Terrain Classification from Aerial Data to Support Ground Vehicle Navigation

Boris Sofman, J. Andrew Bagnell, Anthony Stentz, and Nicolas Vandapel
Robotics Institute Carnegie Mellon University
Pittsburgh, USA
{bsofman, dbagnell, axs, vandapel}@ri.cmu.edu

***Abstract*— Sensory perception for unmanned ground vehicle navigation has received great attention from the robotics community. However, sensors mounted on the vehicle are regularly viewpoint impaired. A vehicle navigating at high speeds in off-road environments may be unable to react to negative obstacles such as large holes and cliffs. One approach to address this problem is to complement the sensing capabilities of an unmanned ground vehicle with overhead data gathered from an aerial source. This paper presents techniques to achieve accurate terrain classification by utilizing high-density, colorized, three-dimensional laser data. We describe methods to extract relevant features from this sensor data in such a way that a learning algorithm can successfully train on a small set of labeled data in order to classify a much larger map and show experimental results. Additionally, we introduce a technique to significantly reduce classification errors through the use of context. Finally, we show how this algorithm can be customized for the intended vehicle's capabilities in order to create more accurate a priori maps that can then be used for path planning.**

I. Introduction

An unmanned vehicle needs to have a complete understanding of the terrain and features around it if it is to be able to navigate complex environments safely. There are cases, however, when it is not possible to get an adequate understanding of the environment from the vehicle-based view of terrain without sacrificing speed or path optimality. For example, a vehicle navigating at high speeds in off-road environments may be unable to react to negative obstacles such as large holes and cliffs. Even when the vehicle can safely navigate an environment, aerial sensing can dramatically improve path planning performance by detecting large obstacles such as buildings and bodies of water as well as areas of preferable terrain such as roads.

Fig. 1. Sample classification results. On the left is an overhead reference image used for ground truth and on the right is the classification of the corresponding high-density, colorized, 3-D data into road, grass, tree, and building. Areas with missing data appear in white.

This work presents techniques to classify terrain from aerial sensor data in order to produce a priori maps to support ground vehicle navigation (see Figure 1). Such maps enable an unmanned ground vehicle to construct an accurate initial path that results in more efficient traversal of the environment. Current automated obstacle classifiers, however, tend to focus predominantly (and often exclusively) on elevation data. We show how utilizing additional sensor information such as color and signal reflectance can significantly improve automated sensory perception and allows accurate classification of a much wider variety of map features.

We describe methods to extract relevant features from this sensor data so that a neural network can successfully train on a small set of labeled data in order to classify a much larger map and show experimental results. Additionally, we introduce a technique to significantly reduce classification errors through the use of context. Finally, we show how these algorithms can model the intended vehicle's capabilities in order to create more accurate obstacle maps for path planning. This work utilizes color, elevation, and signal reflectance data gathered by an unmanned helicopter, but the outlined techniques can be applied to any combination of sensor data gathered through a variety of methods including from satellites. No preset rules or assumptions are made about the environment. Since elevation and image data for the entire world are available at various resolutions, as the quality of global profiling improves, the applications of such research will greatly expand.

In the following sections we outline related work in the area of terrain classification, give an overview of our approach, and give experimental data and results.

II. Related Work

Sensory perception from aerial data has been studied by many researchers from both within and outside the robotics community. In this section we review only work relevant to supporting ground vehicle navigation.

The DARPA Perceptor program contained an important overhead component. Two strategies were implemented by two different teams. The National Robotics Engineering Consortium developed a novel semi-autonomous unmanned ground vehicle (UGV) that utilized a dedicated unmanned helicopter that flew ahead of the UGV to detect holes and other hazards ahead of the vehicle [1]. The helicopter served as a scout to explore terrain before the UGV had to traverse it, allowing the

UGV to re-plan its route to avoid certain areas entirely based on elevation hazards detected by the helicopter. As the UGV adjusted its route, it steered the helicopter to detect terrain features in the direction of its new path. The second team, led by General Dynamics Robotic Systems (GDRS), used a priori data from a manned aircraft to perform path planning and air-ground terrain registration for robot localization. In [2], the authors presented techniques to classify three-dimensional (3-D) points as load bearing surfaces or vegetation. The load bearing surface was then convolved with a vehicle model to produce directional cost maps. The vegetation cover was used to estimate the confidence in the terrain recovery. Those cost maps were then provided to a path planner off-line, prior to the robot mission. During the mission, the robot followed the a priori path while avoiding obstacles using on-board perception. The load bearing surface recovered from the air and from the ground vehicle were also co-registered in 3-D in order to estimate the absolute pose of the robot in the prior map.

Rasmussen and Korah implemented a vision-based approach to autonomous driving on desert roads on a system used during the 2004 DARPA Grand Challenge [3]. Their vehicle used on-board camera video to extract linear direction and midline estimates of roads as well as satellite imagery immediately surrounding the vehicle's GPS position to trace the road ahead for curve and corner anticipation.

Similar research conducted by Charaniya et al. [4] classified terrain into roads, grass, buildings, and trees using aerial LiDAR height data, height texture, and signal reflectance, achieving classification rates in the range of $66\% - 84\%$. Knudsen and Nielson attempted to classify buildings using a previously available GIS database and RGB information for an environment [5]. Even with the prior data, they encountered significant difficulty in classifying accurately without utilizing elevation data as well. Even works attempting to simply identify man-made objects from an aerial image have been only moderately successful [6].

Others have focused on supplementing an UGV's LiDAR range data with color sensors in order to perform ground level feature analysis and improve real-time UGV navigation [7]. Classifier fusion has also been extensively studied in attempts to improve on the quality of classification resulting from just one feature extraction system [8][9]. Finally, methods have been studied to improve UGV utilization of a priori data [10] as well as techniques for testing the quality of maps generated from such data [11].

III. Approach

A. Feature Space

1) Data Representation: The sensor data was discretized into grid cells such that each cell's data values were the average values of all data points that fell within that cell. For small grid cell sizes, many cells contained no data points while for large grid cell sizes, even though map coverage was improved, much of the information was lost in the averaging of the raw data. Even though the actual training and classification was performed with these grids, our features were generated from both raw 3-D data and the binned 2-D versions of the data. This approach was taken in order to improve computation efficiency.

2) Elevation: The elevation related features were computed on elevation relative to local ground levels rather than from absolute elevations. It is common for many environments to have a small, constant slope. As a result, the measured elevation of a building roof at one side of the map may in fact be below the ground level at the other side of the map, invalidating any feature learning based on elevation. The local ground level of each cell was estimated by searching a fixed distance in each direction and treating the lowest found elevation as the ground level for that cell.

3) Color: Raw RGB color data was inadequate for our approach due to its sensitivity to illumination variations. By converting the point data from RGB space to HSV (hue, saturation, value) space, we provided the classifier with values that were much more suitable for learning and classification. A hue of α° was represented by the pair of values, $sin(\alpha)$ and $cos(\alpha)$ to address the continuity problem at hue values close to 0°. Even though most of the effects of varying shadows and lighting were negated by using HSV space, this technique made the assumption that light is purely white, which was not entirely true but was sufficient for this application.

4) Reflectance: The power of the laser signal reflected back to the helicopter's sensors from an object depended mainly on the surface material properties and its relative orientation to the incident angle. No additional pre-processing is needed for signal reflectance measurements.

5) PCA Features: Our three Principal Component Analysis (PCA) features characterized the local distribution of a 3-D point cloud around a point of interest. PCA features dealt directly with the 3-D data and not the binned 2-D versions of the data like the other features. The final results were binned into the grid cells as before to be used with the other features.

Let $\{X_i\} = \{(x, y, z)\}$ be a 3-D point. Let $B(X) = \{X_k; |X_k - X| < r\}$ be the set of 3-D points within the support region centered at X and of radius r and let N be the cardinality of $B(X)$.

With $\overline{X} = \frac{1}{N}\sum_i X_i$ we have the covariance matrix:

$$COV_{B(X)}(X) = \frac{1}{N}\sum(X_i - \overline{X})(X_i - \overline{X})^T$$

This matrix was decomposed into principal components ordered by decreasing eigenvalues:

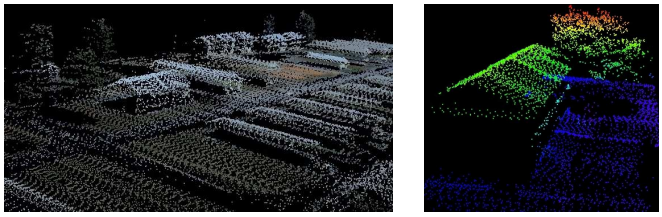
$$|COV_{B(X)}(X) - \lambda \cdot I| = 0$$

with $\lambda_0 \geq \lambda_1 \geq \lambda_2$.

The three PCA features were defined as λ_2, $\lambda_0 - \lambda_1$, and $\lambda_1 - \lambda_2$.

These features captured respectively the scatter-ness, linear-ness and surface-ness of a local point cloud. Two large eigenvalues and one small eigenvalue corresponded to a surface. The difference between the second largest and the smallest eigenvalues corresponded to the saliency of the surface-ness of the structure: the larger the difference, the more planar

the point cloud was locally. Also, if no dominant eigenvalue existed, the points were distributed randomly in space, a characteristic of many tree canopies. Finally, with only one dominant eigenvalue, the point cloud was encompassed in an ellipsoid corresponding most likely to a branch or power line. Such features allowed us to accurately characterize complex 3-D scenes such as the one presented in Figure 2b.



(a) (b)

Fig. 2. Close-up view of 3-D point cloud data (a) and 3-D point cloud data for a tree canopy hanging over a building, color-coded by elevation (b).

The mean and standard deviation for each of these eight parameters were computed in a 5 by 5 window surrounding each cell to be used as features. The standard deviation of data values was often a very good indicator of terrain features (for example, the high standard deviation in elevation readings throughout tree canopies.) Using such a window also ensured that outlier points do not significantly influence the classification while still providing a degree of robustness to missing data. Cells with less than three valid cells within their window were ignored. In order to aid in neural network training, all features were rescaled to the $[-1, 1]$ range where -1 corresponds to the lowest value for that feature in the map and 1 corresponds to the highest.

B. Training and Classification

1) Neural Network Classifier: A neural network with one input node for each feature, one hidden layer containing three nodes, and one output node for each desired classification class was used. The sixteen computed features for each cell were used as inputs to the neural network for classifying that cell. Three hidden nodes proved to be optimal for this particular classification task, but if the environment were to become more complex, additional types of sensor data were used, or the number of classification classes were increased, the size of the hidden layer could be increased accordingly. The highest output node score was used to classify each cell. The neural network was trained on each class by a set of labeled regions selected by the user from the training map. Each cell within a labeled region was used as a training example for that class. Each training example was given a desired output of 1.0 for the output node corresponding to that example's class and a desired output of 0.0 for all other output nodes. The network was trained using backpropagation with a learning rate of 0.05 until improvement stabilized. In order to avoid favoring certain classes, the algorithm alternated picking training examples between the classes. For best results, the sizes of the training sets for each class should be similar.

2) Context-Based Retraining: A cell's neighbors should be considered as context for the task of classifying the cell itself. The classification of an area, therefore, should be strongly correlated with the classification of its neighbors. This trained network was used to classify both the training map and the larger testing map. The average output values of each cell's neighbors in a 5 by 5 window around it was calculated and used as an additional feature. The network was retrained on the training map now containing neighbor classification information. This new network was then used to reclassify the testing map, taking into account neighbors' average initial classifications. This resulted in a smoothing effect that eliminated many outlier classifications. This procedure is similar to mean field inference in a graphical model [12]. This retraining process can be iterated until the desired classification consistency is achieved.

The classification results were then used to create a cost map based on the obstacle costs for each class derived from the mobility model for the UGV.

IV. Results

A. Experimental Data

The capabilities of the described feature classification algorithm are demonstrated in the following example. An unmanned helicopter was used to gather raw point cloud sensor data from the site shown in Figure 4 during the summer season. This urban environment consisted of grass, roads of various surface type, buildings of various sizes, trees of various sizes with dense foliage, and occasional other objects (cars, silos, etc).

Approximately 1 million data points in a 783 x 358 x 48 m^3 area were gathered. Each data point contained the global spacial position of that point, color measurements (RGB), and laser reflectance power measurement.

A grid cell size of 0.3 m^2 was chosen as a good compromise between map coverage and limited loss of information for this particular sensing density. At this grid cell size, the number of points in each cell ranged from 0 to 53 with an average of 1.648 points in each cell that contained at least one point (18.31 points per m^2). The density of data points varied greatly throughout the map with some areas completely lacking data.

This environment was classified using a neural network into four classes (road, grass, tree, and building). The network was trained on parts of the south-west section of the map (see Figure 3). Even though the training map was a subset of the testing map, the cells that belonged to the training set were not used in the calculations of classification rates. The labeled cells used for training constituted 9.24% of the total cells in the map containing enough data to be classified.

Results before context-based reclassification are shown in Figure 5 and final results using all presented features and two iterations of context-based reclassification are shown in Figure 6. Overhead imagery used for visual comparison and ground truth is presented in Figure 4. Notice how the reclassification created sharper feature boundaries and eliminated most outlier

Fig. 4. Overhead imagery of test site. This image was used only in identifying ground truth for our data.

Fig. 5. Classification results using all features before context-based reclassification. The light gray, medium gray, dark gray, and red areas are those classified as roads, grass, trees, and buildings respectively. Cells that did not contain enough data for a valid classification appear in white.

Fig. 6. Classification results using all features after context-based reclassification. The light gray, medium gray, dark gray, and red areas are those classified as roads, grass, trees, and buildings respectively. Cells that did not contain enough data for a valid classification appear in white.

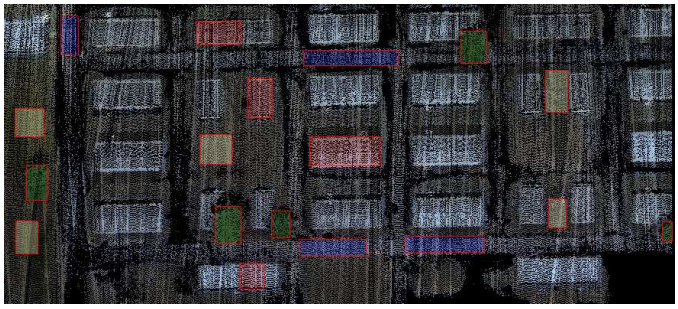


Fig. 3. Selected training examples for each class. The blue, yellow, green, and red areas contain training examples for roads, grass, trees, and buildings respectively.

points. The entire training and classification process completed in 83 seconds on a 3.2 GHz computer with 2.0GB of RAM. A large majority of this computation time was due to the inefficiency of the ground level estimation technique used.

B. Classification Performance

The classified map was compared against a fully hand-labeled map of the environment to test the performance of the algorithm (see Table I). Since ground truth labeling was performed manually, the true accuracy of the classification was likely higher than reported. Deciding on true boundaries (such as boundaries between trees and grass) proved to be very difficult and was the primary cause of error. Despite the reported error, all major terrain features were successfully classified. Additionally, our context-based reclassification technique successfully eliminated a majority of scattered misclassifications and helped generate well-defined object boundaries.

TABLE I

FINAL CLASSIFICATION RATES IN PERCENT BY CLASS USING ALL FEATURES WITH CONTEXT-BASED RECLASSIFICATION (DOES NOT INCLUDE CELLS USED IN TRAINING).

Ground Truth / Classified	Road	Grass	Tree	Building
Road	82.79	11.71	1.06	4.43
Grass	21.51	66.73	2.57	9.18
Tree	7.69	18.35	66.81	7.15
Building	4.94	0.51	0.68	93.87

C. Sensitivity Analysis

Figure 7 shows close-ups of classification results when relying only on subsets of the available features. Because of the high density of this sensor data, the PCA features played a critical role in achieving high classification accuracy. Figure 7a and Table II show the results of classifying without utilizing PCA features. Specifically, the high error rate at the boundaries of trees demonstrated the importance of PCA features when differentiating between buildings and foliage.

Similarly, results were visibly degraded because of a lack of local classification consistency (see Figure 5 and Table III) when context-based reclassification was not performed. Such spurrious classification error could corrupt a cost map and impede the planner.

TABLE II

CLASSIFICATION RATES IN PERCENT BY CLASS WITHOUT PCA FEATURE UTILIZATION WITH CONTEXT-BASED RECLASSIFICATION (DOES NOT INCLUDE CELLS USED IN TRAINING).

Ground Truth / Classified	Road	Grass	Tree	Building
Road	75.13	10.63	6.76	7.48
Grass	22.65	62.24	1.79	13.33
Tree	6.97	26.43	45.61	20.99
Building	3.15	0.39	0.28	96.18

TABLE III

CLASSIFICATION RATES IN PERCENT BY CLASS USING ALL FEATURES WITHOUT CONTEXT-BASED RECLASSIFICATION (DOES NOT INCLUDE CELLS USED IN TRAINING).

Ground Truth / Classified	Road	Grass	Tree	Building
Road	81.42	12.72	1.14	4.72
Grass	24.08	63.12	2.40	10.40
Tree	8.88	17.97	66.53	6.62
Building	6.32	0.78	2.34	90.56

Figures 7b, 7c, and 7d, show classification results when relying on only elevation, color (RGB), and signal reflectance data respectively. The absence of elevation data introduced heavy difficulties in differentiating between classes at ground level as well as detecting boundaries between various objects. Likewise, relying only on color information created confusion in classifying similarly colored objects such as grass and trees. In this particular environment, roads and building roofs were similarly colored as well. Finally, even though signal reflectance was shown to be a weak classifier when used alone, it was helpful in identifying distinct surface types in the environment. These results clearly show that even though these features were extremely effective when used in combination, no single feature type can be used as a successful classifier.

D. Integration with Planner

Cells classified as roads, grass, trees, and buildings were assigning costs of 40, 100, 180, and 255 (infinite) respectively in order to create a cost map that modeled the capabilities of a particular UGV. A path through several way points specified by the user was then planned using the D* path-planning algorithm (see Figure 8) [13].

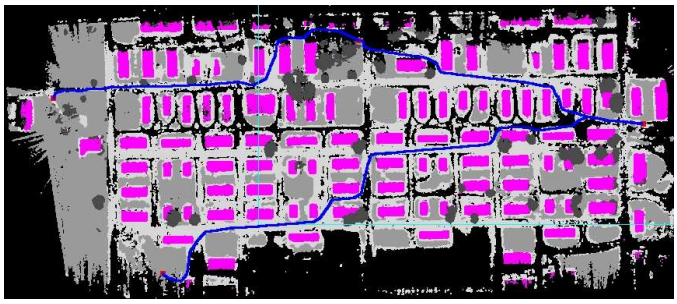


Fig. 8. Planned path through environment for the UGV: notice the strong tendency toward roads.

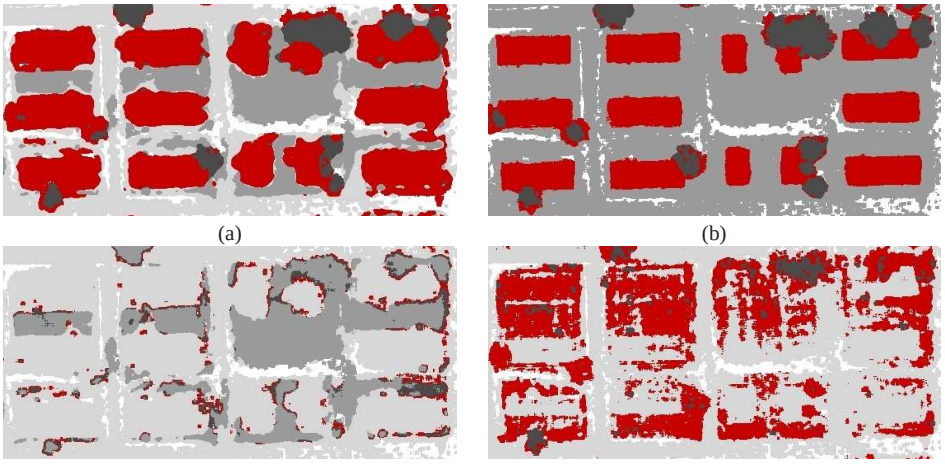


Fig. 7. Sensitivity of classification performance to withholding PCA features (a), relying only on elevation features (b), relying only on color (RGB) features, (c), and relying only on signal reflectance features (d). The light gray, medium gray, dark gray, and red areas are those classified as roads, grass, trees, and buildings respectively. Cells that did not contain enough data for a valid classification appear in white.

V. Conclusions

In this work, we addressed the problem of performing overhead terrain classification to support ground vehicle navigation. We proposed a two-step approach to classifying high-density, colorized, 3-D laser data. In the first step, 2-D and 3-D features were extracted and supervised classification through a neural network was performed. In the second step, the classification context was taken into account by using the classification results as additional features in an approach similar to mean field inference. We showed how such classification results could be used in computing cost maps to performed path planning for an UGV in a dense environment.

We presented results from an urban setting containing two story barracks, grass, and trees with dense canopy. We demonstrated the usefulness of our two-step approach in handling complex 3-D scenes and demonstrated the relative importance of each feature. The classification performance was evaluated against ground truth extracted from an independent overhead data source.

By introducing new features based on local point distributions and by iteratively utilizing classification context information, we obtained high classification rates and accurate 3-D structures delineation. We are currently testing our algorithm on larger data sets and in various environments as well as comparing the computed planned paths against paths executed by an UGV without an a priori map.

Acknowledgment

We would like to thank Omead Amidi for providing the helicopter sensor data that was used throughout this work.

References

[1] A. Stentz, A. Kelly, P. Rander, H. Herman, and O. Amidi, "Real-time, multi-perspective perception for unmanned ground vehicles," in *Proceedings of the Association for Unmanned Vehicle Systems International*, 2003.

[2] N. Vandapel, R. R. Donamukkala, and M. Hebert, "Experimental results in using aerial ladar data for mobile robot navigation," in *Proceedings of the International Conference on Field and Service Robotics*, 2003.

[3] C. Rasmussen and T. Korah, "On-vehicle and aerial texture analysis for vision-based desert road following," in *Proceedings of the IEEE International Workshop on Machine Vision for Intelligent Vehicles*, 2005.

[4] A. P. Charaniya, R. Manduchi, and S. K. Lodha, "Supervised parametric classification of aerial lidar data," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2004.

[5] T. Knudsen and A. A. Nielson, "Detection of buildings through multivariate analysis of spectral, textural, and shape based features," in *Proceedings of IGARSS*, 2004.

[6] G. Cao, X. Yang, and Z. Mao, "A two-stage level set evolution scheme for man-made objects detection," in *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, 2005.

[7] N. Vandapel, D. Huber, A. Kapuria, and M. Hebert, "Natural terrain classification using 3-d ladar data," in *Proceedings of the IEEE International Conference on Robotics and Automation*, April 2004.

[8] C. Dima, N. Vandapel, and M. Hebert, "Classifier fusion for outdoor obstacle detection," in *Proceedings of the IEEE International Conference on Robotics and Automation*, April 2004.

[9] D. McKeown, S. D. Cochran, S. J. Ford, J. C. McGlone, J. A. Shufelt, and D. A. Yocum, "Fusion of hydice hyperspectral data with panchromatic imagery for cartographic feature extraction," in *Proceedings of the IEEE Transaction on Geoscience and Remote Sensing*, May 1999, pp. 1261–1277.

[10] C. Scrapper, A. Takeuchi, T. Chang, T. Hong, and M. Shneier, "Using a priori data for prediction and object recognition in an autonomous mobile vehicle," in *Proceedings of the SPIE Aerosense Conference*, April 2003.

[11] N. Vandapel, R. R. Donamukkala, and M. Hebert, "Quality assessment of traversability maps from aerial lidar data for an unmanned ground vehicle," in *Proceedings of the IEEE/JRS International Conference on Intelligent Robots and Systems*, October 2003.

[12] M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," *Machine Learning*, vol. 37, pp. 183–233, 1999.

[13] A. Stentz, "Optimal and efficient path planning for partially-known environments," in *Proceedings of the IEEE International Conference on Robotics and Automation*, May 1994, pp. 3310–3317.