green poles are the intermediate waypoints computed by the path planner. Figure (b) presents the path (black) overlaid on top of a cost map with from green to red the terrain classified as traversable to non-traversable, and in blue the unknown areas.

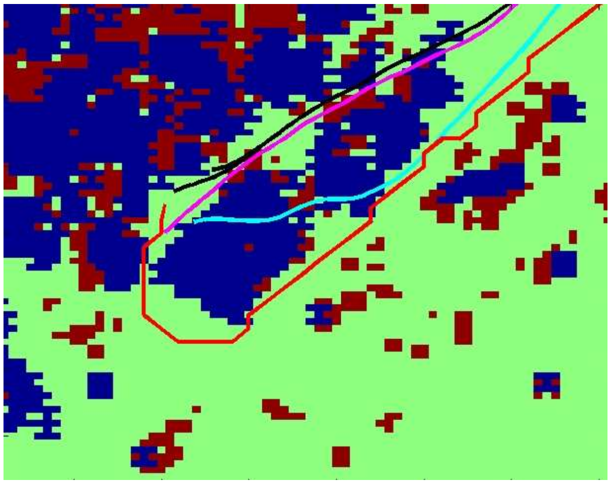


Fig. 20. Area No. 1, *Meadow test*. Actual paths (top) and planned path (bottom, red) overlaid on top of the cost map, terrain No. 1. The map covers 100 × 150 m, the grid resolution is 1 m. Color map: In blue/red/green the unknown/non-traversable/other cells.

robot drove forward 30 meters, stopped then drove backward 7 meters and finally forward another 10 meters. The robot position was sampled at 20 Hz and the data set corresponds to 60 seconds of traverse. Figure 21-(d) shows the trajectory of the robot. The terrain is covered by tall grass and bushes. Figure 21-(c) is a top view of the terrain with in grey the ground and in color the vegetation. Figure 21-(a)/(b) presents respectively the pitch and roll of the robot, computed from aerial lidar data and from the ground mobility ladar data. In the last two cases the DGPS ground truth of the robot position and the heading from the inertial navigation system was used to convolve a model of the robot with the lidar data. From Figure 21, we can see that the frequencies of the curves are the same but the amplitudes are different. Amplitude differences at regime transitions, around frame No. 400 for example, are probably due to the static model of the vehicle we are using. Amplitude differences at static regimes are probably due to the sensor error and terrain recovery process. In either case, in that particular example at least, our traversability map computation has been able to recover the basic trend of the robot attitude. Such analysis could be used to fine tune the cost map generation so as to match actual robot characteristics.

In this section we described an algorithm which computes cost maps from aerial lidar data. Great attention has been given to evaluating various stages in the algorithm. We propose various ways to verify algorithms' performance. We believe that such rigorous post-mission evaluation is required to improve the performance of an autonomous vehicle. We quantitatively evaluated the fidelity of the traversability maps produced from aerial lidar data for an autonomous ground mobile robot navigation. We computed the surface reconstruction error to verify the vegetation filtering algorithm. We tested the influence of vehicle model which can be used to to fine tune the parameters in the model. Finally we compared the robot's attitude collected during one of the missions (using inertial navigation system on the robot) with the attitude for the same position computed using aerial and ground lidar data.

VI. Conclusion

This document presents algorithms which exploits aerial ladar data for autonomous ground vehicle navigation in two different ways. In the first application, the aerial data is used to localize a robot while in the second the data is used to help the vehicle plan global mission paths. The former is useful when GPS localization is unavailable (below the tree canopy, in canyon) while the latter prevents the robot from falling into local traps. Both the algorithms have been successfully tested in actual missions in different types of terrains. Vegetation has been shown to be a major challenge for mobile robot navigation. Algorithms to filter the vegetation have been presented. Also great attention has been given to evaluating various stages in the algorithms. We believe that such rigorous experiment evaluation is required to improve the performance of the algorithms.

However, there are still many research issues in the system that can be improved or extended. One can use the ground-ground registration algorithm in a mission with multiple robots where the ground data from one robot can be used by the other or in a mission where the terrain map are constructed using ground lidar data. In cost map evaluation, a dynamic model of the robot will probably produce more realistic cost maps than using a static model. The terrain registration and path planning algorithms can work in tandem to improve the system performance. The ground-aerial registration can be used to enhance the resolution of aerial data and fill holes. Cost maps (and hence global paths) can be recomputed at mission time using the enhanced aerial data. Also many improvements are possible in ground reconstruction algorithm.

Acknowledgment

This project was supported by DARPA PerceptOR Program, under subcontract to General Dynamics Robotic Systems. This work would not have been possible without the help of W. Klarquist and Jeremy Nett from PercepTEK.

References

[1] A. Stentz, A. Kelly, P. Rander, H. Herman, O. Amidi, R. Mandelbaum, G. Salgian, and J. Pedersen, "Real-time, multi-perspective perception for unmanned ground vehicles," in *AUVSI*, 2003.

[2] P. Rander, T. Stentz, A. Kelly, H. Herman, O. Amidi, and R. Mandelbaum, "Integrated air/ground vehicle systems for semi-autonomous off-road navigation," in *AUVSI*, 2002.

[3] J. Miller, "A 3d color terrain modeling system for small autonomous helicopters," Ph.D. dissertation, Carnegie Mellon University, 2002.

[4] N. Vandapel and M. Hebert, "3d rover localization in airborne ladar data," in *International Symposium on Experimental Robotics*, 2002.

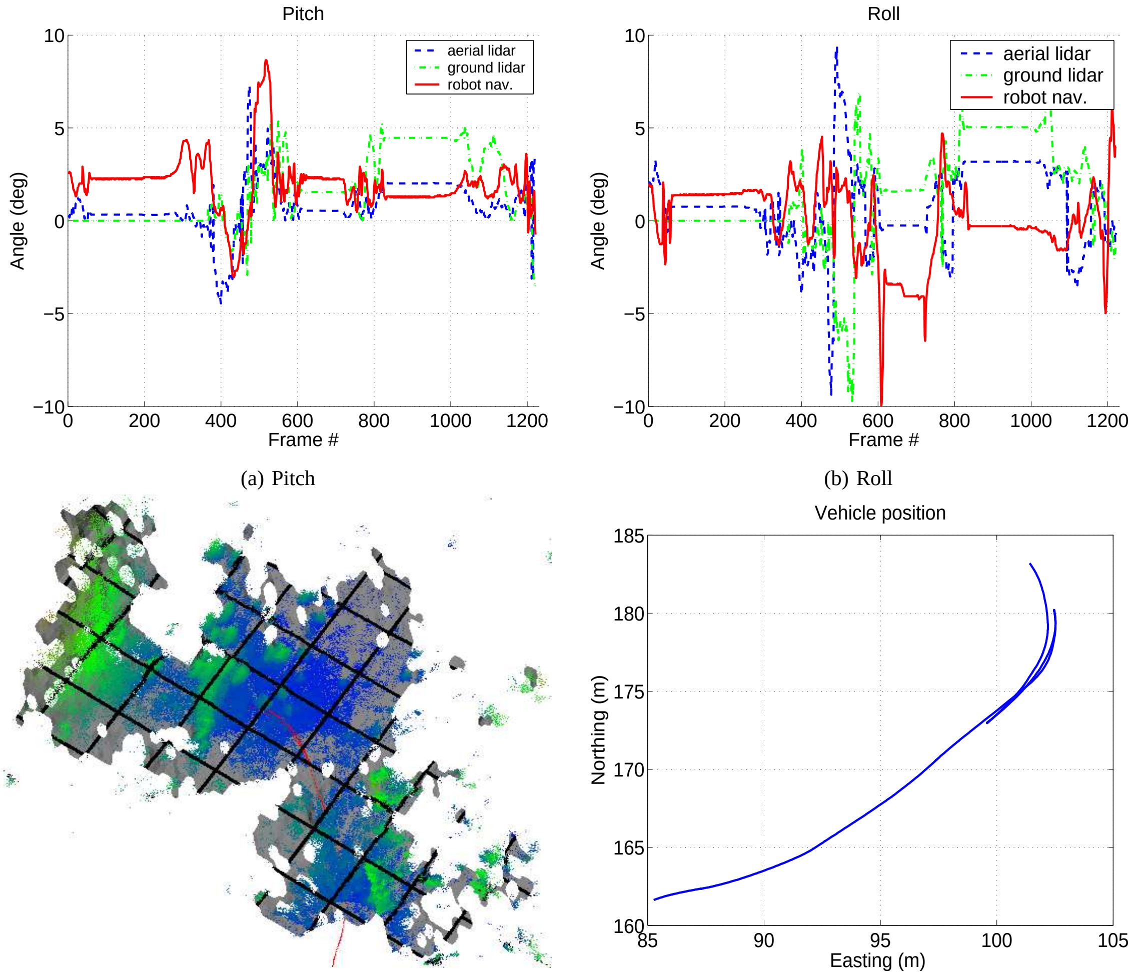
[5] N. Vandapel, R. Donamukkala, and M. Hebert, "Quality assessment of traversability maps from aerial lidar data for an unmanned ground vehicle," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2003, pp. 305–10.

[6] ——, "Experimental results in using aerial ladar data for mobile robot navigation," in *International Conference on Field and Service Robotics*, 2003.

[7] D. Langer and al., "Imaging ladar for 3-d surveying and cad modeling of real world environments," *International Journal of Robotics Research*, vol. 19, no. 11, pp. 1075–88, 2000.

[8] E. Baltsavias, "Airborne laser scanning: existing systems and firms and other resources," *ISPR Journal of Photogrammetry & Remote Sensing*, vol. 54, 1999.

[9] J. Albus and al., "4d/rcs version 2.0: A reference model architecture for unmanned vehicle systems," NIST, Tech. Rep., 2002.



(a) Pitch

(b) Roll

(c) Top view terrain

(d) Vehicle trajectory

Fig. 21. (a-b) Vehicle attitude: red line from robot navigation data, blue dashed line from aerial lidar data, green dash-dot line from ground lidar data. (c) Terrain: in grey, surface terrain 0.5 m resolution, black grid spaced by 2 m, in red the robot trajectory and in color (from blue to green) the vegetation with the elevation color coded. Results are from Area No. 2

[10] K. Krauss and N. Pfeifer, “Determination of terrain models in wooden areas with airborne laser scanner data,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 53, pp. 193–203, 1998.

[11] M. Lefsky and al., “Lidar remote sensing of the canopy structure and biophysical properties of douglas-firwestern hemlock forests,” *Remote Sensing Environment*, vol. 70, 1999.

[12] J. Hyyppa, O. Kelle, M. Lehikoinen, and M. Inkinen, “A segmentation-based method to retrieve stem volume estimates from 3-d tree height models produce by laser scanners,” *IEEE Transaction on Geoscience and Remotre Sensing*, vol. 39, no. 5, pp. 969–75, 2001.

[13] W. Eckstein and O. Munkelt, “Extrating objects from digital terrain models,” in *Remote Sensing and Reconstruction for Three-Dimensional Objects and scenes, SPIE Proceedings*, vol. 2572, 1995.

[14] P. Lohmann, A. Koch, and M. Shaeffer, “Approaches to the filtering of laser scanner data,” in *International Archives of Photogrammetry and Remote Sensing*, vol. XXXIII, 2000.

[15] S. Elberink and H. Mass, “The use of anisotropic height texture measures for the segmentation of airborne laser scanner data,” in *International Archives of Photogrammetry and Remote Sensing*, 2000.

[16] B. Petzold, P. Reiss, and W. Stossel, “Laser scanning surveying and mapping agencies are using a new technique for the derivation of the digital terrain models,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 54, no. 2, pp. 95–104, 1999.

[17] J. Huang, A. Lee, and D. Mumford, “Statistics of range images,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2000, pp. 324–331.

[18] J. Macedo, R. Manduchi, and L. Matthies, “Ladar-based discrimination of grass from obstacle for autonomous navigation,” in *International Symposium on Experimental Robotics*, 2000.

[19] A. Castano and L. Matthies, “Foliage discimination using a rotating ladar,” in *IEEE International Conference on Robotics and Automation*, 2003, pp. 1–6.

[20] A. Lacaze, K. Murphy, and M. DelGiorno, “Autonomous mobility for the demo III experimental unmanned vehicles,” in *Proceedings of the AUVSI Conference*, 2002.

[21] M. Hebert and N. Vandapel, “Terrain classification techniques from ladar data for autnomous navigation,” in *Collaborative Technology Alliance Conference*, 2003.

[22] G. Sithole, “Filtering of laser altimetry data using a slope adaptive filter,” in *ISPRS workshop on Land Surface Mapping and Characterization using laser altimetry*, 2001.

[23] R. Talluri and J. Aggarwal, “Position estimation for an autonomous robot in an outdoor environment,” *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 573–84, 1992.

[24] K. Sutherland and W. Thompson, “Localizing in unstructured environments: dealing with the errors,” *IEEE Transactions on Robotics and Automation*, vol. 10, no. 6, pp. 740–54, 1994.

[25] F. Stein and G. Medioni, “Map-based localization using the panoramic horizon,” *IEEE Transactions on Robotics and Automation*, vol. 11, no. 6, pp. 892–6, 1995.

[26] F. Cozman, E. Krotkov, and C. Guestrin, “Outdoor visual position estimation for planetary rovers,” *Autonomous Robot*, vol. 9, no. 2, pp. 135–50, 2000.

[27] Y. Yacoob and L. Davis, “Computational ground and airborne localization over rough terrain,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 1992, pp. 781–3.

[28] ——, “Ground and airborne localization over rough terrain using random environmental range-measurements,” in *IEEE International Conference on Pattern Recognition*, 1992, pp. 403–6.

[29] R. Li, F. Ma, F. Xu, L. H. Matthies, C. F. Olson, and R. E. Arvidson, “Localization of mars rovers using descent and surface-based image data,” *Journal of Geophysical Research - Planets*, vol. 107, no. E11, 2002.

[30] A. Johnson, “Spin-images: A representation for 3-d surface matching,” Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, August 1997.

[31] D. Huber and M. Hebert, “A new approach to 3-d terrain mapping,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1999, pp. 1121–1127.

[32] A. Johnson, “Surface landmark selection and matching in natural terrain,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2000, pp. 413–20.

[33] K. Iagnemma, D. Golda, M. Spenko, and S. Dubowsky, “Experimental study of high-speed rough-terrain mobile robot models for reactive behaviors,” in *International Symposium on Experimental Robotics*, 2002.

[34] M. Cherif and C. Laugier, “Motion planning of autonomous off-road vehicles under physical interaction constraints,” in *IEEE International Conference on Robotics and Automation*, 1995, pp. 1687–93.

[35] M. Cherif, “Kinodynamic motion planning for all-terrain wheeled vehicles,” in *IEEE International Conference on Robotics and Automation*, 1999, pp. 317–22.

[36] K. Iagnemma, H. Shibly, and S. Dubowsky, “On-line terrain parameter estimation for planetary rovers,” in *IEEE International Conference on Robotics and Automation*, 2002, pp. 3142–3147.

[37] C. Wellington and A. Stentz, “Learning predictions of the load-bearing surface for autonomous rough-terrain navigation in vegetation,” in *International Conference on Field and Service Robotics*, 2003.

[38] S. Singh and al., “Recent progress in local and global traversability for planetary rovers,” in *IEEE International Conference on Robotics and Automation*, 2000, pp. 1194–200.

[39] T. Kubota, Y. Kuroda, Y. Kunii, and T. Yoshinitsu, “Path planning for newly developped microrover,” in *IEEE International Conference on Robotics and Automation*, 2001, pp. 3710–15.

[40] T. Simeon and B. Dacre-Wright, “A practical motion planner for all-terrain mobile robots,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1993, pp. 1357–63.

[41] T. Simeon, “Motion planning for a non-holonomic mobile robot on 3-dimensional terrains,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1991, pp. 1455–60.

[42] D. Pai and L.-M. Reissell, “Multiresolution rough terrain motion planning,” *IEEE Transactions on Robotics and Automation*, vol. 14, no. 1, pp. 19–33, 1998.

[43] S. Balakirsky and A. Lacaze, “World modeling and behavior generation for autonomous ground vehicle,” in *IEEE International Conference on Robotics and Automation*, 2000, pp. 1201–6.

Nicolas Vandapel Dr. Vandapel is a Project Scientist at the Robotics Institute, Carnegie Mellon University. He received his Ph.D from LAAS-CNRS, France, in December 2000. His research efforts focus on scene interpretation from 3-D laser data for autonomous ground vehicle navigation in natural environment containing vegetation. He developed techniques for real-time laser data classification and segmentation on-board vehicle. He is also involved in 2-D and 3-D overhead data processing to support ground vehicle navigation. This includes load bearing surface recovery, air-ground 3-D data alignment for localization, mobility assessment and path planning.

Raghavendra Rao Donamukkala Raghavendra Donamukkala received the B.Tech degree in Computer Science and Engineering from the Indian Institue of Technology, Kharagpur, India in 2001. He received the M.S. degree in Robotics from the Robotics Institute, Carnegie Mellon University in 2003.

Martial Hebert Martial Hebert is a Professor at the Robotics Institute, Carnegie Mellon University. His interests include the development of techniques for representing, comparing, and matching 3-D data from sensors, with application to object recognition and model building. His group has developed techniques for fully automatic modeling of 3D scenes from many views and for recognizing objects in complex scenes. His interests also include the development of perception techniques for autonomous mobility for mobile robots navigation and techniques for object recognition in images.