2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Learning slip behavior using automatic mechanical supervision

Anelia Angelova, Larry Matthies, Daniel Helmick and Pietro Perona

Abstract— **We address the problem of learning terrain traversability properties from visual input, using automatic mechanical supervision collected from sensors onboard an autonomous vehicle. We present a novel probabilistic framework in which the visual information and the mechanical supervision interact to learn particular terrain types and their properties.**

The proposed method is applied to learning of rover slippage from visual information in a completely automatic fashion. Our experiments show that using mechanical measurements as automatic supervision significantly improves the visual-based classification alone and approaches the results of learning with manual supervision. This work will enable the rover to drive safely on slopes, learning autonomously about different terrains and their slip characteristics.

I. INTRODUCTION

Remote prediction of mechanical terrain properties and rover mobility has significant importance in autonomous navigation applications. Recent progress has been made by applying methods based on learning from examples, imitation or experience [15], [20], [26]. A commonly used concept in learning for autonomous navigation, known as *learning from proprioception* [20], [26], is to associate the terrain appearance observed from a distance with the mechanical observations made by the robot (e.g. if the terrain is traversable or not) when the corresponding location is traversed; this association is learned, thus allowing prediction of mechanical traversability properties from vision information only.

Although most navigation systems are targeted towards full vehicle autonomy, they rely mainly on offline training and use heuristics or human supervision to determine the traversability properties of a terrain type [15], [20]. However, the ultimate goal in autonomous navigation is to have a robot which is able to learn *autonomously* about different terrains and its mobility restrictions on them. For example, it is not practical to stop the exploration of a planetary rover, in order to downlink and label training data. Moreover, providing ground truth manually is prohibitive because of the huge volume of data available and using expert knowledge is expensive or might be unreliable. For example, a human operator might not have the best knowledge about soil characteristics and their influence on rover mobility.

This work is supported by NASA's Mars Technology Program funding

A. Angelova is with the Computer Science Department, California Institute of Technology, 1200 E. California Blvd, Pasadena, CA 91125 anelia@vision.caltech.edu

L. Matthies and D. Helmick are with the Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109 lhm,dhelmick@jpl.nasa.gov

P. Perona is with the Electrical Engineering Department, California Institute of Technology, 1200 E. California Blvd, Pasadena, CA 91125 perona@vision.caltech.edu

To automate the training process we propose to use the vehicle's low-level mechanical sensors which measure its slip behavior to provide supervision of the learning of terrain type from visual information. Although mechanical sensor measurements have been used to characterize terrain [5], [8], [20], [26], they have not been used to close the loop in a fully automatic vision-based learning framework and no principled approach for learning using automatic mechanical supervision has been considered.

In this paper, we propose a learning algorithm in which the supervision comes from the mechanical measurements taken by the robot and therefore can be noisy, uncertain or ambiguous. We call this scenario: *learning from automatic supervision*. We show that learning with this weaker form of supervision is more useful than ignoring the supervision and that it can bridge the gap to the performance achieved by manual supervision.

To address the problem of learning with automatic supervision, we extend the Mixture of Experts (MoE) framework [12] to allow for the mechanical measurements to act as supervision to the visual information. We propose a probabilistic framework in which the interaction between the visual and the mechanical sensory information is learned, as well as the parameters of both the terrain classification and the mechanical behavior are estimated. The problem is formulated as a maximum likelihood estimation in which the EM algorithm [7] is used to learn the unknown parameters.

We apply the proposed method to learning rover slippage from visual information. Being able to predict slip from a distance will have significant impact on future Mars rover missions, because slip has been recognized as one of the key limiting factors in the current Mars Exploration Rover (MER) mission [3], [16]. In our previous work [1], [2] we have shown the viability of the approach for prediction of slip at a future location. However, in [1], [2] the learning of terrain classification and slip models is done independently in an offline fashion, using human supervision for providing the ground truth for the terrain type. In this paper we consider learning without any supervision, using only automatic supervision from the terrain.

Previous learning approaches, which attempt to decrease the amount of human supervision involved in data labeling (the so-called *semi-supervised learning* [4], [25]), rely on at least some sort of supervision. For example, some part of the data is required to be reliably labeled in [4], or the supervision is provided in the form of pairwise constraints which are assumed to be known *apriori* [25]. We are not aware of learning methods which work with noisy or uncertain supervision or which can cope with potentially noisy or

unreliable labeling.

Some related work on *self-supervised learning* [17] in the context of autonomous navigation has emerged recently, again motivated by the need to remove human supervision and enable autonomous learning by the rover. Self-supervised learning uses one type of sensor to enhance the performance of another (e.g. learn range information from color features) and has been applied to extending the effective perception range [6], [11], [14], [17], [20], [22]. The above mentioned approaches use manual data labeling [11], [20], assume the sensor used as supervision can provide reliable labeling of terrain types [6], [14], or use heuristically defined traversability cost [22]. Moreover, they focus on only detecting one traversable class (e.g. drivable road) or consider a binary traversability value (i.e. traversable vs. non-traversable), whereas in our framework we learn both the terrain classification for multiple terrains and the *nonlinear* (real-valued output) models of the mechanical behavior for each terrain. The latter is much harder, as in our formulation we do not assume a one-to-one correlation between the mechanical sensor measurements and the corresponding representation in the vision space.

II. LEARNING WITH AUTOMATIC SUPERVISION

A. Problem formulation

Consider the problem of learning and prediction of certain *mechanical behavior* which changes depending on the terrain type, and in which some of the inputs come from some visual space Ω and the others from some mechanical information domain Φ. Denoting the mechanical behavior as $Z = F(\mathbf{x}, \mathbf{y})$, this problem could be formulated in the following way:

$$F(\mathbf{x}, \mathbf{y}) = \begin{cases} f_1(\mathbf{y}), & \text{if} \quad \mathbf{x} \in \Omega_1 \\ \vdots & \vdots \\ f_K(\mathbf{y}), & \text{if} \quad \mathbf{x} \in \Omega_K \end{cases} \quad (1)$$

where $\mathbf{x} \in \Omega$, $\mathbf{y} \in \Phi$, $\Omega \cap \Phi = \emptyset$, $\Omega_i \in \Omega$ are different subsets in the vision space, $\Omega_i \cap \Omega_j = \emptyset, i \neq j$, $f_k(\mathbf{y})$ are (nonlinear) functions which work in the domain Φ and which change their behavior dependent on terrain, and K is the number of terrains. In other words, different mechanical behaviors occur on different terrain types as determined by appearance. The term *mechanical behavior* could stand for different things. For example, f can be a function of slope angles, temperature, or some sensor based input signal (e.g. frequency). The vehicle's sensors are used to do the mechanical behavior measurements, i.e. they are received completely automatically. For example, while traversing a previously seen terrain the vehicle can measure (using an onboard algorithm called Visual Odometry (VO) [19] that estimates the actual rover pose) how much of the commanded distance it has failed to traverse on that terrain as a function of terrain slope (we call the latter 'slip' (Figure 1)). A trivial example of large slip would be a vehicle rotating its wheels on an icy road or on a sandy slope without actually moving, because of lack of traction. Regarding autonomous navigation, we are interested in predicting slip behavior on surfaces like deep sand, packed soil, and gravel, because they affect vehicle mobility differently.

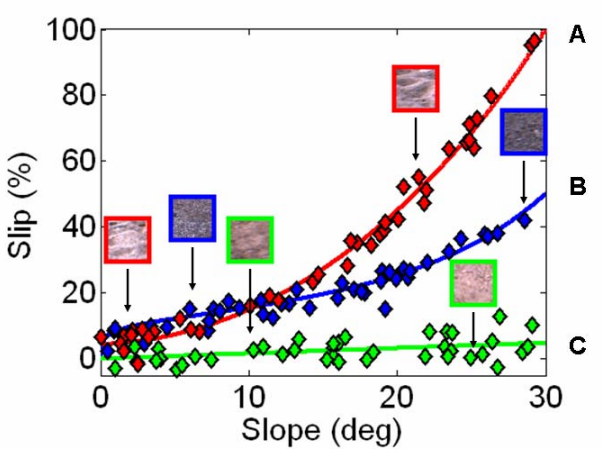

Fig. 1. A schematic of the main learning setup using automatic supervision: several (unknown) nonlinear models describe the mechanical behavior corresponding to different terrain types; each training example consists of a vision part (e.g. an image patch of this terrain) and one single point on the curve (marked with a diamond) describing the mechanical behavior. The system works without human supervision and relies on the goodness-of-fit of the mechanical behavior for automatic supervision to learn both the terrain classification and the nonlinear behaviors. In this paper, we will be using slip measurements from actual robot traversals as in Figure 2.

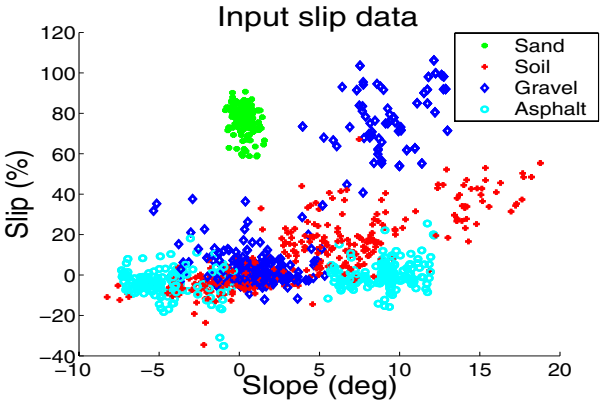

Fig. 2. Slip measurements plotted as a function of the estimated slope angles retrieved from actual rover traversals (the vision patches are not shown). The ground truth terrain types in this figure are provided by human labeling, but the proposed algorithm does not use ground truth. Instead, it learns both the terrain classification and the nonlinear slip behaviors from training data only. The data is very challenging: the slip measurements to be used as supervision are very noisy and can overlap in parts of the domain.

Figure 1 visualizes the problem when measurements of slip as a function of terrain slope are used as supervision. Each terrain measurement is composed of an appearance patch, a terrain slope estimate (from stereo or other range sensor) and a measurement of the amount of slip occurring at the location with this particular appearance and slope (note that one training example is a single point on the nonlinear curve of slip behavior). Figure 2 shows actual slip measurements taken from rover traversals. One can note that they are very noisy. Here, for simplicity, we consider only

the slip in the forward motion direction as dependent on the longitudinal slope, similar to slip measurements done for the Mars Exploration Rover [18].

In this setup, it is possible that some of the models overlap in parts of their domain (i.e. for some $i, j, i \neq j$, $f_i(\mathbf{y}) \equiv f_j(\mathbf{y})$, for $\mathbf{y} \in \Phi_0$, for some $\Phi_0 \subseteq \Phi$). For example, models A, B and C on Figure 1 overlap for $\sim 0^\circ$ slope. This is due to the nonlinearity of mechanical behavior models $f_i(\mathbf{y})$. That is, the automatic supervision for some of the training examples can be inherently *ambiguous*. Moreover, two visually similar terrains might exhibit different slip behavior (e.g. A and C), as a result, the automatic supervision should be forcing a better discrimination in the visual space. Finally, as we are working with actual rover data, the sensor based measurements will have noise from various sources, including occasional outliers due to non-modeled events from the terrain or some ground truth measurement errors.

The goal now is to learn the function $Z = F(\mathbf{x}, \mathbf{y})$ from the available training data $D = \{\mathbf{x}_i, \mathbf{y}_i, z_i\}_{i=1}^N$, where $\mathbf{x}_i$, $\mathbf{y}_i$ are the visual and mechanical domain inputs and z_i are the mechanical measurements collected by the vehicle. Thus, after the learning has completed, the mechanical behavior z for some query input example $(\mathbf{x}_q, \mathbf{y}_q)$ will be predicted as $z = F(\mathbf{x}_q, \mathbf{y}_q)$. We do not want to use manual labeling of the terrain types during training, so the mechanical measurements z_i, which are assumed to have come from one of the unknown nonlinear models, will act as the only supervision to the whole system. The main problem is that using the mechanical supervision as the only ground truth, we have to learn both the terrain classification and the nonlinear functions for each particular terrain. Note that the physical models for the particular mechanical behavior might not be known beforehand, as is the case with slip. The particular difficulty in our formulation lies in the fact that a combinatorial enumeration needs to be solved as a subproblem, which is known to be computationally intractable [13].

Note that in our training setup, the slip measurements come from some unknown nonlinear functions (Figure 2) and could not be simply clustered into well discriminable classes, as previously done for characterizing terrains from mechanical vibration signatures [5], [8], or for learning terrain traversability in self-supervised learning [6], [11], [14], [17]. So, using these slip measurements as supervision is not a trivial extension of supervised learning.

B. Approach

We can consider the problem formulated in (1) as having two parts, a vision part and a mechanical behavior part, which are linked through the fact that they refer to the same terrain type, so they both give some information about this terrain. In other words, during learning, we can use visual information to learn something about the nonlinear mechanical models, and conversely, the mechanical feedback to supervise the vision based terrain classification. The main challenge is how to make those two different sets of information interact.

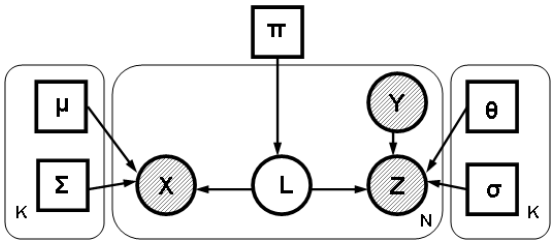

Fig. 3. The graphical model for the maximum likelihood density estimation for learning from both vision and automatic mechanical supervision. The observed random variables are displayed in shaded circles.

We provide a solution to (1) in a maximum likelihood framework. The main problem is that the decision about the terrain types and learning of their mechanical behavior are not directly related (i.e. they are done in different/decoupled spaces) but they do refer to the same terrains. So, we introduce hidden variables L (from a multinomial distribution with a parameter π) which will define the class-belonging of each training example ($L_{ij} = 1$ if the i^{th} training example $(\mathbf{x}_i, \mathbf{y}_i, z_i)$ has been generated by the j^{th} nonlinear model and belongs to the j^{th} terrain class). Now, given the labeling of the example is known, we assume that the mechanical measurements and the visual information are independent. So, the complete likelihood will factor as follows:

$$P(X, Y, Z, L|\Theta) = P(X|L, \Theta)P(Y, Z|L, \Theta)P(L|\Theta),$$

where $\Theta = \{\mu_j, \Sigma_j, \theta_j, \sigma_j, \pi_j\}_{j=1}^K$ contains all the parameters that need to be estimated in the system. μ_j, Σ_j are the means and covariances of the K clusters of vision data, θ_j are the parameters of the nonlinear fit of the mechanical data, σ_j are the covariances (here it is the standard deviation, as the final measurement is one dimensional), and π_j are the prior probabilities of each terrain class. The graphical model corresponding to this case is shown in Figure 3. We have assumed that the number of terrain types K is known and that we have a fixed appearance representation $\mathbf{x}$ which is good enough for our purposes.

Using the hidden variables, the complete log likelihood function (CL) for the whole data could be written as follows:

$$CL(X, Y, Z, L|\Theta) = \sum_{i=1}^{N}\sum_{j=1}^{K} L_{ij} \log P(\mathbf{x}_i|L_{ij} = 1, \mu_j, \Sigma_j) +$$

$$\sum_{i=1}^{N}\sum_{j=1}^{K} L_{ij} \log P(\mathbf{y}_i, z_i|L_{ij} = 1, \theta_j, \sigma_j) + \sum_{i=1}^{N}\sum_{j=1}^{K} L_{ij} \log \pi_j$$

The hidden variables simplify the problem and allow for it to be solved efficiently with the Expectation Maximization (EM) algorithm [7]. The vision information X and mechanical information Y, Z are considered to come from particular probability distributions, conditioned on the label. Those distributions are modeled, so that a tractable solution to the complete maximum likelihood problem is achieved. The vision data is assumed to belong to any of the K clusters (terrain types). For each of them, the mean and covariance parameters need to be estimated. The probability

of a datapoint $\mathbf{x}_i$ belonging to a terrain class j is expressed as:

$$P(\mathbf{x}_i|L_{ij}=1,\mu_j,\Sigma_j)=\frac{e^{-\frac{1}{2}(\mathbf{x}_i-\mu_j)^T\Sigma_j^{-1}(\mathbf{x}_i-\mu_j)}}{(2\pi)^{d/2}|\Sigma_j|^{1/2}},$$

where d is the dimensionality of the vision space. The mechanical measurement data is assumed to come from a nonlinear fit, which is modeled as a General Linear Regression (GLR) [21]. GLR is appropriate for expressing nonlinear behavior and is convenient for computation because it is linear in terms of the parameters to be estimated. For each terrain type j, the regression function $\tilde{Z}(Y)=E(Z|Y)$ is assumed to come from a GLR with Gaussian noise: $f_j(Y)\equiv Z(Y)=\tilde{Z}(Y)+\epsilon_j$, where $\tilde{Z}(Y)=\theta_j^0+\sum_{r=1}^{R}\theta_j^r g_r(Y)$, $\epsilon_j\sim N(0,\sigma_j)$, g_r are several nonlinear functions selected before the learning has started. Some example functions are: x, x^2, e^x, $\log x$, $\tanh x$ (those functions are used later on in our experiments with the difference that the input parameter is scaled first). The parameters $\theta_j^0,...,\theta_j^R,\sigma_j$ are to be learned for each model j. The following probability model for z_i belonging to the j^{th} nonlinear model (conditioned on $\mathbf{y}_i$), is assumed:

$$P(z_i|\mathbf{y}_i,L_{ij}=1,\theta_j,\sigma_j)=\frac{1}{(2\pi)^{1/2}\sigma_j}e^{-\frac{1}{2\sigma_j^2}(z_i-G(\mathbf{y}_i,\theta_j))^2},$$

where $G(\mathbf{y},\theta_j)=\theta_j^0+\sum_{r=1}^{R}\theta_j^r g_r(\mathbf{y})$ and $\theta_j=(\theta_j^1,...,\theta_j^R,\theta_j^0)$. $P(\mathbf{y}_i)$ is given an uninformative (here, uniform over a range of slopes) prior.

The EM algorithm applied to our formulation of the problem is shown in Figure 4. In the E-step, the expected values of the unobserved label assignments L_{ij} are estimated. In the M-step, the parameters for both the vision and the mechanical side are selected, so as to maximize the complete log-likelihood. As the two views are conditionally independent, the parameters for the vision and the mechanical side are selected independently in the M-step, but they do interact through the labels, as they both provide information for estimation in the E-step. Some clarifications of the algorithm in Figure 4: L_j^t is a diagonal NxN matrix which has $L_{1j}^t,...L_{Nj}^t$ on its diagonal, G is a Nx$(R+1)$ matrix, such that $G_{ir}=g_r(\mathbf{y}_i)$, $G_{i(R+1)}=1$ and Z is a Nx1 vector containing the measurements z_i (the derivations follow standard manipulations on normal distributions).

C. Discussion

Within our maximum likelihood framework, it can be seen that the algorithm copes naturally with examples providing ambiguous supervision (i.e. belonging to areas of overlap of several different nonlinear models). For those examples the algorithm falls back to using the visual input only, because the probability of belonging to each of the overlapping models is almost equal.

The EM solution is prone to getting stuck in a local maximum, which is also possible in our formulation (e.g. one can imagine creating adversarial mechanical models to contradict the clustering in vision space). In practice, for the autonomous navigation problem we are addressing, our intuition is that the mechanical measurements are correlated to a large extent to the vision input and will be only improving the vision based classification. This is seen in the experiments in the next section.

Input: Training data $\{\mathbf{x}_i,\mathbf{y}_i,z_i\}_{i=1}^N$, where $\mathbf{x}_i$ are the vision domain data, $\mathbf{y}_i$ are the mechanical domain data, z_i are the mechanical supervision measurements.

Output: Estimated parameters $\mathbf{\Theta}$ of the system

Algorithm:

1. Initialize the unknown parameters $\mathbf{\Theta^0}$. Set $t=0$.

2. Repeat until convergence:

2.1. (E-step) Estimate the expected value of L_{ij}

$$L_{ij}^{t+1}=\frac{P(\mathbf{x}_i|L_{ij}=1,\mathbf{\Theta^t})P(\mathbf{y}_i,z_i|L_{ij}=1,\mathbf{\Theta^t})\pi_j^t}{\sum_{k=1}^{K}P(\mathbf{x}_i|L_{ik}=1,\mathbf{\Theta^t})P(\mathbf{y}_i,z_i|L_{ik}=1,\mathbf{\Theta^t})\pi_k^t}$$

2.2. (M-step) Select the parameters Θ^{t+1} to maximize $CL(X,Y,Z,L|\Theta^t)$:

$$\mu_j^{t+1}=\frac{\sum_{i=1}^{N}L_{ij}^{t+1}\mathbf{x}_i}{\sum_{i=1}^{N}L_{ij}^{t+1}};\ \mathbf{\Sigma}_j^{t+1}=\frac{\sum_{i=1}^{N}L_{ij}^{t+1}(\mathbf{x}_i-\mu_j^{t+1})(\mathbf{x}_i-\mu_j^{t+1})^T}{\sum_{i=1}^{N}L_{ij}^{t+1}}$$

$$\theta_j^{t+1}=(G^TL_j^{t+1}G)^{-1}G^TL_j^{t+1}Z$$

$$(\sigma_j^2)^{t+1}=\frac{\sum_{i=1}^{N}L_{ij}^{t+1}(z_i-G(\mathbf{y}_i,\theta_j^{t+1}))^2}{\sum_{i=1}^{N}L_{ij}^{t+1}};\ \pi_j^{t+1}=\sum_{i=1}^{N}L_{ij}^{t+1}/N$$

2.3. $t=t+1$

Fig. 4. EM algorithm updates.

D. Experimental evaluation of the framework

To evaluate the performance of the proposed algorithm for learning from automatic mechanical supervision we perform a simulation in which the slip behavior models are created from known nonlinear models as in Figure 1. This experiment is partially controlled, to be able to report classification error, comparing to human labeled terrains, and to measure goodness-of-fit to the mechanical behavior models. The models used are generated to simulate actual 'slip vs. slope' behavior, as measured and reported in [18] for MER. The image patches are collected from three actual terrains, while driving on sand, soil and gravel, but the visual representation used is also very simple: it is composed of the average normalized Red and Green of the terrain patch. In the next section we will show experiments on real field-test data collected by the rover.

The experimental setup generally follows Figure 1: a set of slip and slope measurements are generated from each of the curves and are paired with appearance patches coming from actual terrains. It is not known to the algorithm which terrain classes the input examples belong to.

Table I gives a summary of the results when learning with and without mechanical supervision, comparing to human labeled ground truth. The error reported (Abs) is the average absolute difference between the predicted slip and the actual values of the nonlinear fit. The results are averaged over 100 independent runs. Each run uses about 400 training

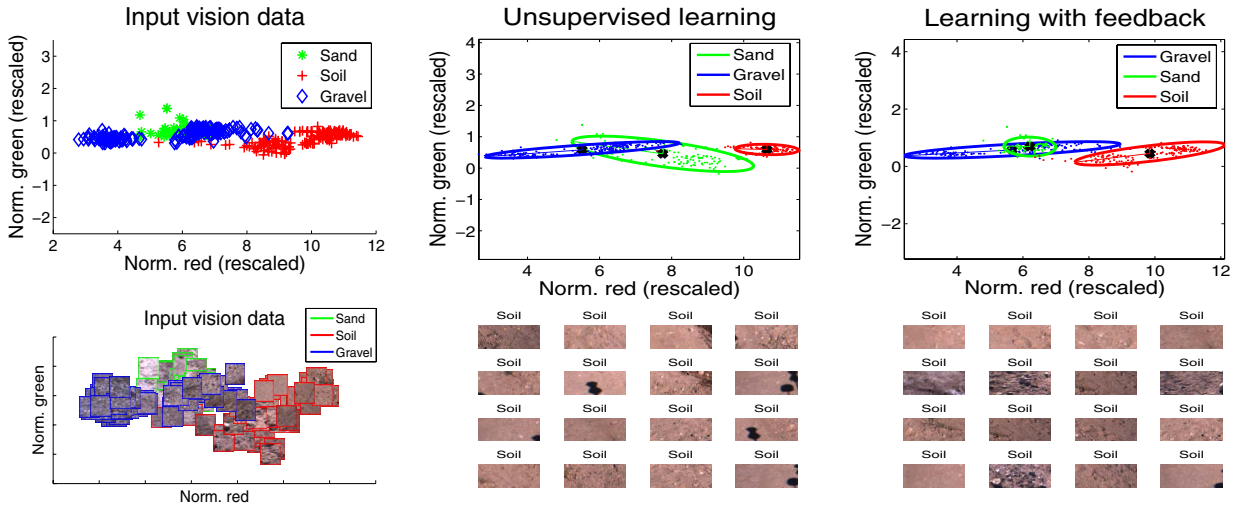

Fig. 5. Initial vision space with ground truth classification and some example terrain patches (left column), the classification in the vision space after learning without supervision (middle column), and after learning with automatic supervision (right column). Some example patches, representatives of the learned soil class corresponding to the two learning scenarios, are shown in the bottom row.

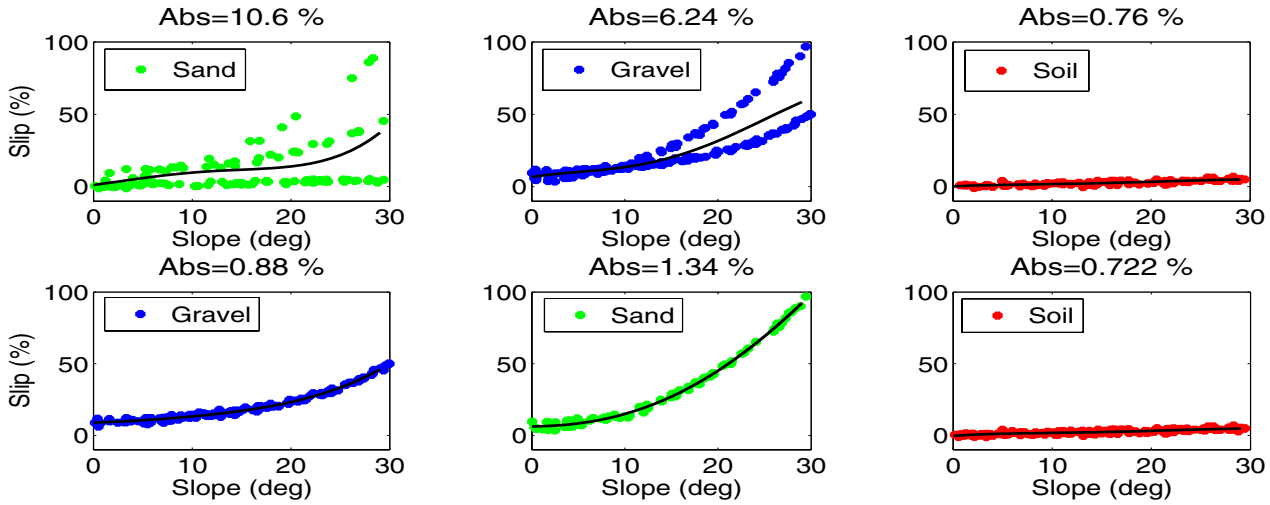

Fig. 6. Learned nonlinear models for the three classes superimposed on the training data. Learning without supervision (top), learning with automatic mechanical supervision (bottom). When learning without supervision, some initial classification errors in the vision space cause examples from the wrong models to be assigned to a class and, as a result of that, the wrong slip models are estimated (top row). This is not the case if automatic mechanical supervision is used during training (bottom row).

TABLE I

SIMULATED EXPERIMENT. SUMMARY OF THE TEST PERFORMANCE.

Learning scenario	Terrain classif. err. (%)	Slip error (Abs) (%)
Unsupervised	41.3	6.94
Autom. supervision	29.5	4.47
Human supervision	15.7	3.49

and 400 test examples, randomly selected from the data. Figure 5 shows the classification in the vision space and some of the appearance patches which have been learned to belong to one of the classes. The learned nonlinear models for the three terrains are shown in Figure 6. In the case of learning without supervision, essentially an unsupervised clustering is done in the visual space. The mechanical models are fit after the classification has converged and the data corresponding to each terrain type is used. As seen on Figure 6, the mechanical models could not be estimated correctly, because of initial errors in the terrain classification (Figure 5). This results in larger slip errors when not using supervision. Using mechanical feedback as supervision helps the classification algorithm and the right models can be estimated, leading to a smaller test error (Table I). Learning with automatic supervision achieves 72% of the possible margin for improvement between the unsupervised and the learning with human supervision. As seen in Figure 5, when learning with automatic supervision, the vision based classifier has been forced to learn that some additional darker patches also belong to the soil class,

which was not immediately apparent in the unsupervised classification. This point is important, as real-life data offers a lot of variability in appearance, and even if some limited supervision is admissible, a human operator would not be able to show to the system all possible illumination or view invariances of a terrain patch, for example. The mechanical supervision can be used to do that instead. As only visual information is used for determining the terrain type in the test mode, the terrain classification errors for all three scenarios (Table I) are much larger compared to the training mode (Figure 6). This is due to a significant overlap in the vision space. From this experiment we can conclude that using automatic supervision outperforms the unsupervised learning and is able to retrieve the correct underlying terrain classification achieving performance comparable to human supervised learning.

III. SLIP LEARNING AND PREDICTION WITH AUTOMATIC SUPERVISION

In this section we apply the algorithm for learning from automatic supervision to the problem of slip prediction. The main idea is to have the rover drive on different terrains collecting visual patches and measuring the rover's slip occurring at particular slope angles at each traversed location. We then apply the algorithm from Section II, training simultaneously the terrain classification and the nonlinear slip models without human supervision and by using the slip measurements as the only supervision. We further evaluate slip prediction based on the learned terrain classification and the learned slip models. We compare the results to learning when the training examples are classified by human and to unsupervised learning, i.e. when using only the input visual features for learning.

A. Experimental setup

The dataset for this experiment is collected on several different terrains (soil, gravel, asphalt) in a natural park with an autonomous robot. Here we consider measurements of the actual slip experienced by the rover at the corresponding slopes. Figure 2 visualizes the mechanical part of the data. As seen, the data is very noisy, as a result of being collected on natural off-road terrains. Because of certain limitations in the mobility of the vehicle on some off-road terrains, not all possible slip angles and behaviors could be collected (e.g. the robot recorded a slip of about 80% on flat sand and could not climb sandy slopes of any degree; so, the 'sand' dataset is not considered in this experiment, as it is very limited in terms of slopes (Figure 2)).

1) Robot platform: This research is targeted for learning and prediction of slip for a Mars exploration rover. For our experiments we used a LAGR[1] robot as it is a more convenient data collection platform (Figure 7). The LAGR robot has two front differential drive wheels and two rear caster wheels. It is equipped with a pair of stereo cameras, wheel encoders, IMU, and GPS (the IMU and GPS are postprocessed into a 'global pose'). It is about 1m tall, 0.75m wide and 1m long.

2) Slip measurements: Slip is defined as the difference between the commanded and the actual velocities between two consecutive steps of the rover. The commanded velocity is computed by the rover's kinematics model, differential drive in this case. The actual velocity is estimated by VO. We also used VO to get the ground truth rover position. The terrain slopes are estimated from range data produced by the stereo system. Additionally a tilt sensor is needed to retrieve gravity leveled slopes. In the case of LAGR we use the provided *global pose* based on the IMU. The appearance information from imagery, the slope from range data, and the tilt of the robot are the necessary inputs for slip prediction. Note that they are onboard sensors, so a remote slip prediction can be performed after the slip behavior has been learned [1].

In these experiments we focused on slip in X (along the forward motion direction) as dependent on the longitudinal slope. In general, rover slip depends on other inputs, such as the lateral slope or terrain roughness. For simplicity, we have selected the longitudinal slope and the measured slip to act as a label for learning the terrain classification during training. After the robot has learned how to visually discriminate the terrains it is conceivable to learn a more complex slip model using more input variables, as in [1].

B. Visual feature representation

The visual part of the data is composed of terrain patches, corresponding to 0.4x0.4 m map cells (i.e. $\sim$120x80 pixel image patches). As visual appearance changes with range, we have collected only the patches which are observed at a particular range (here, 1-2m range). We use a visual representation based on the frequency of occurrence (i.e. a histogram) of visual features, called textons, within a patch [24]. In this case, 90 textons are selected from the data, constructing a 90-dimensional feature vector. This representation, based on both color and texture, has been shown to achieve satisfactory results for classifications of natural terrains [1]

1) Dimensionality reduction: As the proposed representation is very high dimensional, we use a nonlinear dimensionality reduction technique, Isomap in particular [23], to automatically select a smaller number of suitable dimensions to represent the data. Nonlinear dimensionality reduction techniques have been successfully applied to find appropriate patterns in unsupervised fashion for visual or robot sensor data [9], [10]. With a dimensionality reduction tool at hand [23] we are able to apply the proposed in Section II method to more complex visual representations, as the texton-based one [24].

Figure 7 shows the projected by Isomap points in two dimensions (with their ground truth labels) so we can see that a relatively good separation between the classes could be achieved [2]. For comparison we also show an alternative

[1]LAGR stands for Learning Applied to Ground Robotics and is a program funded by DARPA.

[2]In our implementation we use two dimensional projections. An optimal number of dimensions could be selected automatically by Isomap. It might be > 2 and might provide even better separability of the data than shown.

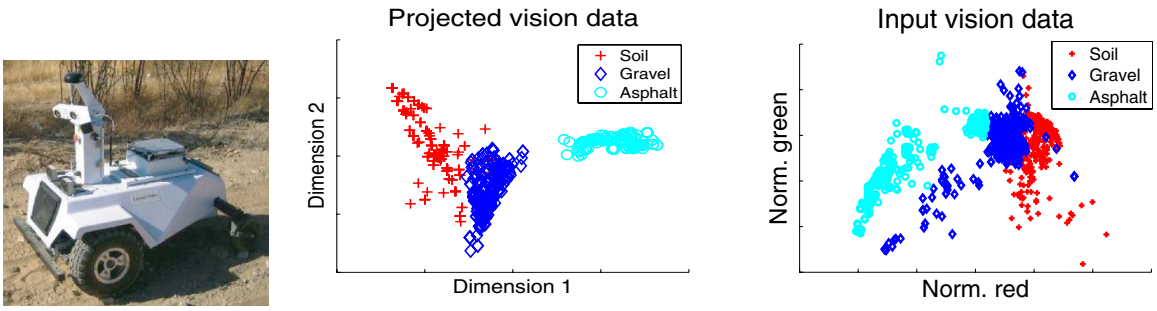

Fig. 7. Experimental setup for the field data test. The autonomous LAGR robot used for data collection (left). The vision data projected by the Isomap algorithm into 2D space from an input 90-dimensional texton representation (middle) and the vision data in 2D normalized color space (right). A better separation of the data is observed when using an enhanced visual space and dimensionality reduction. The measured slip data plotted as a function of the estimated slope angles for the corresponding terrains is shown in Figure 2. The color coding corresponds to human labeled ground truth which is not used by the proposed algorithm.

representation of the same data by relatively simple features in 2D - the average normalized Red and Green in a patch, in which classification, especially without supervision, appears to be more problematic. In the next section, we will show quantitative results for slip prediction, comparing the performance when using both visual representations.

C. Algorithm

The algorithm could be summarized as follows: 1. Build a visual representation using the texton based approach [24]. 2. Do dimensionality reduction using Isomap [23]. 3. Apply learning with automatic supervision (Section II) to discriminate terrains in the reduced dimensionality space and to learn slip models. 4. Predict slip based on the learned visual classes and slip models.

D. Results

The experimental setup is similar to the one in Section II-D, with the difference that the test is done on real field-test data. A set of appearance patches are collected along the traverse together with their corresponding slope and slip measurements. It is not known to the algorithm which terrain classes the input examples belong to; the slip vs slope measurements (Figure 2) will be the only information to be used as automatic supervision. To reflect the monotonic nature of slip, an additional constraint ($\theta_j \geq 0$) is imposed (thus rendering a suboptimal solution). We have about 900 examples which are split randomly into equal training and test sets. As we do not know the correct slip models, the ultimate test of performance is by comparing the predicted slip to the actual measured slip on a test set (not used in training).

The average test errors for 50 runs for learning without supervision, with automatic supervision and with human supervision are shown in Table II. For comparison we also show the results when using the simple normalized Red and Green color space. As seen, learning with automatic supervision outperforms the purely unsupervised learning and closes the gap to the learning with human supervision. More precisely, learning with automatic supervision achieves about 70% of the possible margin for improvement when using the normalized color space and about 82% improvement when using the texton based representation with Isomap. That is, using an enhanced feature representation helps decrease even further the average slip prediction error. This is because the dimensionality reduction technique finds automatically the dimensions which best separate the data. The learned nonlinear models and the corresponding test errors for the three terrain classes for one of the runs are given in Figure 8. We can see that the unsupervised learning could not learn the correct models well because of classification errors in the vision space. One should also note the large slip error even when training on manually labeled terrain types. This is because the field-test data is very noisy.

TABLE II

FIELD EXPERIMENT. AVERAGE SLIP PREDICTION TEST ERROR (%).

Learning scenario	Texton feat.+Isomap	R,G color space
Unsupervised	19.1	18.1
Autom. supervision	11.2	12.2
Human supervision	9.5	9.7

In this experiment we see again that learning with automatic supervision outperforms the unsupervised learning and is close to learning with human supervision. In summary, learning with automatic supervision has the potential to substitute the expensive, tedious and inefficient human labeling in applications related to autonomous navigation.

IV. CONCLUSIONS AND FUTURE WORK

We have proposed a method for learning terrain classification and the nonlinear mechanical behavior on each terrain by using automatic (noisy and uncertain) mechanical supervision which comes from the onboard sensors of an autonomous vehicle. The development of this framework is motivated by the problem of autonomous navigation without human supervision. An important outcome of the algorithm is that the expected mechanical behavior can be predicted from only visual or other onboard sensors and that the learning is done completely automatically. We have shown experiments on a dataset, collected while driving in the field,

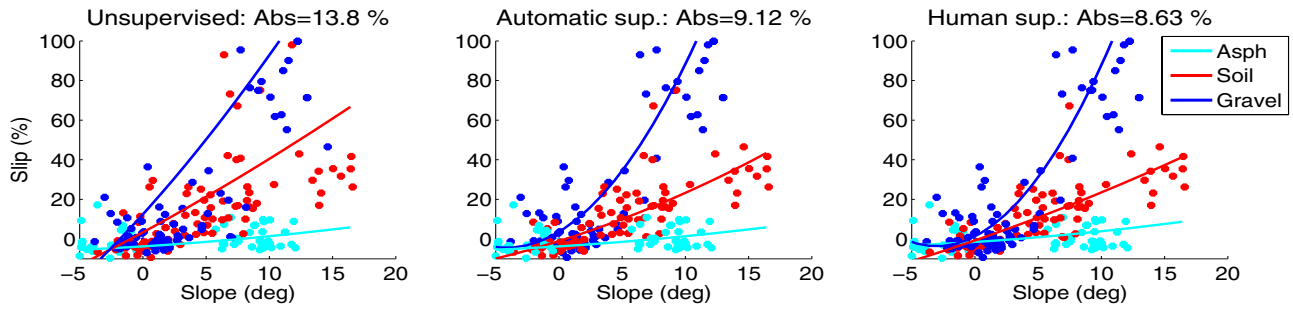

Fig. 8. Field data test results for one of the runs. The learned nonlinear slip models superimposed on the test data when learning without supervision (i.e. unsupervised) (left), when learning with automatic supervision (middle), and when human labeling is used (right). The test errors are given atop each plot.

in which different terrain types are learned better from both vision and slip behavior supervision, than with vision alone. The impact of the proposed method is that it can enable the rover to drive safely on slopes, learning autonomously about different terrains and its mobility limitations on them.

Our future work is targeted towards applying the algorithm to scenarios relevant to the current MER or the future Mars Science Laboratory (MSL) rover missions. The VO algorithm, as a mechanism for measuring slip, is readily available onboard both rovers. However, the use of only grayscale imagery for navigation poses significant challenges and more complex texture based visual representations of the terrain might be needed. After the robot has learned how to visually discriminate the terrains, it is conceivable to model slip as a function of both longitudinal and lateral slopes [1], which will enable more accurate slip prediction. Additionally, learning of lateral or Yaw slip [2] will be very useful while driving on transverse slopes or turning in place.

V. ACKNOWLEDGMENTS

This research was carried out by the Jet Propulsion Laboratory, California Institute of Technology with funding from NASA's Mars Technology Program. Thanks also to the JPL LAGR team for giving us access to the LAGR vehicle and to Nick Hudson for making us aware of reference [13].

REFERENCES

[1] A. Angelova, L. Matthies, D. Helmick, and P. Perona. Slip prediction using visual information. *Robotics: Science and Systems Conference*, 2006.

[2] A. Angelova, L. Matthies, D. Helmick, G. Sibley, and P. Perona. Learning to predict slip for ground robots. *International Conference on Robotics and Automation*, 2006.

[3] J. Biesiadecki, E. Baumgartner, R. Bonitz, B. Cooper, F. Hartman, P. Leger, M. Maimone, S. Maxwell, A. Trebi-Ollennu, E. Tunstel, and J. Wright. Mars Exploration Rover surface operations: Driving Opportunity at Meridiani planum. *IEEE Conference on Systems, Man, and Cybernetics*, October 2005.

[4] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. *Conf. on Computational Learning Theory*, 1998.

[5] C. Brooks, K. Iagnemma, and S. Dubowsky. Vibration-based terrain analysis for mobile robots. *International Conference on Robotics and Automation*, 2005.

[6] H. Dahlkamp, A. Kaehler, D. Stavens, S. Thrun, and G. Bradski. Self-supervised monocular road detection in desert terrain. *Robotics: Science and Systems Conference*, 2006.

[7] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–37, 1977.

[8] E. DuPont, R. Roberts, C. Moore, M. Selekwa, and E. Collins. Online terrain classification for mobile robots. *International Mechanical Engineering Congress and Exposition Conference*, 2005.

[9] D. Grollman, O. Jenkins, and F. Wood. Discovering natural kinds of robot sensory experiences in unstructured environments. *Journal of field robotics*, 2006.

[10] G. Grudic and J. Mulligan. Topological mapping with multiple visual manifolds. *Robotics: Science and Systems Conference*, 2005.

[11] M. Happold, M. Ollis, and N. Johnson. Enhancing supervised terrain classification with predictive unsupervised learning. *Robotics: Science and Systems Conference*, 2006.

[12] M. Jordan and R. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214, 1994.

[13] A. Julosky, S. Weiland, and W. Heemels. A Bayesian approach to identification of hybrid systems. *IEEE Trans. on Automatic Control*, 50(10):1520–1533, 2005.

[14] D. Kim, J. Sun, S. Oh, J. Rehg, and A. Bobick. Traversability classification using unsupervised on-line visual learning for outdoor robot navigation. *International Conference on Robotics and Automation*, 2006.

[15] Y. LeCun, U. Muller, J. Ben, E. Cosatto, and B. Flepp. Off-road obstacle avoidance through end-to-end learning. *Advances in Neural Information Processing Systems*, 2005.

[16] C. Leger, A. Trebi-Ollennu, J. Wright, S. Maxwell, R. Bonitz, J. Biesiadecki, F. Hartman, B. Cooper, E. Baumgartner, and M. Maimone. Mars Exploration Rover surface operations: Driving Spirit at Gusev crater. *IEEE Conference on Systems, Man, and Cybernetics*, October 2005.

[17] D. Lieb, A. Lookingbill, and S. Thrun. Adaptive road following using self-supervised learning and reverse optical flow. *Robotics: Science and Systems Conference*, 2005.

[18] R. Lindemann and C. Voorhees. Mars Exploration Rover mobility assembly design, test and performance. *IEEE International Conference on Systems, Man and Cybernetics*, 2005.

[19] L. Matthies and S. Schafer. Error modeling in stereo navigation. *IEEE Journal of Robotics and Automation*, RA-3(3):239–250, June 1987.

[20] L. Matthies, M. Turmon, A. Howard, A. Angelova, B. Tang, and E. Mjolsness. Learning for autonomous navigation: Extrapolating from underfoot to the far field. *NIPS, Workshop on Machine Learning Based Robotics in Unstructured Environments*, 2005.

[21] G. Seber and C. Wild. *Nonlinear Regression*. John Wiley & Sons, New York, 1989.

[22] B. Sofman, E. Lin, J. Bagnell, N. Vandapel, and A. Stentz. Improving robot navigation through self-supervised online learning. *Robotics: Science and Systems Conference*, 2006.

[23] J. Tenenbaum, V. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, December 2000.

[24] M. Varma and A. Zisserman. Texture classification: Are filter banks necessary? *IEEE Conf. on Computer Vision and Pattern Recognition*, 2003.

[25] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl. Constrained k-means clustering with background knowledge. *International Conference on Machine learning*, pages 577–584, 2001.

[26] C. Wellington and A. Stentz. Online adaptive rough-terrain navigation in vegetation. *International Conference on Robotics and Automation*, 2004.