Robotica (2008) volume 26, pp. 767–779. © 2008 Cambridge University Press
doi:10.1017/S0263574708004360 Printed in the United Kingdom

A study of visual and tactile terrain classification and classifier fusion for planetary exploration rovers

Ibrahim Halatci*, Christopher A. Brooks and Karl Iagnemma

Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139.

(Received in Final Form: February 11, 2007. First published online: April 24, 2008)

SUMMARY

Knowledge of the physical properties of terrain surrounding a planetary exploration rover can be used to allow a rover system to fully exploit its mobility capabilities. Terrain classification methods provide semantic descriptions of the physical nature of a given terrain region. These descriptions can be associated with nominal numerical physical parameters, and/or nominal traversability estimates, to improve mobility prediction accuracy. Here we study the performance of multisensor classification methods in the context of Mars surface exploration. The performance of two classification algorithms for color, texture, and range features are presented based on maximum likelihood estimation and support vector machines. In addition, a classification method based on vibration features derived from rover wheel–terrain interaction is briefly described. Two techniques for merging the results of these "low-level" classifiers are presented that rely on Bayesian fusion and meta-classifier fusion. The performance of these algorithms is studied using images from NASA's Mars Exploration Rover mission and through experiments on a four-wheeled test-bed rover operating in Mars-analog terrain. Also a novel approach to terrain sensing based on fused tactile and visual features is presented. It is shown that accurate terrain classification can be achieved via classifier fusion from visual and tactile features.

KEYWORDS: Mobile robots; robot vision systems; robot sensing systems; extraterrestrial exploration.

1. Introduction

Near-term scientific goals for Mars' surface exploration are expected to focus on understanding the planet's climate history, surface geology, and potential for past or present life. To accomplish these goals, rovers will be required to safely access very rough terrain with a significant degree of autonomy. Terrain areas of interest might include impact craters, rifted basins, and water-carved features such as gullies and outflow channels.[34] Such regions are in general highly uneven and sloped, and may be covered with loose drift material that causes rover wheel slippage and sinkage.

Terrain physical properties can strongly influence rover mobility, particularly on sloped, natural terrain.[18] For example, a rover might navigate up a rocky slope with ease, but slide down a sandy slope of the same grade. Similarly, a rover might easily traverse a region of packed soil but become entrenched in loose drift material. The effect of terrain properties on rover mobility was exemplified in April–June 2005 and again in May–June 2006 when NASA's Mars Exploration Rover (MER) Opportunity became entrenched in loose drift material and was immobilized for several weeks. The ability to detect or estimate terrain physical properties would allow a rover to predict its mobility performance and thereby autonomously avoid terrain regions that are potentially untraversable. Knowledge of terrain properties could also allow a system to adapt its control and planning strategies to enhance performance, by maximizing wheel traction or minimizing power consumption.

Terrain classification methods provide semantic descriptions of the physical nature of a given terrain region (e.g., "sandy terrain," "rocky terrain," "drift material"). Class labels can be associated with nominal numerical physical parameters, and/or nominal traversability estimates, to improve mobility prediction accuracy.

This paper presents a study of multisensor terrain classification for planetary rovers in Mars and Mars-like environments. Performances of two existing "low-level" classification algorithms (based on maximum likelihood estimation and support vector machines) for color, texture, and range features are evaluated in the context of Mars exploration. In addition, classification of terrain based on features derived from rover wheel–terrain interaction is briefly described. Two techniques for merging the results of these low-level classifiers are presented that rely on Bayesian fusion and meta-classifier fusion. The performance of these algorithms is studied using images from NASA's MER mission and through experiments on a four-wheeled test-bed rover operating in Mars-analog terrain. It is shown that accurate terrain classification can be achieved via classifier fusion from visual and tactile features.

1.1. Related work

Numerous researchers have proposed terrain classification methods based on features derived from remote sensor data such as color, image texture, and range (i.e., surface geometry). Most of these algorithms have been developed in the context of terrestrial unmanned ground vehicles where visual features exhibit wide diversity. Moreover, these algorithms typically address the problem of detecting road surfaces or handling tall vegetation. We would note

* Corresponding author: E-mail: ihalatci@alum.mit.edu

that a planetary surface presents a difficult challenge for classification since scenes are often near-monochromatic, terrain surface cover consists mainly of sands of varying composition and rocks of diverse shapes, and sandy "crusts" can form on (and therefore obscure) rocks.

Color-based methods for classification and segmentation of natural terrain have been developed that are accurate and computationally inexpensive. Manduchi has presented a method based on mixture of Gaussians modeling for classifying outdoor scenes using color.[22,23] Kelly *et al*. has demonstrated the effectiveness of multispectral imaging, specifically at the near-infrared range, for terrain classification.[19] Dima *et al*. have employed the LUV color space and computed distribution statistics of each channel over ground patches as color features.[14] Color-based classification is attractive for terrestrial applications because many major terrain types such as soil, vegetation, and rock possess distinct color signatures. Color-based classification is also attractive for planetary exploration rover applications since most past, current, and planned rovers have included multispectral imagers as part of their sensor suites.[32]

Texture-based classifiers have also been studied extensively. Rasmussen used Gabor filters to detect "denseness" of textured surfaces to distinguish roads from surrounding vegetation.[28] Dima employed a Fast Fourier Transform representation of terrain surfaces for texture feature extraction.[14] Angelova utilized a histogram-based method where terrain classes are represented by textons and terrain patches are identified through occurrence statistics of these textons.[1,2] Castano also used texture features for autonomous detection of science targets.[10] Texture-based techniques are generally computationally expensive methods; however, they have been shown to be effective at segmenting natural scenes (Reed & Hans du Buf 1993).[29]

Geometric features acquired through stereo cameras or range finders have also been used extensively for terrain classification and/or obstacle detection. A standard approach for detecting obstacles relies on detecting rapid changes in elevation from the ground plane. Manduchi[23] and Bellutta[5] have developed algorithms for exploiting elevation points while Vandapel[35] preferred to represent 3D point clouds using their statistical distributions in space. Avedisyan[4] was interested in far-field navigation and employed toposemantic techniques and Mandelbaum[20] detected obstacles directly from disparity maps. The result of such obstacle detection is often a traversability map that defines regions of the terrain as traversable or untraversable.[16] Note that such methods allow for detection of "geometric" hazards or terrain features such as rocks; however, they cannot easily detect "non-geometric" hazards or terrain classes that are not characterized by geometric variation such as sand dunes on Mars.

Although nearly all terrain classification methods rely on features derived from remote sensor data, recently methods have been proposed to classify terrain based on "tactile" features, i.e., features derived from sensor data measuring some aspect of physical robot wheel–terrain interaction. A method for terrain classification based on analysis of vibrations arising from robot wheel–terrain interaction was first proposed in ref. [18] by Iagnemma and Dubowsky and then later developed by Brooks and Iagnemma.[9] Similar work was presented in ref. [30]. Another recent work has used a neural network-based classifier to analyze a variety of tactile sensor signals to classify terrain.[26] It was shown that data from various sensor modalities can be fused to produce reliable class estimates. Of course, tactile data can only be used to classify terrain that a robot is currently traversing and cannot predict the class of distant terrain without also employing visual features.

Classifier fusion methods attempt to combine the results from low-level classifiers into class assignments that are (ideally) of higher accuracy than those attainable from any individual classifier. Recent work in classifier fusion has been applied to Mars terrain. Specifically, researchers have developed algorithms that fuse intensity and elevation data to identify scientifically interesting targets.[10,17] Related work applied a vision-based classification approach using features such as color, texture, spatial dependence, and elevation for detection of rocks.[33] McGuire developed a method for real-time detection of geologically interesting points based on color and texture histograms of terrain images.[25] Note that several methods exist that employ a larger set of visual features such as texture and infrared imaging in addition to range data; however, their focus is detecting relatively structured roads and obstacle detection rather than terrain classification.[13,28]

This paper is organized as follows. Section 2 describes the low-level classifiers and the choice of features for color, texture, range, and vibration-based classification. Section 3 describes the proposed classifier fusion methods. Section 4 presents experimental results. MER imagery is used to assess the algorithm's performance, along with experiments conducted with TORTOISE (all-Terrain Outdoor Rover Test-bed for Integrated Sensing Experiments), the Field and Space Robotics Laboratory's four-wheeled test-bed rover. Section 5 concludes the paper and describes directions of current work.

2. Description of Low-Level Classifiers and Features

2.1. Classifier architectures

Low-level classifiers are defined as classification methods that operate solely on a single feature type. As noted in Section 1, such classifiers have been studied extensively for terrain classification based on color, texture, range features, and vibration. Here we study the performance of two distinct classification methods: a maximum likelihood classifier with mixture of Gaussians modeling (MoG) and a support vector machine (SVM).

2.1.1. MoG method. The MoG method models the distribution of data points in the feature space as a mixture of Gaussians.[7] The likelihood of observed features y given the terrain class x is computed as a weighted sum of k Gaussian distributions:

$$f(y \mid x_i) = \sum_{j=1}^{k} \alpha_j G(y, \mu_j, \Sigma_j) \tag{1}$$

Parameters of the model are α_j, the weighting coefficient, μ_j, the mean, and Σ_j, the covariance. These parameters

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

are learned through off-line training using the Expectation Maximization algorithm.[6,7] Here, the selection of k was tuned based on empirical analysis. Similar to ref. [23], good results were obtained using three to five Gaussian modes, with a greater number of modes often leading to over-fitting. The classifier based on this approach will be termed the MoG classifier. This classifier was applied to the remote sensing modes described in Sections 2.2, 2.3, and 2.4. The classifier output is a semantic class label assigned according to maximum likelihood, along with the conditional likelihood of the assigned class.

2.1.2. SVM method. A second classification method studied here was based on an SVM framework.[36] This approach constructs a binary classifier for each pair of classes by building a function that will be positive for one class and negative for the other. This function is constructed as a linear combination of similarity measures between the point to be classified y and the training points x_j:

$$f(y) = \sum_{j=1}^{n} \alpha_j K(y, x_j). \tag{2}$$

The similarity measure K is the kernel function. For this work linear, polynomial, and Gaussian kernels were evaluated. Values for the α_j are calculated during training by minimizing a loss function over the training data set. Complexity of the function $f(y)$ is limited by restricting the values of α_j to lie in the range [0,C], and for the Gaussian kernel by controlling the width of the Gaussian using a parameter γ. Cross-validation over a training data set was used to determine an appropriate choice of kernel and reasonable values for the regularization parameters $C$and γ. Binary classifiers are combined into multiclass classifiers using a voting scheme.

The SVM algorithms used in this work were implemented with the LIBSVM library with additional optimizations for linear classification.[11] The classifier based on this approach will be termed the SVM classifier. This classifier was applied to all remote and tactile sensing modes described later. The classifier output is a discrete class label for each input feature vector.

2.2. Feature selection

In this work features based on color, texture, range (i.e., surface geometry), and vibration arising from robot wheel–terrain interaction are utilized for classification. These features are described here.

2.2.1. Color feature selection. Color is an obvious distinguishing characteristic of many terrain types. Although perceived color is affected by various physical factors such as illumination, color-based classification has yielded accurate results in natural terrain.[5,23] It should be noted, however, that color variation is somewhat limited for the surface of Mars. Mars' lack of moisture (and, therefore, vegetation) leads to a narrow distribution of colors for the distinct terrain types.

In this work red, green, and blue channel intensity values were selected as the 3D color feature vector for every image pixel. Construction of this feature vector for MER imagery was slightly different due to the nature of the rover imaging system, and is detailed in Section 4.1.

2.2.2. Texture feature selection. Texture is a measure of the local spatial variation in intensity in an image. For our present work, the texture length scale of interest is on the order of tens of centimeters. This scale allows us to observe textural appearances of surfaces in the range of 4 to 30 meters, which corresponds to the range of interest for local planetary rover navigation.[16] In this work we employ a wavelet-based fractal dimension signature method, which yields robust results in natural texture segmentation, as demonstrated by Espinal *et al.*[15]

Texture feature extraction consists of two parts: computing the wavelet transform of the image at various resolutions and calculating the fractal dimension signature of every pixel in a given neighborhood. The discrete wavelet transform of an image is the decomposition of the 2D signal at multiple resolutions. Every sublevel of transformation yields fractal information for the horizontal, diagonal, and vertical directions. After the transformation, the texture signature of every pixel is extracted by averaging the amplitudes of texture components in a predefined neighborhood window. Every transformation level utilizes a different neighborhood window that scales with the level of transformation. Here, three levels of transformation were applied using the Haar wavelet kernel and neighborhood windows of 7, 9, and 11 pixels. This feature extraction method yields an HDV (horizontal, diagonal, and vertical) feature space, and a 3D feature vector is generated for every pixel.

2.2.3. Range feature selection. Surface geometry information can be used to distinguish between terrain classes that possess inherent geometric dissimilarity. An example of two such classes is rock and cohesionless sand. Since cohesionless sand can never attain a slope greater than its angle of repose (whereas rock, of course, can), features related to terrain slope could potentially be used to distinguish "rock" and "sand" classes. Similar logic was applied for range feature selection in this work. This is based on the observation that rocky terrain on Mars-like surfaces project from the ground plane while sandy terrain predominantly lies parallel to the ground plane. Figure 1 shows a typical Mars scene in the field of view of an MER rover stereo pair.

In this work, range data was acquired from stereo imaging techniques. To compute range features in a scene, a 20 cm × 20 cm grid-based patch representation of the terrain surface was constructed in the field of view of the stereo pair. This grid size was selected to be similar to one rover wheel diameter, as an ad hoc attempt to capture the minimum-sized terrain patch that could significantly influence a rover's mobility. Best-fit planes were found within every patch using least-squares estimation, and a 2D surface normal vector was extracted. The 3D range feature vector for a given patch was then composed of the coefficients of the surface normal vector, along with the height difference between the maximum and minimum elevation points within the patch.

The primary assumption of this feature selection method is that the terrain is dominated by planar features. Another method for analyzing range data has been described in

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

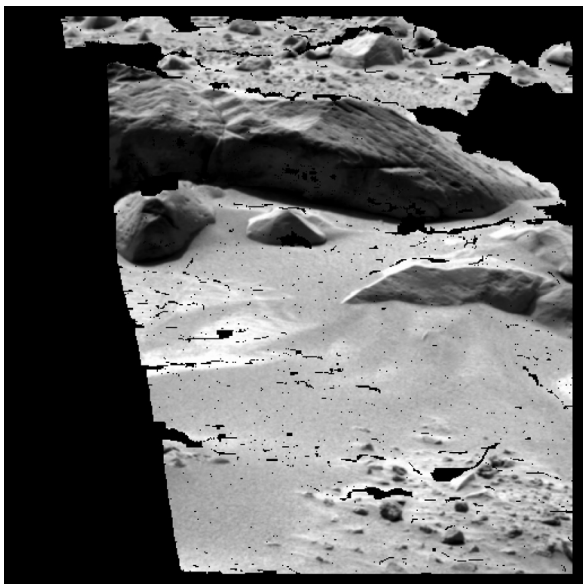

Fig. 1. Visible stereo range of a Martian scene.

ref. [38] where covariance of the data points in a point cloud is used to estimate 3D structure. In this representation, features represent the "point-ness," "curve-ness," and "surface-ness" of a 3D point cloud in a support region. When applied to Mars scenes with a fixed 20 cm × 20 cm support region, it was observed that the "surface-ness" feature is dominant, and hence the above planar features assumption appears justified.

2.2.4. Vibration feature selection. Analysis of vibrations propagating through a rover's wheel/suspension structure can be used to distinguish between various types of terrain the rover is traversing.[9] This classification mode is unique among the low-level classifiers described here in that it relies on a "tactile" sensor signal that is modulated by physical rover wheel–terrain interaction. Such a classifier is immune to illumination variation, making it a potentially attractive complement to vision-based classification techniques. Of course, tactile data can only be used to classify terrain that a robot is currently traversing, and cannot predict the class of distant terrain unless combined with remote sensing elements.

The general classification framework employed here is identical to that in ref. [9]. Vibration signals were processed as the log power spectral density for every one-second time step at 557 frequencies in the frequency range 20.5 Hz to 12 kHz. A linear classifier was trained in this 557-dimensional space using hand-labeled training data. For this work, an SVM with a linear kernel was used as the classifier instead of the Fisher linear discriminant employed in ref. [9].

3. Description of High-Level Classifier Methods

Low-level classifiers can yield poor results when applied individually in certain problem domains. For example, color is sensitive to illumination changes and shadowing, and thus poor color classification performance is possible in scenarios with wide expected lighting variation, although there are certain computer vision techniques designed to reduce this sensitivity.[22] Similarly, texture feature characteristics vary with distance, so poor texture classification performance is possible in scenarios with image geometry such that a single image spans a large distance. In this section, two classifier fusion methods are presented as means for merging results from multiple low-level classifiers, in an attempt to mitigate the weaknesses of each individual classifier.

It should be noted that the class spaces of low-level and high-level classifiers can be different. Since certain class distinctions may be unobservable by low-level classifiers, classifier fusion methods can improve classifier descriptiveness by combining results from multiple sensing modes. Although this difference makes it more difficult to directly compare classifier performance, it should be noted that increase in the number of detectable classes is a performance boost in itself.

3.1. Bayesian classifier fusion

Bayesian fusion has been presented for classification of natural scenes with promising results.[21,31] Here, Bayesian fusion was applied to merge the results of the low-level classifiers presented in Section 2.1. In the work described here, the low-level MoG classifiers' outputs yield conditional class likelihoods. Posterior distributions of conditional class assignments are computed by Bayes' rule, using the assumption that prior likelihoods are equal. Assuming that the visual features are conditionally independent, simple classifier fusion is applied as in Eq. (3). Here $P(x_i|y_j)$ is the posterior probability of terrain class (x_i) given the sensing mode (y_j).

$$P(x_i \mid y_1, \ldots, y_n) = \prod_{j=1}^{j=n} P(x_i \mid y_j) \tag{3}$$

However, this formulation implicitly requires that all classifier functions reside in the same class space (i.e., the set x_i is same for all sensing modes). In the absence of this assumption, the class space of the final fusion is formed as the Cartesian product of the low-level class spaces, which yields a high number of nonphysical terrain classes. Although previous researchers have addressed this problem with an unsupervised dimensionality reduction algorithm,[21] this method did not exploit physical class knowledge that could be inherited from supervised classifiers. In this work the fusion class space was manually grouped into a lower-dimensional space of physically meaningful terrain classes based on physical class knowledge as shown in Eq. (4). Here R, S, M represent the terrain classes "rock," "sand," and "mixed," respectively, and c, t, r represent the visual features "color," "texture," and "range," respectively. *Ro* and *Sm* refer to the texture-based classes of "rough" and "smooth."

$$\begin{aligned} P(R \mid c,t,r) &= P(R \mid c) * P(R \mid r) * P(Sm \mid t) \\ P(S \mid c,t,r) &= P(S \mid c) * P(S \mid r) * P(Sm \mid t) \\ P(M \mid c,t,r) &= [P(R \mid c) * P(R \mid r) + P(S \mid c) \\ &\quad * P(S \mid r)] * P(Ro \mid t) \end{aligned} \tag{4}$$

These relations explicitly encode physical knowledge in the final class decisions. Rocky terrain was generally observed to be darker than sandy or mixed terrain, with pronounced vertical features, and often had a smooth texture. Low-level

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

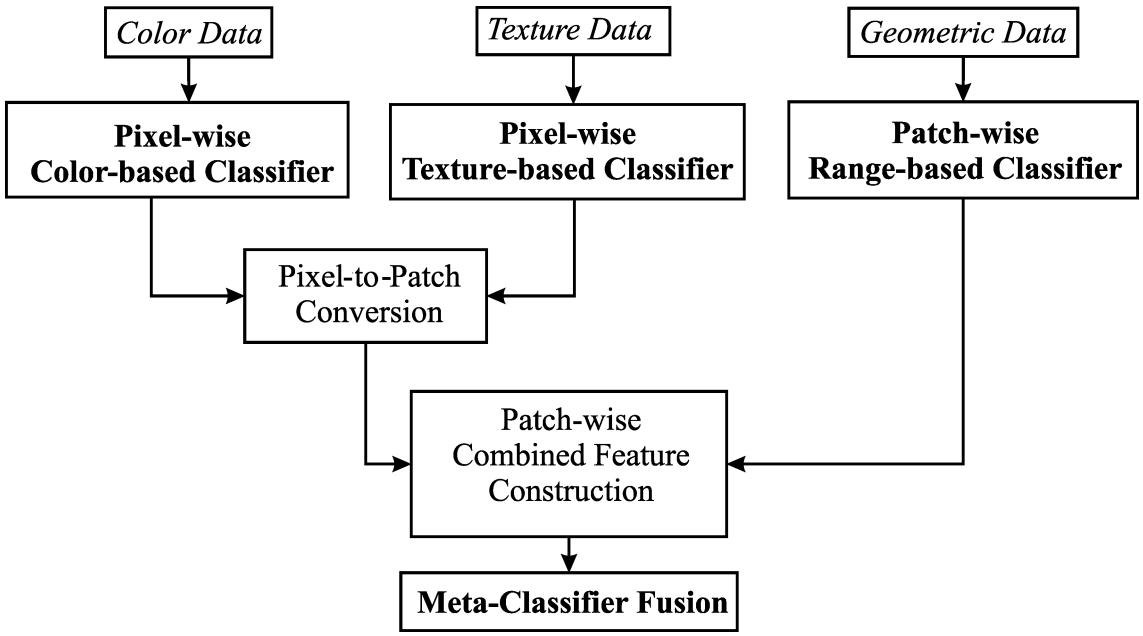

Fig. 2. Classifier flow for meta-classifier fusion.

classifier outputs related to these attributes were thus mapped to the classifier fusion "rock" class. Sandy terrain was generally observed to be smooth and low-lying relative to the ground plane. Mixed terrain exhibited uneven geometric properties and very rough textural properties at the scale of interest.

3.2. Meta-classifier fusion

A second approach to high-level classifier fusion is meta-classifier fusion. As illustrated in Fig. 2, meta-classifier fusion is a classifier with features extracted from the outputs of low-level classifiers. Specifically, it employs as features the continuous class likelihood outputs of the low-level classifiers as an n-dimensional feature vector for every terrain patch. Hand-labeled training data is used to fit a MoG classifier in this meta-feature space by the method described in Section 2.

Meta-classifier fusion is very similar to stacked generalization (SG) presented by Wolpert (1992) and applied for road detection by Dima *et al.*[14] In our method, low-level classifiers described in Section 2 correspond to the "level-0 generalizer" where meta-classifier corresponds to "level-1 generalizer" of SG architecture. Similar to SG, meta-classifier uses the outputs of low-level classifiers as inputs and computes the terrain class assignment for the data points. However, in our work, the data points may not have the same resolution for all low-level classifiers. As described in Section 2, color- and texture-based classifiers are pixel-wise while range-based classifier is a patch-wise classifier. A trivial solution to this data association problem is addressed by a pixel to patch conversion. This conversion computes the continuous class likelihood of a cell by averaging the class likelihood values of every pixel in that particular cell. In short, the "patch-wise feature extraction" step generates a feature vector from the converted patch-wise class likelihood outputs of low-level classifiers and feeds it to the high-level classifier. This high-level classifier is also a supervised classifier which needs to be trained with a different set of training data than the low-level classifiers.

3.3. Data fusion

A straightforward and well-known method to combine different sensing modes is simply concatenating all the sensor outputs into a single feature vector. This method was employed as a baseline to compare the performance of the Bayesian and meta-classifier fusion techniques and as a method for combining wheel vibration and vision data. In the context of color, texture, and geometric data, the feature vectors from the various sensing modes are combined to form a single 9D feature vector (composed of RGB, HDV, and 3D range features). Similar to the classifier flow shown in Fig. 2, data fusion scheme utilizes a pixel-to-patch conversion, but the low-level classifier layer is bypassed. Note that the mixed-resolution problem discussed in Section 3.2 is also relevant for data fusion and it is overcome by averaging the feature vectors over all pixels in a given cell. This 9D feature space is mapped to a probability distribution function using an MoG model. An SVM classifier was also applied to the data fusion framework. Note that the class space for data fusion includes three classes and SVM is implemented as a multiclass classifier using a voting scheme.

Data fusion was also applied as an approach to combine vibration and vision data for improved local terrain classification accuracy. Here, images captured using a camera pointed at the wheel provided visual data corresponding to the terrain being sensed by the vibration sensor, as seen in Fig. 3. The visual data was represented as the mean RGB value of the pixels in a small region below the wheel. This 3-element vector was appended to the 557-element vibration vector using the data fusion framework (see Fig. 4) producing a 560-element combined vision/vibration vector. An SVM classifier was used to identify the local terrain class.

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

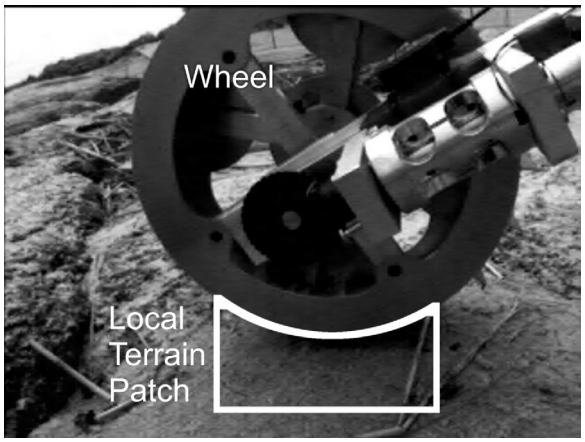

Fig. 3. Image of wheel and terrain from belly-mounted camera.

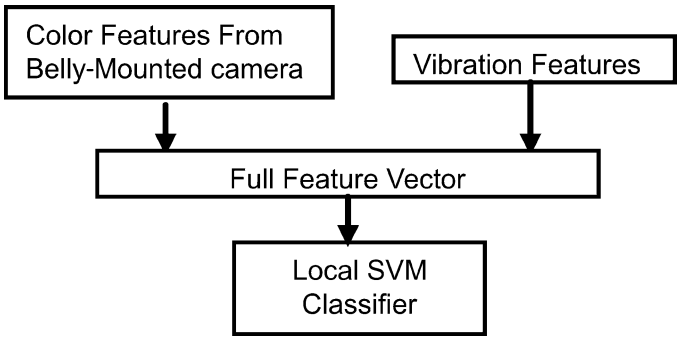

Fig. 4. Classifier flow of data fusion for local terrain classification.

4. Experimental Results

The performance of the low-level classifiers and the classifier fusion algorithms presented in Sections 2 and 3 was studied using images from NASA's MER mission and through experiments on a four-wheeled test-bed rover operating in Mars-analog terrain. These results are described below.

4.1. MER imagery

Publicly available images from the MER mission's Spirit and Opportunity rovers were used to assess the performance of the low-level and high-level classifiers. Fifty images from the rovers' panoramic camera stereo pairs were selected from the Mars Analyst's Notebook database.[24] These images were selected to be representative of the entire image set collected to date during the MER mission. Ten images were used for classifier training and identifying meta-parameters. The remaining 40 images were used to evaluate algorithm accuracy and computation time. Note that five more images were used for meta-classifier fusion and data fusion in addition to the training set to overcome data scaling problem. For MER imagery, the vibration-based classification approach was not employed since only image data was available.

The MER panoramic camera pair has eight filters per camera, with left camera filters distributed mostly in the visible spectrum and right camera filters located in the infrared region (with the exception of filter R1 at 430 nm). For color feature extraction, a combination of images from the left camera filters was employed. The 4th filter at 601 nm, 5th filter at 535 nm, and 6th filter at 482 nm intensities were chosen since they are near to the red, green, and blue wavelengths, respectively. Note that radiometric calibration would be necessary to acquire true color RGB images from the filter intensities (Crotty, 2006). Texture feature extraction was performed on the intensity image from the 2nd filter of the left camera at 753 nm. Range data was extracted by processing stereo pair images using stereo libraries developed at JPL.[3]

For Mars surface scenes, three primary terrain types that are believed to possess distinct traversability characteristics were defined: rocky terrain, composed of outcrop or large rocks; sandy terrain, composed of loose drift material and possibly crusty material; and mixed regions, composed of small loose rocks partially buried or lying atop a layer of sand. Examples of these terrains are shown in Fig. 5 (right). High-level classifiers such as data fusion, Bayesian fusion, and meta-classifier fusion are expected to distinguish these three terrain classes; however, low-level classifiers can distinguish only a subset of these terrain classes (Fig. 5, left, middle) because observable features of the terrain do not present unique signatures for each of the sensing modes. For instance, the color space of mixed terrain class, since it is composed of small rocks scattered on sand, overlaps with the color spaces of rock and sand terrain classes, so a color-based

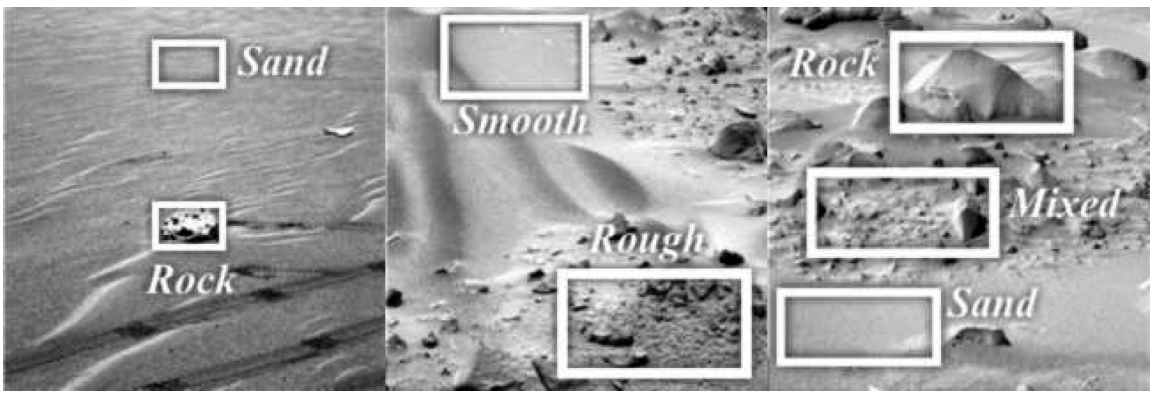

Fig. 5. Observable class distinctions for classifiers: color- and geometry based classes (left), texture-based classes (middle), fusion (final) classes (right).

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

Table I. Low-level classifier performance.

	Average accuracy (%)	95% Confidence interval for average	Standard deviation (%)
Color-based			
MoG	57.2	[52.4 62.1]	15.6
SVM	68.1	[63.4 72.7]	15.0
Texture-based			
MoG	60.9	[56.1 65.7]	15.6
SVM	66.7	[61.4 71.9]	16.8
Geometry-based			
MoG	75.5	[69.0 82.1]	21.2
SVM	70.2	[63.0 77.3]	23.0

classifier cannot identify it. Similarly, texture on the rock surfaces is not observable given the range of observation is 4 to 20 meters. In short, for color- and range-based classifiers, the classes of interest are "rock" and "sand" classes; for texture-based, the classes of interest are "sand" and "mixed" classes.

4.1.1. Low-level classifier results. Quantitative results of low-level classifier are presented in Table I as average performances over the test set. The color-based classifiers produced results close to expectation of random choice between two classes on average. This may be expected due to the monochromatic nature of Martian surface. Almost all images exhibit a uniform brownish-red tone, where rocks generally appear only slightly darker than sandy plains. Texture-based classifier performed better than color since the discrimination between mixed and sandy terrain is more apparent. However, the performance for texture-based classification is still not sufficiently robust since the texture classification accuracy is sensitive to the scaling of the image. This may be expected because the texture features are a visual measure of roughness, tuned to sense the roughness of surfaces in the range of 4 to 20 meters. Poor performance was observed in classifying terrain outside that range. The range-based classifier demonstrated the best performance, with 75% average classification accuracy. Failures in range-based classification were observed when sand was steeply sloped to form ridges and dunes.

Figure 6 shows ROC curves for each low-level classifier, illustrating the accuracy of the MoG and SVM classifiers across a range of confidence thresholds. The threshold levels start at 0.6 and goes up to 0.99 for low-level classifiers while high-level classifiers utilize threshold levels of 0.5 up to 0.99. These results demonstrate the weaknesses of the low-level classifiers. Besides being unable to distinguish between the three terrain classes of interest, low classification accuracy is inherent due to the challenging nature of the classes. It should be observed that SVM and MoG classifiers demonstrated similar behavior for each of the low-level sensing modes as viewed in the ROC curves.

Sample images showing ground truth images and classification results are shown in Figs. 7 and 8. Note that although color- and texture-based classifiers are pixel-wise in nature, their results are also presented as patch-wise (i.e., each patch was assigned to a class, based on the total probability computed from the likelihoods of the pixels in that patch).

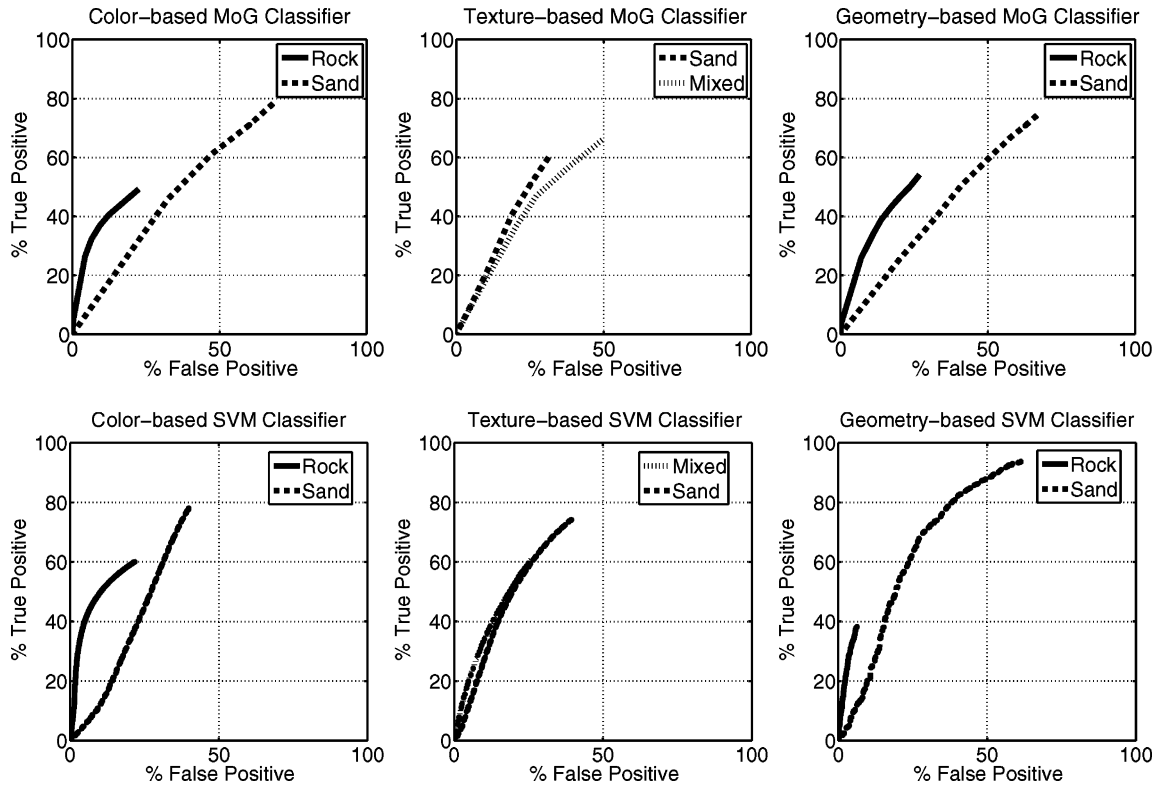

Fig. 6. ROC curves of the low level classifier: MoG (first row), SVM (second row).

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

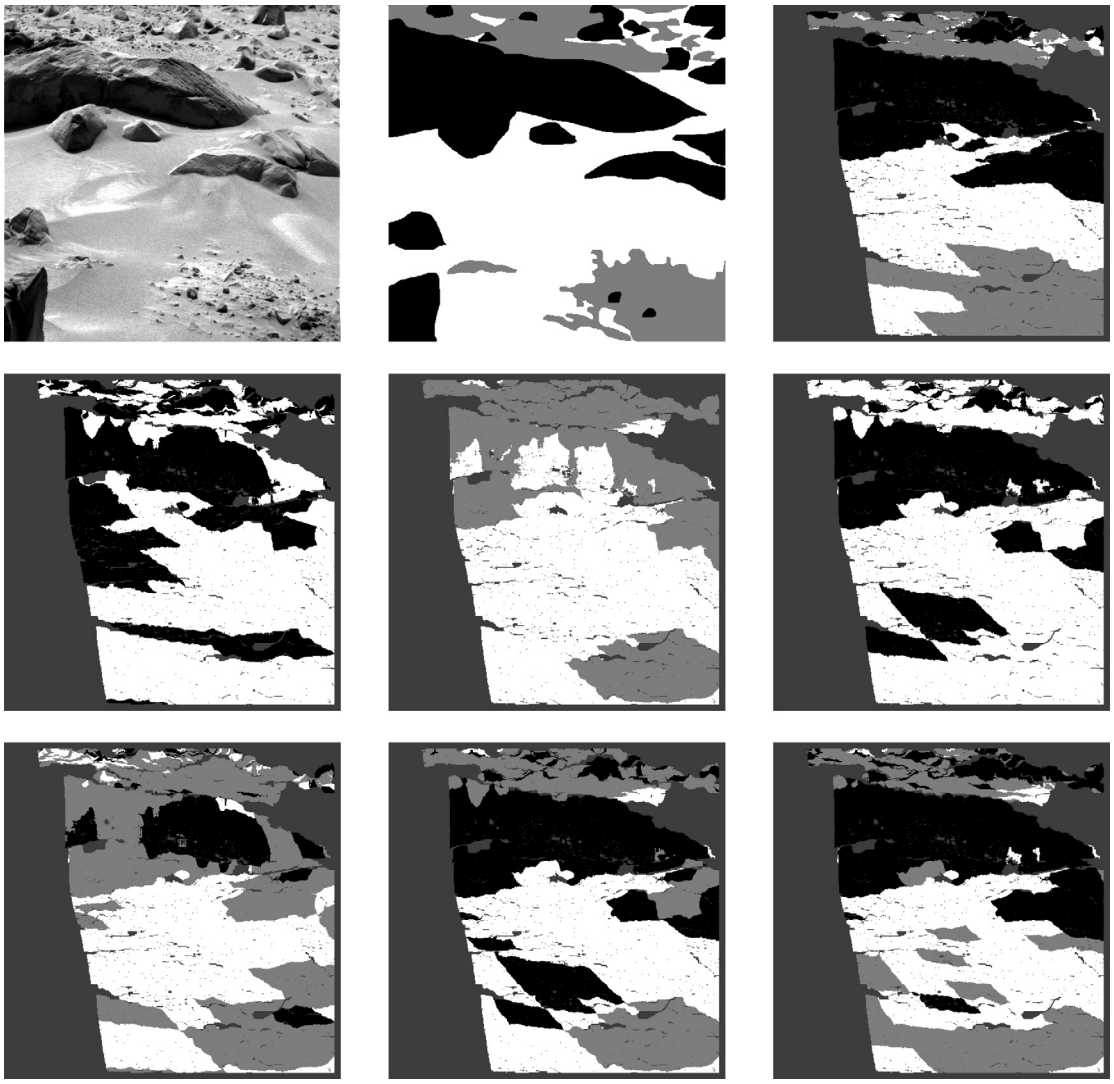

Fig. 7. Sample classification results; First row: Original scene (left), Pixel-wise hand labeled ground truth (middle), Patch-wise ground truth; Second Row: Color-based (left), texture-based (middle), range-based (right); Third row: Data fusion (left), Bayesian fusion (middle), Meta-classifier fusion (right). Black: rock, white: sand, light grey: mixed, dark grey: unknown.

4.1.2. High-level classifier results. As described in Section 3, classifier fusion methods combine the data from multiple sensing modes to compute a class label. By merging the results of color- and range-based classifiers, classifier fusion algorithms aim to compensate the weaknesses of the low-level classifiers (e.g., to decrease the false positives of rock vs. sand detection). Moreover, inclusion of the texture data enables the observation of roughness and allows the definition of a "mixed" class that represents regions of small stones distributed in sand.

Figure 9 shows ROC curves for the data fusion method applied with SVM and MoG as a multiclass classifier. As expected, data fusion performed poorly. This may be due to the difficulty of modeling in the high-dimensional feature space. In each case, it was observed that the classifier tend to have a bias toward a certain terrain class which yields poor average performance. These results also demonstrate the need for high-level classifier fusion for robust classification performance. Table II shows the comparison between the data fusion and classifier fusion methods in terms of global performance results.

Regarding the comparison between low- and high-level classifiers, note that high-level classifiers distinguish between three classes, whereas the low-level classifiers each distinguish between only two. Therefore the performance in terms of average accuracy is not directly comparable. However, the color- and texture-based classifiers perform close to the expectation of random choice, whereas the performances of the classifier fusion methods are much more robust.

Comparing high-level classifiers based on the ROC curves presented in Fig. 10, it was observed that Bayesian and meta-classifier fusion were much more accurate than data fusion. Although scaling of data (from pixel to patch) potentially affects both data fusion and meta-classifier fusion, classifier fusion demonstrates better results than data fusion given the same amount of training data. For this data set, Bayesian fusion demonstrated similar accuracy to meta-classifier fusion. However, meta-classifier fusion requires more training data for the second level of classifier, besides the training set of low-level classifiers. Bayesian fusion, on the contrary, does not require extra training

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

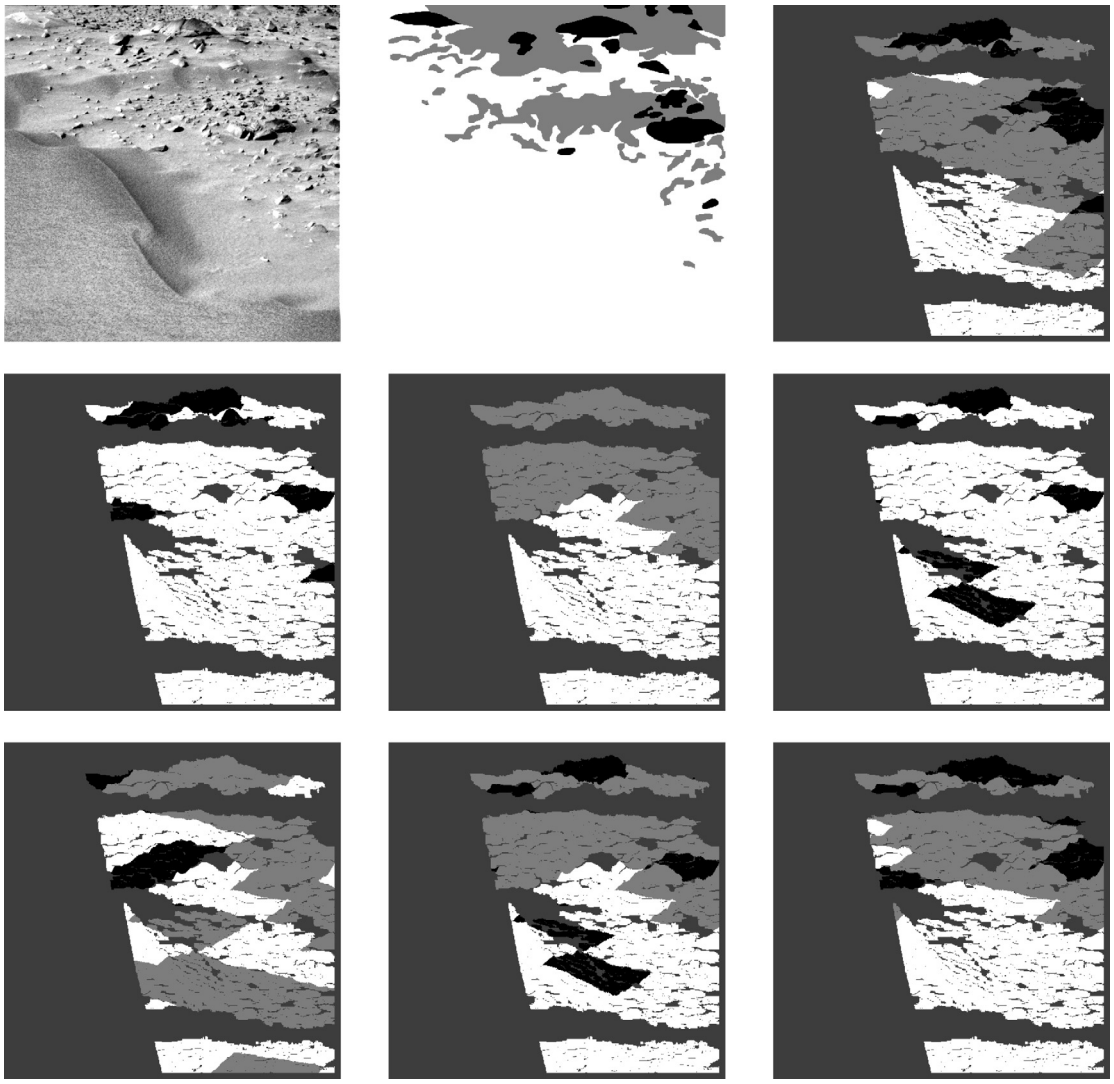

Fig. 8. Sample classification results; First row: Original scene (left), Pixel-wise hand labeled ground truth (middle), Patch-wise ground truth; Second Row: Color-based (left), texture-based (middle), range-based (right); Third row: Data fusion (left), Bayesian fusion (middle), Meta-classifier fusion (right). Black: rock, white: sand, light grey: mixed, dark grey: unknown.

for the second level, but the relationship between low-level classes and high-level classes has to be manually defined based on the environment setting. In short, there is a trade-off between predefined class space and amount of training data to identify the selection of either fusion method.

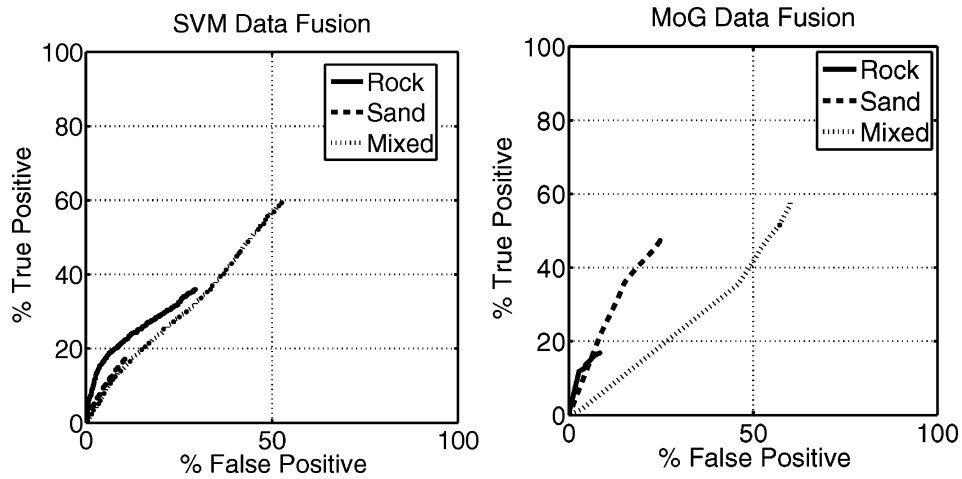

Fig. 9. Data fusion ROC curves using SVM classifier (left) and MoG classifier (right).

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

Table II. High-level classifier performance.

	Average accuracy (%)	95% Confidence interval for average	Standard deviation (%)
Data Fusion			
MoG	38.0	[32.5 43.5]	17.8
SVM	47.0	[41.6 52.3]	17.3
Bayesian Fusion	64.7	[59.9 69.5]	15.5
Meta-classifier Fusion	59.6	[55.3 63.7]	13.6

4.2. Wingaersheek Beach rover experiments

4.2.1. Experimental setup. Additional experiments were performed using a four-wheeled mobile robot developed at MIT, named TORTOISE, shown in Fig. 11. TORTOISE is an 80-cm-long × 50-cm-wide × 90-cm-tall robot with 20-cm diameter wheels. The TORTOISE sensor suite includes the following: a forward-looking mast-mounted Videre Design "dual DCAM" stereo pair with 640 × 480 resolution; a belly-mounted color monocular camera with 320 × 240 resolution to observe local terrain; and a Signal Flex SF-20 contact microphone mounted on the rover suspension near the front right wheel assembly to sense vibrations. The stereo pair is capable of capturing color and grayscale images, which were used for color and texture feature extraction, respectively. Range data was extracted from stereo images using Videre Design's commercial stereo processing software.[37] During experiments, TORTOISE traveled at an average speed of 6 cm/s. It captured monocular images at 2 Hz and vibration data at 44.1 kHz. Stereo images were captured every 1.5 s.

Experiments were performed at Wingaersheek Beach in Gloucester, MA. This is an oceanfront environment dominated by large (i.e., meter-scale) rock outcrops and distributions of rover-sized and smaller rocks over sand. Neighboring areas exhibit sloped sand dunes and sandy flats mixed with beach grass. Figure 12 shows a typical scene

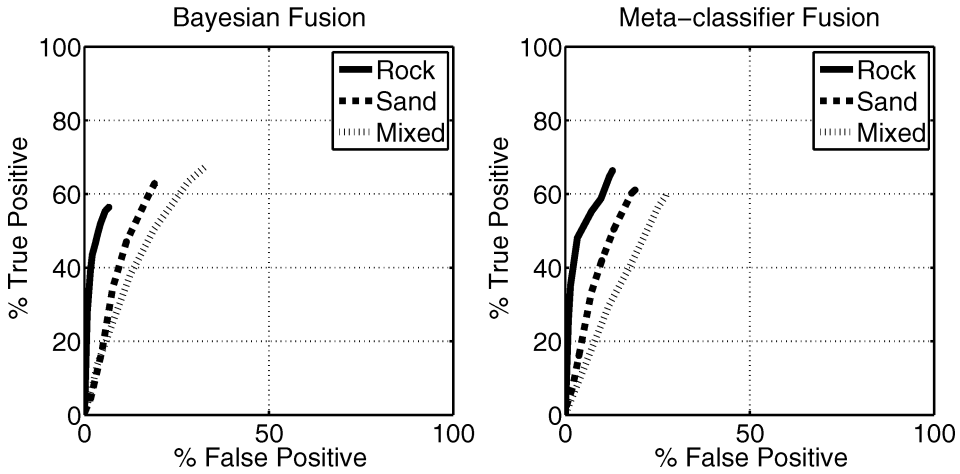

Fig. 10. ROC curves for Bayesian fusion (left) and meta-classifier fusion (right).

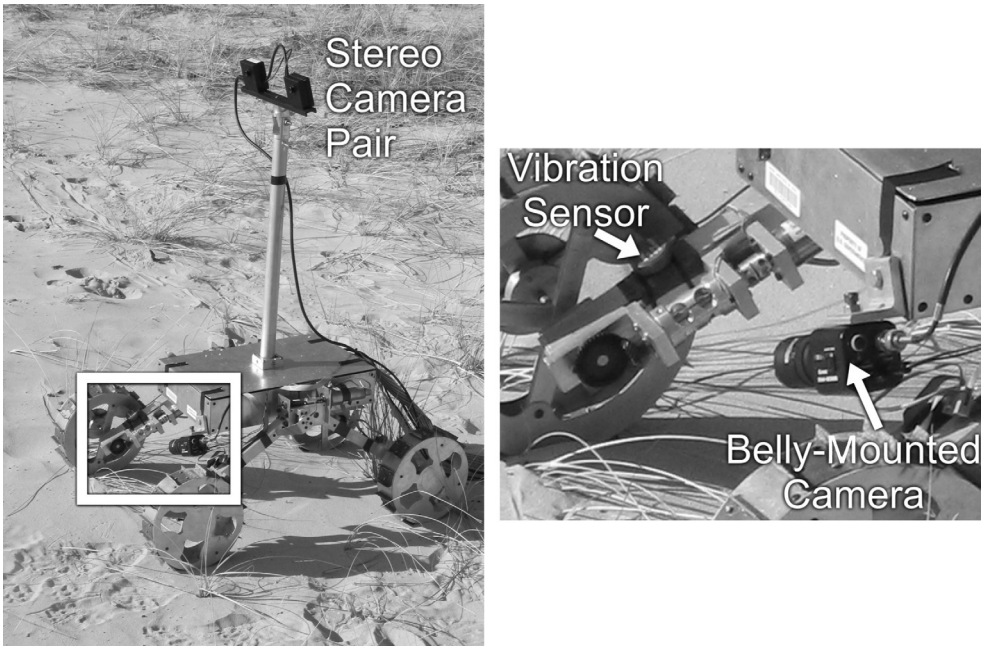

Fig. 11. TORTOISE experimental rover (left), local sensing suite (right).

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

Fig. 12. Sample scene from Wingaersheek Beach.

from the experiment site. This scene shows a large rock in the foreground and scattered, partially buried rocks in the middle range. Sand appears grayish in color while rock features vary from gray to light brown and dark brown. This test site was chosen because of its visual and topographical similarities to Mars surface scenes. For the following experiments, the terrain classes of interest were "rock," "sand," and "beach grass." The "mixed" class was not defined due to lack of scattered small-sized rocks; dry beach grass was used to reflect distinct texture signature in an effort to maintain a consistent number of classes with MER results.

4.2.2. Low-level classifier results. Six days of experiments were conducted, with a total of approximately 50 traverses and a total distance traveled of 500 meters. Every traverse includes around 250 images and every other tenth image is selected for test set to minimize overlap. Data of the first traverse of the day is used for training data. The performance of the low-level classifiers is shown in Fig. 13 as a series of ROC curves. It was observed that the performance of the color-based classifier was improved over that observed in experiments on MER imagery. This was likely due to the greater color variation present in an average beach scene. Relatively poor results were observed from the range-based classifier. The reason for this decrease in performance may be related to the poor accuracy and resolution of stereo-based range data for these experiments relative to MER imagery data, which used state-of-the-art JPL stereo processing software operating on high-quality images. This performance decline illustrates the sensitivity of range-based classification to data quality and strengthens the motivation for classifier fusion.

4.2.3. High-level classifier results. High-level classifier performance is shown in Fig. 14. In keeping with the MER results, the classifier fusion methods perform significantly better than the data fusion approach. Data fusion exhibits a bias toward the "rock" class yielding high false positives and degrading the detection rate for other classes. In this experiment setting, high-level classifiers do not increase the observable terrain classes since the color-based classifier is able to distinguish all terrain classes present in the setting. However, the ROC curves show a performance increase as a result of merging texture- and range-based classifiers

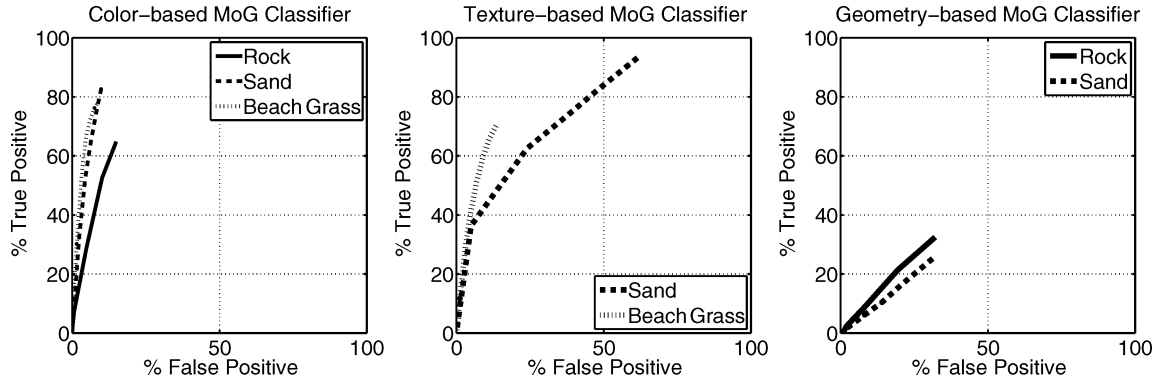

Fig. 13. Low level classifier results for Wingaersheek Beach experiments.

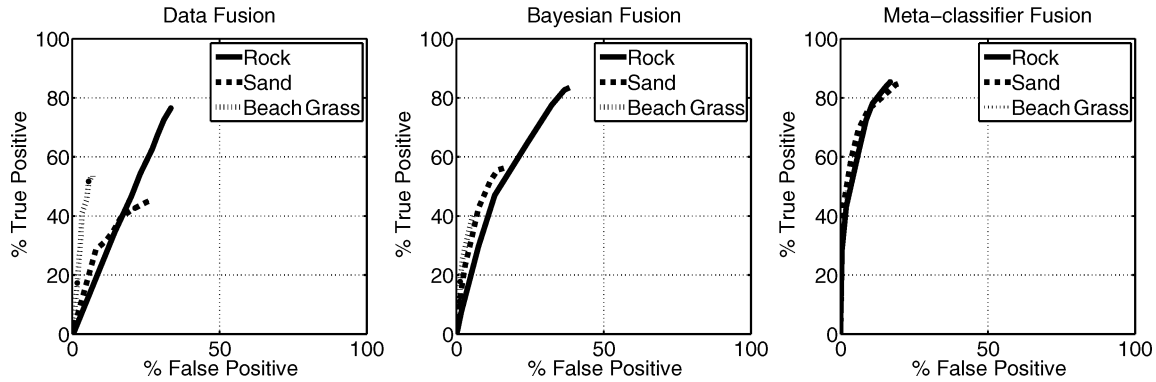

Fig. 14. ROC curves: Data fusion (left), Bayesian fusion (middle), meta-classifier fusion (right).

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

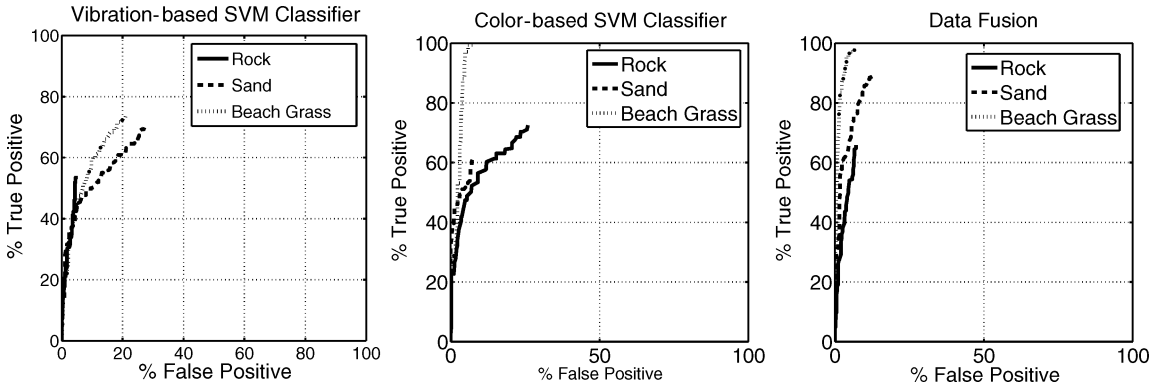

Fig. 15. Classifier results for local vibration-based classification (left), color-based classification (middle), and data fusion of color and vibration (right).

with color-based results. In the meta-classifier fusion results, it is clear that although individual performances of other low-level classifiers are below color-based results, they contribute to the training of meta-classifier yielding improved results.

4.2.4. Data fusion for tactile and visual sensing of local terrain. Local classification of terrain based on fusion of vibration and color features was tested using data captured by the vibration sensor and belly-mounted camera. These data were collected while the rover traversed sand, beach grass, and rock. A total of 21 min of vibration data were collected (1260 one-second segments), with over 2500 associated local images. Half of the data was used for establishing the meta-parameters and training each SVM classifier. The other half was used to test the classifiers.

The results for local terrain classification are shown in Fig. 15. The left plot shows results for pure vibration-based classification. It can be seen that all terrains are moderately well distinguished, with an average accuracy of 65% at full classification. The center plot shows results for pure color-based classification. Here "beach grass" is nearly always detected, with very few false positives. "Rock" and "sand" are also well distinguished. The average accuracy is 77% at full classification. Finally, the right plot shows the results for data fusion of color and vibration. An improvement over vibration-only and color-only classifiers was exhibited, with an average accuracy of 84%. This result suggests that improved classification performance can be derived from fusion of visual and tactile information. This is likely due to the insensitivity of tactile features to variations in illumination.

4.3. Computation times

All algorithms in this work except SVM classification were implemented in Matlab. On a Pentium 1.8 GHz desktop computer, pixel-wise MoG classification of a 512×512 image took an average of 5.2 s. Patch-wise MoG classification (for range-based, data fusion, and meta-classifier fusion) required an average of 2.4 s. Bayesian fusion took 1.2 s to form classifier decisions. The most computationally expensive element of the algorithms is texture feature extraction, which requires approximately 14.8 s of computation time for three levels of Haar wavelet transforms and for computing the pixel-wise texture signature of 512×512 grayscale image. In total, classifying a 512×512 frame takes approximately 29.0 s/frame. These times could be significantly reduced in a C-code implementation.

SVM classification was implemented with C++, using the LIBSVM library, with additional optimization for linear kernels.[11] Classification of a 512×512 color image took an average of 0.61 s using a linear kernel. Classification using a Gaussian kernel took an average of 77.5 s for a 512×512 color image. After feature extraction, texture classification times were identical to those for color classification. Patch-wise classification (for range and data fusion) averaged less than 0.01 s per patch for the linear SVM, and less than 0.04 s per patch for the Gaussian SVM. The number of patches in each image varied from 10 to 400.

5. Conclusion

This paper has compared the performance of various methods for terrain classification based on the fusion of visual and tactile features. Two classification algorithms for color, texture, and range features were presented based on maximum likelihood estimation and support vector machines. In addition, a classification method based on features derived from rover wheel–terrain interaction was briefly described. Two techniques for merging the results of these "low-level" classifiers were presented that rely on Bayesian fusion and meta-classifier fusion. It was shown that the classifier fusion methods improved overall classification performance in two ways. First, classifier fusion yielded a more descriptive class set than any of the low-level classifiers could attain individually. Second, the rate of false positives decreased significantly while the rate of true positives increased. This shows that in challenging planetary surfaces, stand-alone visual features are not sufficiently robust enough for mobile robot sensing; however, classifier fusion techniques manage to elevate the sensing performance significantly. Future research will focus on integrating additional tactile sensing modes such as wheel sinkage and torque with visual classifier fusion algorithms. This will enable improved prediction of the physical properties of distant terrain and lead to generation of safe, feasible traverse routes.

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press

References

1. A. Angelova, L. Matthies, D. Helmick, G. Sibley and P. Perona, "Learning to Predict Slip for Ground Robots," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Orlando, FL (May 2006a).
2. A. Angelova, L. Matthies, D. Helmick and P. Perona, "Slip Prediction Using Visual Information," **In**: *Proceedings of the Robotics: Science and Systems (RSS)*, Philadelphia, PA (Aug. 2006b).
3. A. Ansar, A. Castano and L. Matthies, "Enhanced Real-Time Stereo Using Bilateral Filtering," **In**: *2nd International Symposium on 3D Data Processing, Visualization, and Transmission*, Thesaloniki, Greece (Sep. 2004) pp. 455–462.
4. A. Avedisyan, D. Wettergreen, T. Fong and C. Baur, "Far-Field Terrain Evaluation Using Geometric and Toposemantic Vision," **In**: *8th ESA Workshop on Advanced Space Technologies for Robotics and Automation*, Noordwijk, Netherlands (2004).
5. P. Bellutta, R. Manduchi, L. Matthies, K. Owens and K. Rankin, "Terrain Perception for Demo III," **In**: *Proceedings of the Intelligent Vehicles Symposium*, Dearbon, Michigan (Oct. 2000) pp. 326–331.
6. J. Bilmes, "A Gentle Tutorial on the EM Algorithm and Its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models," Technical Report, University of Berkeley (1997).
7. C. M. Bishop, *Neural Networks for Pattern Recognition* (Oxford University Press, New York, 1995).
8. C. Bouman and B. Liu, "Multiple Resolution Segmentation of Textured Images." *IEEE Trans. Pattern Anal. Machine Intell.* **13**(2), pp. 99–113 (1991).
9. C. Brooks and K. Iagnemma, "Vibration-Based Terrain Classification for Planetary Rovers," *IEEE Trans Robotics* **21**(6), 1185–1191 (2005).
10. R. Castano, M. Judd, T. Estlin, R. C. Anderson, D. Gaines, A. Castano, B. Bornstein, T. Stough and K. Wagstaff, "Current Results From a Rover Science Data Analysis System," **In**: *Proceedings of 2005 IEEE Aerospace Conference*, Big Sky (2005) pp. 356–365.
11. C. Chih-Chung and L. Chih-Jen, "LIBSVM: A Library for Support Vector Machines" (2001). Software retrieved January 2006. Available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.
12. D. Coltry, "Mars Exploration Rover Multispectral Color Imagery" (2006). Retrieved May 21, 2006, from http://www.lyle.org/∼markoff/.
13. C. S. Dima, N. Vandapel and M. Hebert, "Sensor and Classifier Fusion for Outdoor Obstacle Detection: An Application of Data Fusion to Autonomous Road Detection," *Appl. Imagery Pattern Recognition Workshop*, Vol. 1, 255–262 (2003).
14. C. S. Dima, N. Vandapel and M. Hebert, "Classifier Fusion for Outdoor Obstacle Detection," **In**: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 1, (ICRA, 2004), New Orleans, Lousiana, pp. 665–671.
15. F. Espinal, T. L. Huntsberger, B. Jawerth and T. Kubota, "Wavelet-Based Fractal Signature Analysis for Automatic Target Recognition," *Opt. Eng. (Special Section on Advances in Pattern Recognition)* **37**(1), 166–174 (1998).
16. S. Goldberg, M. Maimone and L. Matthies, "Stereo Vision and Rover Navigation Software for Planetary Exploration," **In**: *IEEE Aerospace Conference*, Big Sky, 5 (2002) pp. 2025–2036.
17. V. Gor, R. Castaño, R. Manduchi, R. Anderson and E. Mjolsness, "Autonomous Rock Detection for Mars Terrain," *Space 2001* (*AIAA*, Albuquerque, New Mexico, 2001), pp. 1–14.
18. K. Iagnemma and S. Dubowsky, "Terrain Estimation for High Speed Rough Terrain Autonomous Vehicle Navigation," **In**: *Proceedings of the SPIE Conference on Unmanned Ground Vehicle Technology IV*, Orlando, Florida (Mar. 2002).
19. A. Kelly, A. Stentz, O. Amidi, M. Bode, D. Bradley, A. D. Calderon, M. Happold, H. Herman, R. Mandelbaum, T. Pilarski, P. Rander, S. Thayer, N. Vallidis and R. Warner "Toward Reliable Off Road Autonomous Vehicles Operating in Challenging Environments," *Int. J. Robotics Res.* **25**(5/6) (June 2006), pp. 449–483.
20. R. Mandelbaum, L. McDowell, L. Bogoni, B. Reich and M. Hansen, "Real-Time Stereo Processing, Obstacle Detection and Terrain Estimation Form Vehicle-Mounted Stereo Cameras," **In**: *Proceedings of the 4th IEEE Workshop on Applications of Computer Vision*, Princeton, NJ, 288 (1998).
21. R. Manduchi, "Bayesian Fusion of Color and Texture Segmentations," **In**: *Proceedings of International Conference on Computer Vision (ICCV)*, (ICCV, 1999) Kerkyra 2 (1999) pp. 956–962.
22. R. Manduchi, "Learning Outdoor Color Classification From Just One Training Image," **In**: *Proceedings of European Conference on Computer Vision (ECCV)*, (ECCV, 2004), Prague, Czech Republic, 28(11) (2004) pp. 1713–1723.
23. R. Manduchi, A. Castano, A. Thalukder and L. Matthies, "Obstacle Detection and Terrain Classification for Autonomous Off-Road Navigation," *Autonomous Robots* **18**, 81–102 (May 2005).
24. Mars Analyst's Notebook (2006). Retrieved May 24, 2006, from http://anserver1.eprsl.wustl.edu/.
25. P. C. McGuire, E. D. Martinez, J. O. Ormö, J. G. Elvira, J. A. Rodriguez Manfredi, E. Sebastian Martinez, H. Ritter, R. Haschke, M. Oesker, J. Ontrup, "The Cyborg Astrobiologist: Scouting Red Beds for Uncommon Features With Geological Significance," *Int. J. Astrobiol.* **4**, 101–113 (2005).
26. L. Ojeda, J. Borenstein, G. Witus and R. Karlsen, (2006). "Terrain Characterization and Classification With A Mobile Robot," Journal of Field Robotics **23**(2), pp. 103–122 (2006).
27. K. M. Rajpoot and N. M. Rajpoot, "Wavelets and Support Vector Machines for Texture Classification," **In**: *Proceedings of 8th IEEE International Multitopic Conference*, Lahore, Pakistan (2004) pp. 328–333.
28. C. Rasmussen, "Laser Range-, Color-, and Texture-based Classifiers for Segmenting Marginal Roads," **In**: *Proceedings of Conference on Computer Vision & Pattern Recognition Technical Sketches*, Kauai, HI (Dec. 2001).
29. T. Reed and J. Hans du Buf. "A review of recent texture segmentations nad feature extraction techniques," CVGIP: Image Understanding Vol. 57(3), May 1993, pp. 359–372.
30. D. Sadhukhan, C. Moore and E. Collins, "Terrain Estimation Using Internal Sensors," **In**: *Proceedings of International Conference on Robotics and Applications* (*IASTED)*, Honolulu, Hawaii 84(11) (2004) pp. 1684–1704.
31. X. Shi and R. Manduchi, "A Study on Bayes Feature Fusion for Image Classification," **In**: *Proceedings of the IEEE Workshop on Statistical Algorithms for Computer Vision*, Madison, Wisconsin (2003).
32. S. W. Squyres *et al.*, "Athena Mars Rover Science Investigation," *J. Geophys. Res.*, **108**(E12), 8062 (2003).
33. D. R. Thompson, S. Niekum, T. Smith and D. Wettergreen, "Automatic Detection and Classification of Features of Geologic Interest," **In**: *Proceedings of IEEE Aerospace Conference* , Big Sky, Montana (2005) pp. 366–377.
34. M. Urquhart and V. Gulick, "Lander Detection and Identification of Hydrothermal Deposits," abstract presented at *First Landing Site Workshop for MER*, Mountain View, California (2003).
35. N. Vandapel, D. F. Huber, A. Kapuria and M. Hebert, "Natural Terrain Classification Using 3-D Ladar Data," **In**: *Proceedings of the International Conference on Robotics and Automation (ICRA)*, New Orleans, Lousiana 5 (2004) pp. 5117–5122.
36. V. N. Vapnik, *The Nature of Statistical Learning Theory* (Springer, New York, 1995).
37. Videre Design (2006). Retrieved on May 25, 2006 at http://www.videredesign.com/index.htm.
38. D. H. Wolpert (1992), Stacked Generalization, Neural Networks, Vol. 5, pp. 241–259, Pergamon Press.

https://doi.org/10.1017/S0263574708004360 Published online by Cambridge University Press