

Grid-based Visual Terrain Classification for Outdoor Robots using Local Features

Yasir Niaz Khan, Philippe Komma, Karsten Bohlmann and Andreas Zell

Abstract—In this paper we present a comparison of multiple approaches to visual terrain classification for outdoor mobile robots based on local features. We compare the more traditional texture classification approaches, such as Local Binary Patterns, Local Ternary Patterns and a newer extension Local Adaptive Ternary Patterns, and also modify and test three non-traditional approaches called SURF, DAISY and CCH. We drove our robot under different weather and ground conditions and captured images of five different terrain types for our experiments. We did not filter out blurred images which are due to robot motion and other artifacts caused by rain, etc. We used Random Forests for classification, and cross-validation for the verification of our results. The results show that most of the approaches work well for terrain classification in a fast moving mobile robot, despite image blur and other artifacts induced due to extremely variant weather conditions.

I. INTRODUCTION

The estimation of the ground surface is essential for a safe traversal of an autonomous robot. Employed for a variety of outdoor assignments, such as rescue missions or surveillance operations, the robot must be aware of ground surface hazards induced by the presence of slippery and bumpy surfaces. These hazards are known as non-geometric hazards [28].

Terrain identification techniques can be classified into at least two different groups: retrospective and prospective terrain identification. Whereas retrospective techniques predict the current ground surface from data recorded during robot traversal [14], [16], prospective techniques classify terrain sections, which are located on the current path, i.e. in front of the robot. The latter approaches can rely on the environment's geometry at short and long range acquired using either LADAR sensors [25] or stereo cameras [3]. Yet, classifying terrain based on geometrical reasoning alone gives rise to ambiguities which cannot be resolved in some situations: for example, tall grass and a short wall provide similar geometrical features. Furthermore, stereo cameras only yield little information at long range. This information, however, is important for generating pathways which safely guide the robot toward distant targets. Hence, in this paper, we consider another class of prospective terrain classification techniques which relies on texture features acquired from monocular cameras. In comparison with geometry features, these texture features provide meaningful information about the ground surface even at long-range distances. Using the

extracted visual cues we then apply a Random Forests based approach to the problem of terrain classification: That is, after training a model which establishes the assignment between a visual input and the its corresponding terrain class, this model is then employed to predict the ground surface of a respective visual clue. As in [9], [15], texture features are extracted from image patches which are regularly sampled from an image grid. We perform terrain classification on a patch-wise basis rather than on a pixel-wise basis because the latter tends to produce noisy estimations which complicates the detection of homogeneous ground surface regions [8].

Several authors have addressed the problem of representing texture information in terms of co-occurrence matrices [11], Markov modeling [18], [27], Local Binary Patterns (LBP) [20], and texton-based approaches [26], [2] to name a few. Yet, it remains unclear which approach is suited best for an online application on a real outdoor robot both related to prediction accuracy and run-time performance. Hence, the main motivation of our paper is a thorough comparison of different texture descriptors for representing different terrain types. Here, the data originates from a real robot traversal whose camera images contain artifacts such as noise and motion blur. These data differ from the ones included in the Brodatz data set [7] or Calibrated Colour Image Database [21]. There, the images have been acquired under controlled conditions lacking dark shadows and overexposure. Note that latter sources of noise are often present in images taken outdoors. Furthermore, terrain class prediction should be performed on-board the mobile robot. As a second contribution we introduce five further texture descriptors, the Local Ternary Patterns descriptor (LTP) [23], the Local Adaptive Ternary Patterns descriptor (LATP) [1], the SURF descriptor [4], the Daisy descriptor [24] and the Contrast Context Histograms (CCH) descriptor [13], which, to our knowledge, have not been applied to the domain of terrain identification before.

The remainder of this paper is organized as follows: Sect. II briefly summarizes the adopted techniques for representing acquired terrain patches in terms of meaningful texture descriptors. These texture descriptors constitute the basis on which the terrain classifier relies. In Sect. III, we provide details of our classification experiments whose results are presented and discussed in Sect. IV. Finally, Sect. V gives conclusions.

Y. N. Khan, P. Komma, K. Bohlmann and A. Zell are with the Chair of Computer Architecture, Computer Science Department, University of Tübingen, Sand 1, D-72076 Tübingen, Germany {yasir.khan, philippe.komma, karsten.bohlmann, andreas.zell}@uni-tuebingen.de

II. TEXTURE DESCRIPTORS

A. Local Binary Patterns

Local Binary Patterns (LBP) [20] are very simple, yet powerful texture descriptors. A 3x3 window is placed over each pixel of a grayscale image and the neighbors are thresholded based on the center pixel. Neighbors greater than the center pixel are assigned a value of 1, otherwise 0. Then the thresholded neighbors are concatenated to create a binary code which defines the texture at the considered pixel. We divide the image into a grid and calculate a histogram of binary patterns of each pixel within a patch. Thus each grid cell yields a histogram which is then used to assign a terrain class to the respective cell. Since the 8-bit binary pattern can have 256 values, we have a histogram containing 256 dimensions for classification.

Below is an example of a 3x3 pixel pattern of an image. Thresholding is performed to obtain a binary pattern:

94	38	54
23	50	78
47	66	12

1	0	1
0		1
0	1	0

Binary Pattern = 10110100

B. Local Ternary Patterns

Local Ternary Patterns (LTP) [23] are a generalization of Local Binary Patterns. Here, a ternary pattern is calculated by using a threshold k around the value c of the center pixel instead of generating a binary pattern based on the center pixel. Neighboring pixels greater than $c+k$ are assigned a value of 1, smaller than $c-k$ are assigned -1, and values between $c+k$ and $c-k$ are mapped to 0.

$$T = \begin{cases} 1 & T \geq (c + k) \\ 0 & T < (c + k) \text{ and } T > (c - k) \\ -1 & T \leq (c - k) \end{cases}$$

where c is the intensity of the center pixel.

Instead of using a ternary code to represent the 3x3 matrix, the pattern is divided into two separate matrices. The first one contains the positive values from the ternary pattern, and the second contains the negative values. From both matrices an LBP is determined resulting in two individual matrices of LBP codes. Using these codes two separate histograms are calculated. In this approach, we also divide the image into a grid and calculate histograms for each cell. The two histogram parts are concatenated to form a histogram of 512 dimensions.

Below is an example of a 3x3 pixel pattern of an image. A threshold parameter ($k=5$) is used to obtain a ternary pattern, which is then divided into two binary patterns:

94	38	54
23	50	78
47	66	12

1	-1	0
-1		1
0	1	-1

Ternary Pattern ($k=5$): 1(-1)01(-1)10(-1)

Part1=10010100, Part2=01001001

C. Local Adaptive Ternary Patterns

Local Adaptive Ternary Patterns (LATP) [1] are based on the Local Ternary Patterns. Unlike LTP, they use simple local statistics to compute the local pattern threshold. This makes them less sensitive to noise and illumination changes. LATP have been successfully applied to face recognition in [1]. We test this operator in the domain of texture classification. The basic procedure is the same as LTP. Instead of a constant threshold, the threshold (T) is calculated for each local window using local statistics as given in the equation:

$$T = \begin{cases} 1 & T \geq (\mu + k\sigma) \\ 0 & T < (\mu + k\sigma) \text{ and } T > (\mu - k\sigma) \\ -1 & T \leq (\mu - k\sigma) \end{cases}$$

where μ and σ are mean and standard deviation of the local region, respectively, and k is a constant.

The resulting ternary pattern is divided into two binary patterns like LTP and separate histograms are calculated and concatenated for classification forming a 512 dimensional vector. Below is an example of such pattern calculation:

94	38	54
23	50	78
47	66	12

1	0	0
-1		1
0	0	-1

$\mu=51.33, \sigma=25.74, \mu+k\sigma=77.07, \mu-k\sigma=25.59$

Ternary Pattern ($k=1$): 1001(-1)00(-1)

Part1=10010000, Part2=00001001

D. SURF

Speeded Up Robust Features (SURF) [4] are an extension of the famous SIFT features [17]. SURF is used to detect interest points in a grayscale image and to represent them using a 64- or 128-dimensional feature vector. These features can then be used to track the interest points across images and thus prove suitable for localization tasks. In this paper, we considered SURF features for a new application: texture classification. In SURF interest points are detected across the image using the determinant of the Hessian matrix. Box filters of varying sizes are applied to the image to extract the scale space. Then the local maxima are searched over space and scale to determine the interest points at the best scales. The key-point extraction capabilities of SURF, however, have been omitted. This is because the interest points detected by SURF are usually concentrated around sharp gradients, which are likely not present within homogeneous terrain patches. Instead we manually choose the interest point location and scale from which the SURF descriptor is determined. This renders our approach much faster.

In our approach we divide the image in a grid and use the generated patches or sub-windows to calculate the descriptors. Each image patch is then classified individually. We use 64-dimensional Upright-SURF (U-SURF) descriptors, in which the rotation invariance factor is removed. Still they are rotation invariant up to +/-15 degrees. Furthermore, we only consider a single scale for descriptor extraction which was determined experimentally using a grid-search approach.

We call this modified approach TSURF or Terrain-SURF. The SURF descriptor describes how the pixel intensities are distributed within a scale dependent neighborhood of each interest point. Haar wavelets are used to increase robustness and speed over SIFT features. First, a square window of size 20σ is constructed around the interest point, where σ is the scale of the descriptor. The descriptor window is then divided into 4×4 regular subregions. Within each subregion, Haar wavelets of size 2σ are calculated for 25 regularly distributed sample points. If x and y wavelet responses are referred by dx and dy respectively, then for the 25 sample points,

$$v_{subregion} = \left[\sum dx, \sum dy, \sum |dx|, \sum |dy| \right]$$

are collected. Hence, each subregion contributes four values to the descriptor vector resulting in a final vector of length 64 ($4 \times 4 \times 4$).

E. Daisy

The Daisy descriptor [24] is inspired from earlier ones such as the Scale Invariant Feature Transformation (SIFT) [17] and the Gradient Location-Orientation Histogram (GLOH) descriptor [19] but can be computed much faster for this purpose. Unlike SURF, which can also be computed efficiently at each pixel, it does not introduce artifacts that degrade the matching performance when used densely. For each image, first H orientation maps, $G_i, 1 \leq i \leq H$, are computed, one for each quantized direction, where $G_o(x,y)$ equals the image gradient norm at location (x,y) for direction o if it is bigger than zero, else it is equal to zero. Each orientation map is then convolved several times with Gaussian kernels of different \sum values to obtain convolved orientation maps for different sized regions. Daisy uses a Gaussian kernel, whereas SIFT and GLOH use a triangular shaped kernel.

Originally, Daisy features are calculated as dense features on the entire image. We instead divide the image into a grid of a specific size and calculate daisy features on this grid, like our TSURF approach. We call this approach as TDaisy or Terrain-Daisy denoting the calculation of daisy features across a grid. We then perform classification on these local features. Each local feature is a 200-dimensional vector.

F. CCH

The Contrast Context Histogram [12] and [13] is a new invariant local descriptor for image matching and object recognition. The motivation was to develop a computationally fast descriptor, which uses fewer histogram bins and has a good matching performance. This approach considers a histogram-based representation of the contrast values in the local region around the salient corners. First, corners are extracted from a multi-scale Laplacian pyramid by detecting the Harris corners at each level of the pyramid. For each salient corner p_c , in the center of a $n \times n$ local region R , the center-based contrast $C(p)$ of a point p in R is calculated by the formula:

$$C(p) = I(p) - I(p_c),$$



Fig. 1. Outdoor robot used for experiments

where $I(p)$ and $I(p_c)$ are the intensity values of p and p_c , respectively. A log-polar coordinate system (r,θ) is used to divide the local region R into several non-overlapping regions R_1, R_2, \dots, R_t . This makes it more sensitive to the points closer to the center. The direction of $\theta = 0$ in the log-polar system is set to coincide with the edge orientation of p_c , to ensure rotation invariance. The sub-regions are then represented by histograms. Since summation of positive and negative contrast values can damage the discriminative ability of the bin, separate histograms are calculated for positive and negative values. Hence, for each point p in region R_i , the positive histogram bin with respect to p_c is given as:

$$H_{R_i+}(p_c) = \frac{\sum \{C(p) | p \in R_i, C(p) \geq 0\}}{\#_{R_i+}},$$

where $\#_{R_i+}$ is the number of positive contrast values in the i -th region R_i . The negative histogram bin is calculated as:

$$H_{R_i-}(p_c) = \frac{\sum \{C(p) | p \in R_i, C(p) < 0\}}{\#_{R_i-}},$$

where $\#_{R_i-}$ is the number of negative contrast values in the i -th region R_i .

Finally, histograms of all subregions are combined to form the CCH descriptor of p_c for the local region R as:

$$CCH(p_c) = (H_{R_1+}, H_{R_1-}, H_{R_2+}, H_{R_2-}, \dots, H_{R_t+}, H_{R_t-})$$

The resulting local descriptor is a 64-dimensional vector.

III. EXPERIMENTAL SETUP

A. Testing Platform

Our outdoor robot (Fig 1) is a modified RC-model truck whose body was removed and replaced by a dual-core PC, a 32-bit microcontroller and different sensors attached to the vehicle, including a Point-Grey Firefly color camera with a 6mm lens to capture images at a resolution of 640x480 pixels. It is one of the 8 such robot developed and built at our

department. For our experiments, we ran the robot at about 1 m/s speed while capturing images from the camera, hence not all of the acquired images are sharp due to motion blur artifacts. The height of the mounted camera is approximately 50cm from the ground. The robot is equipped with tractor tires to be able to run on very rough terrain. However, these tires produce an increased amount of vibration while traversing even a smooth surface. The camera is tilted down so as to capture the terrain directly in front of the robot. Hence, the camera captures images starting from a distance of 30cm with respect to the robot's front.

B. Terrain Classes

We drove the robot outdoors in our campus and observed the terrain types visible to the robot through the camera. The outdoor area of the campus consists of roads, meadows and some parking areas covered with gravel or tiles. We were able to identify five different classes: asphalt, gravel, grass, big-tiles and small-tiles. We navigated the robot multiple times over different routes at varying times of the day. One of these experiments was carried out when the sun was about to set which resulted in a direct insolation of the camera. In this case, the image colors were extremely distorted.

The second experimental setup was a heavily clouded sky after rainfall. Some of the terrain types contained wet and dry patches, e.g. asphalt, gravel, etc. In this case, a single terrain type contained different colors. The third scenario was at noon on a sunny day. Note that not all terrain types were captured in each scenario.

While driving on the campus we found that all terrain types contained many different features depending on the location and time at which the pictures were acquired. Fig. 2 shows different terrain types indicating the artifacts introduced under different scenarios. For example, Fig. 2(a) shows a blurred image of the grass terrain type along with small plants and their flowers. Fig. 2(b) shows the asphalt terrain type with a wet patch after rain. In fig. 2(c) the gravel terrain type is depicted after rainfall. Here, water was gathered in a bigger amount. Fig. 2(d) shows a sample image from the big-tiles terrain type. Since parts of the terrain are shadowed, its intensity changes a lot and the boundary also becomes difficult to classify. Similarly, Fig. 2(e) shows an image from the small-tiles terrain type. It is also noticeable that the shadow of a tree induces texture artifacts of its own.

Fig. 3 shows the grass terrain type under two different weather conditions. The image on the left is taken in winter (middle of March) one hour before sunset. The sun was looking into the camera at that time. The image on the right is taken in spring (middle of June) on a cloudy afternoon. Moreover, the image on the left also has a patch of snow among the grass. Both of these scenarios were included in the dataset. This clearly shows that color based descriptors will not work in all cases. Also note that under similar conditions, color based descriptors can misjudge the wet and dry or shaded and open parts of the same terrain type. Other than that, color will only accurately distinguish grass from other terrain types as is obvious from the sample terrain images.

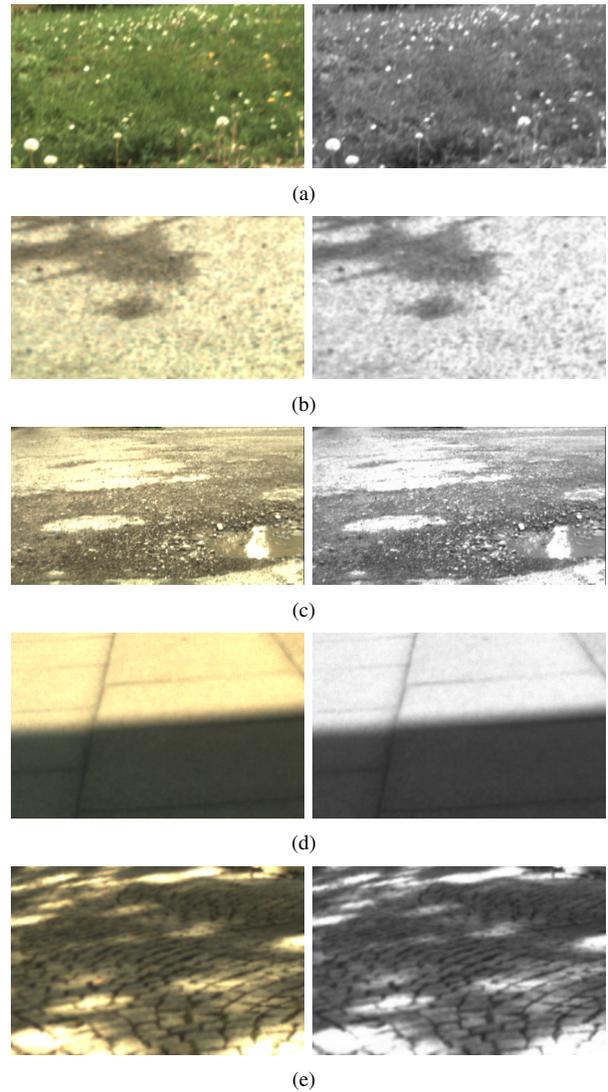


Fig. 2. Sample images of different terrain types: (a) grass, (b) asphalt, (c) gravel, (d) big-tiles and (e) small-tiles, both in color and in grayscale

Finally, Fig 4(a) shows the small-tiles terrain type with blur induced due to robot motion and fig. 4(b) shows the same terrain type with over-exposure due to sun.

All images are characterized by the presence of not only one but multiple terrain types. These images were labeled manually to generate training images for each class. Almost all of the images contained diagonal or irregular boundaries between two terrain types. Hence, even after clipping, most of the images contained other terrain types at the borders. Note, that this interferes with the terrain descriptors which are based on a rectangular grid and hence results in a decrease in classification accuracy. Images containing blur were not filtered out, except in extreme cases where the blur artifacts were too dominant.

C. Classifier

We performed the classification task using several classifiers. Therefore, we used the machine learning software Weka [10] to train and test these classifiers. The adopted



Fig. 3. Difference of grass color under different sun angle and weather conditions



Fig. 4. Samples from small-tiles terrain type under: (a) blur, (b) over-exposure

classifiers were Random Forests, Support Vector Machine (SVM) using the Sequential Minimal Optimization (SMO) training algorithm [22], the Multilayer Perceptron (MLP), LIBLINEAR, J48 Decision Tree, Naive Bayes and k-Nearest Neighbour. From this set, Random Forests gave the best overall performance.

Random Forests [6] are an ensemble classifier consisting of a collection of individual decision tree predictors. These binary trees are grown randomly with controlled variation [5]. That is, for each tree an individual bootstrap sample is drawn. Further, each node of a tree uses a varying feature subset of the complete feature set on which a binary decision is based on. Given an input vector, this test decides whether to traverse the left or right child of the tree. Leaf nodes are assigned the actual class labels. If, during tree traversal, a leaf node is reached, the tree casts a unit vote for the class represented by the leaf label. The class with the largest number of votes is then defined as the predicted class.

Concerning prediction accuracy, using a larger number of trees reduces the generalization error for forests. However, this also increases the run-time complexity of the classification process. Hence, a compromise has to be found between accuracy and speed by varying the number of trees. We found that in our case 100 trees gave good accuracy without a significant decrease in speed.

Free parameters such as the patch size and the hyperparameters for all the applied classification approaches were found using a grid-search. Further, we adopted a 5-fold cross-validation scheme to verify the accuracy of the results.

IV. RESULTS

For each descriptor, we applied the classifiers and obtained the true positive rate (TPR) of the entire dataset. The TPR is the ratio of the correctly classified terrain type instances and the number of all test patterns contained in the data set. Here, we display the results as percentages. Table I presents a summary of accuracy results of the six approaches on the five terrain types using Random Forests.

Descriptor	total	gravel	asphalt	grass	big-tiles	small-tiles
LBP	97.2%	94.6%	98.8%	95.0%	99.8%	97.7%
LTP	97.4%	96.7%	98.3%	94.6%	99.7%	97.5%
LATP	97.0%	93.0%	98.4%	95.8%	99.4%	98.5%
TSURF	92.1%	94.5%	92.4%	94.2%	97.9%	81.7%
TDaisy	70.3%	55.7%	82.9%	59.8%	76.7%	76.3%
CCH	49.7%	28.8%	59.4%	52.3%	72.3%	35.6%

TABLE I
SUMMARY OF CLASSIFICATION RESULTS OF THE SIX FEATURE DESCRIPTORS

For LBP, each image was divided into patches of size 100x100 for histogram calculation. Big-tiles was the best classified terrain type whereas gravel and grass were the worst. The confusion matrix of the LBP classification is presented in Table II. It can be observed that the confusion is greatest between grass and gravel.

For LTP-based classification, each image was again divided into 100x100 patches. We chose different values for the threshold value k described in section II-B. The optimum threshold value was found to be 5. In this case big-tiles and small-tiles were the best identified and grass the worst identified terrain class. Table III presents the confusion matrix of the LTP approach. There are some grass patterns classified as gravel. The LTP descriptor is one of the longest descriptors consisting of a 512 dimensional vector.

Similarly for LATP-based classification, each image was again divided into 100x100 patches. We tried different values of the threshold value k described in the section II-C. The optimum threshold was found to be 0.4. In this case big-tiles was the best identified and gravel the worst identified terrain class. Table IV presents the confusion matrix of the LATP approach. Some confusion occurs between gravel and grass. The LATP descriptor has the same size as the LT descriptor and hence also consists of a 512 dimensional vector.

For TSURF based classification, the descriptors were calculated on a grid of 100x100 pixels. Different scale levels (σ) described in section II-D were tried and $\sigma = 15$ was found to be the best scale. The classification performance of the big-tiles pattern is the best in TSURF while small-tiles proves to be the worst classified terrain type. The confusion matrix of the TSURF approach is presented in Table V. There is a large number of small-tiles patterns identified as gravel. Note that the number of samples is less in this case, since only the grid intersections are used to calculate descriptors as opposed to patches. The TSURF descriptor is one of the smallest descriptors consisting of only 64 dimensions.

For TDAISY based classification the descriptors calculated on a grid of 100x100 pixels gave best results. In this case asphalt was the best identified and gravel the worst identified terrain class. Table VI presents the confusion matrix of the TDAISY approach. There is a large number of gravel patterns classified as grass and vice versa. A grid size of 30x30 produced a slightly better results: Total=72.0%, gravel=58.1%, asphalt=86.7%, grass=66.9%, big-tiles=77.5%, small-tiles=70.9%.

	gravel	asphalt	grass	big-tiles	small-tiles
gravel	610	6	29	0	0
asphalt	2	637	6	0	0
grass	29	3	614	0	0
big-tiles	0	1	0	645	0
small-tiles	1	2	0	12	631

TABLE II
CONFUSION MATRIX FOR LBP

	gravel	asphalt	grass	big-tiles	small-tiles
gravel	624	10	11	0	0
asphalt	3	634	3	4	1
grass	26	9	611	0	0
big-tiles	0	0	0	644	2
small-tiles	0	1	0	15	630

TABLE III
CONFUSION MATRIX FOR LTP

Finally, for the CCH-based classification approach, the descriptors were calculated on a grid of 100x100 pixels. Here big-tiles was also the best identified terrain type while gravel was the worst. The Confusion matrix of the CCH approach is presented in Table VII. The largest confusion occurs between gravel and grass. Finally, the CCH descriptor only consists of 64 dimensions.

V. CONCLUSION

In this paper, we thoroughly investigated the applicability of varying local descriptors for visual terrain classification on outdoor mobile robots. Along with three texture-based descriptors, LBP, LTP, and LATP, we have tested modified forms of two other descriptors: SURF and DAISY and an additional CCH descriptor. SURF and DAISY are modified to be calculated on a grid drawn across the image. The texture-based descriptors performed the best. LTP gave the best performance, however, it has one of the largest feature vector. LATP is the other largest feature vector that also performs well. TSURF has one of the smallest feature vectors and its performance is appropriate, though not among the

	gravel	asphalt	grass	big-tiles	small-tiles
gravel	600	6	39	0	0
asphalt	2	635	6	0	2
grass	24	2	619	0	1
big-tiles	0	0	0	642	4
small-tiles	0	1	0	9	636

TABLE IV
CONFUSION MATRIX FOR LATP

	gravel	asphalt	grass	big-tiles	small-tiles
gravel	309	2	5	8	3
asphalt	3	302	5	15	2
grass	11	0	308	0	8
big-tiles	2	5	0	321	0
small-tiles	26	15	14	5	268

TABLE V
CONFUSION MATRIX FOR TSURF

	gravel	asphalt	grass	big-tiles	small-tiles
gravel	359	29	115	53	89
asphalt	21	535	36	42	11
grass	119	58	386	52	31
big-tiles	17	47	58	495	28
small-tiles	69	5	23	56	493

TABLE VI
CONFUSION MATRIX FOR TDAISY

	gravel	asphalt	grass	big-tiles	small-tiles
gravel	186	83	199	91	86
asphalt	69	383	92	66	35
grass	137	56	338	36	79
big-tiles	63	48	31	467	37
small-tiles	110	63	169	74	230

TABLE VII
CONFUSION MATRIX FOR CCH

best. It is interesting to note that it performs best for terrain types which are worst identified by the three best texture-based descriptors. CCH is also the smallest feature vector, but it gave the worst performance in this application. TDaisy has the second smallest feature vector, but its performance is also not satisfactory.

Furthermore, it is demonstrated that visual terrain classification can be successfully performed even in extreme conditions, such as motion blur induced by a fast moving robot and its vibrating camera, different weather conditions, both wet and dry ground surfaces and a low camera viewpoint. Most current texture classification approaches use sharp images containing a single texture captured from a perpendicular camera angle. We used images from real runs of the robot containing blurred images with non-sharp terrain boundaries.

Future work will focus on the comparison of different machine learning approaches on our dataset and the inclusion of additional terrain types. Outdoor mapping and localization based on terrain classification is also an interesting research direction.

REFERENCES

- [1] Moulay Akhloufi and Abdel Hakim Bendada. Locally adaptive texture features for multispectral face recognition. In *IEEE International Conference on Systems, Man, and Cybernetics*, Istanbul, Turkey, October 2010. IEEE.
- [2] A. Angelova, L. Matthies, D. Helmick, and P. Perona. Learning and prediction of slip from visual information: Research articles. *Journal of Field Robotics*, 24(3):205–231, 2007.
- [3] M. Bajracharya, T. Benyang, A. Howard, M. Turmon, and L. Matthies. Learning long-range terrain classification for autonomous navigation. In *IEEE International Conference on Robotics and Automation, 2008 (ICRA 2008)*, pages 4018–4024, Pasadena, CA, 2008.
- [4] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In *Proceedings of the European Conference on Computer Vision (ECCV 2006)*, pages 404–417, Graz, Austria, 2006.
- [5] Leo Breiman. Bagging predictors. In *Machine Learning*, volume 24, pages 123–140, Hingham, MA, USA, August 1996. Kluwer Academic Publishers.
- [6] Leo Breiman. Random forests. In *Machine Learning*, volume 45, pages 5–32, Hingham, MA, USA, October 2001. Kluwer Academic Publishers.
- [7] P. Brodatz. Textures: A photographic album for artists & designers. New York: Dover, New York, NY, 1966.

- [8] I.L. Davis, A. Kelly, A. Stentz, and L. Matthies. Terrain typing for real robots. In *Proceedings of the Intelligent Vehicles '95 Symposium*, pages 400–405, Detroit, MI, 1995.
- [9] C. Dima, N. Vandapel, and M. Hebert. Classifier fusion for outdoor obstacle detection. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2004)*, pages 665 – 671, New Orleans, LA, USA, 2004.
- [10] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: An update. *SIGKDD Explorations*, Volume 11, Issue 1, 2009.
- [11] R.M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 3(6):610–621, November 1973.
- [12] Chun-Rong Huang, Chu-Song Chen, and Pau-Choo Chung. Contrast context histogram - a discriminating local descriptor for image matching. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 53–56, Washington, DC, USA, 2006. IEEE Computer Society.
- [13] Chun-Rong Huang, Chu-Song Chen, and Pau-Choo Chung. Contrast context histogram-an efficient discriminating local descriptor for object recognition and image matching. *Pattern Recognition*, 41(10):3071–3077, 2008.
- [14] K. Iagnemma and S. Dubowsky. Terrain estimation for high-speed rough-terrain autonomous vehicle navigation. In *Proceedings of the SPIE Conference on Unmanned Ground Vehicle Technology IV*, Orlando, FL, USA, 2002.
- [15] D. Kim, S. Sun, S. Oh, J. Rehg, and A. Bobick. Traversability classification using unsupervised on-line visual learning for outdoor robot navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2006)*, pages 518–525, Orlando, FL, USA, 2006.
- [16] P. Komma, C. Weiss, and A. Zell. Adaptive Bayesian filtering for vibration-based terrain classification. In *IEEE International Conference on Robotics and Automation (ICRA 2009), Kobe, Japan*, pages 3307–3313, May 2009.
- [17] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [18] B.S. Manjunath and R. Chellappa. Unsupervised texture segmentation using markov random field models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):478–482, 1991.
- [19] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(10):1615 – 1630, October 2005.
- [20] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 24(7):971–987, 2002.
- [21] A. Olmos and F.F.A. Kingdom. McGill calibrated colour image database.
- [22] John C. Platt. Using analytic qp and sparseness to speed training of support vector machines. In *Neural Information Processing Systems 11*, pages 557–563. MIT Press, 1999.
- [23] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. In *Proceedings of the 3rd international conference on Analysis and modeling of faces and gestures (AMFG 07)*, pages 168–182, Berlin, Heidelberg, 2007. Springer-Verlag.
- [24] Engin Tola, Vincent Lepetit, and Pascal Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:815–830, 2010.
- [25] N. Vandapel, D. Huber, A. Kapuria, and M. Hebert. Natural terrain classification using 3-d lidar data. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2004)*, pages 5117–5122, New Orleans, LA, April 2004.
- [26] M. Varma and A. Zisserman. A statistical approach to texture classification from single images. *International Journal of Computer Vision*, 62(1-2):61–81, 2005.
- [27] P. Vernaza, B. Taskar, and D.D. Lee. Online, self-supervised terrain classification via discriminatively trained submodular Markov random fields. In *IEEE International Conference on Robotics and Automation (ICRA 2008)*, pages 2750–2757, 2008.
- [28] B. H. Wilcox. Non-geometric hazard detection for a mars microrover. In *Proceedings of the NASA Conference on Intelligent Robots in Field, Factory, Service and Space*, pages 675–684, Houston, TX, 1994.