# Traversability Classification using Unsupervised On-line Visual Learning for Outdoor Robot Navigation

Dongshin Kim Jie Sun Sang Min Oh James M. Rehg Aaron F. Bobick
{dongshin, sun, sangmin, rehg, afb}@cc.gatech.edu
College of Computing, Georgia Institute of Technology
Atlanta, GA 30332-0280

***Abstract*— Estimating the traversability of terrain in an unstructured outdoor environment is a core functionality for autonomous robot navigation. While general-purpose sensing can be used to identify the existence of terrain features such as vegetation and sloping ground, the traversability of these regions is a complex function of the terrain characteristics and vehicle capabilities, which makes it extremely difficult to characterize *a priori*. Moreover, it is difficult to find general rules which work for a wide variety of terrain types such as trees, rocks, tall grass, logs, and bushes. As a result, methods which provide traversability estimates based on predefined terrain properties such as height or shape will be unlikely to work reliably in unknown outdoor environments. Our approach is based on the observation that traversability in the most general sense is an *affordance* which is jointly determined by the vehicle and its environment. We describe a novel on-line learning method which can make accurate predictions of the traversability properties of complex terrain. Our method is based on autonomous training data collection which exploits the robot's experience in navigating its environment to train classifiers without human intervention. This is in contrast to other learning methods in which training data is collected manually. We have implemented and tested our traversability learning method on an ummaned ground vehicle (UGV) and evaluated its performance in several realistic outdoor environments. The experiments quantify the benefit of our on-line traversability learning approach.**

## I. Introduction

Navigation in an unknown and unstructured outdoor environment is a fundamental and challenging problem for autonomous mobile robotics. In this paper we consider the following navigation task for a UGV : Given a goal location and a starting point, the vehicle should be able to traverse an unknown outdoor environment and reach the designated goal safely without human intervention. A key aspect of this task is to determine the *traversability* of the outdoor terrain. A standard approach to this problem uses ranging sensors such as stereo vision or radar to recover the 3-D shape of the terrain. Features of the terrain such as slope, roughness, or discontinuities are then analyzed to determine traversable regions [10], [12], [13]. Additional visual cues such as color, shape, and height above the ground plane have also been employed at [2], [5], [16]. Although these methods can be effective for some tasks such as path-following, autonomous navigation in highly-vegetated complex terrain remains a challenging problem.

Our approach is based on the observation that traversability in the most general sense is an *affordance* which is jointly determined by the vehicle and its environment. As originally conceived by Gibson [4], an affordance is an actionable property which exists between the environment and an actor. While general-purpose sensing can be used to identify the existence of terrain features such as vegetation or sloping ground, the traversability of these regions is a complex function of the terrain characteristics and vehicle capabilities, and it is extremely difficult to characterize *a priori*. Moreover, the difficulty is compounded by the wide variety of terrain types such as trees, rocks, tall grass, logs, and bushes. As a result, methods which provide traversability estimates based on predefined terrain properties such as height or shape will be unlikely to work reliably in unknown outdoor environments.

We adopt an *on-line learning approach* in which the vehicle learns the affordance of traversability through guided interactions with terrain features. The learning process produces a classifier which makes traversability predictions for new terrain regions. On-line classification enables the vehicle to generalize from a small set of interactions to a characterization of the traversability of previously-unseen terrain. A key property of our approach is that the collection of labeled training examples for traversability classification is done completely autonomously with no human intervention. This is in contrast to other learning methods in which training data is provided either implicitly or explicitly by human actions [8], [11].

Our framework for on-line learning is achieved by establishing a correspondence between the vehicle's navigation experience and the visual terrain data acquired from a pair of stereo heads. Navigation experiences such as successful traverses, slippages, and collisions are assessed automatically by means of on-board sensors such as IMU, motor current, and bumper switch. Successes and failures of the navigation provide positive and negative traversability labels for cells in a grid-based representation of the terrain surrounding the vehicle. By establishing the correspondence between labeled terrain cells and the visual features in the stereo imagery, we can automatically label visual features which correspond to traversable and nontraversable terrain examples. The automatically labeled data provide the input to an on-line classifier learning algorithm. As the robot explores its environment, the classifier is trained incrementally. At any point in time, the classifier can make predictions about the traversability of the visible terrain based on its past experiences. These predictions are used to continuously plan an optimal path to the goal.

We have implemented and tested our traversability learning

method on a UGV and evaluated its performance in the several realistic outdoor terrains. The experimental results demonstrate that the learner successfully learns traversability concepts such as traversable tall grass and nontraversable logs/trees. We show that this new learning capability allows the robot to accomplish navigation tasks which cannot be achieved with a conventional stereo-based obstacle detector. This paper makes three main contributions:

- A novel on-line algorithm for learning the traversability of unknown outdoor terrain
- A method to automatically acquire labeled training examples without human supervision
- Experimental demonstration of successful learning of complex terrain classes and the resulting improved ability to finish navigation tasks in challenging outdoor environments.

## II. Related work

We believe that this work is the first to cast traversability estimation as a problem of on-line affordance learning in an unstructured environment. Previous work on affordance learning has primarily addressed more static domains such as robotic manipulation [3], [14]. Most previous work on outdoor navigation either characterizes the concept of traversability *a priori,* or learns the concept of traversability from manually acquired data. A preliminary version of this work is presented in [15].

Previous efforts in which the characteristics of traversability or obstacles are defined *a priori* include [1], [5], [10], [12], [13], [17]. The classification functions are hand-designed based on knowledge of the properties of terrain features such as 3-D shape, roughness, image edges or discontinuities. While these works deal effectively with a class of terrain features, they are insufficient to address the broad range of terrain and vegetation types found in natural outdoor environments. For example, it would be difficult to use these methods to define characteristics of traversable vegetation such as tall grass or small bushes. In contrast, our approach is more broadly applicable to any environment in which the robot can be safely driven as it uses the interaction between the vehicle and its environment to ground the problem of traversability classification.

Other previous work has adopted a learning approach to robot navigation. For example, the ALVINN system [11] for road-following used a neural network to learn steering angle corrections. Other examples include the detection of open areas using learned visual features [6], [8]. These works demonstrate that learning methods can be successfully applied to the navigation task. However, our approach differs from them in that it is an unsupervised method which does not require human action to train the learner. Moreover, our approach provides the robot with the ability to adapt to a previously unseen environment through on-line learning.

There are other works which are closely related to ours which develop on-line learning approaches where the data are gathered through the interaction between the robot and the environment. For example, [16] describes an on-line learning method for indoor robot navigation where stereo sensing in the vicinity of the robot provides the training data for a color-based ground plane classifier. However, this method uses a flat ground assumption and color-based classification to predict traversable (i.e. flat) regions of the terrain. A recent work [18] on outdoor navigation simultaneously collects training data and learns a terrain model through real-time interaction with the outdoor environment. Our work differs from theirs in that we learn a direct mapping from observations to affordances. In contrast, the traversability sensing in [18] is based upon the analysis of an estimated terrain surface.

We also note that our method uses a grid-based representation of terrain, which is common in many navigation methods for mobile robotics. While we employ a single level of resolution in our current approach, it may be interesting to consider multi-resolution techniques described in [9].

## III. Learning Traversability with Autonomous Data Collection

### A. Learning a Mapping from Appearances to Affordances

Our approach to inferring traversability is premised on four key assumptions. First, we assume that visual features derived from stereo vision and color imagery are sufficient to discriminate between terrain regions from the standpoint of the traversability affordance. Specifically, we assume that two terrain locations which produce similar feature values will have similar properties of traversability. Note that this does not imply that all traversable terrain locations look similar— there is no simple description of traversable regions in terms of image appearance or geometry. As is standard in classification problems, the question of whether two regions with different traversability properties can be discriminated will depend upon the representational capabilities of the feature space. While it is always possible to identify extremely challenging situations, like logs or stumps hidden behind dense grass or ferns, we demonstrate in Section IV that features based on image texture and stereo ranging are able to discriminate between the several terrain classes commonly encountered in outdoor navigation.

Our second assumption is that we can determine the navigation experience of the robot reliably enough to label the traversability of terrain regions as the robot attempts to drive over them. This requires the ability to differentiate slippage and collisions from successful forward motion. We employ a standard suite of on-board navigation sensors, such as IMU and motor current, to assess the progress of the robot automatically. We show that this is sufficient to accurately determine the robot's progress in a number of off-road driving situations. We maintain a grid-based map of the robot's local environment and label all grid cells over which the robot attempts to drive as either traversable or nontraversable, based on the navigation sensor values. Note that this label assignment depends critically upon the particular characteristics of the vehicle— a large vehicle may be able to drive over small saplings that would present an insurmountable obstacle to a smaller vehicle. This is exactly why traversability should

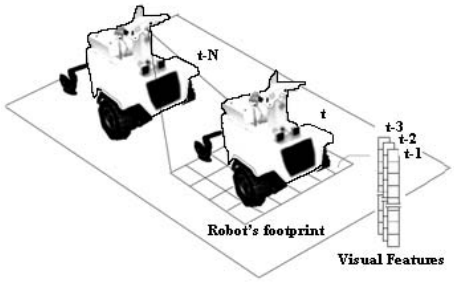


Fig. 1. Illustration of autonomous data collection. The robot images cells in a terrain map at time $t - N$. These cells lie under the robot's footpoint at time $t$.

be treated as an affordance and not simply as a predefined property of different types of terrain.

The third assumption is that we can establish the correspondence between terrain regions in the local neighborhood of the robot and visual features that result from imaging these terrain regions using a standard stereo rig. We use this correspondence to associate traversability labels which are obtained from the robot's experience with visual features of the corresponding terrain structures. For example, as the vehicle drives towards a patch of bushes, it may be able to image it from several different viewpoints, resulting in a collection of visual feature measurements that correspond to a single terrain object. When the robot finally encounters the bush and determines that it is traversable, this label can be propagated to all of the corresponding feature measurements already collected. Therefore, errors in localizing the robot to its local terrain environment will impact the ability to establish this correspondence. We demonstrate that a standard GPS-based localization scheme is sufficient for many outdoor situations (essentially everywhere except under dense tree or building cover).

Our fourth assumption is that the robot can afford to explore the terrain features in its environment without endangering the overall success of its mission. For some high-risk applications like planetary exploration or volcano exploration, where a single accident can be fatal, this may be an unreasonable assumption. However, in many practical situations we believe the advantages of learning the traversability properties of complex terrain outweigh the dangers. Encroaching vegetation is a ubiquitous property of trails and clearings in many outdoor environments. The ability to learn, for example, that the tall grass which lines a trail or borders a clearing is traversable can also mean the difference between success and failure. Navigation techniques which are based on the assumption that any terrain structure with sufficient vertical extent is an obstacle may bar the vehicle from the most simple and desirable path to its objective. An example of this situation is given in Section IV.

### *B. Approach and Architecture*

There are two key elements of the traversability learning algorithm: a mechanism to autonomously collect and label training data as the robot interacts with its environment, and

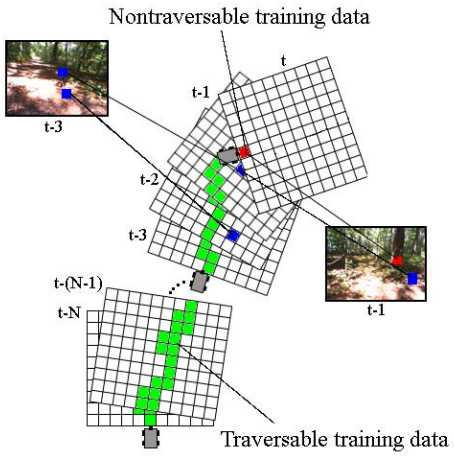


Fig. 2. Autonomous labeling of terrain cells and corresponding visual features using local maps. The green cells were found to be traversable by the robot while the red cell encountered at time $t$ is a nontraversable obstacle.

an online method to learn the concept space for traversable and non-traversable terrain. Our data collection approach is illustrated in Fig. 1. The robot images the terrain in front of it at time $t - N$. The robot stores the resulting image patches in a data pool. Each image patch is an observation of a single cell in a grid-based terrain map. Initially all of this data is unlabeled, because the robot has not yet interacted with the terrain, and its traversability is unknown. At time $t$, the robot will attempt to drive over the terrain that it previously observed, thus discovering the traversability properties of the environment. Cells under the robot footprint that can be driven over are traversable and result in positive training examples, while those that hinder the robot's motion are non-traversable and therefore yield negative examples. To summarize, image data obtained in the past is associated with traversability labels obtained in the present. In order to exploit newly-acquired training data in making traversability predictions about unknown terrain, we propose an on-line learning method. The learned traversability concepts are incrementally updated with new data only, with the advantage that the updated classifier is immediately available for navigation. Another advantage of this on-line approach in comparison to an off-line method is that it reduces the memory requirements for learning.

The key technical issue in our approach is to establish a correspondence between the collected image patches and the cells in the terrain map that they represent. Mis-correspondence may result in mislabeled training data, thus affecting the accuracy of the learned traversable and non-traversable concepts. In principle we could solve the correspondence problem by collecting all of the image patches into a global map as described in [15]. In practice, however, several problems can occur. The localization of the robot has occasional errors that are caused by GPS jump and wheel slippage which introduce registration errors that result in mis-correspondence. For example, if the estimated pose has $30cm$ error, then the ground close to a tree can be labeled as non-traversable and the tree can be marked as traversable. More importantly, such registration

error accumulates through time, which makes the global map practically unattractive. To solve this, we maintain a local map instead, which only accounts for the observations that are in a certain time window with size $N$ (see Fig. 2). This implies that only data that are labeled within time $N$ are used for training, unlabeled data after time $N$ are simply disregarded. This greatly alleviates the commutated registration error.

We now describe the three key methods in our overall learning framework: autonomous data collection, online traversability learning, and terrain classification. First, our system performs autonomous data collection continuously as the robot moves through its environment, and labels collected data with local map approach. Second, the online traversability learning algorithm develops the concepts of traversability and non-traversability based on automatically labeled training data. In our approach, feature vectors constructed from image texture and stereo disparities are clustered using an on-line algorithm to obtain a compact representation of the complete set of observations. There are two groups of clusters corresponding to traversable and nontraversable feature measurements. Third, after each iteration of on-line learning, the result is a classifier which can be applied to input data to make traversability predictions. This terrain classification is performed simultaneously with on-line learning, and provides cues for navigation which guide the planner that is responsible for generating the robot's path through its environment.

### C. Autonomous Data Collection

The training data consists of feature vectors of quantized visual and geometric information from image patches. Each image patch is encoded by a 13 dimensional feature vector $F$, and assigned to a grid cell in the egocentric local map. The autonomous data collection is achieved by labelling collected features in the past using the current interacting feedbacks. Through the proposed correspondence matching process, the robot is able to label the features as being either traversable or non-traversable.

The system builds an egocentric local map $M_t$ with associated features at each time step $t$. Then, the newly built local map $M_t$ is fed into a queue of size $N$. A square grid local map which is aligned in front of the robot is illustrated in Fig. 2. And an expanded view of each cell in Fig. 2 is depicted in Fig. 1. Each local map is built in such a way that all the data at time $t$, i.e., features $F$ of an image patch at time $t$, is mapped into the proper grid cells.

The features stored in local maps are labeled at every time step based on feedback from GPS, bumper, and motor current sensors. Using pose information generated by GPS, it is possible to label the grid cells that coincide with the current robot's position as being either traversable or non-traversable. However, as the robot maintains a queue with fixed size of $N$, the robot does not maintain the features which were observed more than $N$ time steps ago. Once the features are labeled, they are input to the on-line learner which is described in Section III-D.

**Algorithm 1** Online learning algorithm

1. For each feature $f$ of a patch on the image, if the number of cluster is zero, then make new cluster center $f$.
2. If the number of cluster reaches a given maximum number, then set fixed radius, $r = r \times 2$. It enforces the criteria of adding new center.
3. If the minimum distance between $f$ and a cluster center $c_i$, $D(f, c_i) \leq r$ as well as the label of $f$ and $c_i$ is the same, then update the center of $c_i$.
4. If the label of $f$ and $c_i$ is different, then reduce the number of example in $c_i$ by 1. If the number of examples in $c_i$ reaches zero, then remove $c_i$.
5. If $D(f, c_i) > r$, then make new cluster center $f$.

The incorporation of a queue of local maps gives two practical advantages. First, it reduces the effect of the cumulative localization error upon the correspondence matching between features and feedbacks. Since our system maintains a queue of $N$ local maps, GPS errors which occurred more than $N$ time steps ago are disregarded. Note that $N$ should be set to be appropriately large so that sufficient number of samples are labeled.

Secondly, the computational overhead in terms of memory space to store visual features and CPU running time can be reduced. Note that there is a trade-off : even though larger queue size may incur more mis-labeled data and computational overhead, it increases the amount of labeled training data which will result in accelerated learning.

### D. On-Line Learning

The classifier consists of two modules : a learning module which clusters appearance feature vectors from training examples, and a classification module where novel image patches are classified based on the learned models. In order to achieve on-line learning, the training method should be incremental, and it is required to have finite storage. We note, for example, that one minute of operation of our system generates more than 60,000 automatically acquired training vectors. We quantize the input data by clustering labeled feature vectors according to a fixed hyper-spherical radius. Separate sets of clusters are maintained for positive and negative examples. Pseudo-code for the clustering method is given in Algorithm 1.

The distance function which we adopt for clustering is based on the $\chi^2$ distance measure given as follows:

$$D(f, c_i) = w \sum_{j=1}^{8} \frac{(c_i^j - f^j)^2}{2(c_i^j + f^j)} + (1 - w) \sum_{j=9}^{13} \frac{(c_i^j - f^j)^2}{2(c_i^j + f^j)}$$

Our feature space consists of two different feature sets, one with geometric characteristics and the other with appearance characteristics. They are weighted with different factors because distinct feature sets provide different classification power. The weight factor $w$ allows us to control the relative contributions of these two components, e.g., we set $w = 0.7$ because the appearance features have more discriminative characteristics.

### E. Online classifier

The online classifier must also be computationally efficient. Usually, multiple feature vectors accumulate over time in one grid cell, and we use simple majority voting to predict the traversability of each cell . The online classifier is :

$$H(g) \quad = \quad I\left[\frac{\sum_{i=1}^{n} h(f_i)}{n} \geq \theta\right] \tag{1}$$

Where $g$ is a grid cell in the local map, and the cell is associated with number of feature vectors, $f$. $n$ is the number of feature vectors in the cell. The cell $g$ is traversable if $H(g)$= 1, and otherwise if $H(g) = 0$. The equation below represents our base classifier, $h(f)$:

$$h(f) \quad = \quad I\left[\frac{D(f, c_{NT})}{D(f, c_T)} \geq \theta^{'}\right] \tag{2}$$

where $c_{NT}$ is the center of closest non-traversable cluster to $f$, and $c_T$ is the center of closest traversable cluster.

### F. Visual Features for Classification

We assume that the traversability of a terrain region can be reliably estimated from two types of information: geometry and appearance. We adopt a patch-based method where we divide the image at regular intervals and associate each image patch with a feature vector. This is done to aggregate potentially noisy sensor information. In our method a $192 \times 256$ pixel image is divided into 336 image patches of $12 \times 12$ pixels.

*Geometric* features are obtained from stereo, which provides three dimensional coordinates $(x, y, z)$ for a subset of image pixels. In order to reduce the dimensionality of our feature space and gain robustness to noise in the stereo measurements, we quantize the height estimates for each pixel using a histogram. We use a height histogram $f_H$ with 5 bins and generate feature data only from image patches with more than 50% stereo returns. The $x$ and $y$ information from each stereo measurement determines its correspondence with the terrain map (after projection through our calibration model).

While the geometric histogram features provide information about the traversability of the terrain, they are not sufficient to measure the affordance of traversability. For example, a tall (nontraversable) tree trunk and a patch of tall (traversable) grass will result in a similar height histogram. However, these patches differ in visual appearance.

The *appearance* characteristics of image patches are obtained by filtering the patches to obtain a compact representation of *texture*. We build a normalized eight dimensional texture histogram $f_T$ for each image patch by applying maximum Laws mask [7]. First, the input image is converted to a gray scale image $I$. Then, $I$ is convolved with eight Laws masks $\cup_{i=1}^{8} LM_i$ to obtain texture energy outputs $\cup_{i=1}^{8} \{I \otimes LM_i\}$. Among the eight outputs at each pixel, the index of the masks with the maximum response is chosen to be the appearance value $p_j \triangleq \text{argmax}_{\ i} \{I \otimes LM_i\}_{i=1}^{8}$ of the corresponding $j$th pixel in a patch. Consequently, the appearance values $p_j$'s are binned to obtain the appearance feature histogram $f_T$.

Consequently, we obtain a compact 13 dimensional feature vector $f \triangleq [f_H, f_T]$ for each patch where $\sum f_T = \sum f_H = 1$. The proposed representation embeds both geometric and appearance based characteristics of the area associated within each image patch.

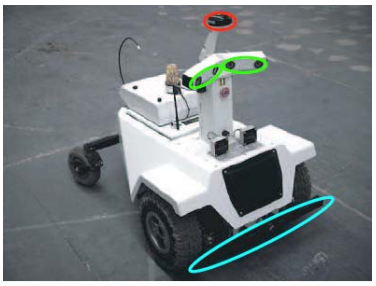

Fig. 3. Robot system. The robot is equipped with a GPS system (red), two stereo heads (green) and a bumper (cyan).

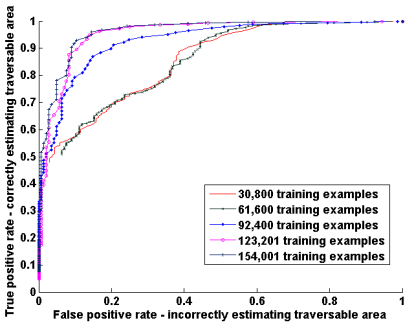


Fig. 4. ROC curves showing the results of off-line classification. As the number of training data increases, the accuracy of the proposed classifier improves.

## IV. Experimental Results

### A. Robotic System Specification

Fig. 3 shows the robot platform used in our experiments. The robot is equipped with two pairs of stereo vision cameras which collect visual and geometric data from the environment. The robot is localized using a combination of GPS readings and Inertial Measurement Unit (IMU) measurements, processed via Kalman filtering. Additionally, a bumper switch at the front of the vehicle is used along with motor current sensors to signal 'stuck' and 'slip' events, respectively, during navigation. These signals are used to identify non-traversable terrain regions.

### B. Off-line Classification Experiment

We performed an experiment to measure the performance of our classifier directly, independent of its use for on-line navigation. We collected 215,602 example patches using our autonomous data collection method. In practice, errors in pose and orientation estimation of the robot lead to the misregistration of some feature vectors. This can be viewed as

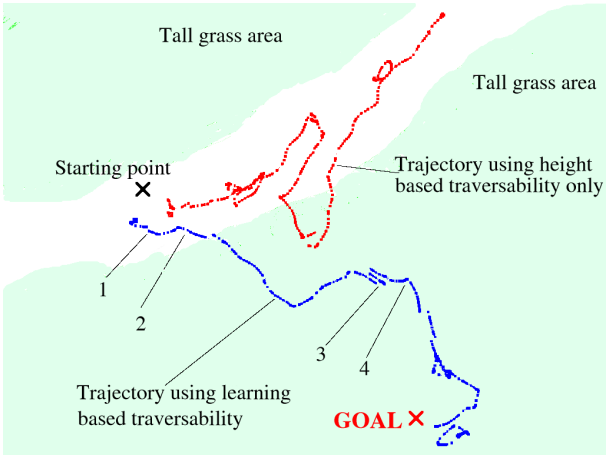


Fig. 5. Trajectories of robot on tall grass area. The blue path shows the trajectory of the robot with on-line learning capabilities and the red path is the resulting run without traversability estimator. The images corresponding to locations marked 1~4 are shown in Fig. 6. The grass area was drawn by manually based on the scene video.

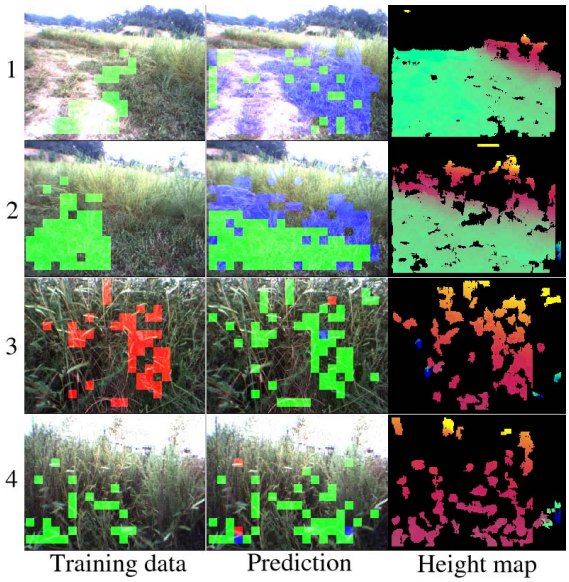


Fig. 6. Images of the test which corresponds to Fig. 5. The numbers on the left of image rows are associated with the numbers in Fig. 5. These numbered labels indicate the position of the robot where the images above are captured. Green, red, and blue colors represent traversable, non-traversable, and uncertain patch, respectively. The color map order in the height map is blue to green to red to yellow. Blue represents the lowest elevation, and yellow represents the highest elevation.

label noise which affects the quality of the classifier. For this controlled experiment, we manually filtered the example patches to limit the amount of label noise to assess the best case performance of the method. In contrast, our experiments on the robot involved no manual intervention whatsoever.

Fig. 4 shows ROC curves produced by testing the trained classifier on 61,601 independently held-out examples. A family of curves were generated by incrementally increasing the amount of training data used. The evolution of the ROC curves demonstrates that our method improves its ability to distinguish traversable and non-traversable areas as time passes and more data is collected. In this experiment, the true positive rate is about 92% at a false positive rate of 10%, which we believe to be an acceptable level of accuracy for real-world robot navigation.

### C. On-line Experimental Setting

In our experiments, $6m \times 6m$ local maps with $10cm \times 10cm$ grid cells were used; hence, there were 3,600 cells in a single local map. The size of the queue of local maps was 50, and the radius parameter, $r$, was set to 0.05. The footprint of the robot is specified as an array of cells in the map located in front of the vehicle. At every position of the vehicle in the map where navigation was successful, a $5m \times 5m$ grid of cells is labeled traversable. When robot fails to make progress, a $3 \times 5$ grid of cells is labeled as nontraversable.

A standard path planning algorithm is used to compute the best cost path to the goal that avoids colliding with any obstacles. The planner has two operational modes: conventional and learning-based. In the conventional planning mode, the cost map is based solely on the elevation map of terrain in the robot's environment which is computed through stereo ranging. Regions that exceed a slope threshold are marked as nontraversable.

In the learning-based mode, however, traversability information provided by the on-line classifier has the ability to over-ride obstacle information from stereo. Cells in which the traversability prediction is either "traversable" or "unknown" are then available to the planner. However the planner is designed to only explore unknown regions (which correspond to features vectors that are sufficiently far from all available training samples) if they happen to lie along the shortest path to goal. For example, a region of tall grass would be avoided by the conventional planner, but might be estimated to be traversable by the classifier. Then the slope-based obstacles are suppressed by the learning-based planner, allowing the robot to enter the grass area. Tall grass is an example of a terrain type that has significant vertical extent and therefore looks like a substantial obstacle to stereo, but is in fact easily traversable by our vehicle.

### D. On-line experimental results

We tested the performance of our method in two types of unstructured outdoor terrain. The first test was designed to demonstrate the benefit of traversability learning in enabling the robot to reach a goal location surrounded by tall grass, which is inaccessible to the conventional planner. The second test was designed to show that our robot can navigate successfully in an environment with obstacles that is well-suited to conventional approach. In this test, the conventional planner can reach the goal without hitting any obstacles, while the new planner based on traversability may interact with nontraversable obstacles initially, and then plan around them once it learns the correct concept class.

The result of the first experiment is shown in Fig. 5. The blue path indicates the trajectory followed by the robot using with our new learning-based method. The red path shows the path of the robot when running with the conventional planner

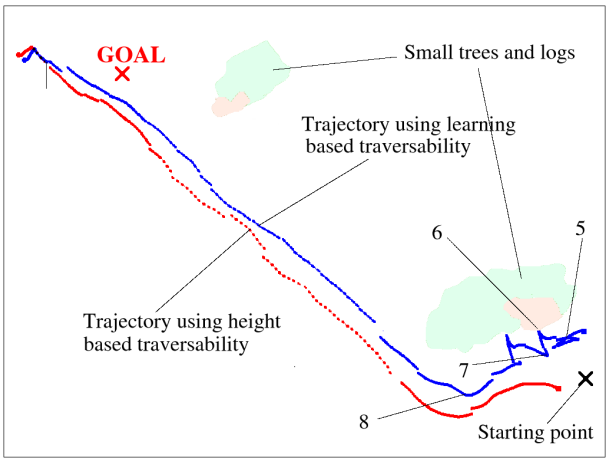


Fig. 7. Trajectories of robot on the second test site. The blue path shows the trajectory of robot with on-line learning capabilities and the red path is the resulting run without traversability estimator. The images corresponding to locations marked 5~8 are shown in Fig. 8. The area colored in light green is mostly occupied with small trees, and the area colored in light brown represents logs on the ground.

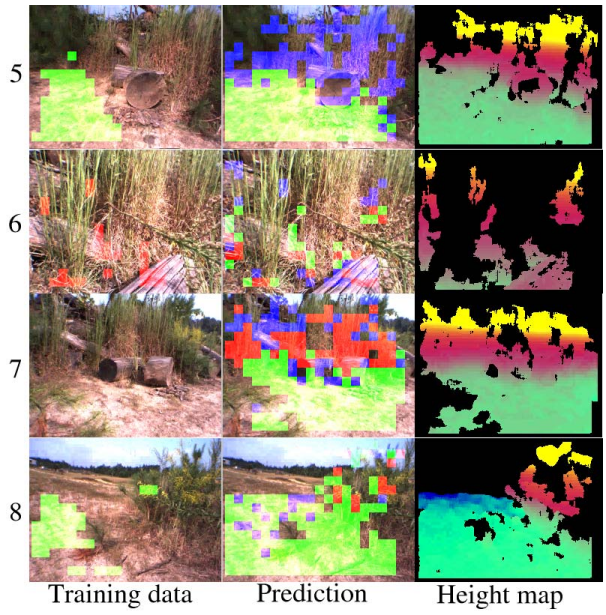


Fig. 8. Images of the test which is corresponding with Fig. 7. The numbers on the left of each image corresponds to the numbers in Fig. 7. These numbered labels indicate the position of the robot where the images above are captured. Green, red, and blue colors represent traversable, non-traversable, and uncertain patch, respectively.

which does not have traversability estimator. Note that all other aspects of the robot configuration and software were identical in the two runs.

This demonstrates that our planner with traversability prediction reached the goal which was surrounded by tall grass. Although the stereo returned high elevations in that area, the planner successfully suppressed these elevations. As a result the robot could pass through the tall grass area and reach the goal. In contrast, the conventional planner was unsuccessful. Without suppressing the elevation map from stereo vision, the planner kept searching for alternative paths around the goal - the path was closed to the goal. Note that the blue path does not lead straight to the goal, since there were some particularly dense tall grass areas on the way to the goal which proved to be nontraversable for our vehicle. The process of encountering and avoiding these regions resulted in a curved trajectory terminating at the goal position.

The numbers 1 - 4 which are labeled on the blue path in Fig. 5 indicate the position of the robot at which each image in Fig. 6 was captured. In Fig. 6, the images in rows 1 and 2, which are captured at the beginning of the run, demonstrate that the on-line learning method correctly learns the traversable grass concept class over time. The patches which are depicted in blue correspond to regions for which no prediction can be made, and the number decreases in frequency over time (looking at the middle column of images). The images in row 3 are captured at the moment before the robot hits a tall grass area which is too dense for the robot to pass through. The visual appearances of the non-traversable patches in the leftmost column are almost same to the previous trained traversable patches. In this case, it is even hard for human to distinguish between traversable area and non-traversable area. Because the number of already trained traversable patches dramatically exceeds the number of newly added non-traversable patches in this case, the method does not generalize to learn the non-traversable grass concept. In this situation, the inability to learn two conflicting grass concepts can be attributed to a lack of representational power in our choice of feature space. It remains to be seen whether other representation methods and classifier designs could distinguish between these two extremely difficult concept classes. However, we expect in general that similar terrain forms will possess similar traversability properties, allowing our method to be successful useful in practice.

The second test scenario was designed so that all height-based obstacles in the environment are true non-traversable obstacles. In this case, the robot with a conventional planner should find an optimal path to the goal easily. Also our planner with traversability estimator should not demonstrate significantly degraded performance in comparison to the conventional planner, in spite of the fact that it must learn from the scratch the concept class which was designed into the conventional planner explicitly. The result of the second experiment is shown in Fig. 7. The blue path is the result of the run with our new method, and the red path is the result of the run with the conventional planner. The numbers 5 - 8 which are labeled on the blue path indicate the position of the robot at which each image in Fig. 8 was captured. We see that both the conventional planner and our new planner could successfully complete the navigation task. At the beginning of the run, our new planner with traversability estimator interacted with logs on the ground and some small trees. Once it learned the concept class for these two obstacle types, it proceeded to the goal without any further interactions with other terrain obstacles.

In Fig. 8, the images in row 5 are captured at the beginning of the run where the learner has not yet acquired the concept

class. The image in the middle shows some blue (invalid) patches corresponding to logs and small trees for which no predictions are available. But it successfully classifies the ground region as traversable. The images in row 6 are captured at right after the robot has interacted with the log and it begins to predict that logs are non-traversable areas. The images in row 7 are captured when the robot backs up more, after hitting the log, and the classifier starts to detect non-traversable areas.

## V. Conclusions and Future Work

We have described a novel method to learn the traversability affordance between an autonomous robot and the terrain properties of an unconstrained and unknown outdoor environment. Our framework is fully automatic, and labeled training data is acquired through the experiences of the robot as it interacts with its environment. We believe this framework can be generalized to learn other affordances of outdoor terrain such as the identification of strategic positions from a navigation standpoint and the ability of the terrain to provide protection from observation (in a multi-robot scenario). We have demonstrated the successful performance of our system on two scenarios for realistic outdoor environments. One source of performance improvements which we plan to explore in future work is the use of alternative localization methods to supplement the GPS and IMU sensors used in our current work. In spite of using a highly localized map, we still have label noise due to localization and pose errors. We plan to explore visual odometry along with strategic low-level control to ameliorate these sources of error. We also plan to explore the classic tradeoffs of exploitation and exploration in the context of our new learning approach.

## Acknowledgments

This work was funded in part by DARPA under the IPTO/LAGR program (contract #FA8650-04-C-7131), and by the National Science Foundation under NSF Grant IIS-0133779. The third author is supported by a fellowship from Samsung Lee Kun Hee Scholarship Foundation.

## References

[1] P. Bellutta, R. Manduchi, L. Matthies, K. Owens, and A. Rankin. Terrain perception for demo iii. In *Intelligent Vehicle Symposium*, 2000.

[2] G. N. DeSouza and A. C. Kak. Vision for mobile robot navigation: A survey. In *IEEE Trans. Pattern Anal. Machine Intell.*, pages 24(2):237–267, 2002.

[3] P. Fitzpatrick, G. Metta, L. Natale, S. Rao, and G. Sandini. Learning about objects through action - initial steps towards artificial cognition. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Taipei, Taiwan, May 12 - 17 2003.

[4] J. J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, 1979.

[5] A. Huertas, L. Matthies, and A. Rankin. Stereo-based tree traversability analysis for autonomous off-road navigation. In *IEEE Work-shop on Applications of Computer Vision*, 2005.

[6] T. M. Jochem, D. A. Pomerleau, and C. E. Thorpe. Vision-based neural network road and intersection detection and traversal. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 344–349, 1995.

[7] K. I. Laws. Rapid texture identification. *Proc. of the SPIE - Intl. Soc. Opt. Eng.(SPIE)*, pages 376–380, 1980.

[8] J. Michels, A. Saxena, and A. Y. Ng. High speed obstacle avoidance using monocular vision and reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2005.

[9] M. Montemerlo and S. Thrun. A multi-resolution pyramid for outdoor robot terrain perception. In *AAAI Nat. Conf. on Artificial Intelligence*, 2004.

[10] R. Pagnot and P. Grandjea. Fast cross-country navigation on fair terrains. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 2593–2598, 1995.

[11] D. Pomerleau. Alvinn: An autonomous land vehicle in an neural network. In *Advances in Neural Information Processing Systems (NIPS)*, pages 305–313, 1989.

[12] A. Rieder, B. Southall, G. Salgian, R. Mandelbaum, H. Herman, P. Rander, and T. Stentz. Stereo perception on an off-road vehicle. In *Proceedings of the Intelligent Vehicles*, San Jose, CA, 2002.

[13] S. Singh, R. Simmons, T. Smith, A Stentz, V Verma, A Yahja, and K Schwehr. Recent progress in local and global traversability for planetary rovers. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 1194–1200, 2000.

[14] A. Stoytchev. Behavior-grounded representation of tool affordances. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Barcelona, Spain, April 18-22 2005.

[15] J. Sun, J. M. Rehg, and A. Bobick. Learning for ground robot navigation with autonomous data collection. Technical Report GIT-GVU-05-29, Georgia Institute of Technology, Atlanta, USA, 2005.

[16] I. Ulrich and I. Nourbakhsh. Appearance-based obstacle detection with monocular color vision. In *AAAI Nat. Conf. on Artificial Intelligence*, pages 866–871, 2000.

[17] N. Vandapel, D. F. Huber, A. Kapuria, and M. Hebert. Natural terrain classification using 3-d ladar data. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2004.

[18] C. Wellington and A. Stentz. Online adaptive rough-terrain navigation in vegetation. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2004.