Terrain Understanding for Robot Navigation

Robert E. Karlsen, *Member, IEEE* and Gary Witus, *Member, IEEE*

***Abstract*— This paper presents a method to forecast terrain trafficability from visual appearance. During training, the system identifies a set of image chips (or exemplars) that span the range of terrain appearance. Each chip is assigned a vector tag of vehicle-terrain interaction characteristics that are obtained from on-board sensors and simple performance models, as the vehicle traverses the terrain. The system uses the exemplars to segment images into regions, based on visual similarity to the terrain patches observed during training, and assigns the appropriate vehicle-terrain interaction tag to them. This methodology will therefore allow the online forecasting of vehicle performance on upcoming terrain. Currently, we are using fuzzy c-means clustering and exploring a number of different features for characterizing the visual appearance of the terrain.**

I. Introduction

Most, if not all, unmanned ground vehicles currently in use are teleoperated. Typically, the operator relies exclusively on visual input from a video camera to select the route and speed. Teleoperation is robust and effective. Vision processing for autonomous and semi-autonomous navigation has not matched the human operator's visual terrain understanding. Current approaches to autonomous navigation employ a wide gamut of sensors including 3D imaging LIDAR, ground penetrating radar, multi-spectral stereovision, ultrasound, and other sensor modalities to detect potential obstacles and forecast trafficability. Inspired by the ability of human operators, our research is focused on methods to assess terrain trafficability directly from image appearance. We do not address obstacle detection, which is an important, but separate cognitive process.

We are also not attempting to characterize physical properties of the terrain that are independent of the vehicle. Parameters of theoretical models for smooth, semi-infinite, homogeneous soil, such as cohesion and shear angle, are not well defined for natural terrain. Natural terrain is a complex amalgam of layers of different materials (e.g. grass and root mass or loose sand and stones over a mixture of loam, sand, clay, rocks and tree roots) each layer having spatially varying thickness, composition, and moisture gradient. Two vehicles with different ground pressures will interact with different layers of the terrain. A mobile robot was used in [1] to demonstrate the estimation of terrain properties.

We are interested in forecasting terrain trafficability for use in automated driving and navigation, e.g., route selection, decisions to cross or avoid particular terrain, and speed limits for the terrain. Route and speed selection algorithms seek to limit or minimize some combination of travel time, fuel consumption, and absorbed power from shock and vibration (a proxy for damage and wear). In this paper, we have focused on two aspects of trafficability: roughness and resistance, which are functions of the vehicle-terrain interaction.

Various researchers have worked to develop methods to forecast traversability based on estimates of geometrical properties inferred from non-contract sensors. A fuzzy-rule-based system [2] was developed for planetary rover environments to mimic human "high/medium/low" trafficability assessment based on measures of roughness, slope and distance between obstacles, computed from stereo imagery. A stereo color vision system, together with a single axis LADAR, was used to classify terrestrial terrain cover and detect obstacles in [3]. It was noted that the color-based classification system could be made more robust by considering the texture of regions and the shape features of objects. A rule-based system for terrain classification from LADAR and color camera imagery was developed in [4].

Appearance-based approaches do not estimate geometrical properties and then infer traversability. Instead, they classify the terrain appearance and then assign the associated trafficability vector measured while traversing similar terrain, reflecting terrain properties, such as friction, resistance and sinkage. The research in [5] has similar goals as our work and uses a clustering approach with color, texture and geometric features. Although further advanced in terms of implementation, the classification is binary (Go/NoGo). The approach in [6] also considers color, texture and geometric features, but uses a support vector machine classifier to predict vibration attributes.

We present an approach to automated image segmentation and terrain classification using exemplars, or small image samples, to represent the variety of terrain appearance. Each chip is assigned a set of measured vehicle-terrain interaction (VTI) parameters that describe the vehicle's performance while driving over that particular terrain. Previous work [7] has been performed in determining meaningful and robust VTI parameters, such as vehicle slip, ground resistance and terrain roughness. An exemplar-based approach was used in [8] to segment terrain into Go and NoGo regions and compared a heuristic clustering method with fuzzy c-means clustering and support vector machines.

Exemplar models assume that intact stimuli are stored in memory, and that classification or recognition is determined by the degree of similarity between a stimulus and the stored exemplars. Exemplar methods admit evolution of similarity metrics, since the entire sample is stored intact in memory and not merely a feature vector summary. Exemplar models are the most parsimonious models of categorization in terms

Robert E. Karlsen is with the U.S. Army – Tank-Automotive Research, Development and Engineering Center, Warren, MI, 48397-5000, robert.karlsen@us.army.mil.

Gary Witus is with Turing Associates, Ann Arbor, MI, 48103, witusg@umich.edu.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
29 OCT 2007	**N/A**	**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Terrain understaning for robot navigation	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
Robert E. Karlsen; Gary Witus	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
US Army RDECOM-TARDEC 6501 E 11 Mile Rd Warren, MI 48397-5000 Turning Associates, Ann Arbor, MI 48103, USa	**17535**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	TACOM/TARDEC
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
	17535

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

Postprint: This is published in: Intelligent Robots and Systems, 2007 IROS 2007, IEEE/RSJ Publication date Oct. 29 2007-Nov. 2 2007 On pages(s): 895-900 ISBN: 978-1-4244-0912-9, The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**SAR**	**6**	

of the underlying associative mechanism [9]. Exemplar based learning has been proposed as a model of human learning [10] and has since been shown to explain both human and animal visual classification performance significantly better than alternative hypotheses of feature-based and prototype-based processing [11], [12].

II. Technical Approach

A. Data Processing

The proposed learning process requires three main functions: segmenting the terrain into areas that are visually similar, measuring and computing appropriate measures of the vehicle-terrain interaction (VTI), and matching the resulting parameters to the correct image area.

All VTI measures that we are interested in have a dependence on vehicle speed and, therefore, this is an important parameter to measure accurately. We currently use a wheel encoder attached to a fifth wheel trailing the vehicle to provide the speed of the vehicle. Based on previous experiments [7], we assume that vehicle speed is linearly proportional to the voltage drop measured across the vehicle's drive motor, $v = \alpha V$. Our measure of ground resistance is inversely proportional to the constant α, with high α corresponding to low ground resistance and low α to high ground resistance, as seen in Fig. 1.

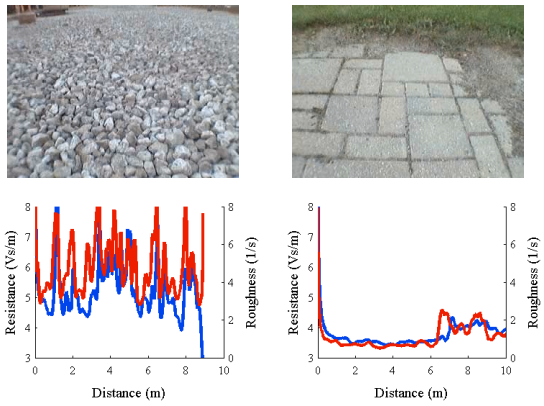

Fig. 1. Input training images and VTI parameters (resistance ($1/\alpha$) = blue, roughness (ρ) = red).

The second VTI measure of interest is ground roughness. We use the output of an accelerometer positioned over the front axle to collect disturbance data, which we assume is linearly proportional to speed, $D = \rho v$. The proportionality constant ρ is used as a measure of ground roughness, with high ρ for rough terrain and low ρ for smooth terrain.

In the absence of range information, we use the "flat earth" assumption to associate the sensor data from the vehicle to the image data that the vehicle has not traversed yet. By measuring the camera height off the ground and the distance from the front axle of the vehicle to the apparent top and bottom rows of the images, we can estimate the distance to all points in the images. For this work, we assumed that the terrain was homogenous in the horizontal direction and the data was taken and processed to keep that essentially true. A more realistic approach would identify those portions of the image that the wheels actually traverse. In addition, the vehicle was commanded to travel in a straight line. In actual operation, where the vehicle turns, where the terrain is not flat and where there are objects in front of the vehicle, more complex processing and data handling procedures will be required. Examples of image and VTI data are shown in Figure 1.

B. Image Processing

An essential function in the proposed approach is to numerically compare two image chips as to their similarity or contrast. However, there is currently no obviously correct metric for measuring this difference. This can be seen in image compression, where it is easy to measure the amount of compression and the encoding/decoding time, but difficult to measure image quality. Different image characteristics are important depending on the image content, the questions at hand, and who is looking at the image. Before an image is chopped into chips, it can be processed to balance relevant image characteristics. In principle, therefore, simple measures of the aggregate difference are all that are needed. Even so, there are many different ways to calculate the difference between two image chips. Some metrics are computed from the pixel-by-pixel difference between two chips, others are calculated from the difference in statistics computed from the individual chips.

In addition, various image processing functions, such as transformation to various color spaces or multi-resolution bandpass filtering, can be used to extract additional information. Another option is to process the images through a bank of spatial filters, such as edge and corner filters at different spatial scales and orientations, with each filter producing a single-plane output image.

Image processing can be used to remove attributes of the imagery that can lead to misclassification, such as noise, color balance, and brightness. Automated features in cameras attempt to compensate for different lighting conditions and produce more life-like imagery. However, they are sometimes only partially successful, resulting in a time lag before compensation or applying the correction over the entire image when only a portion of the image needs correction. We were interested in applying a transform to the imagery such that consistent results would be obtained, irrespective of the lighting conditions. As an initial attempt at separating the luminance component from the color component, we tried the HSV (hue, saturation, value) color space. Although this resulted in some improvements over the RGB color space, the HSV system is unsatisfactory due to the cyclical nature of hue and the fact that HSV is far from perceptually uniform. This led to the implementation of an L*a*b* color space transform, where L^* refers to luminance and the a^* and b^* components encode the color information (red/green and yellow/blue color opponency, respectively). The transformation to L*a*b* is nonlinear, resulting in components that are closer to perceptually uniform. All the results depicted in this paper use the L*a*b* color space transformation.

Our image sequences do show evidence of spurious color effects, most likely due to automated features of the camera system. We are considering ways to alleviate this problem. In previous work [13], we tried having the system learn the color changes. For the current paper, we decided not to use color as a feature, even though it is an important visual cue. However, ideally one would like a vision system that is able to recognize terrain even in the presence of changing lighting conditions or color shifts.

Since membership in a terrain class is considered to be a bulk property of a local region, not a point-location property, we know that texture [14] will play an important role in our analysis. We have explored two primary measures of texture, the standard deviation and entropy. For the former, we created a texture image by computing the standard deviation over all patches of a given shape, centered on each pixel in the image. Because this also picked up the strong edges of objects and other texture boundaries, we employed a Canny edge detector to find these strong edges and suppress them in the texture image. Fig. 2 shows examples of texture for the images in Fig. 1. Each texture image has three planes, with the red and green planes containing the output of horizontal and vertical one-dimensional filters, respectively. The blue plane is computed from a two-dimensional standard deviation filter.

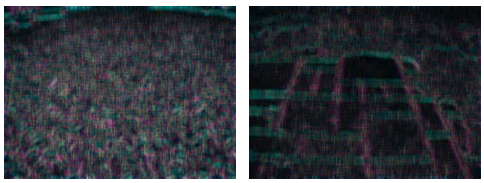

Fig. 2. Texture images computed via standard deviation with 11-pixel filter.

We also computed a texture measure based on entropy ($\Sigma\, x \log(x)$). Examples of this are shown in Fig. 3, where the different color planes correspond to different resolutions (5, 11, 17 pixels). In this case, we did not use Canny edge detection to suppress strong edges, since they appeared less pronounced.

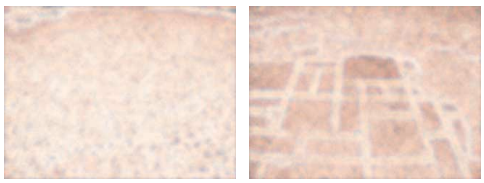

Fig. 3. Texture images computed via entropy.

For the current paper, based on a number of runs with a reduced data set, we found that the most effective features consisted of the mean luminance (L^*), the mean of the standard deviation texture images with a two-dimensional filter at resolutions 5 and 11 pixels, and the standard deviation of the entropy texture images at resolutions 5 and 11 pixels. It was found that the color information coded in a^* and b^* provided little to classification accuracy and actually degraded the results in most cases. The addition of the horizontal and vertical standard deviation filters also did not help the results significantly. The median of the image plane chips was also explored, but added little to the classification accuracy. If an even smaller set of features was desired, the texture at resolution 11 pixels was more important than the resolution at 5 pixels. Eliminating the mean luminance caused only a small decrease in classification accuracy and would result in a feature vector computed entirely from grayscale texture.

C. *Learning Algorithm*

The software is organized into two routines: one for offline training and one for online learning, although the same algorithm could be used for both. At the end of the offline training, an exemplar bank is created that contains image and parameter identification data. During online learning, the exemplar bank is updated.

If an image difference metric based on statistical measures is used, one can employ one of the various learning algorithms, such as neural networks, fuzzy logic or clustering. The current system uses a fuzzy c-means clustering (FCM) algorithm [15]. A heuristic method for learning was developed in [8] that is suitable for online learning using either direct chip or statistical comparison.

The user must provide a set of representative training images and associated vehicle-terrain interaction (VTI) parameters. Ideally, the training images would be drawn from the same distribution as the downstream application images. In practice, it may not be possible to ensure this. The effect that different conditions between the training image set and test/application image set, such as different terrain, foliage, season, lighting, and weather, has on segmentation and parameter identification performance is a question for empirical investigation.

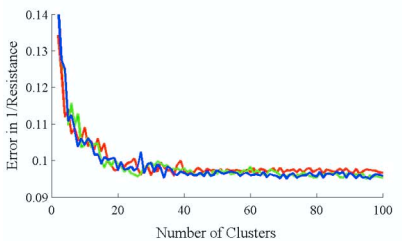

Fig. 4. Error in α, using 5 features (red), 4 features (green) and 3 features (blue).

For the offline portion of the system, we are using the most basic form of FCM clustering with spherical clusters of the same size. Future work may look at non-spherical clusters of different sizes. As described earlier, our initial data set consisted of three features computed from each of the three L*a*b* image planes and twelve texture planes (nine multi-resolution standard deviation and three multi-resolution entropy). However, it was determined through experimentation that a five-element feature vector would suffice and could even be reduced to three or four elements with little loss of accuracy, as seen in Fig. 4.

The FCM algorithm provides a list of cluster centers and a matrix with the distance of each data point to each

cluster. Since the cluster centers have no direct connection to the data, we move each cluster center to the location of the nearest exemplar in feature space and recompute the distances. From the resulting matrix, we can identify which exemplar (cluster center) should be assigned to each image chip in the data.

Since each chip has an associated set of VTI parameters, this gets naturally carried along with the corresponding exemplar. However, for the results in this paper, instead of using the VTI parameters for the particular exemplar, we have averaged the parameters over all chips within the cluster and used the resulting values to tag each exemplar. This helps smooth out some of the variability in the data. We modified existing computer code [16] for our implementation of the FCM algorithm.

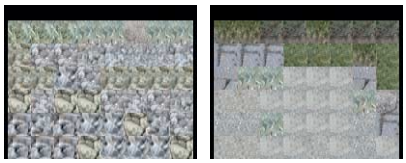

Fig. 5. Reconstruction of images with exemplars.

There is no obvious and correct way to represent the different segments for purposes of visualization. To provide direct visual insight into the basis for the segmentation, the software replaces each image chip with the exemplar chip to which it is associated. By using the exemplar chips themselves, the visualization image shows what the exemplars look like, and which image chips they are associated with. Comparing the visualization to the original image gives prima fascia evidence of the credibility of the segmentation. See Fig. 5 for reconstruction of the images in Fig. 1, which is based on the image feature set discussed earlier that does not include color.

IV. Results

The data collection that forms the basis for the results in this paper consists of 34 runs over different types of terrain, such as concrete, asphalt, dirt, grass, bricks, gravel, sand and rocks. Each run was between 15-25 seconds, with periods at the beginning and end where the vehicle was motionless; vehicle motion occurred for between 10-15 seconds. The vehicle-terrain interaction (VTI) parameters that we are currently exploring are for quasi-steady state conditions and so we are not considering effects due to acceleration or deceleration. For each run over a given terrain segment, we also had another run in the opposite direction.

For this paper, as in a previous work [13], we chose five runs to train the system and used the companion runs in the opposite direction for testing. Terrain 1 consisted of rocks, terrain 2 was brick pavers and grass, terrain 3 was cement and grass, terrain 4 was asphalt and cement, and terrain 5 was rough sand.

We smoothed the voltage data with a Hamming-like filter of length 0.5 s. The acceleration data was converted to disturbance by a Hamming-like standard deviation filter of length 0.5 s. We used a heuristic algorithm to remove spikes from the wheel encoder data, which was then filtered by a derivative filter of length 0.5 s to produce vehicle speed.

We extracted every fifth frame in the training sequences, resulting in 325 images, and every frame in the test sequences, resulting in 1575 images. We cropped the 320x240 images to 200x160 by taking 60 pixels off each side and 80 pixels off the bottom. We chose an image chip size of 24x24, which resulted in 48 chips per frame. The resulting training set had 15,520 samples and the test set had 75,560 samples. A five-element feature vector was computed for each of the image chip samples in the training and test sets.

We chose to use 40 clusters for this test, although test error results were relatively flat beyond 20 clusters, as seen in Fig. 4. This resulted in a training error of 6.2% and 34.2% and a test error of 9.7% and 47.0%, for the ground resistance and ground roughness predictions, respectively. The error was computed as the absolute difference between prediction and measurement divided by the average of the two. The large error in ground roughness can be attributed in large part to the significant variation in ρ when traveling over even homogenous rough terrain, such as in Fig. 1. In fact, the average ratio of mean-normalized variation between the roughness and resistance parameters was about 5, similar to the ratio of errors.

We also ran the same data through a decision tree algorithm [17, 18] and found that the error on the training set was 5.9% and 30.9%, and the error on the test set was 9.3% and 50.3%, for the ground resistance and ground roughness, respectively. Based on these results, we are anticipating that the decision tree algorithm may provide a more suitable online learning algorithm, while maintaining the classification performance of the FCM clustering algorithm.

Fig. 6. Measured vehicle-terrain interaction parameters (α = center and ρ = right).

We implemented a color-coding scheme to graphically illustrate the predicted VTI measures using the image data. The color red corresponds to the least desirable end of the parameters (0.2 for α and 2.7 for ρ), while green corresponds to the most desirable end of the parameters (0.3 for α and 0.7 for ρ). Quantities outside that range were truncated. Image chips that were determined to be too far from any exemplar were color-coded blue, with those having desirable VTI parameter values being cyan-hued and those that were least desirable were magenta-hued. The unknown chips were included in the error computations.

Fig. 6 shows an example of extrapolating the measured values for the VTI parameters to specific image locations via the "flat earth" assumption, which are input to the FCM algorithm for the training image on the left. The center image contains the terrain resistance parameter and the right

image contains the terrain roughness parameter. This figure illustrates where errors can enter the process: synchronizing the onboard data with the image data. Errors can enter due to faulty range estimations, but here the terrain is fairly flat and the problem is due to distances being computed from the front axle of the vehicle. The terrain resistance is maximized when both the front and rear tracks are on the terrain, while the roughness manifests when the front track encounters the boundary. This lag between terrain roughness and resistance can also be seen in the right plot of Fig. 1.

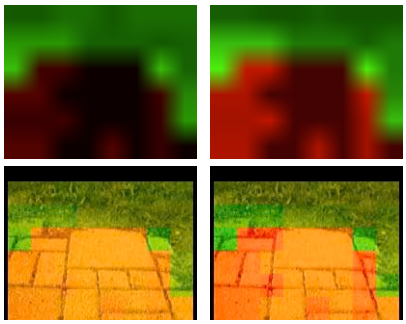

Fig. 7. Predicted vehicle-terrain interaction parameters (top) and color-coded images (bottom) (α = left and ρ = right).

Fig. 7 shows a prediction for the VTI parameters for the training image in Fig. 1 along with the color-coding scheme. The terrain resistance images are on the left and the terrain roughness images are on the right. Note that the predictions are not very accurate, they show the pavers being rough with high resistance and the grass being smooth with low resistance. The error can be traced with the aid of the reconstruction image on the right side of Fig. 2, where a poor choice for exemplars is shown for the image chips. In this case, our training database did not contain enough samples of brick pavers in different lighting conditions. In fact, the exemplar bank contained only two exemplars from the paver portion of the dataset.

Fig. 8. Test image reconstruction (top right) and VTI predictions (bottom, α = left and ρ = right).

Fig. 8 shows data from the vehicle being run over rough rocks. Here the results are generally good and correspond to expectations for both the terrain resistance and the terrain roughness. Because of the high variability in the rock and sand images, they tend to dominate the exemplar bank, with 11 exemplars each out of 40. This is reflected in the reconstruction, which is reasonably close to the original. Exemplars derived from cement portions of the images were next with 8 exemplars and those with grass had 7 exemplars. Note the color variations within the reconstruction image in Fig. 8, which is due to an absence of a color element in the feature vector. In previous work, where color was a part of the feature vector, the reconstruction was more accurate in regards to color, but the overall VTI parameter prediction was less accurate.

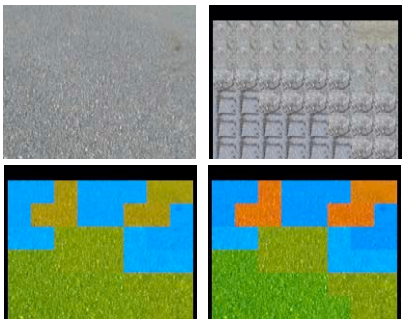

Fig. 9. Test image reconstruction (top right) and VTI predictions (bottom, α = left and ρ = right).

Fig. 9 shows results for an image that contains asphalt. The system provided erroneous results for this whole image sequence since the exemplar bank only contained one exemplar derived from asphalt, likely due to a shortage of asphalt images in the training sequences. As the reconstruction image shows, the system tended to pick sand exemplars as the best match, which resulted in modest agreement with the terrain resistance parameter and poor agreement for the terrain roughness parameter. The blue areas indicate that the image patches were too far from any exemplar, but the closest exemplar was one that was average in regards to both ground resistance and roughness. The reconstruction image in fact indicates that the top part of the resistance image would have been yellow and the top part of the roughness image would have been orange.

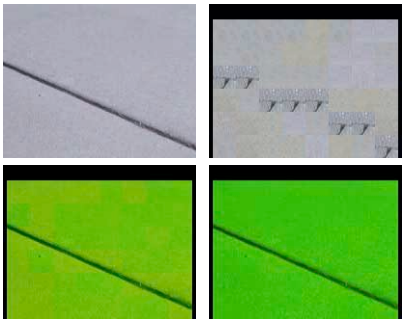

Fig. 10. Test image reconstruction (top right) and VTI predictions (bottom, α = left and ρ = right).

Fig. 10 shows some interesting results for the cement images, which generally had good agreement with reality, although the cracks were mistaken for pavers, which still resulted in accurate predictions since pavers and cement have the same VTI characteristics. In past work [13], with

less emphasis on texture, the cracks were often mistaken for rocks, resulting in poor agreement in those specific portions.

V. Findings and Observations

This paper has demonstrated an approach to image-based terrain segmentation using exemplars, as applied to vehicle-terrain interaction (VTI) prediction. Exemplars provide a simple way to represent the characteristic color/luminance and spatial patterns of terrain. Since the exemplars are drawn from training images in such a way as to span the appearance of the training images, they are well suited to represent the variations of appearance without an a priori model of terrain appearance. Preliminary results indicate the approach has potential to segment terrain in a manner that is consistent with subjective perception. The segmentation appears to provide some robustness over changes in lighting, specific terrain, and automatic camera gain and contrast adjustments, but still needs some additional work. We continue to explore methods to compensate for automatic gain and color distortions.

Although humans do not have a specific range sensing capability, they use many clues and past experience to infer estimates of range from the environment and to recognize obstacles. The current "flat earth" assumption is not viable for real-world application. So, until we can replicate a human's image-based approach, solutions include using internal sensors to measure pitch and roll and then correct for them, although this just provides a correction for the "flat earth" assumption. More costly methods would use a stereo camera system or laser range finder, which are already available on many autonomous vehicles.

Other subjects to explore in the current system include shape filtering and multi-resolution processing, which could yield improvements to the clustering without much change to the existing architecture. Implementing bandpass multi-resolution techniques, such as wavelets or Gaussian-Laplacian pyramids, or other more complex texture metrics [14] may also be fruitful. In addition, we can analyze the spatial organization of image variance to differentiate structure from texture. Tracking image chips or using an evidence grid would provide multiple looks at the same terrain, something not done now.

Based on the experience in [18], we are now exploring the use of a decision tree algorithm [17] for both the training and online portions of the system. Preliminary tests indicate that the decision tree algorithm provides comparable offline performance to the fuzzy c-means algorithm, with shorter training times. Future work includes investigating the ability to add new exemplars to the data bank without retraining the entire system, something difficult to achieve with fuzzy c-means (FCM) clustering. While the heuristic online algorithm [8] worked reasonably well, it was less accurate than FCM clustering and generated substantially more exemplars. Using the decision tree algorithm, we can partition the data, based not only on the independent variables, but also on the dependent variables. This may allow the discovery of hidden structure in the independent variables.

We are reasonably satisfied with the VTI parameters for roughness and ground resistance. A simple and reliable technique for measuring wheel slip would also be of interest. However, this can be complicated by non-steady state events, such as accelerating and braking

The system as a whole shows promise and we intend to explore further elements in order to provide more accurate predictions. The visualization tools that have been developed for this project have been very valuable in determining where the system performs correctly and where it does not and will greatly aid with upcoming enhancements.

References

[1] L. Ojeda, J. Borenstein, G. Witus and R. Karlsen, "Terrain characterization and classification with a mobile robot," J. Field Robotics 23, 103-22 (2006).

[2] A. Howard and H. Seraji, "Vision-based terrain characterization and traversability assessment," *J. Robotic Systems* **18**(10), 577-587 (2001).

[3] R. Manduchi, A. Castano, A. Talukder and L. Matthies, "Obstacle detection and terrain classification for autonomous off-road navigation," *Autonomous Robots* **18**, 81-102 (2005).

[4] A. Sarwal, D. Simon and V. Rajagopalan, "Terrain classification," *Unmanned Ground Vehicle Technology V*, SPIE Proc. **5083**, 156-163 (2003).

[5] D. Kim, J. Sun, S. Oh, J. Rehg and A. Bobick, "Traversability classification using unsupervised on-line visual learning for outdoor robot navigation," Proc. IEEE Int. Conf. Robotics and Automation, Orlando, FL (2006).

[6] C. Brooks and K. Iagnemma, "Self-supervised classification for planetary rover terrain sensing," Proc. IEEE Aerospace Conf., Big Sky, MT (2007).

[7] R.E. Karlsen, J.L. Overholt and G. Witus, "Run-time assessment of vehicle-terrain interactions," Proc. 24th Army Science Conference, Orlando, FL (2004).

[8] R.E. Karlsen and G. Witus, "Vision-based terrain learning," *Unmanned Ground Vehicle Technology VIII*, SPIE Proc. **6230**, 33-42 (2006).

[9] S. Chase and E.G. Heinemann, "Exemplar memory and discrimination," *Avian Visual Cognition*, R.G. Cook, Ed., (2001).

[10] D. Medin and M. Schaffer, "Context theory of classification learning", *Psychological Review* **85**(3), 207-238 (1978).

[11] R.M. Nosofsky, "Tests of an exemplar model for relating perceptual classification and recognition memory," *J. Experimental Psychology: Human Perceptual Performance* **17**(1), 3-27 (1991).

[12] C.W. Werner and G. Rehkamper, "Categorization of multidimensional geometrical figures by chickens (Gallus gaullus f. domestica): fit of basic assumptions from exemplar, feature and prototype theory," *Animal Cognition* **4**, 37-48 (2001).

[13] R.E. Karlsen and G. Witus, "Image understanding for robot navigation," Proc. 25th Army Science Conference, Orlando, FL (2006).

[14] P. Howarth and S. Rüger, "Evaluation of texture features for content-based image retrieval," *Proc. Int. Conf. Image and Video Retrieval*, 326-34 (2004).

[15] F. Hoppner, F. Klawonn, R. Kruse and T. Runkler, *Fuzzy Cluster Analysis: Methods for Classification, Data Analysis and Image Recognition*, John Wiley & Sons, (1999).

[16] B. Balasko, J. Abonyi and B. Feil, "Fuzzy clustering and data analysis toolbox" (www.fmt.vein.hu/softcomp/fclusttoolbox).

[17] S.R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," IEEE Trans. on Systems, Man and Cybernetics **21**(3), 660-74 (1991).

[18] G. Witus, "System to build fuzzy logic models from databases and application to multi-sensor data," *Data Mining and Knowledge Discovery: Theory, Tools, and Technology III*, SPIE Proc. **4384**, 171-79 (2001).