Terrain Classification Through Weakly-Structured Vehicle/Terrain Interaction

Amy C. Larson, Richard M. Voyles, and Guleser K. Demir
Department of Computer Science, Univerisity of Minnesota
Minneapolis, MN 55455
Email: larson@,voyles@,demir@cs.umn.edu

***Abstract*— We present a novel terrain classification technique both for effective, autonomous locomotion over natural, unknown terrains and for the qualitative analysis of terrains for exploration and mapping. Our straight-forward approach requires a single camera with little processing of visual information. Specifically, we derived *gait bounce* and *gait roll* measures from visual servoing errors that result from vehicle-terrain interactions during normal locomotion. Characteristics of the terrain, such as roughness and compliance, manifest themselves in the spatial patterns of these signals and can be extracted using pattern classification techniques. For legged robots, different limb-terrain interactions generate gait bounce signals with different information content, thus deliberate limb motions can effect higher information content (i.e. the robot is an active sensor of terrain class). Segmentation of the gait cycle based on the limb-terrain interaction isolates portions of the gait bounce signal with high information content. The decoding of, then sequencing of, this content from each cycle segment yields a robust classification of terrain type from known benchmarks. This is analogous to word recognition in which the spatial pattern of speech encodes phonemes and the sequence of the phonemes encodes the word. To extract the spatiotemporal pattern of the gait bounce signal, we developed a meta-classifier using discriminant analysis and hidden Markov models. In this paper, we present the gait bounce and gait roll derivation, terrain classification using both spatial discriminants and our meta-classifier, and we describe how terrain classification can be used for gait adaptation, particularly in relation to an efficiency metric. We also demonstrate that our technique is generally applicable to other locomotion mechanisms such as wheels and treads.**

I. Introduction

Creating adept robots for the real world requires adaptation to the environment. Adaptability is especially key when operating in environments where *a priori* information is approximate or unavailable, for example rescue missions inside felled buildings or planetary exploration. When operating in such environments, adaptation efficiency can generally be improved by first characterizing the current operational environment, then using this information to optimize some performance metric. We use environmental assessment to select appropriate behahaviors and their parameters. Assessment of terrain conditions can be particularly useful, because they impact both path planning and motion control. Operating outdoors highlights this point, where encounters with sand, rocks, grass, or pavement impact efficiency and speed of forward progress. These surfaces do not necessarily obstruct the path, rather require adjustments to motion control for more effective progress.

There are a variety of approaches for terrain assessment, which we review in Section II, but prior efforts require special equipment or are computationally expensive. We have developed a terrain classification technique using visual servoing errors obtained from a single camera tracking features in the image plane. Intuitively, it is analogous to estimating the roughness of the road, for example, based on the severity of the rocking motion of a vehicle. Our straight-forward approach provides a rich source of terrain information, yet requires only a single camera, relatively little processing of visual data, and is applicable to legged and to nonlegged platforms.

In Section III, we present the derivation of *gait bounce* and *gait roll*, which are estimates of the pitch and roll (using visual information) of the vehicle as it interacts with the terrain. The spatial pattern of the gait bounce encodes terrain conditions, such as roughness and compliance, which we can identify with pattern classification techniques using spatial discriminants, as shown in Section IV-A.

For our legged robot, TerminatorBot, different limb-terrain interactions generate gait bounce signals with different information content, thus deliberate limb motions can effect higher information content (i.e. the robot becomes an active sensor of terrain class). Segmentation of the gait cycle based on the limb-terrain interaction isolates portions of the gait bounce signal with high information content. The decoding of, then sequencing of this content from each cycle segment produces yet another signal whose sequence encodes terrain class. This is analogous to word recognition in which the spatial pattern of the speech encodes phonemes and the sequencing of the phonemes encodes the word. To extract the spatiotemporal pattern of the gait bounce signal, we developed a meta-classifier using discriminant analysis and hidden Markov models. The development and use of this approach is detailed in Sections IV-B and V.

Our testing platform is TerminatorBot (Figure 1), which is a small-scale robot designed for applications of search-and-rescue, planetary exploration, and surveillance. Its versatility in arm motion allows for a variety of gait classes, as well as fine-manipulation. Subsequently, TerminatorBot is an excellent platform for our studies. As an active sensor, it can effect a variety of limb-terrain interactions to investigate and define different characteristics of the terrain. As a platform for use in search-and-rescue or planetary exploration, it is sufficiently mobile to traverse the natural, rough terrains of

Fig. 1. *Image of TerminatorBot emerging from a cinder block, which it can traverse using its narrow-passage gait.*

these applications. As a study in adaptation, TerminatorBot can provide terrain assessment as an active sensor, then apply that knowledge to adapt its gait for improved locomotion.

II. Related Work

Several techniques have been developed for classification, for traversability assessment, and for characterization of terrain. Motivation of these techniques in the area of mobile robotics splits into two basic areas – for navigational planning and for control. In other words, terrain classification is motivated by the two questions – 1) "Where is the robot going?" (navigation), and 2) How is it getting there?" (control). These questions are closely tied in outdoor environments, where terrain is varied and ease-of-control defines the path, as opposed to indoor environments, where the terrain is assumed constant and obstacles define the path.

Elevation maps and vision techniques can be used to assess terrain for navigation. In [1], [2], [3], the author's employed similar approaches using elevation maps, which are derived from stereo vision or laser rangefinders, to assign a traversability measure within a grid map cell based on height, slope, height variability, and/or terrain knowability. A search algorithm generated an optimal path from these measures. Similarly in [4], [5], [6], shape, texture, and color analysis aided in generating a path traversability measure. These works *predict* future vehicle-terrain interactions for the purposes of finding a global path and for determining an open-loop locomotion strategy. Our approach *measures* local vehicle-terrain interactions for the purpose of dynamically adapting locomotion, thus it provides a more accurate assessment of the terrain in real-time.

Other approaches for measuring vehicle-terrain interaction have been developed for wheeled vehicles for use in planetary exploration. Yoshida and Hamano [7] developed a wheel-terrain interaction simulation model and a control law based on estimated wheel-slip to prevent the robot from digging itself into a trench. Iagnemma et al. estimated terrain cohesion and internal friction to calculate drawbar pull (torque that pulls a wheel forward) [8], and in [9] maintained stability using ground contact angle estimates. Talukder et al. [6] used a spring-mass model to estimate terrain compliance both to maintain a safe velocity and to predict vehicle dynamics. In each case, the robot required several sensors to estimate terrain conditions, additionally these techniques are applicable to wheeled vehicles only. Our technique requires a single forward-looking camera, and it applies to legged vehicles with the potential for use on wheeled vehicles.

Fig. 2. *Lift phase of the swimming gait.*

In the realm of legged vehicles, there are a variety of sensors applied to measuring limb-terrain interaction, including tactile, force, inclinometer, and gyroscopes. In the simplest case, tactile sensors detect the presence or absence of terrain, for example in Hirose's seminal work on control of a quadruped walker [10], or in [11], in which a robot traversed a slatted surface. Other sensors can be used to estimate compliance or slope of the terrain for adaptive locomotion, as shown in [12], [13], [14], [**?**], to name a few. Our approach is similarly motivated. Its primary benefit over these other techniques is the sensor requirement. Cameras are typically in use for other purposes, thus no additional hardware is required. This is particularly beneficial for planetary exploration and applications for small-scale robotics, in which minimal weight is crucial. Additionally, our technique is complementary to these others. For example, contact sensors are precisely localized to the footfall, whereas gait bounce measures limb-terrain interaction over a path segment. Combining these techniques using sensor fusion provides a more complete model of the terrain.

III. Gait Bounce and Gait Roll

Our testplatform, TerminatorBot, can locomote with a variety of gait classes, including swimming, narrow-passage, and bumpy-wheel for forward motion, and differential and body-shift for turning, as outlined in [15], [16]. The swimming gait is its conventional gait, whose cyclic trajectory can be defined with continuous-valued parameters. To execute, the body is lifted to a desired pitch angle, as shown in Figure 2, then the shoulders are rotated back to propel the body forward. In a variation, the shoulders are rotated at different speeds to turn the body, analogous to turning a wheeled vehicle.

TerminatorBot navigates using visual servoing, which can be used to home in on an object by keeping it central in the image plane during forward locomotion. Any displacement of

the object (feature) off-center that is attributable to the robot's movement and not the feature's, is a visual servoing error. We employ sum of squared differences (SSD) template matching to track a feature (*see* [17] for more details). The horizontal displacement, Δx, of the tracked feature from the center of the image frame is used to provide a corrective turning angle, $\theta = \arcsin(\Delta x/f)$, where f is the focal length, to orient the body to the homed object. With ground-based robots, we are typically only interested in the horizontal displacement while the vertical displacement is thrown away, but in this work, we retain the latter information to derive gait bounce for terrain classification.

A. Gait Bounce Derivation

During locomotion, the vertical displacement of a feature across time provides a relative measure of the upward tilting motion of the camera, thus the *bounce* of the robot. In turn, gait bounce provides an estimate of terrain conditions, analogous to estimating the roughness of a road being traveled based on the intensity of your bounce as a passenger, or to estimating the stiffness of a terrain based on sinkage as you step. To use gait bounce as a terrain classifier, we derived a formal, quantitative measure of the vertical motion of the robot, which is essentially an estimate of the pitch angle. The gait bounce "sensor" is active during limb-terrain contact portions of the gait cycle, when the motion of the limb against the terrain effects variation in the vertical tilt of the body.

In the simplest case, the robot is moving across a smooth, hard surface and the vision system is tracking a single, fixed feature with a camera. Assuming orthographic projection, the visual servoing error in the vertical plane of the tracked feature (Δy) can be used to derive body angle relative to the ground plane by $\theta = -\arctan(\Delta y/f)$, where f is the focal length. The top, dark dashed line of Figure 3 shows the raw gait bounce derived from tracking a single feature as TerminatorBot moves across carpet. Before time 90 (time is measured in number of images processed at 30Hz), the arms are positioning themselves for ground contact. From time 100 to 350, TerminatorBot is in ground contact – lifting, dragging, then dropping the body to effect forward motion. After time 350, the arms are repositioning for the next cycle and are not in ground contact. The signal generated after time 350 is not due to body tilt, rather body roll, which is problematic.

Body roll is attributable to TerminatorBot's cylindrical shape, which can roll when neither arm is in ground contact, but more generally, it is also attributable to variations in elevation between limb-ground (or wheel-ground) contact points commonly seen in rough terrain. A feature in the image plane is displaced vertically when the body rolls, thus it generates an erroneous gait bounce signal. In other words, roll increases the signal strength with no corresponding increase of pitch angle. To compensate for this erroneous increase, we estimate body roll using two fixed features in the image plane.

To estimate body roll, we first calculate the angle of the line segment formed by two tracked features relative to the horizontal plane,

$$\phi_i = \arctan(\frac{y2_i - y1_i}{x2_i - x1_i}), \tag{1}$$

where $\{x1_i, y1_i\}$ is the $\{x, y\}$ position in the image frame of feature 1 at time i. This angle is attributable both to body roll and to misalignment of the two features in the horizontal plane, but it is only the former that affects gait bounce. An estimate of the latter (or nominal roll) is subtracted from ϕ_i, resulting in a zero-mean sequence of deviations from nominal:

$$\hat{\phi}_i = \phi_i - \frac{\sum_{j=1}^{N} \phi_j}{N}\,, \tag{2}$$

where N is the total number of image frames.

To derive roll-compensated gait bounce at each time step, we calculate the position of the center of the line $\{x_i, y_i\}$ relative to the center of the image plane $\{XC, YC\}$

$$\{x_i, y_i\} = \{\frac{x1_i + x2_i}{2} - XC, \frac{y1_i + y2_i}{2} - YC\}\,, \tag{3}$$

then we virtually roll the robot back to an upright position by rotating the point $\{x_i, y_i\}$ about the center of the image plane by $-\hat{\phi}_i$, which is roll at time i. The roll-compensated y-value, $\hat{y}$, is calculated as:

$$\hat{y}_i = x_i sin(-\hat{\phi}_i) + y_i cos(-\hat{\phi}_i)\,. \tag{4}$$

Then we calculate the roll-compensated bounce angle θ using $\hat{y}$ and the focal length f:

$$\theta_i = -\arctan(\frac{\hat{y}_i}{f}). \tag{5}$$

In compensating for roll, there is an underlying assumption that the centimeter-scale surface patch traversed during a gait cycle is locally planar. While this is clearly invalid for the rough terrains in which we are interested, it is a reasonable assumption for this calculation. The arm and body contact points form a triangle of support over the local terrain patch. Since the arm contact points typically stay fixed during forward motion (neglecting slip), they do not contribute to the roll during the forward stroke of the gait. The rear (body) contact point is dragged across the nonplanar surface patch, but the resulting erratic motion only manifests itself in gait bounce, not in body roll. Therefore, the planar assumption is reasonable for extracting body roll.

Roll-compensation results in the extraction of an additional characteristic function, similar to gait bounce, that contains terrain-specific information. In the current formulation we ignore "gait roll", but we may be discounting some potentially useful information. We surmise that roll may be a complementary discriminator of rocky and level surfaces, as well as of compliant and noncompliant surfaces. In future work, we will attempt to incorporate this additional distinctive signature into our terrain characterization, but in this paper, terrain classification is based only on gait bounce.

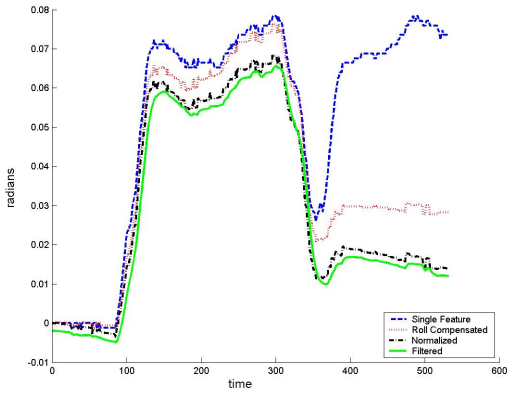

Fig. 3. *Gait bounce from tracking single object (dash), roll-compensated gait bounce (dot), normalized (dash-dot) for perspective distortion, and filtered (solid) (purposely offset for easier viewing).*

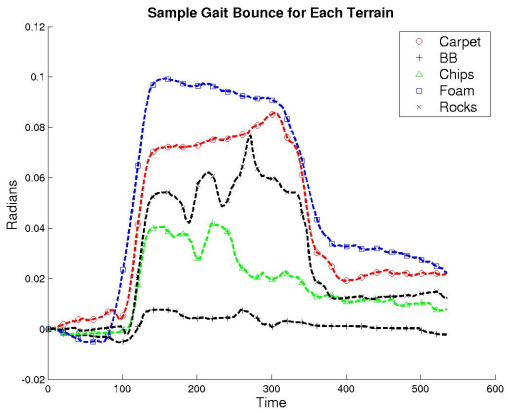

Fig. 4. *A gait bounce signal from a single gait cycle across each terrain type.*

The red dotted line of Figure 3 graphs the roll-compensated gait bounce. This new bounce measure reveals another concern, namely the upward data trend due to perspective distortion from what is more accurately modeled as a pinhole camera. As the robot moves toward tracked features, they move from the focus of expansion, which is at the center of the image. Since we do not know depth to the features, we can't explicitly compensate for perspective. Instead, we crudely approximate the perspective distortion as a linear function of time. We estimate the slope of this function by fitting a line to a complete cycle of gait bounce data in the least squares sense. We then normalize for this distortion by subtracting the approximation from the original gait bounce. This results in a new, normalized gait bounce $\hat{\theta}$:

$$\hat{\theta}_i = \theta_i - \theta_{\approx i} \,, \tag{6}$$

where $\theta_{\approx i}$ is the linear least squares approximation of θ_i. In the case that features get closer than 1 meter, the nonlinearity of the distortion becomes apparent, but the system continues to perform quite well.

The black dash-dot line of Figure 3 graphs the normalized gait bounce. The final issue that we address is local disturbances to the gait bounce, which is a high-frequency component relative to the gait cycle frequency that is filtered out. The filtered gait bounce measure is the green, solid line of Figure 3 (which is artificially offset from the normalized gait bounce for better viewing). This final gait bounce measure is the result of tracking two features to compensate for roll, of then normalizing the affects of perspective distortion, and finally of filtering out the high frequency component. Figure 4 shows a sample gait bounce signal generated from a single gait cycle while TerminatorBot traversed each of the five terrains.

IV. Terrain Classification

As can be seen in Figure 4, terrains have visually differentiable gait bounce, but we require quantifiable discrimination. Pattern classification techniques provide such a mechanism (*see* [18]), but successful application requires an appropriate match of feature space and classifier, as well as the optimization of user-defined parameters. We experimented with a variety of feature vectors and spatially discriminating classifiers, which are summarized in Section IV-A. We found appropriate combinations of feature space and classifier, but we were unsatisfied with the robustness and extensibility of these techniques. Gait bounce contains a temporal component, therefore we applied hidden Markov models (HMM), which classify based on the probability of a sequential event. For terrain classification, we generated an observation sequence based on spatial classifiers, as outlined in Section IV-B. This approach, which we call a meta-classifier, is analogous to speech recognition.

To test our terrain classification techniques, we walked TerminatorBot across a variety of terrains, including carpet, foam, BB's, woodchips, and rocks. The current prototype is insufficiently hardened for use in some natural terrains, therefore we selected artificial terrains to simulate terrain characteristics. BB's simulate highly compliant surfaces such as sand, foam simulates semi-compliant surfaces such as grass, and woodchips are both compliant and rough. TerminatorBot's vision system tracked and recorded the position of two stationary features while the robot moved across these terrains using the periodic swimming gait. We collected approximately 200 gait cycles over each terrain, for a total of 1000 cycles. Offline analysis produced a pattern of gait bounce for each cycle using the techniques described in the previous section. Data was randomly split into training and test sets for verification of the classifiers.

A. Classifiers Using Spatial Discriminants

To find an appropriate feature space and classifier combination, we first mapped gait bounce into three different feature spaces. In the first method, features were determined by applying the Fast Fourier Transform (FFT) over the entire cycle. The magnitude components of the first fifteen frequencies formed the feature vector and were used to train a variety of

	Total	Carpet	BB	Chip	Foam	Rock
Raw						
10 hu	**64**	88	88	29	82	30
15 hu	**62**	86	83	31	83	24
FFT all						
10 hu	**63**	95	76	27	85	33
15 hu	**62**	88	69	32	89	33
FFT seg						
10 hu	**60**	81	80	17	93	30
15 hu	**60**	78	80	17	92	29

TABLE I

Mean classification rates over 50 random test sets. RAW, FFTALL, *and* FFTSEG *indicate feature space*. HU *indicates number of hidden units.* TOTAL *is mean classification rate across all terrains.*

	Total	Carpet	BB	Chip	Foam	Rock
Raw						
LDA	**66**	91	89	57	85	31
QDA	**79**	90	72	56	89	87
LogDA	**67**	90	88	40	84	35
FFT all						
LDA	**73**	97	90	43	89	46
QDA	**83**	96	89	70	92	67
LogDA	**76**	95	86	49	88	64
FFT sec						
LDA	**69**	93	88	36	89	40
QDA	**81**	91	81	61	90	81
LogDA	**75**	93	86	50	90	55

TABLE II

Mean classification rates over 50 random test sets. RAW, FFTALL, *and* FFTSEC *indicate feature space.* LDA, QDA, *and* LOGDA *indicate classifier.*

classifiers. We also applied an FFT over segments of the cycle, concatenating the magnitude components of each to create the feature vector. The segments correspond to key moments of vehicle-terrain interaction, as discussed in the following section. Finally, we subsampled raw gait bounce data.

As a first attempt to classify terrains, we used the ubiquitous back-propagation artificial neural network (ANN). The logarithmic sigmoid transfer function was used as the activation function for all units. Training of the network incorporated both momentum and an adaptable learning rate. The number of input units equaled the length of the feature vector, and we tested both 10 and 15 units for the hidden layer. Output units corresponded to the 5 terrains. A gait bounce signal was assigned to the class that corresponded to the output unit with maximal activation. ANN classification was implemented with Matlab's standard toolbox.

To test the classification rate of the ANNs, we randomly divided the data into test and train subsets 50 times. We then mapped each data subset into each of the three feature spaces – FFT over all data (FFTall), FFT over segmented data (FFTseg), and raw data (RAW). For each of these 150 (50x3) training sets, two ANNs were trained (differing by number of hidden layer units), and then tested with the corresponding test set. Every ANN had random weight initialization. Table I shows the average classification rate of these 300 (50x3x2) experiments across all terrains, and within each terrain class. The best result is obtained from the *Raw* feature vector and the ANN with 10 hidden units, with an average 64% classification rate across all terrains.

Next, we classified terrains using a well-known statistical method – discriminant analysis [19]. Linear discriminant analysis (LDA) finds a linear subspace that maximizes class separability among the feature vector projections in this space. The basic criterion is to enhance the difference between the class means relative to some measure of standard deviation. This is accomplished by introducing the *within-class*, *between-class*, and *total* scatter matrices. Similarly, QDA and LogDA find a quadratic and logarithmic subspace, respectively.

To test these classification methods, we used the same 50 ANN random train and test subsets mapped into the 3 feature spaces. These training sets were used to find appropriate subspaces using LDA, QDA, and LogDA methods, based on Ripley's book [20] and implemented with Kiefte's Matlab *discrim* toolbox [21]. Mean classification rates over the 450 (50x3x3) experiments of the corresponding test sets are presented in Table II. The best result is obtained from the *FFTall* feature vector and the QDA method, with an average 83% classification rate.

Finally, we experimented with Support Vector Machines (SVM) which show high performances in practical applications [22]. SVM corresponds to a linear method in a very high dimensional feature space that is nonlinearly related to the input space. With the use of kernels, all necessary computations are performed directly in input space instead of in high dimensional space. There are three common kernels that we used in our work, namely polynomial, radial basis function, and sigmoid kernels.

To test these classification methods, again we used the 50 ANN random train and test subsets mapped into the three feature spaces. Each of these sets were trained and classified with an SVM based on the three kernel types. SVMs were implemented on Matlab using the OSU SVM Toolbox [23]. Mean classification rates over the 450 (50x3x3) experiments of the corresponding test sets are presented in Table III. The best result is obtained from the *Raw* feature vector and the RBF kernel, with an average 64% classification rate.

The results from the above three classifiers demonstrate the impact of feature space, of classifier, and of user-defined parameter selection on the success rate. In all cases, the FFT applied over all data provided the best feature vector, and we consider an 83% classification rate satisfactory. We did have concerns as to their flexibility and robustness, therefore we developed a meta-classifier that extracts terrain class from the spatiotemporal pattern of the gait bounce signal.

B. Meta-Classifier

Variations in gait bounce result from interactions of the limbs with the terrain. For example, gait bounce is dampened due to sinkage during initial contact on a highly compliant surface and it is chaotic due to random slippage on a rough surface. Different limb-terrain interactions generate

	Total	Carpet	BB	Chip	Foam	Rock
Raw						
Poly	**38**	33	89	17	34	17
RBF	**64**	93	91	31	80	21
Sig	**63**	93	91	31	81	20
FFT all						
Poly	**36**	43	70	20	29	18
RBF	**48**	53	90	18	69	10
Sig	**44**	56	87	20	51	6
FFT sec						
Poly	**35**	40	71	20	27	18
RBF	**46**	49	93	16	65	8
Sig	**43**	57	88	21	40	12

TABLE III

Mean classification rates over 50 random test sets. RAW, FFTALL, *and* FFTSEC *indicate feature space.* POLY, RBF, *and* SIG *indicate kernels.*

gait bounce signals with different information content, thus deliberate limb motions can effect higher information content. Segmentation of the gait cycle based on the limb-terrain interaction isolates portions of the gait bounce signal with high information content. It is useful to think of these cycle segments as motion primitives, which are typically defined in relation to terrain interaction. For example, *TouchGround* is defined as rotation of the shoulders until ground-contact is made. The decoding of, then sequencing of gait bounce from each primitive produces yet another signal whose temporal ordering encodes terrain class. This is analogous to word recognition in which the spatial pattern of the speech encodes phonemes and the sequencing of the phonemes encodes the word. To classify terrain in this manner, we developed a meta-classifier with discriminant analysis and hidden Markov model classifier components.

Hidden Markov models (*see* [24] for a tutorial) are a method to model stochastic sequential events. A model (λ) consists of states and their corresponding probabilities of observations, as well as probabilities of transitions between states. Given a sequence of observations, O, and a model λ, one can ask what is $P(O|\lambda)$, the probability of observations O given λ. Essentially, this is a measure of how well the model represents the event. To create a model, training data (a set of observations O_i for $i = 1..n$) is used to modify an initial estimate of model parameters with the goal of maximizing $P(O|\lambda)$ across all observations. For classification, a model is created for each class (λ_c for $c = 1..m$), and class membership is assigned to a novel observation O, based on the the model with the highest probability $P(O|\lambda_c)$.

To construct and train our meta-classifier, we employed a multi-step process using the data collected for the experiments described above. We first defined eight overlapping segments of the gait bounce signal based on motion primitives of the swimming gait. Each segment was mapped into frequency space using an FFT to generate a feature vector. The feature vectors from a subset of all data were used to train eight segment classifiers (i.e. quadratic discriminant analysis functions), which were subsequently used to generate an 8-element observation sequence. The observation sequences from a different subset of all data were used to train 5 hidden Markov models (one for each terrain class) consisting of 8 states (one for each cycle segment). We tested the classification models using the remaining data that was not used in any of the training phases of the classifier. Again, we achieved a high success rate of 84%.

While the advantages of the meta-classifier over other techniques does not seem immediately obvious in this small example, there are many reasons we think it is superior. First, the observation sequence can be of any length without altering the classifier, which is advantageous in classifying over multiple gait cycles. Using spatial techniques, either the results from each individual cycle would have to be combined in some weighted fashion, or a separate classifier would have to be trained and maintained that used a larger feature vector. We performed classification over three gait cycles without any alteration of the meta-classifier, which increased our classification rate to 90%. Additional merits of the meta-classifier will present themselves in our discussion below regarding classification and adaptation.

V. CLASSIFICATION AND ADAPTATION

Adaptation is key in dynamic and unknown environments, and the terrain classification techniques above can provide essential information for effective locomotion in these domains. Based on past experience, gait parameters can be tuned to optimize performance while traversing the identified terrain, however, parameter tuning complicates terrain classification. Traversing the same terrain using different gait parameters will generate distinct gait bounce signals. This necessitates broad generalization within each terrain classifier, while maintaining separability (distinction) across classifiers.

In preliminary work to address this problem of parameter tuning, we collected additional data following the procedures outlined above, using a different parameter set of the swimming gait. (Subsequently in the text, *distinct gaits* will refer to swimming gaits with distinct parameter values.) Using the combined data from the two distinct gaits, we trained classifiers to recognize either 5 terrains, such that the data from the distinct gaits was combined, or 10 terrains, such that each gait-terrain pair defined its own class. We experimented with both the QDA classifier and the meta-classifier. The results are presented in Table IV.

From these results, we can see the first performance improvements of our meta-classifier over the QDA technique. We attribute this to better terrain classification using the second gait, which increases the average over the two. We believe that the meta-classifier will outperform QDA across a variety of gaits due both to the segmenting of the gait bounce signal which isolates moments with high information content and to an HMM's ability to pair two disparate observations with the same class while discounting an observation situated between them. Spatial discriminants cannot distinguish a class that is spatially enveloped by another. Further investigation of this hypothesis is reserved for future work.

	Total	Carpet	BB	Chip	Foam	Rock
5 class						
Quad DA	**78**	90	84	62	93	62
meta-class	**82**	91	84	66	93	74
10 class						
Quad DA	**82**	92	88	72	92	66
meta-class	**84**	94	85	67	94	78

TABLE IV

Mean classification rates from 2 distinct swimming gaits. The meta- and QDA-classifiers were trained based on 5- and 10-way classification (i.e. either 5 terrain classes or 10 terrain-gait classes).

	Energy (J)			J/cm		
carp	63.6	71.9	106.5	3.70	1.43	2.38
BB	62.4	75.4	116.4	2.12	3.57	3.46
foam	63.7	90.3	112.7	1.76	1.59	2.51

TABLE V

Energy and Efficiency calculated from current draw and distance traveled while TerminatorBot executed three gait cycles of two distinct swimming gaits (columns 1 and 2) and the bumpy wheel gait (column 3).

In looking at Table IV, we can also note that we have maintained good classification rates across distinct gaits. It is reasonable to assume that we can maintain this standard if we continue to train separate classifiers for each gait-terrain class, but the viability of this approach is dubious due to space and time constraints. Ideally, we would like to move away from the notion of terrain classes, because it is not that the label "sand" is useful, rather it is knowing that sand is a compliant surface and that the robot interacts in a specific way when traversing sand. In other words, we are interested in measuring a terrain's characteristics, such as compliance, friction, or roughness, and these characteristics will manifest themselves in the gait bounce signal at different moments of the gait cycle. Our meta-classifier provides a framework to separate out those segments and to identify portions of the gait bounce signal with high information content relative to a specific terrain characteristic.

VI. Mapping Terrains to Gaits

To demonstrate how adaptation using terrain classification might work, we have run some preliminary tests to map terrains to gaits and gait parameters. An appropriate gait selection is made within the context of a performance measure. With small scale robots, efficient motion is often desirable as battery resources are at a premium. To evaluate gaits under this criteria, we walked TerminatorBot across carpet, woodchips, and foam using two distinct swimming gaits ($S1$ and $S2$), as well as the bumpy wheel gait. Current draw and distance traveled for three consecutive gait cycles was recorded over multiple runs. From this data, we calculated average total energy, J (*Joules*), necessary to execute the three gait cycles. Energy is defined over a continuous space, but we approximated this area as:

$$E = \frac{1}{f}\sum_{t=1}^{n-1} vI_t \tag{7}$$

where f is our sampling rate, n is the number of samples, v is volts (which is constant), and I_t is current draw of sample t. We also calculated Power, which is defined as $P = E/\Delta t$ providing a measure of J/sec. Finally, we calculated the amount of energy required to travel a cm,

$$F = \frac{E}{\Delta d}, \tag{8}$$

where Δd is total distance traveled. F is a measure of efficiency, although it does not follow the physics definition of efficiency. Table V outlines the results of these experiments.

In Table V, data for the two distinct swimming gaits, $S1$ and $S2$ are in the first two columns and the bumpy wheel gait data is shown in the third. These data highlight a couple of points. First, the bumpy wheel gait takes significantly more energy to execute, but this is due to its long retraction-and-placement phase that accommodates the current lack of sliprings in the prototype. The excessive power of the bumpy wheel gate exemplifies the need for minimizing movement, but it also makes a performance comparison difficult. We focus our discussion on the other two swimming gaits.

Data in the J/cm table highlight the importance of adaptation when optimizing performance for efficiency. Gait $S2$ takes, on average, 61% less J/cm than $S1$ when traversing carpet, yet takes 41% more J/cm than $S1$ when traversing woodchips. This signifies the importance of adapting the gait for efficient locomotion, because the optimal gait of one terrain is not necessarily the optimal gait of another.

The dependence of efficiency on terrain class can be partially explained by the compliance of the surface. The difference between the two gaits is the body lift (pitch angle), which is achieved by rotating the arms about the shoulder joint. Woodchips are compliant, therefore greater force exerted on the surface results in more sinkage (up to a threshold). The further the arms are buried, the more difficult it is to move. If the traversed terrain is known, then the gait can be adapted accordingly to prevent excessive sinkage.

VII. Viability of Nonlegged Platforms

Although our terrain classification technique has been investigated here in the context of a legged vehicle, our claim is that it is generally applicable to most modes of locomotion, including wheels and tracks over nonplanar environments. To support this claim, we performed a brief test on a radio-controlled tank to assess viability. We drove the tank across three of the toughest terrains – rocks, woodchips, and BB's. Tracked and wheeled vehicles do not inject periodicity into the gait bounce the way a legged vehicle does, therefore we also created a BB terrain with periodicity in the contour. This often occurs, for example, in gently rolling hills or a terraced hillside. We wanted to ensure that our technique would not be fooled by this contour-induced periodicity.

Gait bounce was derived using the equations outlined in Section III except for the low-pass filter. For TerminatorBot,

the periodicity of the gait generates the low frequency component, which is large relative to the local contour of the terrain. In contrast, the global contour of the terrain generates the low frequency component of the tank, which is large relative to the local contour of the gait. Therefore, we did not apply the low-pass filter to the tank data, because the distinguishing signatures appear at high frequencies.

Following previous protocol, the gait bounce signal was mapped into frequency space and classified with QDA. The data sets for this experiment were relatively small (11-18), therefore a different testing method was applied. For each feature vector, we determined class membership using a QDA classifier trained on all other feature vectors. The classification rates for rocks, woodchips, and BB's were 78, 91, and 94%, respectively. Note that BB's yeilded the highest accuracy despite the mixture of terrain topologies (i.e. some level, some periodic). While this small set is not intended to be conclusive, we think it convincingly demonstrates the potential for general applicability of the technique.

VIII. Conclusions and Future Work

We presented a method for terrain classification based on vehicle-terrain interaction. It provides valuable feedback for effective locomotion over unknown, natural terrains and for qualitative analysis of the terrain, both of which are important skills for autonomous operation during search-and-rescue and planetary exploration. Our straight-forward approach requires only a single, forward-looking camera with relatively little processing of the visual data. In fact, no additional image processing is required if feature tracking is implemented for other purposes, such as visual servoing. A rigorous study with more than 700 trials from a real robot demonstrated the viability of the approach for legged vehicles. The average accuracy of our meta-classifier over a single gait cylce (approximately 3cm) was 84% and over three gait cycles was 89%. Preliminary work with a treaded vehicle demonstrated its potential for non-legged platforms. The average accuracy over a smaller benchmark was over 80%.

The motivation of this work is to develop a tool to assess the terrain for intelligent adaptation. We presented some preliminary work that describes how terrain classification can be used to adapt the gait to optimize performance relative to an efficiency metric. The next obvious progression of this work is to adapt the gait *in situ*. A look-up table that maps terrain conditions to gait parameters is the simplest approach, but it lacks versatility. Instead, we plan to implement learning-based "performance servoing" to improve an evaluation metric by hypothesizing gait parameter adjustments learned from prior teleoperations and adjustments, and then, feeding back those adjustments through the evaluation metric.

IX. Acknowledgements

We thank Seth Hulst and Andrew Willouer for their time-consuming efforts in data collection. We also thank Monica LaPoint and Berk Yesin for their contributions in implementing visual servoing.

References

[1] D. Langer, J. Rosenblatt, and M. Hebert, "A behavior-based system for off-road navigation," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 6, pp. 776–783, 1994.

[2] R. Simmons, E. Krotkov, L. Chrisman, F. Cozman, R. Goodwin, M. Hebert, L. Katragadda, S. Koenig, G. Krishnaswamy, Y. Shinoda, and W. Whitaker, "Experience with rover navigation for lunar-like terrain," in *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, vol. 1, 1995, pp. 441–446.

[3] D. Gennery, "Traversability analysis and path planning for a planetary rover," *Autonomous Robots*, vol. 6, pp. 131–146, 1999.

[4] A. Howard, H. Seraji, and E. Tunstel, "A rule-based fuzzy traversability index for mobile robot navigation," in *Proc. of the IEEE Int'l Conference on Robotics and Automation*, 2001, pp. 3067–3071.

[5] D. Huber, L. Denes, M. Hebert, M. Gottlieb, B. Kaminsky, and P. Metes, "A spectro-polarimetric imager for intelligent transportation systems," in *Proc. of SPIE - The International Society for Optical Engineering*, vol. 3207, 1998, pp. 94–102.

[6] A. Talukder, R. Manduchi, R. Castano, K. Owens, L. Matthies, A. Castano, and R. Hogg, "Autonomous terrain characterisation and modelling for dynamic control of unmanned vehicles," in *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, 2002, pp. 708–713.

[7] K. Yoshida and H. Hamano, "Motion dynamics and control of a planetary rover with slip-based traction model," in *Proc. of SPIE - The International Society for Optical Engineering*, vol. 4715, 2002, pp. 275–286.

[8] K. Iagnemma, H. Shibly, and S. Dubowsky, "On-line terrain parameter estimation for planetary rovers," in *Proc. of the IEEE Int'l Conference on Robotics and Automation*, vol. 3, 2002, pp. 3142–3147.

[9] K. Iagnemma, A. Rzepniewski, S. Dubowsky, and P. Schenker, "Control of robotic vehicles with actively articulated suspensions in rough terrain," *Autonomous Robots*, vol. 14, no. 1, pp. 5–16, 2003.

[10] S. Hirose, "A study of design and control of a quadruped walking vehicle," *Int'l Journal of Robotics Research*, vol. 3, pp. 113–133, 1984.

[11] K. S. Espenschied, R. D. Quinn, R. D. Beer, and H. J. Chiel, "Biologically based distributed control and local reflexes improve rough terrain locomotion in a hexapod robot," *Robotics and Autonomous Systems*, vol. 18, no. 1–2, pp. 59–64, 1996.

[12] D. Wettergreen, H. Pangels, and J. Bares, "Behavior-based gait execution for the Dante II walking robot," in *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, vol. 3, 1995, pp. 274–279.

[13] M. Lewis and G. Bekey, "Gait adaptation in a quadruped robot," *Autonomous Robots*, vol. 12, pp. 301–312, 2002.

[14] R. Kurazume, K. Yoneda, and S. Hirose, "Feedforward and feedback dynamic trot gait control for quadruped walking vehicle," *Autonomous Robots*, vol. 12, pp. 157–172, 2002.

[15] R. Voyles, A. Larson, K. Yesin, and B. Nelson, "Using orthogonal visual servoing errors for classifying terrain," in *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, vol. 2, 2001, pp. 772–777.

[16] R. Voyles and A. Larson, "Terminatorbot: A novel robot with dual-use mechanism for locomotion and manipulation," *IEEE/ASME Transactions on Mechatronics, to appear*.

[17] B. Nelson, N. Papanikolopoulos, and P. Khosla, *Visual servoing – real-time control of robot manipulators based on visual sensory feedback*. World Scientific Publishing Co. Pte. Ltd, 1993, ch. Visual servoing for robotic assembly, pp. 129–164.

[18] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. Jonh Wiley and Sons, Inc., 2001.

[19] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Academic Press, New York, 1990.

[20] B. Ripley, *Pattern recognition and neural networks*. Cambridge University Press, 1996.

[21] M. Kiefte, "*discrim* Matlab toolbox," Availabe at http://www.mathworks.nl/matlabcentral/fileexchange in Statistics and Probability.

[22] M. Hearst, B. Scholkopf, S.Dumais, E. Osuna, and J. Platt, "Trends and controversies – support vector machines," *IEEE Intelligent Systems*, vol. 4, no. 13, pp. 18–28, 1998.

[23] J. Ma and Y. Zhao, "Ohio State University SVM Classifier Matlab toolbox (v3.0)," Available at http://eewww.eng.ohio-state.edu˜maj/osu_svm.

[24] L. R. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.