

Fusing Laser Reflectance and Image Data for Terrain Classification for Small Autonomous Robots

Keith Sullivan
Exelis, Inc.
McLean, VA USA

Wallace Lawson and Donald Sofge
Naval Research Laboratory
Washington, DC USA

Abstract—Knowing the terrain is vital for small autonomous robots traversing unstructured outdoor environments. We present a technique using 3D laser point clouds combined with RGB camera images to classify terrain into four pre-defined classes: grass, sand, concrete, and metal. Our technique first segments the point cloud into distinct regions and then applies a simple classifier to determine the classification of each region. We demonstrate three classification and four segmentation algorithms on five outdoor environments. Classification and segmentation algorithms which use more information outperform information poor combinations.

I. INTRODUCTION

Autonomous robot navigation through outdoor environments is challenging due to the difficulty representing the unstructured environment. While it is possible to extract geometric models in smooth environments (i.e., desert, planetary exploration [1], highway [2]), difficulties arise when dealing with environments that are not describable via piecewise smooth surfaces such as grass and bushes. The porous nature of natural environments encourages the use of laser point clouds rather than reconstructing the environment with planar surfaces. Sensing, modeling, and interpreting such environments is critical for successful outdoor autonomous navigation and will require multiple, interlocking tasks. One task is terrain classification, which is the process of associating regions of terrain with well-defined categories such as grass, sand, or concrete.¹

We are interested in developing terrain classification algorithms for small autonomous mobile robots which have a low perspective, limited power, and limited payload capacity (see Fig. 1). These restrictions limit us to low power, lightweight sensors, and a maximum range of approximately 5 meters. Contrast these robot characteristics to typical terrain classification work which uses large autonomous ground vehicles with sensors mounted high above the ground.

Terrain classification for small autonomous robots provides information to navigation systems, which when combined with a priori assumptions about traversability (e.g., it's more efficient to walk across concrete than through sand), allows the robot to intelligently plan a local path through the environment. Additionally, knowledge of the current terrain can guide on-line gait selection for legged robots. Static locomotion al-

¹This paper is not concerned with terrain characterization [3], which aims to identify characteristics of the terrain which affect traversability such as slickness and firmness.

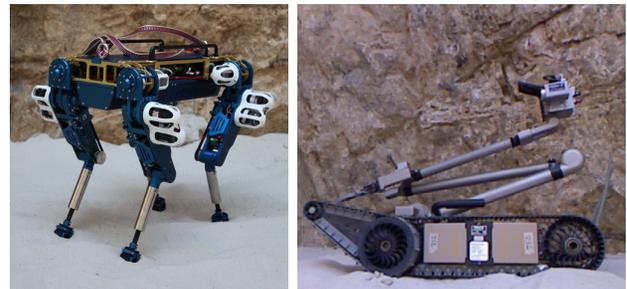


Fig. 1. Our recently acquired quadruped robot and iRobot Packbot.

gorithms completely ignore the robot's motion (e.g., Center of Mass projection technique [4]), while dynamic algorithms consider the robot's acceleration (e.g., Center of Pressure [5] or Zero Moment Point [6]). However, both static and dynamic locomotion use an overly simplified terrain model that does not account for terrain variability. Our work aims to guide the locomotion algorithm by providing information about the current material the robot is moving through, thus (potentially) improving both static and dynamic stability.

Our approach to terrain classification combines information from an on-board camera and 3D laser scanner. After partitioning the laser point cloud, we classify each segment by exploiting different spectral reflectance properties of different materials. In particular, chlorophyll [7] and metal strongly reflect near-IR light while concrete does not reflect nearly as strongly.

II. RELATED WORK

A large body of work studied terrain classification for large, wheeled robots where the goal was to identify areas where the vehicle could drive. Researchers typically use a combination of laser and camera data [8], [9] to classify terrain into predefined classes [10], [11]. However, wheeled vehicles offer the ability to use non-traditional sensors such as vibration sensors [12] and microphones [13]. Additional work has focused on identifying non-foliage obstacles within laser scans with the intent of driving through foliage [14], [9], [15]. Finally, Manz et al. fused vision and LIDAR data for intersection detection in rural road networks [16].

State-of-the-art research via the DARPA learning locomotion project [17], [18] showed impressive results for developing terrain classification algorithms for legged robots. However,

Algorithm 1 Terrain Classification Algorithm

Require: Camera image I **Require:** Registered 3D laser point cloud P

- 1: $S \leftarrow \mathcal{S}(P, I)$
 - 2: **for all** $s \in S$ **do**
 - 3: $p \leftarrow$ Points from P which are contained in s
 - 4: $c \leftarrow \mathcal{C}(p, I)$
 - 5: Assign all point in p class c
 - 6: **end for**
-

these robots used external sensors and pre-computed digital terrain models to generate exact state information, thus reducing the locomotion problem to a planning problem. In truly autonomous systems, the robot’s perceptual system should generate this information to guide its locomotion and high-level path planning decisions. Outside of the DARPA learning locomotion project, minimal work has focused on terrain classification for legged robots. Hoepflinger et al. investigated using haptic feedback in a quadruped for terrain classification [19], [20]. Belter and Skrzypczynski used a laser scanner to create a 2D height map for footfall placement for a hexapod robot [21].

Closest to our work, Wrum et al. developed a classifier for distinguishing between grass and concrete using laser intensity [22]. Similarly, Kirchner et al. used laser intensity to classify materials in an industrial setting [23].

III. METHOD

Our terrain classification algorithm partitions a 3D laser point cloud into unique segments and then assigns a classification to each region. Given our goal of determining traversable terrain for autonomous robots, we only consider concrete, sand, grass, and metal as possible terrain classes. The high-level algorithm is straightforward by design with the complexity coming in the segmentation and classification algorithms presented later.

Our approach starts with a camera image and registered 3D laser point cloud (see Algorithm 1). First, we use a segmentation algorithm \mathcal{S} to segment the 3D point cloud into a set of similar regions S . For each region s , we collect the laser points that fall within s and then apply a classification algorithm \mathcal{C} to determine the class of these points.

A. Segmentation Algorithms

We developed four different segmentation algorithms to segment the laser point cloud (Line 1 in Algorithm 1). The first two use information only from the point cloud while the second two algorithms combine information from the point cloud and the registered RGB image. The simplest segmentation algorithm, *Laser*, splits the 3D point cloud into uniform square patches of size 7×7 . The *Intensity* algorithm splits the point cloud into segments with similar laser intensity values: two points are considered in the same segment if their intensity values are within a threshold β of each other (β was set to 1.0 for the experiments).

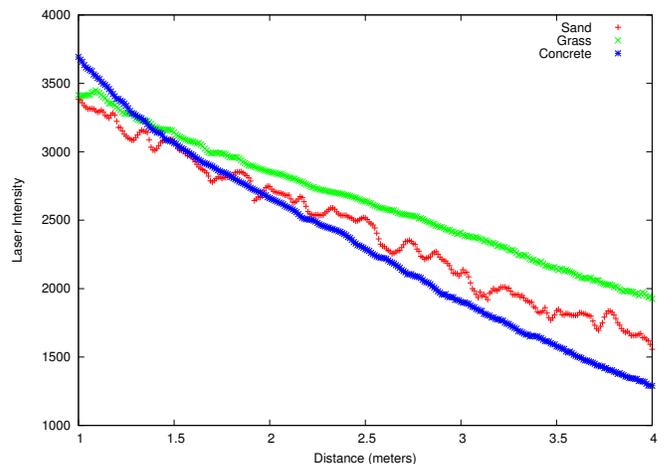


Fig. 2. Average intensity values for grass, concrete, and sand.

Algorithm 2 SimpleClassifier (p, M)

```
error  $\leftarrow$  0
for all  $m \in M$  do
   $d \leftarrow$  DISTANCE( $p$ )
   $error[m] \leftarrow |p.intensity - m[d]|$ 
end for
return arg min  $error$ 
```

Both the *Cloud* and *Color* algorithms use the registered RGB image to help segment the cloud. The *Cloud* algorithm projects each 3D laser point into the 2D image to determine the associated RGB pixel value. The collected RGB values are then segmented by similar values (similar to the *Intensity* algorithm). The final segmentation algorithm, *Color*, uses the color segmentation algorithm from Felzenszwalb and Huttenlocher [24] to determine unique color segments. The color segmentation algorithm defines a predicate for determining the boundary between two regions using a graph-based representation of the image. Once we have the unique color segments, we project the laser cloud into the coordinate frame of the image and determine which laser points fall within each segment.

B. Classification Algorithms

Our classification algorithms determine the appropriate terrain class based on laser intensity and distance (Line 4 in Algorithm 1). The first classification algorithm works on individual points, while the remaining two classification algorithms assign a class to an individual segment (determined using the previously discussed segmentation algorithms). The three classification algorithms are based on collected data for each material of interest. Using a Hokuyo UTM-30LX laser scanner, we collected 1,000,000 - 3,000,000 readings over different distances and incidence angles for each terrain of interest, and then binned the data into 1 cm buckets and computed the average for each bucket (each bucket contained at least 500 points). Fig. 2 shows the resulting distance-based intensity means for multiple materials.

Algorithm 3 MajorityClassifier (S, M)

```
votes  $\leftarrow$  0
for all  $p \in S$  do
   $cls \leftarrow$  SIMPLECLASSIFIER( $p, M$ )
  votes[ $cls$ ]  $\leftarrow$  votes[ $cls$ ] + 1
end for
return arg max votes
```

Algorithm 4 ProbabilityClassifier (S, D)

```
function PROBABILITY( $p, D$ )
   $prob \leftarrow$  0
  for all  $dist_s \in D$  do
     $d \leftarrow$  DISTANCE( $p$ )
     $prob[d] \leftarrow$  SAMPLE( $dist_s[d]$ )
  end for
  return arg max  $prob$ 
end function

votes  $\leftarrow$  0
for all  $p \in S$  do
   $cls \leftarrow$  PROBABILITY( $p, D$ )
  votes[ $cls$ ]  $\leftarrow$  votes[ $cls$ ] + 1
end for
return arg max votes
```

Our first classification algorithm, *Simple*, classifies an individual laser point p based on which material curve is the closest (see Algorithm 2, where the function DISTANCE computes the Euclidean distance from the laser scanner to the point). The *Majority* algorithm works with an individual segment S and the set of intensity curves for each material M : we classify all the points within a segment using the *Simple* algorithm, and then assign the segment’s class as the plurality of individual points (Algorithm 3).

Like *Majority*, *Probability* uses a plurality of votes to determine a segment’s class, but uses probability distributions to determine a point’s class rather than distance to the closest curve. Instead of using the computed average for a bin as before, we determined an appropriate distribution for each bin. Preliminary analysis suggested the data followed a normal distribution, so we fit a distribution to each bin – selected from normal, Cauchy, log normal, gamma, Weibull, and logistic – and assigned the bin the distribution with the lowest Kolmogorov-Smirnov statistic. This results in a set of distributions D for each 1 cm bucket per material. Algorithm 4 classifies each point in a segment S by computing the material with the highest probability from the computed probability distributions. The function SAMPLE samples from the appropriate distance-based probability distribution.

IV. EXPERIMENTS

The goal of our experiments was to determine how much information is required to accurately classify terrain. To that end, we tested our four segmentation algorithms and three

Dataset Name	Length (seconds)	Number of Point Clouds
Sidewalk1	79	23
Sidewalk2	60	20
Outdoor1	45	42
Outdoor2	70	67
Outdoor3	45	43

TABLE I
INFORMATION ABOUT EACH DATASET USED IN THE EXPERIMENTS.

classification algorithms on five outdoor datasets (Fig. 8). Table I shows the number of pointclouds and length of time for each dataset. The datasets were collected with a Hokuyo UTM-30LX laser scanner mounted on a tilt unit, and a RGB camera at 640x480 resolution. The 3D point cloud was limited to a field of view to match the camera, and, due to the tilt unit’s period, resulted in clouds containing approximately 25,000 points. The sensors were mounted on a cart 38 cm above ground. For the Sidewalk datasets, the cart remained stationary while in the Outdoor datasets, the cart was pushed at approximately 0.75 meters per second.

To determine accuracy of our terrain classification algorithm, we hand labeled images from each dataset which corresponded to the point clouds we classified. After computing the classifications, we projected each point into the hand-labeled camera image and determined if the point was correctly classified. Accuracy results are the percentage of correctly classified points across all the point clouds within the dataset.

Fig. 3–7 show the results for all combinations of environment, segmentation algorithm, and classification algorithm. In each figure, every cluster of columns represents a single segmentation algorithm while each column corresponds to an individual classification algorithm. The results show that segmentation and classification algorithms which take advantage of the available information perform best. In particular, the *Color* segmentation algorithm outperformed the other segmentation algorithms especially when combined with the *Probability* classification algorithm. This combination of classification and segmentation algorithms correctly classified terrain 75 - 95% of the time.

Algorithms which do not utilize all available information performed poorly. The *Simple* classification algorithm faired poorly (less than 10% classification accuracy) across all environments since it considers the entire cloud as single segment. One reason the *Simple* algorithm performs poorly is noisy data: while Fig. 2 shows clean, easily separable curves, the variance in each curve is high enough that its easy to confuse different materials particularly at shorter distances. The *Majority* algorithm performs better since it considers each segment as a whole rather than individual points; however, it still suffers from effects of noisy data. The *Probabilistic* method performs best due to its ability to consider each segment in its entirety, and the use of probability distributions accounts for noisy data.

The segmentation algorithms show similar results. The *Cloud* algorithm performs poorly since the transformation of laser points to camera pixels results in pixels not being covered due to sparseness in the transform and multiple laser points

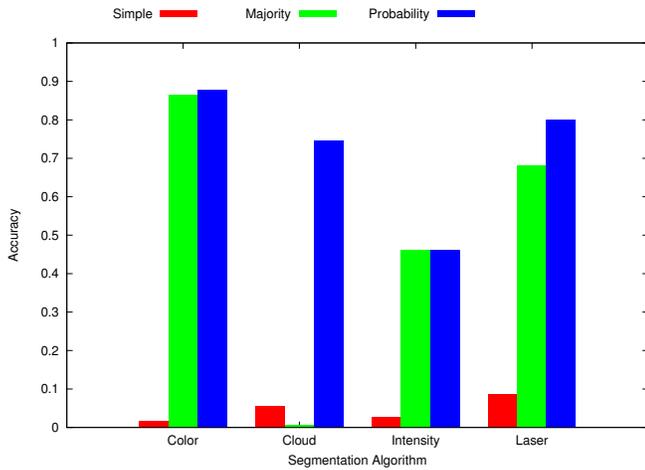


Fig. 3. Terrain classification accuracy results for the Sidewalk1 dataset.

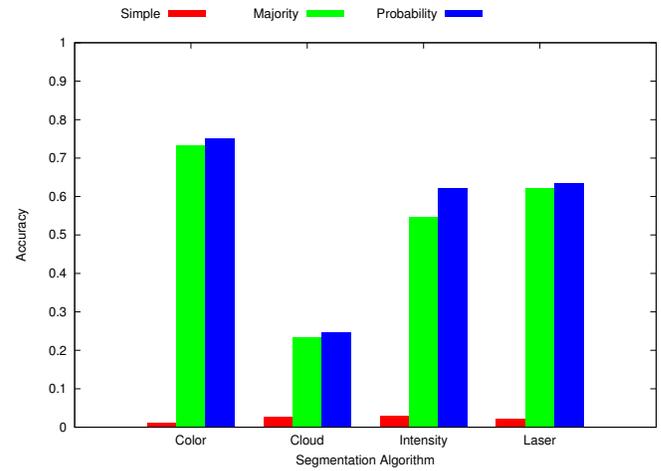


Fig. 5. Terrain classification accuracy results for the Outdoor1 dataset.

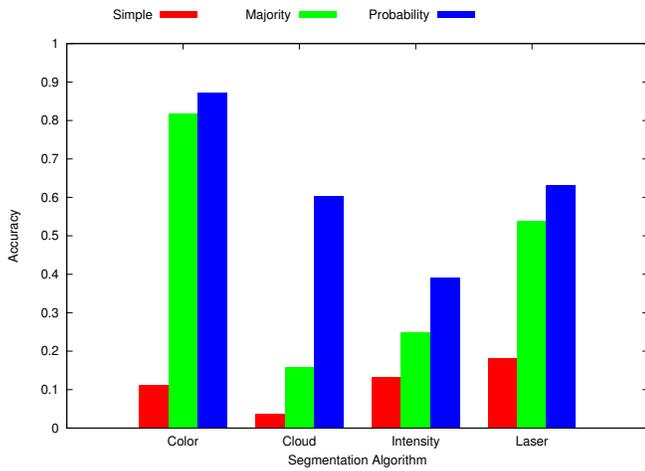


Fig. 4. Terrain classification accuracy results for the Sidewalk2 dataset.

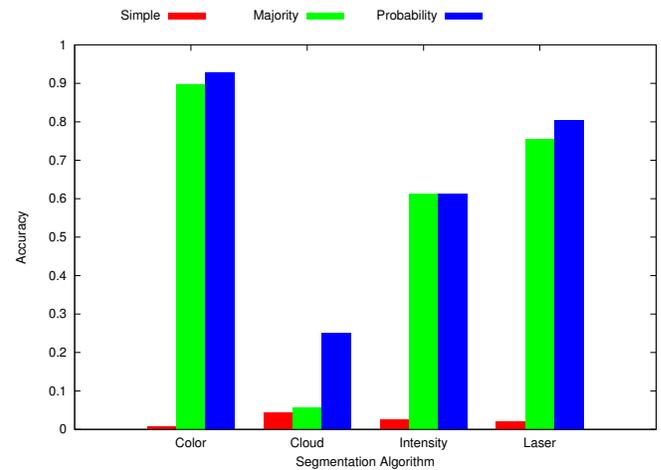


Fig. 6. Terrain classification accuracy results for the Outdoor2 dataset.

being cast to the same pixel. Segmenting via patches (*Laser*) or similar intensity values (*Intensity*) results in similar accuracy with *Intensity* performing better in some environments. However, both these approaches do not leverage the camera data, which is why the *Color* segmentation algorithm performs best: it uses the most information.

V. DISCUSSION

Why did the *Color* segmentation algorithm perform so much better than the other three segmentation algorithms? The *Color* algorithm exploited the dense information in the registered image, while the other segmentation algorithms operated on the relatively sparse data in the laser point cloud. The *Cloud* algorithm attempted to alleviate this by projecting the cloud into the coordinate frame of the image; however, the coordinate transform resulted in multiple laser points being projected onto the same pixel and many pixels not corresponding to any laser point. These results suggest that either increasing the laser point cloud density (say, by using a Velodyne laser scanner) or applying interpolation methods to the laser cloud would increase information density, thus

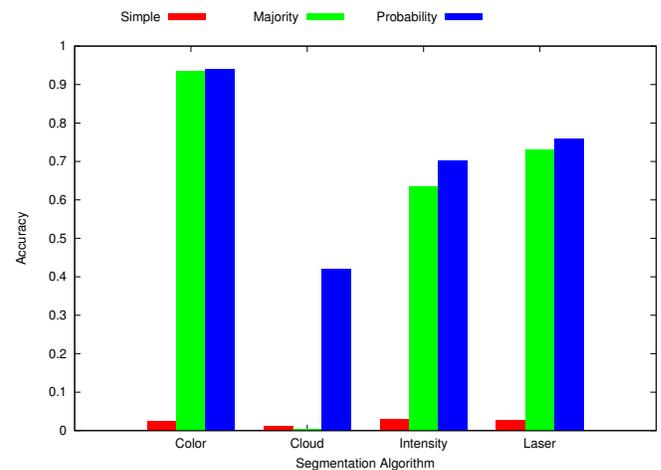
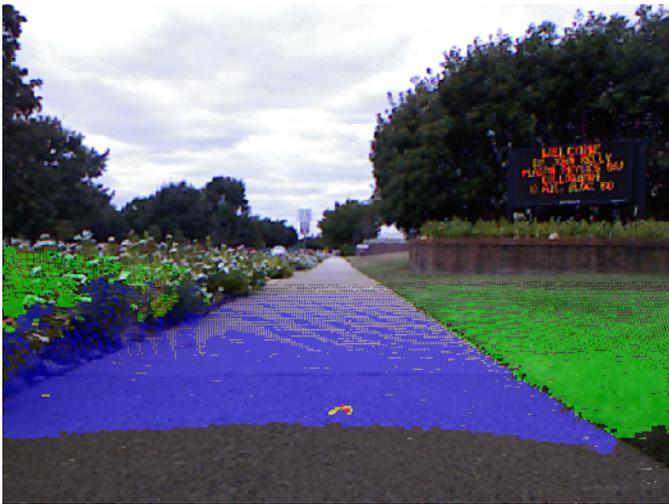
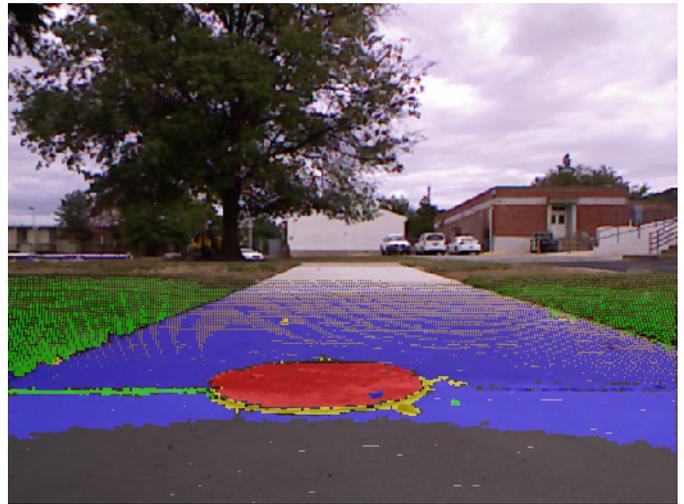


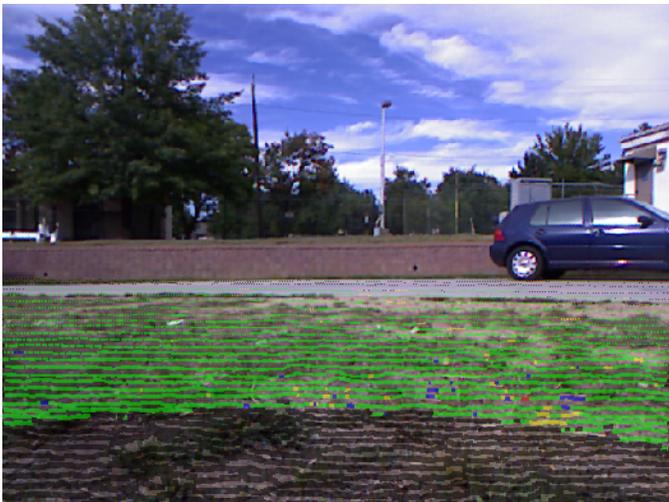
Fig. 7. Terrain classification accuracy results for the Outdoor3 dataset.



Sidewalk1



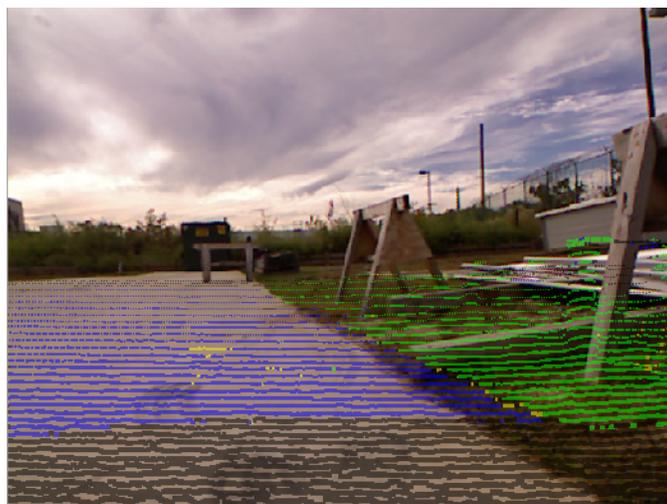
Sidewalk2



Outdoor1



Outdoor2



Outdoor3

Fig. 8. Screen shots of results. The overlaid point cloud shows the output of our approach where the colors correspond to the terrain classification decision: blue is concrete, green is grass, yellow is sand, and red is metal. In Sidewalk2 the red indicates a manhole cover, not a false positive classification.

improving terrain classification. Part of our future work will examine terrain classification accuracy as a function of laser point cloud density.

VI. CONCLUSIONS AND FUTURE WORK

We presented a terrain classification algorithm which segments a 3D laser point cloud and then classifies each segment into a pre-defined terrain class. Adding a registered RGB camera image improved classification accuracy by improving the segmentation of the laser cloud. Further improvements came from combining a voting scheme with choosing the most probable terrain class for each segment.

Future work will expand the choice of different terrains to include asphalt, rock, and non-grass vegetation. Additionally, we plan to include geometric information extracted from the point cloud to improve classification accuracy.

ACKNOWLEDGMENT

This work was performed at the Naval Research Laboratory and was funded by the US Department of Defense, Office of Naval Research under grant number N0001413WX21045, Mobile Autonomous Teams for Navy Information Surveillance and Search (MANTISS). Additional funding came from Naval Research Lab to JGT. Wallace Lawson was also funded by the Naval Research Laboratory under a Karles Fellowship. The views, positions and conclusions expressed herein reflect only the authors opinions and expressly do not reflect those of the US Department of Defense, Office of Naval Research, or the Naval Research Laboratory.

REFERENCES

- [1] S. Goldberg, M. Maimone, and L. Matthies, "Stereo vision and rover navigation software for planetary exploration," in *Proceedings of the IEEE Aerospace Conference*, vol. 5, 2002, pp. 2025 – 2036.
- [2] C. Thorpe, M. Hebert, T. Kanade, and S. Shafer, "Vision and navigation for the carnegie-mellon navlab," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 3, pp. 362 – 373, May 1988.
- [3] L. Ojeda, J. Borenstein, G. Witus, and R. Karlsen, "Terrain characterization and classification with a mobile robot," *Journal of Field Robotics*, vol. 23, no. 2, pp. 103–122, 2006.
- [4] R. McGhee and A. Frank, "On the stability properties of quadruped creeping gaits," *Mathematical Biosciences*, vol. 3, pp. 331 – 351, 1968.
- [5] D. Orin, R. McGhee, and V. Jaswa, "Interactive compute-control of a six-legged robot vehicle with optimization of stability, terrain adaptability and energy," in *Proceeding of the Conference on Decision and Control*, vol. 15, Dec 1976, pp. 382–391.
- [6] M. Vukobratovic, A. Frank, and D. Juricic, "On the stability of biped locomotion," *IEEE Transactions on Biomedical Engineering*, vol. BME-17, no. 1, pp. 25–36, Jan 1970.
- [7] R. Myneni, F. Hall, P. Sellers, and A. Marshak, "The interpretation of spectral vegetation indexes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 33, no. 2, pp. 481–486, Mar 1995.
- [8] J. Macedo, R. Manduchi, and L. Matthies, "Ladar-based discrimination of grass from obstacles for autonomous navigation," in *Experimental Robotics VII*. Springer, 2001, vol. 271, pp. 111–120.
- [9] A. Castano and L. Matthies, "Foliage discrimination using a rotating ladar," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, 2003, pp. 1 – 6.
- [10] C. E. Rasmussen, "Laser range-, color-, and texture-based classifiers for segmenting marginal roads," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2001.
- [11] D. Anguelov, B. Taskarf, V. Chatalbashev, D. Koller, D. Gupta, G. Heitz, and A. Ng, "Discriminative learning of markov random fields for segmentation of 3d scan data," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, vol. 2, 2005, pp. 169–176 vol. 2.
- [12] C. Weiss, H. Frohlich, and A. Zell, "Vibration-based terrain classification using support vector machines," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems (IROS)*, Oct 2006, pp. 4429–4434.
- [13] L. Ojeda, J. Borenstein, G. Witus, and R. Karlsen, "Terrain characterization and classification with a mobile robot," *Journal of Field Robotics*, vol. 23, no. 2, pp. 102 – 123, 2006.
- [14] C. Dima, N. Vandapel, and M. Hebert, "Classifier fusion for outdoor obstacle detection," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1. IEEE, April 2004, pp. 665 – 671.
- [15] J.-F. Lalonde, N. Vandapel, D. Huber, and M. Hebert, "Natural terrain classification using three-dimensional ladar data for ground robot mobility," *Journal of Field Robotics*, vol. 23, no. 10, pp. 839 – 861, November 2006.
- [16] M. Manz, M. Himmelsbach, T. Luettel, and H. J. Wuensche, "Detection and tracking of road networks in rural terrain by fusing vision and LIDAR," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems (IROS)*, Sept 2011, pp. 4562–4568.
- [17] K. Byl, A. Shkolnik, S. Prentice, N. Roy, and R. Tedrake, "Reliable dynamic motions for a stiff quadruped," in *Experimental Robotics*, ser. Springer Tracts in Advanced Robotics, O. Khatib, V. Kumar, and G. Pappas, Eds. Springer Berlin Heidelberg, 2009, vol. 54, pp. 319–328.
- [18] J. Rebula, P. Neuhaus, B. Bonnländer, M. Johnson, and J. Pratt, "A controller for the LittleDog quadruped walking on rough terrain," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems (IROS)*, April 2007, pp. 1467–1473.
- [19] M. Hoepflinger, C. Remy, M. Hutter, L. Spinello, and R. Siegwart, "Haptic terrain classification for legged robots," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2010, pp. 2828–2833.
- [20] C. Remy, O. Baur, M. Latta, A. Lauber, M. Hutter, M. Hoepflinger, C. Pradalier, and R. Siegwart, "Walking and crawling with ALoF: a robot for autonomous locomotion on four legs," in *Proceedings of Conference on Climbing and Walking Robots*, 2010.
- [21] D. Belter and P. Skrzypczynski, "Rough terrain mapping and classification for foothold selection in a walking robot," in *Proceeding of IEEE International Workshop on Safety Security and Rescue Robotics*, July 2010, pp. 1–6.
- [22] K. Wurm, R. Kummerle, C. Stachniss, and W. Burgard, "Improving robot navigation in structured outdoor environments by identifying vegetation from laser data," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems (IROS)*, Oct 2009, pp. 1217–1222.
- [23] N. Kirchner, D. Liu, and G. Dissanayake, "Surface type classification with a laser range finder," *IEEE Sensors Journal*, vol. 9, no. 9, pp. 1160–1168, Sept 2009.
- [24] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient grad-based image segmentation," *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167 – 181, 2004.