EXAMINATION OF SURFACE FEATURE ANALYSIS AND TERRAIN TRAVERSABILITY FOR A WALL-CLIMBING ROBOT

D. SCHMIDT*, M. JUNG and K. BERNS

Robotics Research Lab, University of Kaiserslautern
D-67663 Kaiserslautern, Germany
**E-mail: dschmidt@cs.uni-kl.de*
http://agrosy.cs.uni-kl.de/cromsci/

Terrain traversability is a common problem in outdoor robotics. The main task is to determine if the current or upcoming terrain can be overcome by the locomotion system of the robot. But, this problem increases in terms of climbing robots which do not only have a locomotion, but also an adhesion system which has to be considered. This paper addresses these aspects and presents an approach to identify and analyze important surface features and to estimate the system's behavior. Supervised learning techniques are used to classify surfaces into different categories depending on their traversability. As test-platform, a wall-climbing robot using negative pressure adhesion and an omnidirectional drive system is considered which is able to drive on flat concrete buildings.

Keywords: climbing robot, surface features, feature extraction, supervised learning, support vector machine

1. Introduction

Whether a terrain is traversable by a robot or not generally depends on two aspects: The condition or characteristics of the surface and the robot's ability to overcome this surface with its locomotion system. Commonly, the terrain is either classified into different categories like concrete, gravel or sand[1] or it is analyzed based on aspects like inclination or roughness.[2,3] Basis of this is the environmental perception via laser range sensors, cameras or other sensors. The goal is to adapt vehicle motion to this information and e. g. slow the robot down or evade specific regions. In theory, it is also possible to determine the needed amount of sensor accuracy based on vehicle settings and to guarantee operability of the system depending on its parameters and the expected environmental features.[4]

In the field of climbing robots, the traversability additionally relies on the adhesion system and its ability to handle different terrains. In the present case, the climbing robot CROMSCI (figure 1) using an omnidirectional drive system in combination with negative pressure adhesion has been considered.[5] This robot has been developed to inspect concrete buildings like bridges or dams area-wide and semi-autonomously. Although the negative pressure system is highly sophisticated with optimized control components and seven adhesion chambers, it may still fail on certain surfaces.

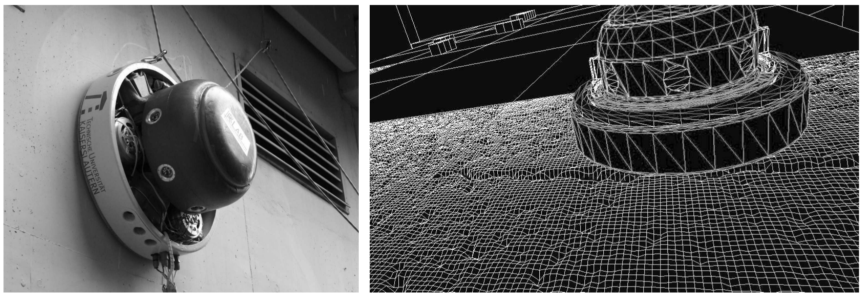

Fig. 1. Real prototype (left) and wireframe view of simulated environment (right).

This paper presents ways to analyze the upcoming terrain by a coupling of surface parameters and reactions of the system by using techniques of supervised learning. Figure 2 illustrates the concept and the internal components which are used for feature extraction and traversability estimation. The offline training has to be executed beforehand based on training examples which include a depth image and a rating of the current robot adhesion to determine the influence of the surface on the adhesion system. The goal is to predict the reaction of the adhesion system on the upcoming terrain and to determine, if it can be overcome by the robot or not.

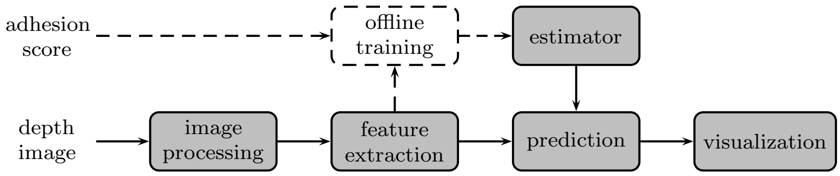

Fig. 2. Components of the traversability estimation including offline elements (dashed).

Next section 2 presents surface parameters and steps of feature extraction. Section 3 contains the surface analysis itself as well as approaches of supervised learning. A comparison of these methods is given in section 4, section 5 sums up the findings of this work.

2. Feature Extraction and Surface Parameters

The feature extraction itself depends on the devices for environmental perception. Here, a stereo camera system is considered which will be placed at a height of about 50 cm at the robot chassis with a pitching angle of 35°. This sensor setup has been created in simulation (figure 3) due to two reasons: At first, it is not possible to use the current version of CROMSCI for sufficient training examples since its hardware is still not reliable enough for long-term applications. Additionally, it was not clear if it is possible at all to acquire useful information out of 3D surface data for traversability information and what image resolution and accuracy would be needed.

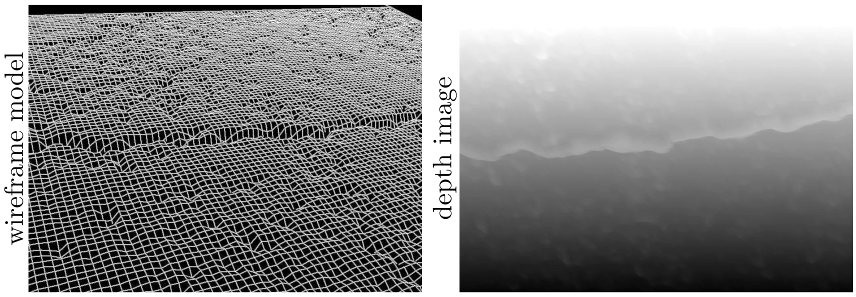

Fig. 3. Basic surface mesh (left) and acquired depth image (right) for feature extraction.

The *image processing* part of figure 2 consists of the height calculation of each pixel based in the acquired depth image. This step also includes a linear interpolation for areas which are e. g. hidden behind superstructures and a perspective transformation into a top-view of the perceived terrain.

For an analytic description of the 3D surface (figure 3) ten different parameters according to upcoming standard ISO 25178-2[6] are calculated: Statistical parameters like *arithmetic mean height* S_a and *root mean square height* S_q describe significant height deviations like cracks or holes, the *skewness* S_{sk} determines the symmetry of the height distribution, and *kurtosis* S_{ku} indicates the presence of either peak or valley defects. The topographical properties are defined by *root mean square gradient* S_{dq} and *developed area ratio* S_{dr} to differentiate surfaces with a similar average roughness. The volume is analyzed via *peak material volume* V_{mp} and *core material volume* V_{mc} indicating the clinging capability, the *core void volume* V_{vc} provides information about the resulting void volume, and the *dale void volume* V_{vv} indicates the remaining void volume after clinging. Additionally, three other characteristics are taken into account: S_v and S_p indicate the *deepest valley* and *highest peak* respectively, S_z is the *maximum height.*

3. Surface Analysis and Supervised Learning

Since it is not useful to calculate the surface parameters for the complete area they are determined for ring masks corresponding to the outer circular sealing area (figure 4). This mask has been chosen because the outer sealing is the most important connection between the adhesion system of the robot and the ground. The idea is to calculate the surface parameters of these masks beforehand and to measure the reaction of the adhesion system when the robot reaches this area. Of course, it is also possible to apply other shapes like rectangles, pie slices (representing chamber areas) or spokes (sealing areas). The main challenge is to find a meaningful rating of the analyzed area to get a suitable rating of its characteristics. Experiments have shown that the circular shape in combination with a so-called adhesion score delivers the most significant information, so this is considered here.

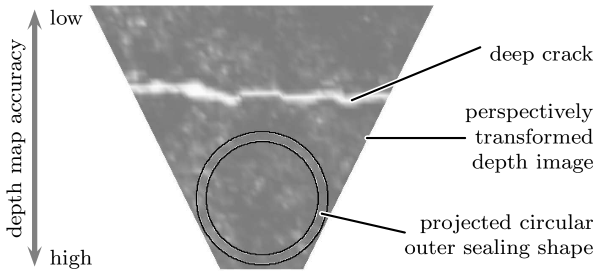

Fig. 4. Top-view of camera field-of-view with circular shape for feature analysis.

In the present case this adhesion score function $S_A \in [0,1]$ evaluates downforce F_z and point of downforce (x_F, y_F) of the adhesion system[7] according to equation 1. Here, d^{max} denotes a maximal distance between point of downforce to robot center, F_z^{min} and F_z^{max} are thresholds for minimal and maximal forces. The final training sets consist of the surface features of one circular area calculated beforehand and the corresponding adhesion score when the negative pressure chambers reach this area.

$$S_A = \max\left(0, \min\left(1, \max\left(\frac{\sqrt{x_F^2 + y_F^2}}{d^{max}}, \frac{F_z^{max} - F_z}{F_z^{max} - F_z^{min}}\right)\right)\right) \tag{1}$$

The goal is, again, a prediction of the reaction of the adhesion system based on the extracted features. Therefore, an *estimator* (compare figure 2) needs to be set up which is able to evaluate the current depth map in driving direction of the robot and to determine whether the upcoming terrain can be overcome by the robot or if it would drop off.

For the given application of feature classification, methods of supervised learning seem to be most effective. In total, three different machine learning approaches have been used and evaluated here:

- **Linear Regression** assumes a linear function $f(\vec{x})$ according to equation 2 with inputs $\vec{x} = (x_1, x_2, ..., x_p)$. The goal is to estimate the unknown coefficients β_j based on the feature vectors $\vec{x}_i$ and desired outputs y_i. The most popular method is *least squares* which minimizes the residual sum of squares[8] over all N training sets.

$$f(\vec{x}) = \beta_0 + \sum_{j=1}^{p} x_j \cdot \beta_j \tag{2}$$

- **Decision Trees** are binary trees in which each internal node corresponds to a test of one feature[8] – e. g. if $S_a > 0$ or not. Therefore, each branch represents an outcome of the test separating the remaining set (subtree) in the best way. Finally, each leaf node holds a class label which represents the prediction. A popular algorithm for the construction of a decision tree is CART using the gini coefficient as measure for a suitable division, as it is also applied here.
- **Support Vector Machines** create one or more hyperplanes in a more-dimensional space.[9] In cases of non-linear functions the input vectors need to be transformed into higher dimensions via kernel functions K to allow a linear separation in this space. The final decision function (equation 3) uses N resulting support vectors $\vec{x}_i$, coefficients α_i, y_i, and intercept β_0 to classify the input vector $\vec{x}$:

$$f(\vec{x}) = \sum_{i=1}^{N} \alpha_i \cdot y_i \cdot K(\vec{x}_i, \vec{x}) + \beta_0 \tag{3}$$

The implementation has been done using *scikit-learn*[a] which is a Python module providing machine learning algorithms. In a first step, several terrain types in form of surface patches have been created. To identify general features the training sets use normed surfaces, whereas the test sets use more realistic structures. Then, the simulated robot collects surface features and the corresponding adhesion scores during motion, serving as the desired outputs of machine learning. A total number of 3 300 datasets was recorded with a depth image resolution of 480x360 pixels.

---

[a] `http://scikit-learn.org/`

4. Experimental Results and Application

For comparison of the introduced learning methods several experiments were executed. Figure 5 illustrates the results once for the training data (left) and for the datasets for testing (right) by applying the trained regressions. For comparison, two different indicators are considered here: The *mean-square error* (MSE) should be close to zero if the prediction nearly fits the ground truth value S_A from the data sets. Additionally, the R^2 score is used which is a statistical measure of how well the regression line fits the given data. This score should be around one in an optimal case.

In terms of *linear regression* third order polynomials of each feature delivered best results, although the test examples are still to scattered which makes this method inapplicable for the given task. The *decision tree regression* delivers very promising results in the training phase (second row in figure 5) with a maximal tree depth of five. But, the resulting scores for prediction of S_A of the test data are much worse which can be seen in a mean-square error (MSE) of 0.123 compared to 0.015 in the training phase. The *support vector machine* delivers best results for the test data as it can be seen in the MSE and R^2 values in the last line of figure 5. Basis of this method was a grid search to optimize the tuning parameters and a cross-validation, which splits the training data into k subsamples of equal size, one for training and one for validation.

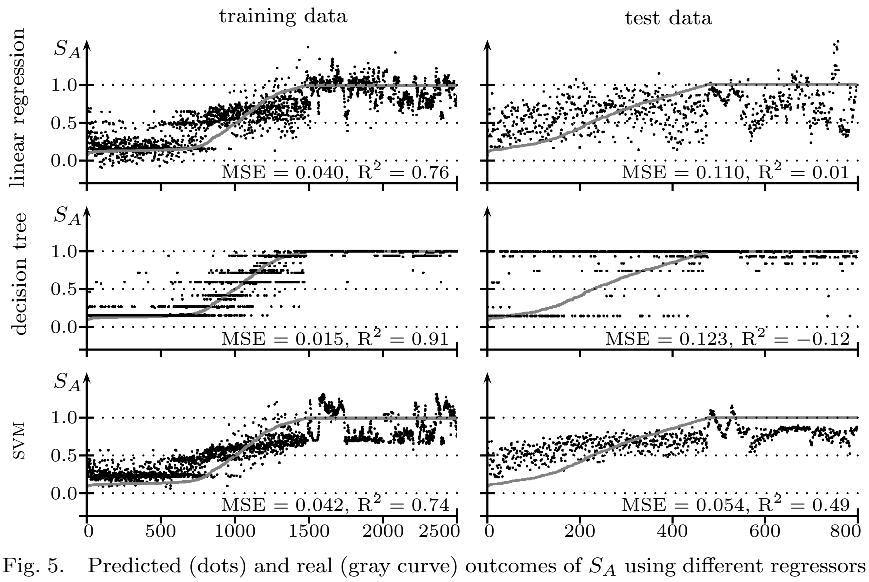

Fig. 5. Predicted (dots) and real (gray curve) outcomes of S_A using different regressors in terms of 2 500 training data (left) and 800 sets for validation (right).

Beside the presented regression methods also *decision tree classification* and *support vector classification* were tested. Here, the adhesion score has been divided into the three classes *safe* ($S_A \leq 0.25$), *drop-off* ($S_A > 0.99$) and *risky* (else). Both classificators delivered very good results in prediction of a drop-off, but the overall precision rate lies only at 0.57 in case of the decision tree classification and 0.40 for using support vectors.

Since the results seemed goal-leading but not sufficient so far, 500 additional training sets were used to test if the results could be improved by more datasets. The resulting predicted values S_A were expanded and shifted via function T (equation 4) to get more significant values.

$$T(S_A) = \max\left(0, \min\left(1, 4 \cdot S_A - 1.5\right)\right) \tag{4}$$

The result for the scaled SVM regressor can be seen in figure 6. Again, the predicted values and the ground truth of the 800 test samples are shown. Finally, a mean-square error of 0.032 and a R^2 score of 0.71 could be achieved which is a significant improvement compared to figure 5.

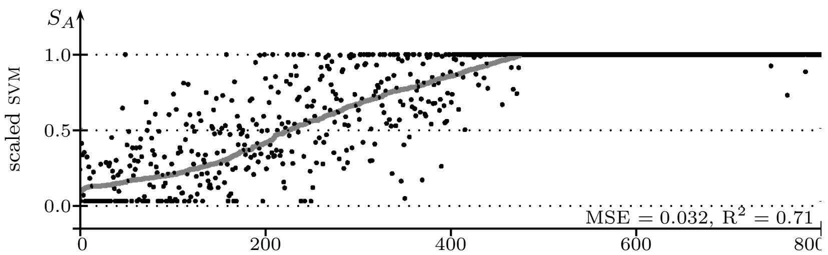

Fig. 6. Scaled validation results of predicted (dots) and real (gray curve) test data of the final support vector machine regressor.

It can be stated that each of the examined methods performs rather well on the training datasets, but none of the examined methods could achieve good results for the untrained datasets used for testing. Among the regression approaches, the support vector machines especially in combination with the scaling transformation delivered the best results. The scatter plots of the decision tree regressor e. g. show that this method predicts the state riskier than it really is. Based on these results further experiments and trainings seem to be necessary. On the one hand the training surfaces were constructed fine-grained structures whereas the test environment is coarser. On the other hand it seems that the camera resolution might be too low during data acquisition.

5. Conclusion

The presented paper aimed at aspects of terrain traversability in the range of climbing robots. Starting from a depth map captured by a simulated stereo camera, several surface features were extracted that distinguish among varying terrain structures. The traversability information are acquired using methods of supervised learning based on training sets. To determine the best supervised learning method, an investigation of a couple of different approaches was undertaken. It turned out that a modified support vector regressor reached a high prediction quality although there is still space for further improvements.

Next steps are the implementation of an improved rating value to characterize the adhesion state and the testing of further surface characteristics. Finally, more statistical data as well as training data from the real robot and stereo vision system are needed to bring this approach into application.

References

1. M. Castelnovi, R. Arkin and T. R. Collins, Reactive speed control system based on terrain roughness detection, in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, (Barcelona, Spain, 2005).
2. T. Braun, H. Bitsch and K. Berns, Visual Terrain Traversability Estimation using a Combined Slope/Elevation Model, in *Proceedings of the German Conference on Artificial Intelligence (KI)*, (Kaiserslautern, Germany, 2008).
3. S. Thrun, M. Montemerlo and A. Aron, Probabilistic terrain analysis for high-speed desert driving, in *Robotics Science and Systems Conference*, (Philadelphia, USA, 2006).
4. A. Kelly and A. Stentz, Rough terrain autonomous mobility - Part 1: A theoretical analysis of requirements, in *Journal of Autonomous Robots*, (2) (Springer, 1998) pp. 129–161.
5. D. Schmidt, C. Hillenbrand and K. Berns, Omnidirectional locomotion and traction control of the wheel-driven, wall-climbing robot, cromsci, in *Robotica Journal*, (7) (Cambridge Univ Press, 2011) pp. 991–1003.
6. T. Torims, J. Vilcans, M. Zarins, I. Strazdina and A. Ratkus, Implications of the new iso surface roughness standards on production enterprises, in *Annals of DAAAM and Proceedings*, (DAAAM International Vienna, 2010)
7. D. Schmidt and K. Berns, Risk prediction of a behavior-based adhesion control network for online safety analysis of wall-climbing robots, in *Proceeding of the 9th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, (Rome, Italy, 2012).
8. T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction* (2001).
9. H. Yu and S. Kim, Svm tutorial - classification, regression and ranking, in *Handbook of Natural Computing*, (Springer, 2012)