HD Ground – A Database for Ground Texture Based Localization

Jan Fabian Schmid[1,2], Stephan F. Simon[1], Raaghav Radhakrishnan[1,3], Simone Frintrop[4], Rudolf Mester[2,5]

***Abstract*— We present the HD Ground Database, a comprehensive database for ground texture based localization. It contains sequences of a variety of textures, obtained using a downward facing camera. In contrast to existing databases of ground images, the HD Ground Database is larger, has a greater variety of textures, and has a higher image resolution with less motion blur. Also, our database enables the first systematic study of how natural changes of the ground that occur over time affect localization performance, and it allows to examine a teach-and-repeat navigation scenario. We use the HD Ground Database to evaluate four state-of-the-art localization approaches for global localization, localization with the approximate pose being known, and relative localization.**

I. INTRODUCTION

The use of ground images from a downward-facing camera is a promising, low-cost approach to achieving millimeter-level localization [1][2][3][4][5]. An agent that uses the ground instead of surrounding landmarks to localize itself has several advantages: (1) it works in dynamic environments with frequently changing surrounding; (2) it works with an occluded surrounding, e.g. in a busy pedestrian zone; (3) it observes only the ground reducing privacy concerns.

Recent developments show that ground texture based localization is suitable for self-contained approaches that can comply with all requirements of localization: global map-based localization without any knowledge of the current positioning [3][5], subsequent map-based local pose refinement using a prior pose estimate [2][3][6], as well as map-less relative localization in form of visual odometry between subsequently recorded ground images [7][8][9].

This work contributes the HD Ground Database[6], a large set of high-resolution ground images, recorded with a downward-facing camera shielded from external light sources. It enables the examination of localization under varying conditions, such as clean versus *dirty*, and dry versus wet ground. Also, in comparison to existing ground image datasets [4], the HD Ground Database provides larger area coverage, higher resolution images with less motion blur, and image sequences from a teach-and-repeat scenario in which the robot is supposed to follow a previously learned path. To examine the impact of natural wear and tear, as well as weather, on the localization performance, we recorded weekly test sequences of similar paths over a period of 24 weeks. An evaluation of state-of-the-art localization methods shows that this time interval between mapping and localization is highly relevant for outdoor environments.

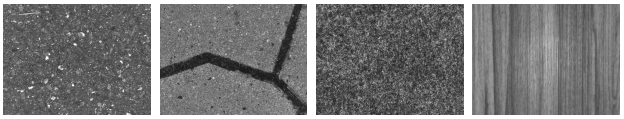


Fig. 1: The four main textures of the HD Ground Database. From left to right: asphalt, cobblestone, carpet, and laminate.

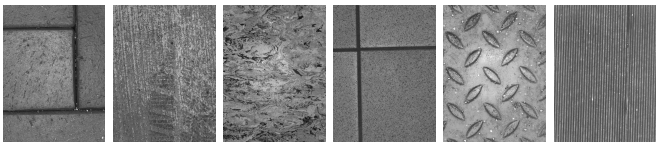


Fig. 2: Additional textures to test generalization. From left to right: pavement, concrete, linoleum, tiles, steel, and rubber.

II. RELATED WORK

A. Datasets for ground texture based localization

To the best of our knowledge the *Micro-GPS* databases of Zhang et al. [4] are the only publicly available ground image databases suited for ground texture based localization. The authors present a database recorded with a PointGrey CM3 camera, and a second one recorded with an iPhone 6. In [5] and [6] the PointGrey CM3 database was used to compare localization approaches. But for many evaluated localization problems, e.g. localization with prior pose estimate (on all textures but wood), every method reaches close to perfect success rate, raising the question of whether the database covers all challenges of the task. For example, it is not systematically covering ground changes that occur over time.

Another publicly available dataset of ground images was created by Xue et al. [10]. It contains more than 30000 images of 40 outdoor ground textures. The dataset is used to show the effectiveness of differential angular imaging for in-place material recognition. Accordingly, it provides many images of the same places from varying camera angles, but not the area coverage required to examine localization tasks.

Alternatively, Rodriguez and Castano-Cano [11] propose to generate image data with a virtual camera from simulated vehicle drives over a single high-resolution terrain image. This allows to generate a virtually infinite number of different sequences. However, while some image conditions can be simulated, this does not allow to examine the effects of actual changes that appear on the ground. In contrast, the HD Ground Database explicitly captures these changes.

[1]Robert Bosch GmbH, Hildesheim, Germany
SchmidJanFabian@gmail.com
[2]VSI Lab, CS Dept., Goethe University, Frankfurt am Main, Germany
[3]Fraunhofer Institute for Applied Information Technology FIT, Germany
[4]Department of Informatics, University of Hamburg, Germany
[5]Norwegian Open AI Lab, CS Dept., NTNU Trondheim, Norway

[6]https://github.com/JanFabianSchmid/HD_Ground

TABLE I: A comparison of the sizes of publicly available ground image datasets.

Database	Total area	Largest single area	#Reference images	#Test images	#Textures	Resolution	mm/pixel
Micro-GPS (PointGrey) [4]	$145.85\,\mathrm{m}^2$	$41.76\,\mathrm{m}^2$	23 487	28 929	6	1288×964	0.16
Micro-GPS (iPhone 6) [4]	$40.27\,\mathrm{m}^2$	$27.52\,\mathrm{m}^2$	2 525	2 483	2	1280×720	NA
HD Ground (ours)	$347.73\,\mathrm{m}^2$	$106.12\,\mathrm{m}^2$	129 965	71 463	11	1600×1200	0.1

B. Methods for ground texture based localization

We introduce the methods that are evaluated in this paper, and distinguish between methods for relative localization, and absolute localization with or without available prior pose estimate (hereafter referred to as *prior*). Absolute localization without prior is performed once at start-up and is referred to as *global localization*, while subsequent *local localization* exploits the available prior. All evaluated methods are feature-based and designed for absolute localization. They estimate the query image pose based on a map consisting of a set of reference images with known poses. However, the methods can also be used for relative localization, considering only the previous query image as reference image.

Chen et al. [3] developed StreetMap, which is based on SURF [12] features and treats global and local localization separately. It also has a variant specifically for floors with tile structure, but it is not considered by us. For global localization, StreetMap uses Bag of Words (BoW) image retrieval [13] to consider only reference images similar to the query image, while for local localization only the reference images from the local vicinity of the prior are considered.

Radhakrishnan et al. [14] propose an alternative to BoW for ground image retrieval, which is based on Deep Metric Learning (DML). They train a convolutional neural network in Siamese fashion to predict the overlap of ground image pairs. At inference time its last layer activations are used as image embedding, and overlapping images of a query image are retrieved as the ones with most similar embedding.

Kozak and Alban propose Ranger [2], a local localization method that computes ORB [15], respectively BRIEF [16], descriptors for CenSurE [17] keypoints. Ranger iteratively considers the spatially closest reference image to the given prior to match its features with that of the query image. The matches validated by the *cross check* constraint are used for RANSAC-based pose estimation, localization terminates if the resulting pose estimate is supported by a sufficient number of matches; otherwise, the next closest image is used.

Schmid et al. [5] proposed a method based on SIFT [18] keypoints that uses only the first 15 bit of LATCH [19] as compact binary descriptor, and a cost-effective matching technique called *identity matching*, which considers only identical descriptor values as matches. In order to deal with the large number of outlier matches, they use a voting procedure for spatial verification of matches [4], and use the remaining matches for RANSAC-based pose estimation.

In a follow-up work, Schmid et al. reduce the computation time of their method substituting SIFT with keypoint sampling [6], where keypoints are determined arbitrarily, regardless of the image content. The authors argue that this is a viable strategy if the ground is assumed to be locally planar, reducing the localization task to a 2D problem and if an adequately accurate prior is available to define feature orientations relative to the map coordinate system.

TABLE II: Details of the 4 main textures. Regular sequences are the test sequences that we recorded on a weekly basis.

	Asphalt	Cobblestone	Carpet	Laminate
Area covered m^2	106.12	59.28	90.15	16.18
Reference images	32251	25337	33456	5812
Test images	17483	14442	16579	9052
Regular sequences	12 dry, 9 wet	12 as it is, 12 cleaned	22	22

III. THE HD GROUND DATABASE

We present a large database of ground images for ground texture based localization: the HD Ground Database. For eleven textures, it contains reference images, covering the application areas, and test images that are to be localized.

The four main textures (see Fig. 1) are footpath asphalt, parking place cobblestone, office felt carpet, and kitchen laminate. For these textures, test image sequences were captured systematically by recording a similar trajectory on a weekly basis (more details on the timings are given in the evaluation section). Additionally, separate sets of trajectories were recorded in quick succession. These trajectories are following quite precisely the same path on the coverage area, which allows to evaluate a teach-and-repeat scenario in which a robot is steered along a specific path once, and subsequently is supposed to follow the taught path autonomously in both directions. Table II presents further details for the main textures. Typically, localization methods are adapted to a database or a specific texture through training or parametrization. For this purpose, we provide additional *training areas*: a separate square meter was recorded for each of the main textures (see Fig. 3), and also a $2\,\mathrm{m}^2$ door mat. For six further textures (see Fig. 2) reference and test images are captured on the same day: terrace pavement ($24.8\,\mathrm{m}^2$), garage concrete ($18.2\,\mathrm{m}^2$), workroom linoleum ($17.1\,\mathrm{m}^2$), bathroom tiles ($3.8\,\mathrm{m}^2$), checker plate steel ($3.3\,\mathrm{m}^2$), and ramp rubber ($2.8\,\mathrm{m}^2$). We call these *generalization textures*, as we use them to evaluate generalization capabilities.

A. Setup of the recording platform

Our platform is a modified RT3-2 VolksBot [20]. The only sensor being used for this database is its ground-facing camera. The recording area is shielded from external lighting and illuminated by a 24 V, 72 Watt LED ring. Pulsed LED lighting is synchronized with the camera exposure, allowing

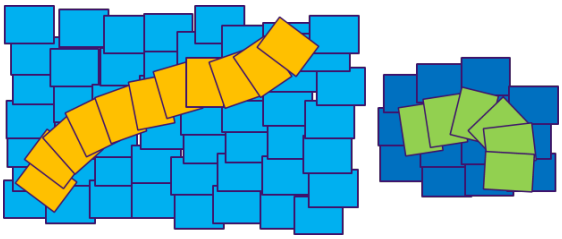


Fig. 3: Application areas of the HD Ground Database are covered by their reference images (light blue). They form the map in which separately acquired query images (orange) are to be located. For the main textures, additional training areas (dark blue) and query images (green) are recorded.

us to provide bright illumination during recording, enabling exposure times of only $0.1\,\mathrm{ms}$. A frame rate of $50\,\mathrm{Hz}$ is possible, but we retain only every fourth frame.

Some of the most important design parameters for a recording setup with a downward-facing camera are the exposure time τ, the longitudinal length of the recorded image l_{long}, and the camera height h. We can derive guidelines for those quantities given some requirements for the platform: a vehicle speed of up to $v_{\max} \leq 20\,\mathrm{km/h} \approx 5.56\,\mathrm{m/s}$ should be supported; for visual odometry, consecutive images should have a longitudinal relative overlap of at least $o_{\min} \geq \frac{1}{3}$; and motion blur, i.e. the traveled distance during exposure b, should be smaller than $b_{\max} \leq 0.5\,\mathrm{mm}$. Three more constraints are given by the maximum recording speed of our AVT Manta G-235C camera (Sony IMX174 global shutter CMOS sensor) with $f = 50\,\mathrm{Hz}$; the recording opening angle of our lens (Schneider Kreuznach Cinegon 1.4/12-0906) of $\alpha \approx 48°$ image diagonal; and the image aspect ratio of $4:3$.

The exposure time is derived from the maximum allowed motion blur $b_{\max}$ and the supported vehicle speed $v_{\max}$ as

$$\tau = \frac{b_{\max}}{v_{\max}} \approx \frac{0.0005\,\mathrm{m}}{5.56\,\mathrm{m/s}} \approx 0.09\,\mathrm{ms}. \tag{1}$$

The longitudinal image length is defined by the vehicle speed $v_{\max}$, the recording frequency f, and the image overlap $o_{\min}$:

$$l_{\mathrm{long}} = \frac{v_{\max} \cdot 1/f}{1 - o_{\min}} \approx \frac{5.56\,\mathrm{m/s} \cdot 0.02\,\mathrm{s}}{1 - 1/3} \approx 0.167\,\mathrm{m}. \tag{2}$$

Finally, the camera has to be mounted high enough to capture the diagonal of our coverage area with length l_{long} and width $l_{\mathrm{lat}} = l_{\mathrm{long}} \cdot 3/4$, given the camera opening angle α:

$$h = \frac{0.5 \cdot (l_{\mathrm{long}}^2 + (3/4 \cdot l_{\mathrm{long}})^2)^{0.5}}{\tan(\alpha/2)} \approx 0.234\,\mathrm{m}. \tag{3}$$

We consider the effort for the buildup of this setup and the resulting good image quality to be realistic for scenarios where a robot is equipped with a dedicated ground camera.

B. Data recording

We differentiate three systematic setups of data recording.

- **Initial scanning of the whole coverage area (reference images).** The application area is recorded lane-by-lane, with each lane having an offset of 3.8 to $5\,\mathrm{cm}$ to the previous one. That way images have approximately $2/3$ overlap with neighboring images from the previous and next lane, as well as with the previous and next image of the same lane. Accordingly, every point on the ground is covered by about 9 reference images, which allows us to properly align the images during mapping.
- **Recording of regular test sequences**. For each main texture, we define a regular test path. Weekly test sequences are recorded by roughly following the respective paths. For cobblestone, two regular test paths are recorded: one where the area is cleaned before recording and one where it is not. For asphalt, additional sequences are recorded with weather-caused wet surface.
- **Recording of teach-and-repeat sequences**. A $20\,\mathrm{m}$ rope is put in a curved shape on the application area. Then, we closely follow this rope two times in forward and backward driving direction. Five different rope configurations are recorded per texture, examples are presented in Fig. 6.

For the training areas and generalization textures, test images are recorded on arbitrary paths directly after the initial scanning. We calibrate the camera once using a pinhole model with two radial distortion parameters and use the rectified images. Also, we compensate for vignetting by normalizing each image with an average brightness image.

C. Mapping

We create a map for each application and training area. They are created offline with an image stitching process similar to that of Zhang et al. [4], aligning the reference images in a common map coordinate system.

A first image is put to the origin of the coordinate system and then we compute relative poses of consecutively recorded images. This yields us the initial reference image pose estimates. Relative poses are estimated with a simple feature-based approach, using SIFT features, a ratio test based brute-force matching strategy, and final RANSAC-based pose estimation. This incremental pose estimation quickly accumulates drift, so only a small set of 5 to 50 images is added to the map at each iteration of the mapping process. Once the initial pose estimates of the added images are available, we estimate their poses relative to all their (potentially) neighboring images. It is crucial to avoid incorrect estimates at this stage. Therefore, we require that each relative transformation of image n to one of its neighbors (image x) is confirmed by the relative transformation of the $(n-1)$-th or the $(n+1)$-th image to image x. Let $[\mathrm{R}, \mathbf{t}]_n^x$ denote the transformation from image n to image x, consisting of a rotation R and a translation $\mathbf{t}$, then we require

$$[\mathrm{R}, \mathbf{t}]_n^x \approx [\mathrm{R}, \mathbf{t}]_{n\pm1}^x [\mathrm{R}, \mathbf{t}]_n^{n\pm1}. \tag{4}$$

Furthermore, we require the number of RANSAC inliers to exceed 100, which is an empirical threshold that depends on the employed feature extractor and its parametrization. Unconfirmed image pose relations are discarded.

At the final step of each mapping iteration, the set of all reference image poses $\{[\mathrm{R}, \mathbf{t}]\}$ is jointly optimized, consid-

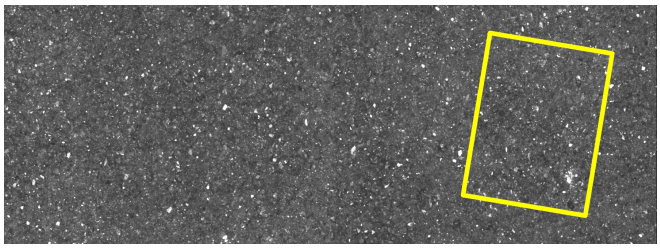


Fig. 4: A small section from the stitched asphalt map. The yellow rectangle corresponds to a single image.

ering pairs of corresponding features (f_i^k, f_j^k) between all pairs of neighboring images (i,j), for which the relative pose estimation was confirmed. Similar to Zhang et al., we formulate the optimization as a non-linear least-squares optimization problem in Ceres [21], using the loss function:

$$\mathrm{E} = \min_{\{[\mathrm{R},\mathbf{t}]\}} \sum_{(i,j)} \sum_{(f_i^k, f_j^k)} ([\mathrm{R},\mathbf{t}]_i \cdot f_i^k - [\mathrm{R},\mathbf{t}]_j \cdot f_j^k)^2. \quad (5)$$

With $[\mathrm{R},\mathbf{t}]_i$ denoting the transform mapping image i into the map. Map correctness is confirmed by visually inspecting the stitched images. We observe only small amounts of smearing. Otherwise, image transitions are smooth as in Fig. 4.

D. Comparison with existing databases

One of the most important novel aspects of our database is the recording of regular test sequences for a systematic evaluation of localization performance over time. Also, we enable the evaluation of a teach-and-repeat scenario and our database is larger than existing ones. Table I compares the sizes of HD Ground with the Micro-GPS databases. The largest coverage area recorded for the HD Ground Database is 2.5 times larger than that of the Micro-GPS databases ($41.76\mathrm{m}^2$ of wood for Micro-GPS compared to $106.18\mathrm{m}^2$ of asphalt for HD Ground). Larger areas can be used to evaluate the effect of visual aliasing when considering a larger number of reference images during localization. In this context, visual aliasing means that different places have similar visual appearances, leading to confusion during localization.

While Micro-GPS provides a minimal set of reference images covering the application area, we provide overlapping reference images. This means, for example, that our asphalt dataset, with 32251 images, contains more than 8 times as many reference images as the wood dataset of Micro-GPS, with 3826 images, while covering only a 2.5 times larger area. For instance, having overlapping images available, a localization method could reduce its memory footprint storing only those features that consistently appear in the overlap of multiple reference images, as suggested by Schmid et al. [6].

Our images present the ground at a higher resolution which allows to examine the extent to which this is beneficial. Also, our exposure time of $0.1\,\mathrm{ms}$ reduces motion blur compared to the Micro-GPS database with exposure times of 3–5 ms [22].

IV. EVALUATIONS ON THE DATASET

As an example of the examinations possible with HD Ground, we evaluate Ranger [2], StreetMap [3], and the two Ground Texture Based Localization (GTBL) methods of Schmid et al. [5][6], one using SIFT keypoints, hereafter called *GTBL SIFT*, and one using randomly sampled keypoints, hereafter called *GTBL RND*. All four methods estimate query image poses with respect to a set of reference images with known poses. This can be all reference images to perform global localization or a subset if some can be disregarded as unlikely overlapping with the query image. We examine two approaches to select a subset of reference images: (1) image retrieval; and (2) consideration of an available prior pose estimate. Image retrieval is a technique to find similar images, in this case the overlapping ones, to a given query image in a database of reference images [23]. If a prior is available, it is sufficient to consider its spatially closest reference images. The radius in which reference images are considered depends on the confidence in the prior.

A. Image retrieval for ground images

We consider two approaches to image retrieval: Bag of Words (BoW) [13] as proposed by Chen et al. [3], and the Deep Metric Learning (DML) method of Radhakrishnan et al. [14]. In both cases, the method retrieves the most similar reference images as the ones being mapped to image descriptors with shortest distance to that of the query image.

1) BoW image retrieval: We use the FBOW library [24] to compute image descriptors in form of BoW representations. First, vocabularies are created based on the extracted SIFT features of the training area images. Afterwards, they are used to compute BoW representations of the images.

2) DML image retrieval: We train the DML method first on the Micro-GPS (PointGrey) dataset, and then jointly on the training areas of the HD Ground Database.

To avoid bias, a similar number of positive examples of overlapping ground images (with overlap of at least 20%), and negative pairs of non-overlapping images are considered.

B. Parameter optimization

The parameters of the localization methods are adapted by repeating two steps: (1) randomly sample a configuration from a defined parameter space, (2) if it has a higher success rate, or a similar success rate but a faster computation time than the previous best, perform a gradient descent like optimization by evaluating configurations with slightly adapted values. Using an E3-1270 Intel Xeon CPU at 3.8 GHz, we find texture-specific parameters for each main texture using their respective training areas, and a set of generalized parameters jointly optimized on all training areas.

Two of the most important parameters for the evaluated localization methods are the scale at which images are processed and the number of extracted features per image. Table III presents the texture-specific values that we obtained. For most combinations of texture and localization method, a much lower image resolution would suffice. On carpet, for example, best performance is reached using only 0.20 to 0.35 of our recording resolution of $0.1\,\mathrm{mm}$ per pixel. However, in other cases having an image scale of up to 0.88 of our native image scale is beneficial to the success rate.

TABLE III: Texture specific parameter settings for **R**ang**er**, **S**treet**M**ap, GTBL **SIFT**, and GTBL **RND**.

	Asphalt				Cobblestone				Carpet				Laminate			
	Rgr	SM	SIFT	RND	Rgr	SM	SIFT	RND	Rgr	SM	SIFT	RND	Rgr	SM	SIFT	RND
Image scale	0.20	0.20	0.60	0.20	0.88	0.38	0.34	0.23	0.20	0.20	0.35	0.23	0.28	0.70	0.20	0.15
#Features	400	200	600	3500	650	400	600	2200	350	600	600	2100	350	900	1100	2300

C. Results

We evaluate (initial) global localization, local localization (with available prior), and relative localization.

Our main performance metric is the localization success rate. Similar to Zhang et al. [4], we consider a pose estimate to be correct if it is confirmed by a second one, which is computed using the closest reference image to the original pose estimate. Here, the pose is estimated in the same way as for relative pose estimation during mapping. The original pose estimate is confirmed if both estimates are close to each other (less than $4.8\,\mathrm{mm}$ in distance and $1.5°$ in orientation). We store the confirming pose estimates as *ground truth* poses, which are required for localization with prior.

1) Global localization: We evaluate StreetMap and GTBL SIFT with their texture-specific parameters in the variants:

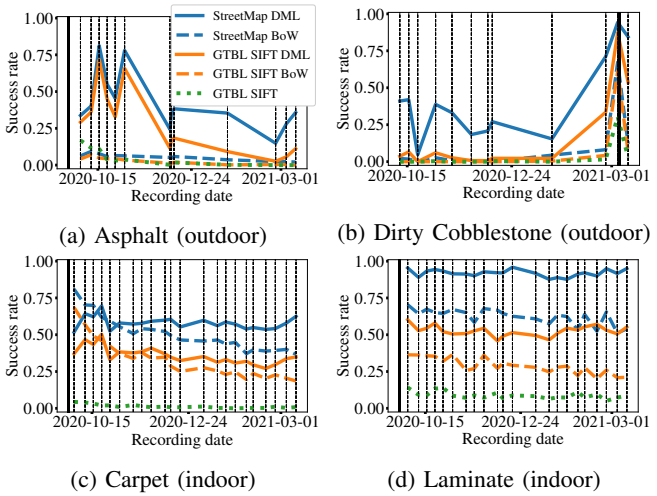
- **GTBL SIFT:** All reference images are considered.
- **StreetMap BoW** and **GTBL SIFT BoW:** Using BoW image retrieval, only the 15 reference images with most similar BoW representations are considered.
- **StreetMap DML** and **GTBL SIFT DML:** Using DML image retrieval, only the 15 reference images with most similar DML embeddings are considered.

Fig. 5 presents success rates on the **regular test sequences**. The success rate is relatively stable over time for both indoor textures, while it highly depends on the date of recording for the outdoor textures, with higher success rates closer to the date of mapping. Using image retrieval to reduce the number of considered reference images improves the success rates. StreetMap and GTBL SIFT mostly achieve their highest success rates with the DML image retrieval approach, while StreetMap reliably outperforms GTBL SIFT.

We observe lower success rates on the **cleaned** cobblestone area, compared to the area being recorded **without cleaning**. The mean success rate of GTBL SIFT changes from 8.1% to 7.7% on the cleaned area and with DML image retrieval it changes from 16.6% to 14.1% for GTBL SIFT, and from 40.9% to 33.7% for StreetMap. However, these changes are much smaller than the changes in success rate that occur through varying recording dates. Therefore, cleaning might have little influence and the lower performance on the cleaned variant might be explained by fluctuations that are to be expected when examining different parts of an area.

On the **wet** asphalt sequences, average success rates drop to 2% for all examined methods. It seems to be difficult to identify feature correspondences between the wet asphalt test images and the map that was recorded at dry condition.

We also evaluate GTBL SIFT without image retrieval for global localization on the **teach-and-repeat scenario**, using one sequence of images recorded while following a certain rope configuration as reference images, and the other sequences following the same rope configuration as query images. Here, global localization is easier, as sequences are recorded in quick succession, and the number of reference images is smaller. We observe mean success rates of 92.9% on cobblestone, 97.6% on carpet, 92.4% on laminate, and 95.1% on asphalt. For each texture, Fig. 6, illustrates a teach-and-repeat path and the corresponding localization results.



(a) Asphalt (outdoor) (b) Dirty Cobblestone (outdoor)

(c) Carpet (indoor) (d) Laminate (indoor)

Fig. 5: Global localization success rates. The thick vertical line highlights the date of map creation, while the dashed vertical lines show the recording dates of the test sequences. Note that for cobblestone the map was created later.

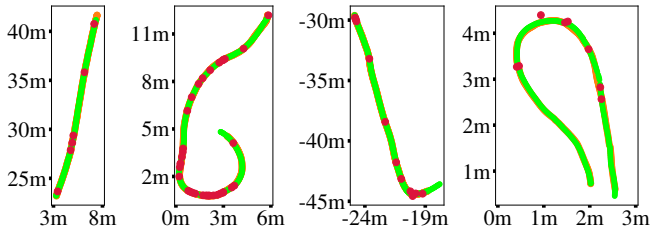


Fig. 6: Evaluation of GTBL SIFT for global localization on the teach-and-repeat scenario. From left to right: asphalt, cobblestone, carpet, and laminate. Orange dots present the true positions of the path that was driven during the *teach* phase. These are overlaid by the estimated positions of the *repeat* phase recordings. Here, green dots represent successful and red dots unsuccessful localization attempts. Positions on the plot axes correspond to metric map coordinates.

2) Localization with available prior: We examine the local localization performance of Ranger, StreetMap, GTBL SIFT, and GTBL RND that is achieved if an approximate

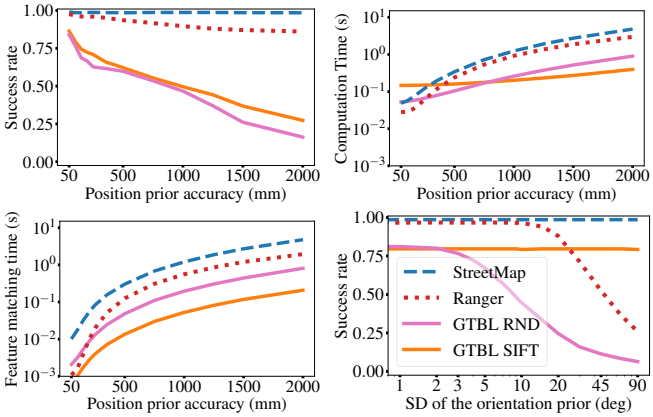


Fig. 7: Local localization results averaged over the main textures: success rate, overall computation time, and time used for feature matching with varying position prior accuracies, and success rate for varying orientation prior accuracies.

pose estimate is available as prior. The prior is generated by taking the ground truth pose of the query image, translating it with a specified distance d into a randomly sampled direction and rotating it with an orientation angle sampled from a zero-mean normal distribution. Depending on d, we adjust the number of considered closest reference images. Let a_P denote the possible area in which we are located, a_I the area covered by an image, and n_inc the number of images we expect each point on the ground to be included in. Then, we compute the number of considered closest images as:

$$\frac{a_\mathrm{P}}{a_\mathrm{I}} \cdot n_\mathrm{inc} = \frac{\pi d^2}{a_\mathrm{I}} \cdot n_\mathrm{inc} = \frac{\pi d^2}{0.12\,\mathrm{m} \cdot 0.16\,\mathrm{m}} \cdot 9. \qquad (6)$$

We perform three experiments. The first two are evaluated on the main textures with texture-specific parametrization, using the two regular test sequences with shortest time interval to mapping. Their results are presented in Fig. 7.

In our first experiment, the standard deviation (SD) of the orientation prior is set to $3.0°$, while we vary the position prior accuracy d between 50 and $2000\,\mathrm{mm}$. StreetMap achieves very high success rates, independently of the position prior accuracy. For Ranger, success rates decrease slowly for d values above $250\,\mathrm{mm}$. GTBL SIFT and GTBL RND achieve lower success rates and suffer more from inaccurate priors. We observe for small numbers of considered reference images that the overall computation time is dominated by the required time for feature extraction, while it is dominated by the time for feature matching for larger numbers of considered reference images. This is why, GTBL SIFT is slow for accurate position priors. For less accurate position priors, the GTBL methods have an advantage using the identity matching technique instead of linear feature matching.

In a second experiment, we fix d to $100\,\mathrm{mm}$ and vary the orientation prior SD between $1°$ and $90°$. StreetMap and GTBL SIFT determine feature orientations on their own and are not affected by this. But Ranger and GTBL RND use the orientation prior as orientation of the query features.

TABLE IV: Local localization success rates with jointly optimized parameters on the main and generalization textures.

Texture type	Ranger	StreetMap	GTBL SIFT	GTBL RND
Main	0.964	0.986	0.888	0.696
Generalization	0.988	0.898	0.700	0.528

TABLE V: Relative loc. success rates with jointly optimized parameters on the main and generalization textures.

Texture type	Ranger	StreetMap	GTBL SIFT	GTBL RND
Main	0.947	0.948	0.830	0.926
Generalization	0.976	0.956	0.485	0.624

Finally, we assess the **generalization capabilities** of the methods, using the jointly optimized parameters. Table IV presents the success rate on the main textures as well as on the six generalization textures with a position prior accuracy of $100\,\mathrm{mm}$ and an orientation prior SD of $3°$. For comparison, the corresponding average success rates on the main textures using the texture-specific parameters are 0.961 for Ranger, 0.986 for StreetMap, 0.775 for GTBL SIFT, and 0.716 for GTBL RND. This means that the performance is similar for Ranger, StreetMap, and GTBL RND, using the jointly optimized parameters, while it improved significantly for StreetMap, which could mean that its texture-specific parametrization overfitted to the training areas. On the generalization textures, Ranger performs better as on the main textures, while the others perform worse, suggesting that Ranger has the best generalization capabilities.

3) Relative localization: We perform a similar evaluation as that for the third experiment of local localization, but query image poses are estimated in respect to their predecessor image of the sequence, which is projected onto its ground truth position. Therefore, we can evaluate the localization success rate, and, since we know the actual poses of the query images, also the translation and orientation errors. We observe an average movement between consecutive images of $92.2\,\mathrm{mm}$ and $3.1°$. Success rates are presented in Table V.

Again, the GTBL methods perform better on the main textures than on the generalization textures. However, GTBL RND has higher success rates than for local localization, while the success rates of Ranger and StreetMap are similar. On the other hand, when comparing the average displacement errors of successful localization attempts, we observe with $0.77\,\mathrm{mm}$ a larger value for GTBL RND, than for StreetMap ($0.34\,\mathrm{mm}$), Ranger ($0.38\,\mathrm{mm}$), and GTBL SIFT ($0.39\,\mathrm{mm}$).

V. CONCLUSION

We introduced the HD Ground Database, on which we evaluated four state-of-the-art localization methods in various scenarios. For the first time, we systematically evaluate the difficulty of localization with increasing time discrepancy between mapping and localization, which we identify as a major challenge on outdoor areas. Also for the first time, we evaluate teach-and-repeat, which we find to be a simpler problem that may be sufficient in many practical scenarios.

REFERENCES

[1] A. Kelly, B. Nagy, D. Stager, and R. Unnikrishnan, "Field and service applications - an infrastructure-free automated guided vehicle based on computer vision - an effort to make an industrial robot vehicle that can operate without supporting infrastructure," *IEEE Robotics and Automation Magazine (RAM)*, vol. 14, no. 3, pp. 24–34, Sept 2007.

[2] K. C. Kozak and M. Alban, "Ranger: A ground-facing camera-based localization system for ground vehicles," in *IEEE/ION Position, Location and Navigation Symposium (PLANS)*, April 2016, pp. 170–178.

[3] X. Chen, A. S. Vempati, and P. Beardsley, "StreetMap - mapping and localization on ground planes using a downward facing camera," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 1672–1679.

[4] L. Zhang, A. Finkelstein, and S. Rusinkiewicz, "High-precision localization using ground texture," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6381–6387.

[5] J. F. Schmid, S. F. Simon, and R. Mester, "Ground texture based localization using compact binary descriptors," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 1315–1321.

[6] ——, "Ground texture based localization: Do we need to detect keypoints?" in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 4575–4580.

[7] S. Nakashima, T. Morio, and S. Mu, "AKAZE-based visual odometry from floor images supported by acceleration models," *IEEE Access*, vol. 7, pp. 31 103–31 109, 2019.

[8] M. Zaman, "High precision relative localization using a single camera," in *IEEE International Conference on Robotics and Automation (ICRA)*, April 2007, pp. 3908–3914.

[9] J. F. Schmid, S. F. Simon, and R. Mester, "Features for ground texture based localization - a survey," in *Proceedings of the British Machine Vision Conference (BMVC)*, 2019.

[10] J. Xue, H. Zhang, K. Dana, and K. Nishino, "Differential angular imaging for material recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 6940–6949.

[11] J. Rodriguez and D. Castano-Cano, "SimSLAM 2D: A simulation framework for testing and benchmarking of two-dimensional visual-SLAM methods," in *International Conference on Advanced Robotics (ICAR)*, 2019, pp. 141–147.

[12] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *IEEE European Conference on Computer Vision (ECCV)*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 404–417.

[13] D. Galvez-L'opez and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, Oct 2012.

[14] R. Radhakrishnan, J. F. Schmid, R. Scholz, and L. Schmidt-Thieme, "Deep metric learning for ground images," *arXiv preprint arXiv:2109.01569*, 2021.

[15] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *IEEE International Conference on Computer Vision (ICCV)*, Nov 2011, pp. 2564–2571.

[16] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *IEEE European Conference on Computer Vision (ECCV)*, 2010, pp. 778–792.

[17] M. Agrawal, K. Konolige, and M. R. Blas, "CenSurE: Center surround extremas for realtime feature detection and matching," in *IEEE European Conference on Computer Vision (ECCV)*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 102–115.

[18] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision (IJCV)*, vol. 60, no. 2, pp. 91–110, Nov 2004.

[19] G. Levi and T. Hassner, "LATCH: Learned arrangements of three patch codes," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016, pp. 1–9.

[20] H. Surmann, A. Bredenfeld, T. Christaller, R. Frings, U. Petersen, and T. Wisspeintner, "The volksbot," in *Workshop Proceedings of SIMPAR*, 2008, pp. 551–561.

[21] S. Agarwal, K. Mierle, and Others, "Ceres solver," http://ceres-solver.org.

[22] L. Zhang, A. Finkelstein, and S. Rusinkiewicz, "High-precision localization using ground texture," *CoRR*, vol. abs/1710.10687, 2017.

[23] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 22, no. 12, pp. 1349–1380, 2000.

[24] R. Muñoz-Salinas and R. Medina-Carnicer, "UcoSLAM: simultaneous localization and mapping by fusion of keypoints and squared planar markers," vol. 101, 2020, paper 107193.