Slip Prediction Using Visual Information

Anelia Angelova
Computer Science Dept.
California Institute of Technology
Email: anelia@vision.caltech.edu

Larry Matthies, Daniel Helmick
Jet Propulsion Lab (JPL)
California Institute of Technology
lhm, dhelmick@jpl.nasa.gov

Pietro Perona
Electrical Engineering Dept.
California Institute of Technology
perona@vision.caltech.edu

***Abstract*— This paper considers prediction of slip from a distance for wheeled ground robots using visual information as input. Large amounts of slippage which can occur on certain surfaces, such as sandy slopes, will negatively affect rover mobility. Therefore, obtaining information about slip before entering a particular terrain can be very useful for better planning and avoiding terrains with large slip.**

The proposed method is based on learning from experience and consists of terrain type recognition and nonlinear regression modeling. After learning, slip prediction is done remotely using only the visual information as input. The method has been implemented and tested offline on several off-road terrains including: soil, sand, gravel, and woodchips. The slip prediction error is about 20% of the step size.

I. Introduction

Slip is a measure of the lack of progress of a wheeled ground robot while driving. High levels of slip can be observed on certain terrains, which can lead to significant slow down of the vehicle, inability to reach its predefined goals, or, in the worst case, getting stuck without the possibility of recovery. Similar problems were experienced in the Mars Exploration Rover (MER) mission in which one of its rovers got trapped in a sand dune, experiencing a 100% slip (Figure 1). In future missions it will be important to avoid such terrains, which necessitates the capability of slip prediction from a distance, so that adequate planning could be performed. This research is relevant to both Mars rovers and to Earth-based ground robots.

While some effort has been done in mechanical modeling of slip for wheeled ground robots [2], [8], [14], no work, to our best knowledge, has considered predicting slip, or other properties of the vehicle-terrain interaction, remotely. In this paper we use vision information to enable that.

We propose to *learn* a mapping between visual information (i.e. geometry and appearance coming from the stereo imagery) and the measured slip, using the experience from previous traversals. Thus, after learning, the expected slip can be predicted from a distance using only stereo imagery as input. The method consists of: 1) recognizing the terrain type from visual appearance and then, after the terrain type is known, 2) predicting slip from the terrain's geometry. Both components are based on learning. In our previous work we have shown that the dependence of slip on terrain slopes when the terrain type is known (termed 'slip behavior') can be learned and predicted successfully [1]. In this paper we describe the whole system for slip learning and prediction, including the texture recognition and the full slip prediction from stereo imagery.

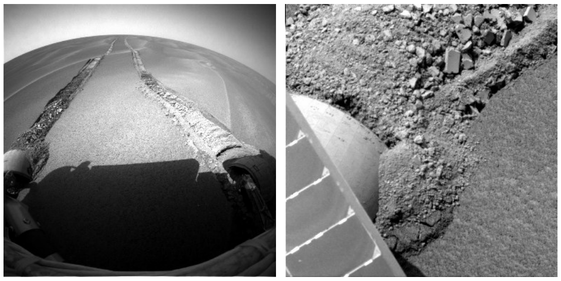

Fig. 1. The Mars Exploration Rover 'Opportunity' trapped in the 'Purgatory' dune on sol 447. A similar 100% slip condition can lead to mission failure.

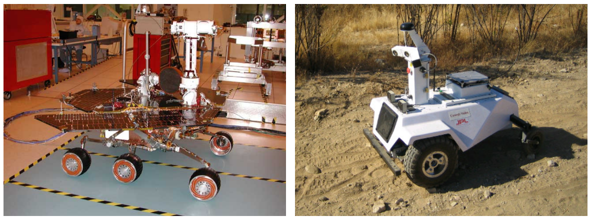

Fig. 2. The Mars Exploration Rover 'Spirit' in the JPL Spacecraft Assembly Facility (left). The LAGR vehicle on off-road terrain (right).

The output of the slip prediction algorithm is intended to be incorporated into a traversability cost to be handed down to an improved path planner which, for example, can consider regions of 100% slip as non-traversable or can give higher cost to regions where more time is needed for traversal due to large slip. Second to tip-over hazards, slip is the most important factor in traversing slopes. Automatic learning and prediction of slip behavior could replace manual measurement of slip, as the one performed by Lindemann et al. [17], which has been used successfully to teleoperate the 'Opportunity' rover out of Eagle Crater. One additional problem which occurred in [17], and which learning could easily solve, is that slip models were available only for angles of attack of 0°, 45°, 90° away from the gradient of the terrain slope [7], [17].

A. Testbed

This research is targeted for planetary rovers, such as MER (Figure 2). For our experiments, however, we used an experimental LAGR[1] testbed (Figure 2), as it is a more convenient

[1] LAGR stands for Learning Applied to Ground Robots

data collection platform. It has two front differential drive wheels and two rear castor 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 can run in autonomous mode or be manually joysticked using a radio controller. The vehicle can achieve speeds of up to 1.2 m/s. It is about 1m tall, 0.75m wide and 1m long.

B. *Definition of slip*

Slip s is defined as the difference between the velocity measured by the wheel (wr) and the actual velocity v: $s = wr - v$, where w is angular wheel velocity and r is the wheel radius. It can also be normalized: $s = \frac{wr-v}{wr}$ (thus the results can be reported in percent). Similarly, the slip for the whole rover is defined as the difference between the actual vehicle velocity and the velocity estimated from the kinematic model for each DOF of the rover per step (i.e. between two consecutive stereo pairs) [10]. We will use interchangeably the change in position and velocity per step, normalizing so that each step takes the same time. A differential drive model is used for the kinematic estimate of the LAGR vehicle. The actual position (ground truth) can be estimated by visually tracking features, a method called Visual Odometry (VO) [19], [20], or measured with some global position estimation device. VO is the preferred method for ground truth estimation because it is a convenient self-contained sensor on the vehicle, so data collection for training could be done automatically and the whole learning could be done online onboard the vehicle, which coincides with the goals of planetary exploration missions. Furthermore, global positioning devices are not always available, especially regarding planetary missions. VO position estimation error is less than 2.5% of the distance traveled for runs of 10-30 meters [1], [10], but stepwise, larger errors could occur [1]. Random effects from the terrain could result in significant additional noise in the slip measurements, see [1] for details.

We have adopted a macro-level (of the whole rover) modeling of slip, in the spirit of [10], [17]. This modeling is justified, as the slip prediction will be used in a first, quick evaluation of terrain traversability. We consider the slip in the previous rover frame (corresponding to the beginning of the step) which is defined as follows: the X coordinate is along the direction of forward motion, Y is along the wheel axis, and Z is pointing down. As the LAGR vehicle has only three kinematically observable DOFs, we define slip in X and slip in Y as the components of slip along the X and the Y axes, respectively. Slip in Yaw is the rotation angle around the Z axis. In this paper, we focus on predicting slip in X. Prediction of also slip in Y and Yaw has been considered in [1].

C. *Previous work*

Mechanical modeling and estimation of slip has been done at various levels of complexity [2], [4], [8], [14], [15], [26] and for various vehicle architectures. These methods are rather complicated and need to be performed at the particular traversed location, as they require detailed knowledge of the wheel-soil interactions. Additional estimation of terrain parameters (such as cohesion and friction angle) [12], [15], and soil behavior modeling [2], [4], [23] are needed. Regarding planetary exploration, in-situ soil parameter estimation has been done for the Mars Sojourner rover [18] and MER [3]. These methods are not predictive either.

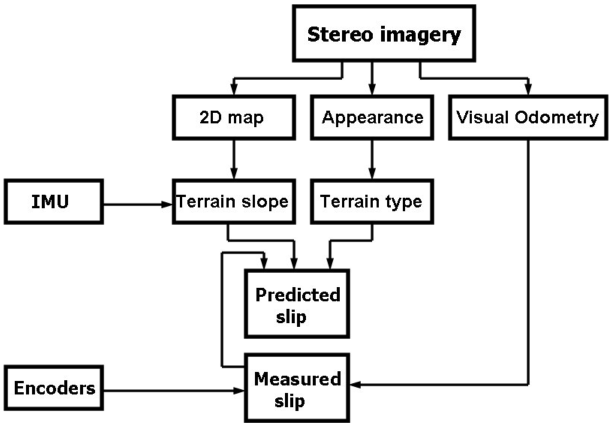

Fig. 3. Slip learning and prediction algorithm framework.

Several authors considered counteracting slip for improving vehicle mobility: e.g. the slip compensation algorithm of Helmick et al. [10], [11], or improving traction control, proposed by Iagnemma et al. [12]. Those methods, again, work at the traversed rover location. The only approach we are aware of which has been used to evaluate slip at a not yet traversed location is the one of Lindemann [17]. However, it is limited to evaluating the behavior of the rover for several isolated terrain slopes, requires manual measurements, and, in general, needs careful selection of the soil type on which the tests are performed to match the target Mars soil.

D. *Outline*

In Section II we propose a general framework for learning of slip from stereo imagery, in which the problem is subdivided into 1) visual recognition of terrain type (Section III) and 2) learning/prediction of slip behavior, once the terrain type is known (Section IV). Final experimental results of slip prediction are given in Section V.

II. Slip learning and prediction

In this section we give a general framework to learn the functional relationship between visual information and the measured slip using training examples.

A. *General framework*

The amount of slip for a particular terrain is a highly nonlinear function of the terrain slopes [17]. This dependence (called earlier *slip behavior*) changes on different terrains, so we can cast the problem into a framework similar to the Mixture of Experts (MoE) framework [13] in which the input space is partitioned into subregions (corresponding to different terrain types) and then different functions (corresponding to different slip behaviors), are learned for each subregion. Our inputs will be terrain geometry (G) (captured by slopes) and

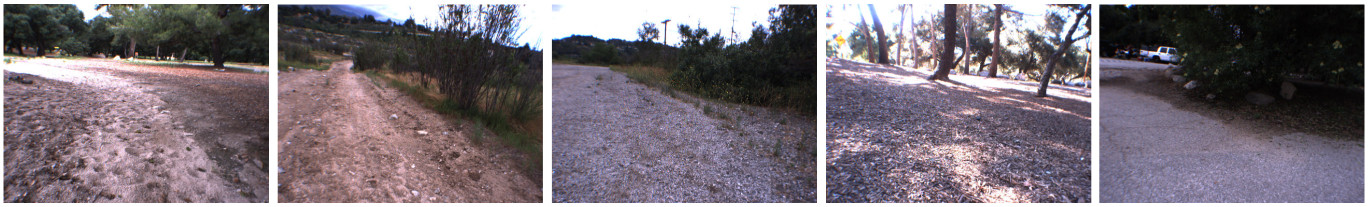

Fig. 4. Example images from some of the terrains collected by the LAGR vehicle: sand, soil, gravel, woodchips, asphalt.

terrain type (described by its appearance (A), such as texture and color); the output will be the slip in X (the slip along the direction of forward motion).

More formally, let I be all the information available from stereo pair images, $I = (A, G)$. Let $f(S|I)$ be the regression function of slip S on the input variables A, G. Now considering that we have several options for a terrain type T, each one occurring with probability $P(T|A, G)$ (given the information from the image in question A, G), we can write $f(S|I)$ as follows:

$$f(S|I) = f(S|A, G) = \sum_T P(T|A, G) f(S|T, A, G),$$

where $\sum_T P(T|A, G) = 1$. This type of modeling admits one exclusive terrain type to be selected per image, or a soft partitioning of the space, which allows for uncertainty in terrain classification. We assume that the terrain type is independent of terrain geometry $P(T|A, G) = P(T|A)$ and that, given the terrain type, slip is independent of appearance $f(S|T, A, G) = f(S|T, G)$. Assuming independence of appearance and geometry is quite reasonable because, for example, a sandy terrain in front of the rover, would appear approximately the same, no matter if the rover is traversing a level or tilted surface. So we get:

$$f(S|I) = \sum_T P(T|A) f(S|T, G).$$

In summary, we divide the slip learning problem into a terrain recognition part ($P(T|A)$, i.e. the probability of a terrain type, given some appearance information) and a slip prediction part ($f(S|T, G)$, i.e. the dependence of slip on terrain geometry, given a fixed terrain type T). The mixing coefficients $P(T|A)$ will be learned and predicted by a terrain texture classifier (Section III). The regression functions $f(S|T, G)$ for different terrain types will be learned and predicted by a nonlinear regression method (Section IV).

B. Architecture

In this section we briefly describe the architecture of our system, summarized in Figure 3. We will be using the stereo imagery as input, as well as the IMU of the vehicle and its wheel encoders (the latter is needed only for training). Stereo imagery is used to create a 2D cell map of the environment from its range data. It also provides appearance information which can be associated to certain locations (cells) in the map. The 2D map can give us geometry information about the terrain G and, as we are interested in terrain slopes with respect to gravity, we use the vehicle's IMU to retrieve an initial gravity leveled pose (in fact a *global pose*, which is the postprocessed IMU and GPS signals, is used instead). The appearance information A will be used to decide which terrain type corresponds to a cell or a neighborhood of cells. This is all the necessary information to perform slip prediction with our algorithm.

In order to learn slip we have added a slip feedback. The mechanism to measure slip is as follows. The actual motion (relative position) between two frames is estimated by Visual Odometry [19] which only needs two consecutive stereo pairs as input. The motion which the vehicle thinks it has performed is given by the vehicle's forward kinematics. As the LAGR vehicle has a differential drive model the wheel encoders are sufficient to compute its full kinematics. A more complex kinematic model is needed for a MER type of vehicle [22], but it is well understood how to compute it [10], [22]. Differencing the actual motion and the motion estimated by the kinematic model gives a measurement of slip for a particular step [10]. This feedback is used for collecting training examples to learn slip from stereo imagery.

The slip prediction coming from appearance and geometry information is based on frames which observe a particular location from a distance, i.e. those stereo frames will come much earlier in time than the frames which measure the slip feedback (using VO). Both types of information come from a single stereo imagery sensor, as shown in Figure 3. The advantage of such a system is that it can sense the terrain remotely and that it needs only passive, cheap and self-contained sensors on the vehicle, such as stereo vision. Now, the main challenge is understanding this visual information.

C. Dataset

For our slip prediction experiments we have collected datasets on off-road terrains. There are five major terrain types which the rover has traversed: soil, sand, gravel, asphalt and woodchips (Figure 4). In addition to that, there are several other terrain types which appear in the sequences, such as green or dry grass. We considered those as one additional 'grass' class in the terrain classification. Although we have good variability in the terrain relief in our dataset (level, upslope and down-slope areas on soil, asphalt and woodchip terrains, transverse slope on gravelly terrain, flat sandy terrain, etc.), not all possible slip behaviors could be observed in the area of data collection. For example: there was no sloped

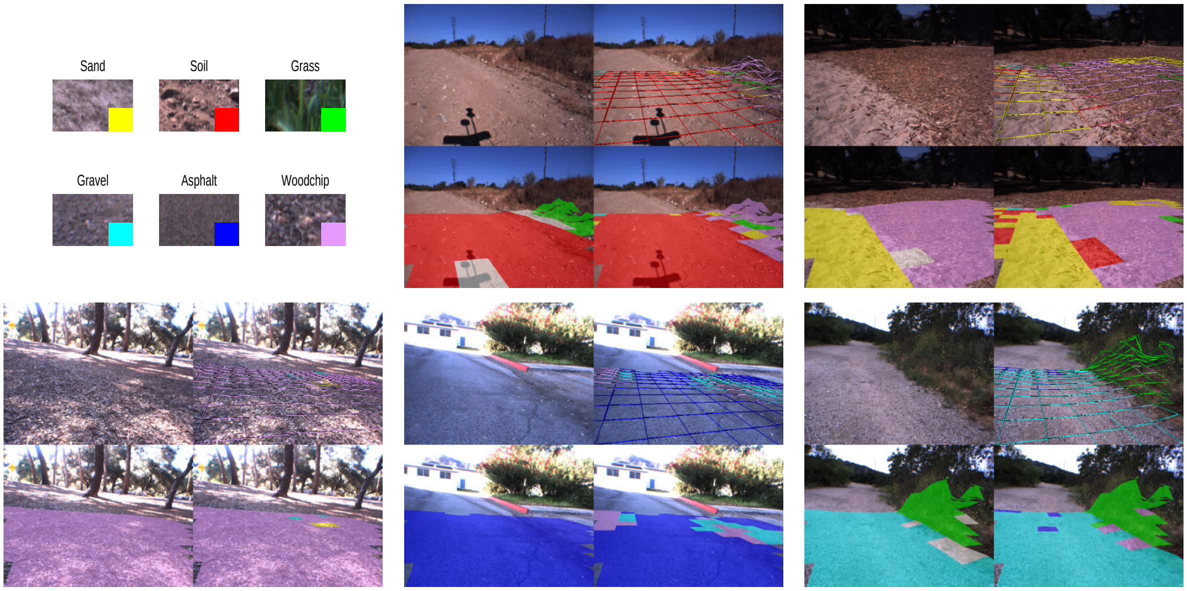

Fig. 6. Example texture classification results from each of the datasets. Patches from the six terrain types considered in the texture classification and the corresponding color coding assigned are shown at top left. Each composite image contains the original image (top left), the ground truth terrain classification (bottom left) and the results of the terrain classification algorithm represented in two different ways (top right and bottom right). Ambiguous terrain type in the ground truth is marked with white (those regions are not required to be classified correctly).

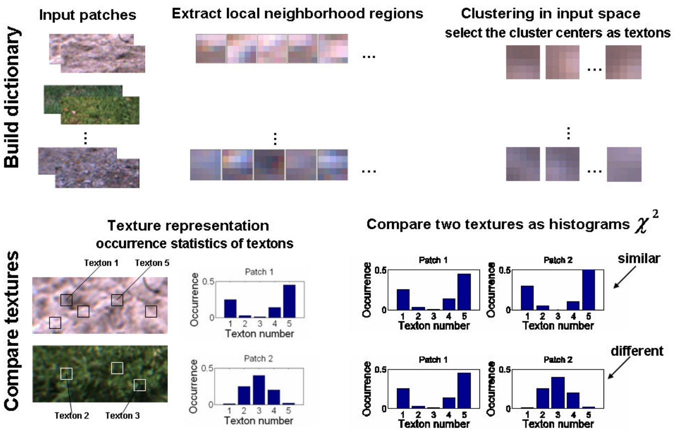
Build dictionary
Input patches
Extract local neighborhood regions
Clustering in input space
select the cluster centers as textons
Compare textures
Texture representation
occurrence statistics of textons
Texton 1
Texton 5
Texton 2
Texton 3
Patch 1
Patch 2
Occurrence
Texton number
Compare two textures as histograms χ^2
similar
different

Fig. 5. Schematic of the terrain classification algorithm [16], [24].

terrain covered with sand, besides, the rover showed poor mobility on flat sand (about 80% slip [1]); the gravelly terrain available was only possible to be traversed sideways for safety reasons; there was no transverse slope for the soil dataset.

III. Terrain Classification

This section describes the terrain classification using vision information, which is the first step of our algorithm. For the purposes of slip prediction, we consider only the part of the image plane which corresponds to the robot's 2D map of the environment. That is, for now, we are not interested in regions beyond the distance where range data is available, because we simply cannot retrieve any reliable slope information and therefore cannot predict slip. A reasonable map for the LAGR vehicle is of size 12x12m or 15x15m, centered on the robot. Note that the MER panoramic camera has a considerably higher resolution and look-ahead [5]. Our goal is to determine the terrain type in each cell of the map (the cell is of size 0.4m). In fact, we will be classifying the patches corresponding to the back-projections of map cells to the image plane.

Note that the patches at close range and at far range have considerably different appearances, so a single texture based classifier could not be used for both. This is due to the fact that the spatial resolution decreases rapidly with range. This could also be clarified by looking at the amount of information in the image plane which corresponds to different areas in the 2D map. For the LAGR vehicle the estimates are: about 70% of the image plane is mapped to ranges below 10m, about 7% - to ranges between 10m and 50m, and about 2% - to ranges between 50m and the horizon [27]. So, for our experiments we build five independent classifiers which are active at different ranges (ranges up to 2m, 2-3m, 3-4m, 4-5m, and 6m and above).

A. *Terrain classification algorithm*

As we are interested in classifying patches, the approach we use considers the common occurrence of texture elements (called 'textons') in some region (a patch). This representation is more appropriate because a texture is defined not by a single pixel neighborhood, but rather by the co-occurrence of visual patterns in larger areas. The idea follows the texton-based texture recognition methods proposed by Leung et al. [16] and Varma et al. [24]. The approach is summarized in Figure 5.

Five different texture classifiers are trained, each one specialized at different range. For each classifier and for each terrain type class (we have six terrain classes), a set of patches in the image plane (corresponding to the map cells at the

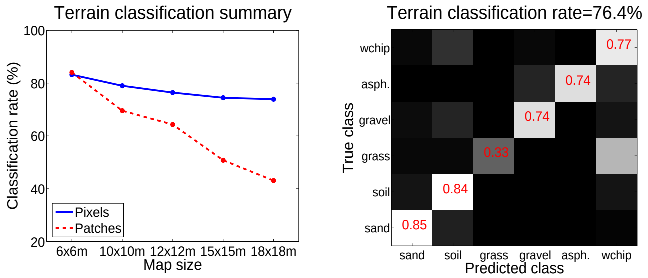

Fig. 7. Terrain classification results for different map sizes (left). Different ways of representing the classification rate by counting correctly classified patches or pixels are shown. Confusion matrix for the 12x12m map (right). The classification rate for each class is displayed on the diagonal.

appropriate ranges) are collected. All the training patches belonging to some range are processed by extracting a set of 5x5 RGB regions forming a 75 dimensional vector representation of a local pixel neighborhood. Those vectors are clustered with k-means and the cluster centers are defined to be the textons for this class. We extracted k=30 textons per class. As a result, a total of 180 textons (called 'texton dictionary') are collected for the whole training set. Working in a feature space composed of local neighborhoods allows for building statistics of dependencies among neighboring pixels, which is a very viable approach, as shown by Varma et al. [24]. Now that the dictionary for the dataset has been defined, each texture patch is represented as the frequencies of occurrences of each texton within it, i.e. a histogram (instead of searching for each texton within a patch individually, each pixel location of the patch is assigned to the closest in Euclidean distance sense texton). In other words, the patches from the training set are transformed into 180 dimensional vectors (each dimension giving the frequency of occurrence of the corresponding texton in this patch). All vectors are stored in a database to be used later for classification. Similarly, during classification, a query image is transformed into a 180 dimensional vector (i.e. a texton occurrence histogram) and compared to the histogram representations of the examples in the database, using a Nearest Neighbor method and a χ^2-based distance measure [24]. The majority vote of N=7 neighbors is taken as the predicted terrain type class of the query patch. The result of the classifier will be one single class. To determine the terrain type in the region the robot will traverse (Section V) we select the winner-take-all patch class label in the region (a 4x4 cell neighborhood). In both decisions, a probabilistic response, rather than choosing a single class, would be more robust. Addressing more advanced probabilistic inference within a patch and among neighboring patches is a subject of our future work.

B. *Terrain classification results*

In this section we report results of the terrain classification algorithm. As mentioned earlier, we consider the patches in the original color image, which correspond to different cells of the map. Each patch is classified into a particular terrain type and all the pixels which belong to this patch are labeled with the label of the patch (Figure 6). The classification performance is measured as the percent correctly classified pixels in the image plane. This is more meaningful than counting the correctly classified patches, as the patches at far range are composed of very small image area and cannot be expected to be classified correctly with this approach (Figure 7). Moreover, an error in the terrain classification at close range affects much more the slip prediction, so those patches should be given more weight in measuring the classification performance.

Our dataset is composed of five different image sequences which are called soil, sand, gravel, asphalt and woodchip after the prevailing terrain type in them (Figure 4). Each of the following six terrain classes: gravel, soil, sand, asphalt, woodchip and grass, can appear in those sequences. To measure test performance we take a total of ~30 frames in each sequence, which are separated by at least 10 frames within the sequence, so as not to be similar to one another. So, the test set contains about 150 frames which includes about 10^4 patches and 10^7 pixels. The ground truth terrain type in the test set is given by a human operator. Example classification results are shown in Figure 6. Note that most terrain classification errors occur at far range. Summary results of the terrain classifier for the five sequences for different look-ahead maps are given in Figure 7. Naturally, a larger map is preferred, as it allows the robot to see farther, but the patches at this range are very small (with little information content) and therefore much more likely to be misclassified. So, for the benefits of slip prediction, a tradeoff between accuracy of classification and seeing farther should be made. To be concrete, in our further experiments we fix the map size at 12x12m. The confusion matrix for terrain classification for the 12x12m map is shown in Figure 7.

C. *Discussion*

The texton-based method for terrain classification has been selected for its potential to be fast and robust to intra-class variability, often observed in natural terrains. Slip prediction also requires a fine discrimination between visually similar terrains (such as soil, sand, and gravel), which has not been considered in the previous approaches for terrain classification in the context of autonomous navigation [6] (for example, those three terrain types fall into one 'soil/rock' class which needs to be discriminated from 'grass' and 'sky' classes in [6]). Naturally, a normalized color space (instead of the full RGB we are currently using) would make the algorithm robust to certain illumination variations. The approach can be easily extended to consider that, as well.

IV. Learning slip behavior on a fixed terrain

In this section we describe the method for learning to predict slip as a function of terrain geometry when the terrain type is known (i.e. the *slip behavior*). The work in this section has been considered in more detail in a previous contribution of ours [1]; we include it here for completeness.

A. *Learning algorithm*

Slip S (i.e. $f(S|T,G)$) is approximated by a nonlinear function of terrain geometry G, because previous experimental evidence shows that slip behavior is a nonlinear function of terrain slopes [17]. We use a model based on the receptive field regression method [21], [25]. Locally linear functions are learned in small neighborhoods, which gives a good tradeoff between memory based nonlinear regression methods [9] and global function approximation methods, such as Neural Networks. $\hat{S}(\mathbf{x}) = f(S|G = \mathbf{x})$ is estimated as:

$$\hat{S}(\mathbf{x}) = \sum_c^C K(\mathbf{x}, \mathbf{x}_c)(b_0^c + \sum_{i=1}^r b_i^c \langle \mathbf{p}_i^c, \mathbf{x} \rangle),$$

where $\mathbf{x}$ are the 2-dimensional input slopes, $K(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2/\lambda)$ is a smoothing kernel, $\mathbf{x}_c$ is a training example which serves as a receptive field center, and $\mathbf{p}_i^c$ are several local projections in each receptive field c, r is the number of local projections (here $r = 2$), and λ is a parameter which controls the receptive field size ($\lambda > 0$). In other words, the slip S, corresponding to a query point $\mathbf{x}$, is computed as a linear combination of C linear functions (one per each receptive field), where the weights are computed according to the distance from $\mathbf{x}$ to the centers of the receptive fields.

We briefly describe below how to estimate the unknown parameters of the nonlinear regression. The centers $\mathbf{x}_c$ are allocated wherever needed in the input space as the data arrive in an online fashion [21]. To estimate the parameters b_i^c, $\mathbf{p}_i^c$ in each receptive field, a Partial Least Squares (PLS) linear fit [9] is performed, in which the training points are weighted according to their distance to the receptive field center [25]. In our case of only 2-dimensional inputs, one can also use the Weighted Linear Regression [21] or some other locally linear projection. However, by using PLS, the algorithm can be easily extended to working with higher dimensional inputs, because of the dimensionality reduction capabilities of PLS [25]. The parameter λ is selected using a validation set.

The input for slip prediction, i.e. the terrain geometry G, is represented by the longitudinal and lateral slopes which are the terrain slopes decomposed along the X and Y axes of the current position of the robot, respectively. They are named roll and pitch angles, as they correspond to the vehicle's roll and pitch. The terrain slopes are estimated by performing a linear fit to the average elevation of the neighboring cells [1].

B. Results

In this section we give experimental results of learning and prediction of slip in X from terrain slopes (estimated from visual information) when the traversed terrain type is known. Our dataset is composed of long stereo sequences (1000-2000 frames) which were taken on one terrain type at a time. The sequence is split consecutively: the first part is used for training, the second part for testing. Some small portion of the data (between the training and testing sets) is held out for validation. We report below both training and test error. The training data are used to learn the regression function. After learning, the function is tested on the same data (training error) and also on data not used in training (test error). Naturally the training error will be smaller, but the test error is a criterion for the learning method's *generalization* abilities, i.e. how well it will perform on new, unseen data. To be able to measure the test error, we predict slip only on locations traversed by the rover. Slip prediction error is measured by the RMS, RMS=$\sqrt{\sum_{i=1}^n (P_i - T_i)^2/n}$, where P_i is the predicted and T_i is the target slip at a particular step i.

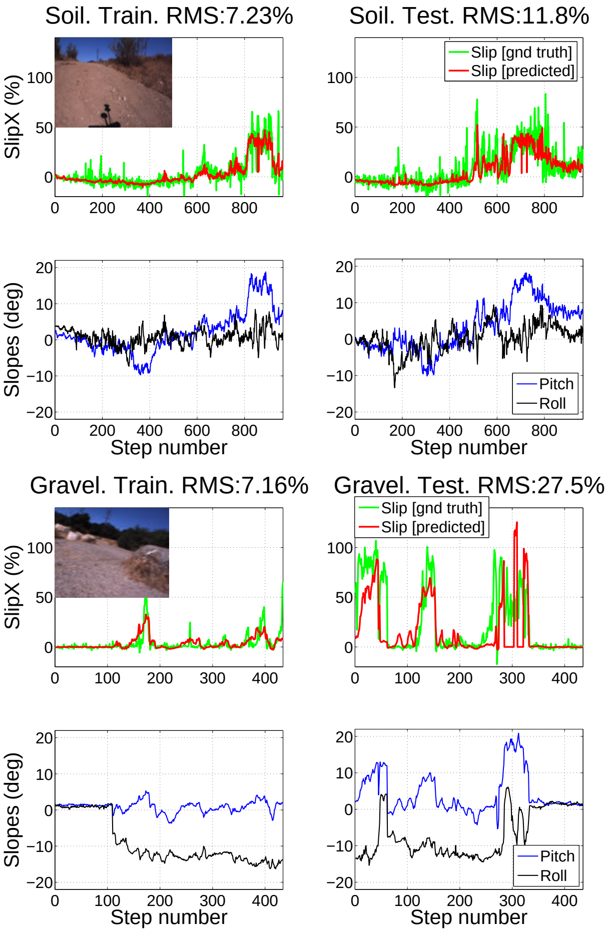

Fig. 8. Example results of prediction of slip in X on soil (top) and on gravel (bottom). Training mode (left), test mode (right). Note the significant noise in the measured slip signal (see [1] for discussion).

Experimental results for slip prediction for the soil and gravel datasets are given in Figure 8. The RMS test error achieved on all the datasets (see [1] for more results) is $\sim 15\%$. The gravel dataset (RMS=27%) is an exception as the training data did not have combinations of roll and pitch slope angles which were seen later in the test set. Still, it could generalize quite well from the available training data (Figure 8).

V. EXPERIMENTAL RESULTS

In the previous sections we saw that terrain type (which entails the intrinsic mechanical properties of the soil) can be predicted from visual features and that slip can be estimated satisfactorily, given the terrain type. This gives us experimental evidence that we can predict slip from visual information only. In this section we perform full slip prediction from stereo imagery for various off-road terrains, as described in Section II, using the already learned terrain type classifier from

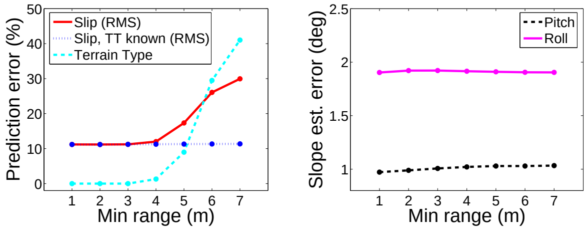

Fig. 9. Slip prediction and terrain type classification errors (left), and slope estimation errors (right), as a function of the minimum range at which prediction is performed. Slip prediction error, if the terrain type were known, is also shown to the left. This experiment is done for a 15x15m map on a subset of the soil terrain dataset.

Section III, and the learned slip behavior predictors for each terrain type from Section IV. The algorithm works as follows: first the terrain type is estimated from appearance information and then the learned slip model for that terrain is activated to produce slip results, given the measured terrain slopes. For these experiments we perform a winner-take-all among the terrain types in neighboring cells i.e. the terrain type with most votes is selected. We present the final quantitative results by comparing the actual measured slip to the predicted slip.

A. Test procedure

We briefly summarize the test procedure. A 2D cell map of the environment is built using range information from the stereo pair images (the cell size is 0.4x0.4m). Each cell keeps information about its extents, average elevation, and a pointer to an image (or images) which have observed this cell. At a remote (future rover) location, we perform a plane fit to the neighborhood of the cell and retrieve the roll and pitch angles of the terrain [1]. We also predict the terrain type by applying the terrain classification algorithm in the neighboring cells. The terrain type which has collected most votes is selected. A 4x4 cell neighborhood is used for both terrain classification and plane fit. All those operations are performed remotely using visual information only. Here, again, to be able to report a test error, we predict slip only on the path which was later traversed by the rover, but, in principle, slip prediction could be done on the whole visible map.

Vehicle localization is very important for this method to succeed and localization is still not a completely solved problem in robotics. VO [19] is used for the vehicle's localization. It is appropriate to be used here, as we are not interested in an accurate absolute global position, but in an accurate relative position within short to mid-size (20m long) sequences (i.e. to be able to map correctly the position of the location seen from a distance to the location traversed later on) and VO is an algorithm which can achieve that [20].

There is one more issue of deciding at what range to start reporting the predicted slip and accumulating information as a particular location is being approached (we call it 'minimum range'). Naturally, a potential path planner would benefit more, the farther we can make a good slip prediction. On the other hand, locations observed at a large distance might give unreliable or noisy slope estimates, or provide very little information for the terrain classifier to be correct. Results of the slip prediction error, as a function of the minimum range at which prediction has started, are given in Figure 9. Note that a much better slip prediction is received for smaller minimum ranges and that the deterioration in slip prediction is mainly due to terrain classification errors occurring at far ranges (the slope angle estimation seems to be much more stable with range for this dataset). The slope angle errors are computed against the roll and pitch angles received from the vehicle's IMU, which are approximations of the actual slope angles. So, for our further experiments we will fix the range at which we start reporting predicted terrain type and slip at 3 meters as a trade-off between a good enough slip prediction and a far enough initial range (preferred from the point of view of the planner). This means that if a location is seen at a closer than 3m range we would not use any information we acquire about it (through vision or other sensors) to improve our slip prediction. Additionally, we need to accumulate information as the rover approaches some location because this location will be seen multiple times at ranges farther from the minimum range. We do that by averaging measurements, weighting them by the inverse of the range at which they were obtained. So, all estimations or predictions about slope angles, terrain type, and slip will be accumulated between the ranges of 3m and possibly 8.4m (8.4m is the diagonal distance from the center to the corner of a 12x12m map; in practice, very few cells will occur at ranges larger than 6m).

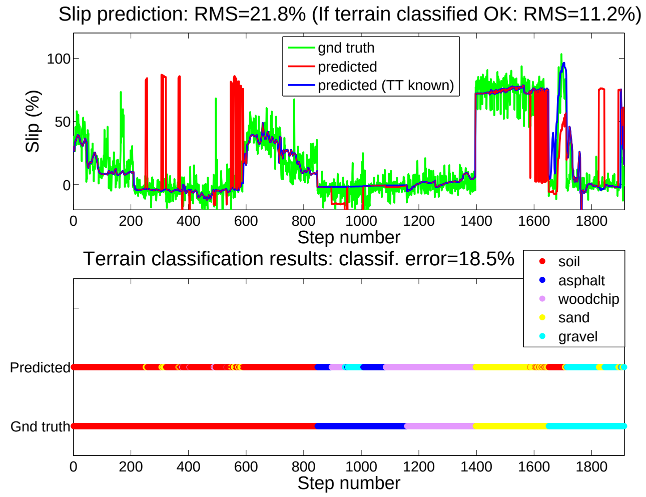

Fig. 10. Results of slip prediction from stereo imagery (terrain geometry and appearance) on the whole dataset. Top: The predicted and measured slip for the corresponding test sequences. Slip prediction, assuming correctly recognized terrain type, is also shown (naturally, it coincides with the final slip prediction, whenever the terrain type is classified correctly). Bottom: The predicted and correct terrain types across the dataset.

B. Results

The test dataset in this section is a composite of sequences of frames in which the terrain type is the same within a sequence but can change to another terrain for the next sequence. In this way a human operator can specify the terrain type of a long image sequence, instead of giving ground truth for each image. The terrain classification algorithm does not have the knowledge that the terrain is continuous for some number of frames and then can abruptly change. The algorithm which estimates the slopes, however is aware of that change because a new frame sequence has to come with a different initial gravity leveled (IMU based) pose. A sequence size varies between 60 and 200 frames and the whole composite dataset contains about 2000 test frames. The terrains are quite challenging, as the dataset is collected in the field. We have made sure that the test dataset has not been used for training.

The results of the full slip prediction experiment for the abovementioned large 'composite' dataset are shown in Figure 10. The figure shows the color coded terrain type classification results, the measured slip, the predicted slip, and the predicted slip if the terrain type were known. The final slip prediction error for the whole dataset is 21.8%. When the terrain type is classified correctly, the slip prediction error is 11.2%. As seen in the figure, large slip errors come from misclassified terrain types (usually soil and gravel are misclassified for sand). In this dataset the error is artificially increased as the slip measured for level sandy terrain is about 80%, which gives a rather large slip error due to terrain misclassification (compare to the error if the terrain type were correct). This result also shows that some errors are more dangerous than others. In other words, that the terrain classification algorithm should be applying different penalties for different types of error, i.e. terrain misclassification which leads to large slip errors should be given larger cost.

VI. Conclusions and future work

In this paper we have proposed to predict slip (a property of mechanical vehicle-terrain interaction) remotely from visual information only. We have achieved very promising results, given the fact that there is a large noise component in the measurement of slip (see [1] for discussion). The predicted slip will be used to enhance a path planning algorithm so that areas of large slip are avoided as potential hazards. The individual components of the slip learning and prediction framework could have independent applications. Learning of slippage for a fixed terrain type could be used instead of tedious manual slip measurements [17]. Predicting terrain type from visual information could be used to supply information *remotely* about soil parameters which are needed by advanced mechanical terrain and rover-terrain interaction modeling [12].

For the time being, we do offline training of both the terrain type classifier and the slip behavior predictors for each terrain type, but our future work is targeted at slip learning in an online fashion, which has influenced the selection of the algorithms and methods in this work.

Further efforts are needed to develop a better terrain classification algorithm, to avoid erroneous slip prediction due to terrain type classification errors. Visual information might not be sufficient to distinguish various terrain types and properties, especially considering Mars terrains. It can be complemented with multispectral imaging or other sensors to resolve some inherent visual ambiguities and improve on the classification results. A more advanced algorithm to consider spatial continuity of terrain classification over neighboring patches or dependent on terrain geometry also needs to be investigated.

Acknowledgment

This research was carried out by the JPL, California Institute of Technology, under a contract with NASA, with funding from the Mars Technology Program. Thanks also to the JPL LAGR team for giving us access to the vehicle and to the reviewers of the paper for many useful comments.

References

[1] Angelova, A., Matthies, L., Helmick, D., Sibley, G., Perona, P., Learning to predict slip for ground robots, *ICRA*, 2006

[2] Andrade, C., Ben Amar, F., Bidaud, P., Chatila, R., Modeling robot-soil interaction for planetary rover motion control', *IROS*, 1998

[3] Arvidson, R., et. al., Localization and physical properties experiments conducted by Opportunity at the Meridiani Planum, *Science*, vol. 306, No. 5685, 2004

[4] Bekker, M., *Introduction to Terrain-vehicle Systems*, Univ. of Michigan Press, 1969

[5] Bell, J., et al., Mars Exploration Rover Athena Panoramic Camera investigation, *J. of Geophysical Research*, vol. 108, No E12, 8063, 2003

[6] Bellutta, P., Manduchi, R., Matthies, L., Owens, K., Rankin, A., Terrain perception for DEMO III, *IEEE Intelligent Vehicles Symp.*, 2000

[7] Cheng, Y., Maimone, M., Matthies, L., Visual odometry on the Mars Exploration Rovers, *Int. Conf. on Systems, Man and Cybernetics*, 2005

[8] Farritor, S., Hacot, H., Dubowsky, S., Physics based planning for planetary exploration, *ICRA*, 1998

[9] Hastie, T., Tibshirani, R., Friedman, J., *The Elements of Statistical Learning*, Springer, 2001

[10] Helmick, D., Cheng, Y., Roumeliotis, S., Clouse, D., Matthies, L., Path following using visual odometry for a Mars rover in high-slip environments, *IEEE Aerospace Conference*, Big Sky, Montana, 2004

[11] Helmick, D., Roumeliotis, S., Cheng, Y., Clouse, D., Bajracharya, M., Matthies, L., Slip compensation for a Mars rover, *IROS*, 2005

[12] Iagnemma, K., Shibly, H., Rzepniewski, A., Dubowsky, S., Planning and control algorithms for enhanced rough-terrain rover mobility, *Int. Symp. on Artificial Intelligence, Robotics, and Automation in Space*, 2001

[13] Jacobs, R., Jordan, M., Nowlan, S., Hinton, G., Adaptive mixtures of local experts, *Neural Computation*, **3**, 79-87, 1991

[14] Kraus, P., Fredriksson, A., Kumar, V., Modeling of frictional contacts for dynamic simulation, *IROS*, 1997

[15] Le, A., Rye, D., Durrant-Whyte, H., Estimation of track-soil interactions for autonomous tracked vehicles, *ICRA*, 1997

[16] Leung, T., Malik, J., Representing and recognizing the visual appearance of materials using three-dimensional textons, *IJCV*, 43 (1), 2001

[17] Lindemann, R., Voorhees, C., Mars Exploration Rover mobility assembly design, test and performance, *Int. Conf. on Systems, Man and Cybernetics*, 2005

[18] Matijevic, J., et al., Characterization of Martian surface deposits by the Mars Pathfinder rover, Sojourner, *Science*, Vol. 278, 5, 1997

[19] Matthies, L., Schafer, S., Error modeling in stereo navigation, *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 3, June, 1987

[20] Olson, C., Matthies, L., Shoppers, M., Maimone, M., 'Stereo ego-motion improvements for robust rover navigation', *ICRA*, 2001

[21] Schaal, S., Atkeson, C., Constructive incremental learning from only local information, *Neural Computation*, 10, **8**, 2047-2084, 1998

[22] Tarokh, M., MacDermott, G., Hayati, S., Hung, J., Kinematic modeling of a high mobility Mars rover, *ICRA*, 1999

[23] Terzaghi, K., *Soil Mechanics in Engineering Practice*, John Wiley & Sons, New York, 1948

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

[25] Vijayakumar, S., D'Souza, A., Schaal, S., Incremental online learning in high dimensions, *Neural Computation*, **12**, 2602-2634, 2005

[26] Wong, J., *Theory of Ground Vehicles*, John Wiley & Sons Inc., 1993

[27] The JPL LAGR team, *private communication*