

# Features for Ground Texture Based Localization - A Survey

Jan Fabian Schmid<sup>12</sup>  
SchmidJanFabian@gmail.com  
Stephan F. Simon<sup>1</sup>  
Stephan.Simon@de.bosch.com  
Rudolf Mester<sup>2</sup>  
mester@vsi.cs.uni-frankfurt.de

<sup>1</sup> Robert Bosch GmbH  
Hildesheim, Germany  
<sup>2</sup> VSI Lab  
Goethe University  
Frankfurt, Germany

---

## Abstract

Ground texture based vehicle localization using feature-based methods is a promising approach to achieve infrastructure-free high-accuracy localization. In this paper, we provide the first extensive evaluation of available feature extraction methods for this task, using separately taken image pairs as well as synthetic transformations. We identify AKAZE, SURF and CenSurE as best performing keypoint detectors, and find pairings of CenSurE with the ORB, BRIEF and LATCH feature descriptors to achieve greatest success rates for incremental localization, while SIFT stands out when considering severe synthetic transformations as they might occur during absolute localization.

## 1 Introduction

Highly accurate localization capabilities are required to enable the use of autonomous robots and vehicles. Available solutions such as GPS for outdoor applications are not able to reliably provide accurate positioning in urban environments, and systems for indoor applications such as Ultra Wideband require installation of costly infrastructure [10, 11, 13]. Visual localization using environmental landmarks can achieve centimeter precise localization in some indoor applications, but might suffer from occlusions and can deviate meters from the correct position in outdoor scenarios [26]. Localization approaches based on ground texture using a downward-facing camera, on the other hand, present promising results for reliable centimeter precise localization [10, 40]. Suitable texture types like concrete, asphalt, or carpet are prevalent and remain sufficiently stable in most application areas of autonomous agents [15, 40]. Therefore, ground texture based solutions have the potential to enable infrastructure-free high-accuracy localization. Furthermore, they enable localization in environments without static landmarks and can help to avoid privacy issues of household robots.

State-of-the-art methods use feature-based localization [10, 11, 28, 34] that relies on extraction of similar features from varying views of the same location. While several feature extraction methods were evaluated in these works, our survey is an extension. We evaluate additional methods for feature detection (e.g. AKAZE [9] and LIFT [37]) and description (e.g. DAISY [35] and LATCH [19]), and we consider different techniques for keypoint selection (non-maximum suppression (NMS), adaptive NMS, and bucketing).

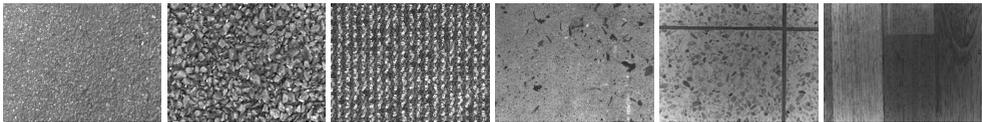


Figure 1: Examples of [40]: fine asphalt, coarse asphalt, carpet, concrete, tiles, and wood.

This paper contributes an extensive survey using an elaborate evaluation framework for ground texture based localization performance. For this purpose, we examine relevant synthetic transformations of ground images, perform pose estimation in respect to separately taken ground images, and introduce appropriate performance metrics to evaluate keypoint detector performance on ground images. Section 2 presents the localization task based on ground texture and introduces the evaluated methods. Then, Section 3 summarizes other surveys of features for ground texture. Sections 4 and 5 describe and evaluate our experiments.

## 2 Ground texture based localization

Ground texture based localization builds on the observation that image patches of ground texture can be used as fingerprint-like identifiers [40]. For most application areas, it is reasonable to assume that the ground is flat and therefore that the distance to the ground is known. Accordingly, with a downward-facing camera, pose estimation is reduced to determine two coordinates for the position and one orientation angle. This corresponds to a standard Euclidean transformation of rotation and translation in two dimensions.

Ground texture based localization can be performed with *appearance-based approaches* [6, 15, 27, 58], e.g. using normalized cross-correlation to find reoccurring image patches, and with *feature-based approaches* that find feature correspondences [11, 17, 28, 54, 40]. Furthermore, localization methods can be divided into map-based *absolute localization* and *incremental localization* for visual odometry [12]. Incremental localization can be performed both with appearance-based [6, 58] and with feature-based approaches [28, 54]. If a sufficiently accurate localization prior is available, appearance-based approaches can be used for absolute localization [15, 27]. However, using image features is potentially more robust to natural degradation of ground texture [17] and can work without localization prior [11, 40].

*Features* are used to describe characteristic image regions [30]. Feature positions are represented by their *keypoints*. We use the term *keypoint object*, if in addition to the position a size and possibly an orientation is included. In addition to its keypoint object, features are represented by their *descriptor*, which describes the local environment of a keypoint.

Feature-based localization can be divided into 5 subtasks:

1. **Detection:** Find the same keypoint objects from different viewpoints and under varying photometric conditions like illumination, noise, and blur.
2. **Selection:** Select a certain number of keypoint objects for further processing.
3. **Description:** Compute descriptors that robustly take similar values for corresponding keypoint objects, and distinctively different values for non-corresponding ones.
4. **Matching:** Propose correspondences between the features found in the current camera image and previously found reference features.
5. **Pose estimation:** Based on the proposed correspondences, estimate the current pose.

For the first three tasks, we examine a range of popular approaches available in OpenCV [8],

as well as LIFT, a state-of-the-art deep learning approach [57]. For matching and pose estimation, we revert to standard techniques. For matching, we compute the Euclidean distance for real-valued descriptors and the Hamming distance for binary ones. Then, features are matched with linear matching and the *ratio test constraint* as suggested by Lowe [20]. This means that for each feature descriptor from the test image the two closest reference descriptors are found. The closest one is suggested as a match if it is significantly closer than the second one. Finally, we estimate the relative poses of test images using the proposed feature matches and RANSAC-based estimation of a Euclidean transformation.

## 2.1 Evaluated keypoint detectors

Keypoint detection approaches can be split into corner detectors and scale-space detectors [1]. Corners mark suitable keypoints as they tend to be robust to view changes. The **Harris** detector [14] and Good Features To Track (**GFTT**) [53] approximate the second derivative of the sum-of-squared-differences with respect to the shift of a circular image patch to detect edges and corners. **FAST** [53] compares intensities of center pixels with their surrounding pixels on a circle. A corner is detected if the circle contains a contiguous sequence of pixels with significantly larger or lower intensity values. If this condition can no longer be fulfilled it can be rejected early. To do this, a decision tree defines the order of comparisons. Mair *et al.* [21] adapt this concept for **AGAST**. Instead of using a single decision tree, they switch between multiple ones according to observed local image characteristics.

Scale-space detectors exploit image scale pyramids to find scale invariant keypoints. Mikolajczyk and Schmid [23] extended with **Harris Laplace** the Harris corner detector to search for corners in multiple scales using a Gaussian scale-space. **SIFT** [24] detects blobs using a Difference-of-Gaussian (DoG) pyramid as local minima and maxima of the intensity values in scale and space. Candidates located on edges or with low contrast are suppressed. Orientation is determined by the dominant local intensity gradients. **SURF** [7] and **CenSurE** [1] approximate the DoG with bi-level Laplacian of Gaussian like Difference-of-Boxes (DoB) or Difference-of-Octagons (DoO), which can be computed efficiently using integral images. While SIFT and SURF find keypoints using the Hessian measure, CenSurE relies on the Harris corner response. **BRISK** [18] and **ORB** [52], on the other hand, use efficient corner detectors like FAST on a scale pyramid to identify repeatable keypoints in scale-space. Alcantarilla *et al.* [8, 9] argue that Gaussian scale-space pyramids and its approximations do not only remove noise, but interesting image details as well. Therefore, they suggest with **AKAZE** to find keypoints as maxima of the Hessian in non-linear scale-space.

For **MSER** [22] the image is thresholded by an increasing illumination value. Regions with illumination values below the threshold emerge and grow during this process. Keypoint objects are identified as regions at their point of slowest growth. In **MSD** [56] image regions that differ from their surrounding in a large neighborhood are considered as keypoint objects.

## 2.2 Evaluated keypoint selection methods

For feature-based localization it is necessary to extract a certain number of keypoints even on weakly textured images. Detection parameters should be chosen with respect to this case or need to be adapted texture dependently. Parameterizing detectors to be able to deal with feature-poor images is difficult. This problem is emphasized for ground texture images as appearance and frequency of features vary dependent on the type of ground. Still, using a

constant set of parameters is desirable, but results in large numbers of keypoints on feature-rich textures. Therefore, in order to limit the required processing time for localization, keypoint selection has an important role for feature extraction on ground texture images. One approach to keypoint selection is Non-Maximum Suppression (NMS): only the  $N$  keypoints with largest response are kept. In order to improve the spatial distribution of keypoints NMS can be combined with **bucketing**, where keypoints are detected independently for areas defined by a regular grid [16]. An alternative approach is adaptive non-maximum suppression (ANMS), where keypoints with strong response suppress keypoints in a local neighborhood.

## 2.3 Evaluated feature description methods

Historically, feature descriptors are real-valued. **SIFT** [10] describes keypoint objects using a histogram of gradient directions. Similarly, **DAISY** [15] uses quantized orientation histograms. However, histogram bins are distributed radially around the keypoint and smoothed increasingly with the distance to the keypoint. **SURF** [7] relies on Haar-Wavelet responses that are efficiently to compute using integral images.

More recently, research started to focus on the more compact binary descriptors. Most of them construct descriptors as concatenated results of pairwise intensity comparisons. **BRIEF** [9] compares randomly paired pixels from a smoothed image patch. **ORB** [12] employs a training algorithm to determine a set of pixel comparisons and rotates this pattern according to the keypoint object orientation. **BRISK** [13] samples pixel pairs around the keypoint. While short-distance pairs are evaluated for the descriptor, long-distance pairs are used to determine an orientation. A similar approach is employed by **FREAK** [2], but appropriate intensity comparisons are found in a training process. In **LATCH** [19] triplets of image patches are compared to each other instead of pixel pairs to increase robustness. **AKAZE** [8] performs pairwise comparisons of first-order gradients.

Most recently, deep learning approaches for feature extraction are developed. We examine **LIFT** [17], which is a state-of-the-art method that provides solutions for keypoint detection, orientation estimation, and feature description. The authors construct an end-to-end trainable Siamese network, which is trained with features from a Structure-from-Motion (SfM) algorithm. However, in practice they train the network separately for the three tasks. Training samples consists of four image patches, two corresponding ones for which the network learns to produce similar output and two other patches that should result in distinctively different network outputs.

## 3 Surveys of features for ground texture images

Zhang *et al.* [11] evaluate the use of SIFT, SURF, ORB, and HardNet [14] for their ground texture based localization pipeline. The authors receive the best results for keypoint regions and descriptors provided by SIFT. In a follow-up paper [18], the authors develop a fully convolutional neural network trained on ground texture images that achieves higher repeatability than SIFT, but has increased computational cost.

Kozak *et al.* [10] evaluate combinations of detector and descriptor methods on pairs of partially overlapping ground texture images, measuring the number of correctly matched keypoints. They find the combination of CenSurE keypoints and SIFT descriptors to lead to the largest number of successfully matched features. Pairings of CenSurE with ORB descriptors, as well as SIFT detector with SIFT descriptor, also show good performance.

Task	Transformation	Performance metrics
Keypoint detection	Synthetic	Repeatability, Ambiguity, < N KPs
Feature matching	Synthetic	Number of correct matches, Precision
Pose estimation	Synthetic & Real	Success rate

Table 1: Experimental setups.

FAST, SURF, and GFTT keypoints, as well as descriptors provided by BRISK, FREAK, or SURF, present significant weaknesses for at least one of the three evaluated road surface texture types: worn asphalt, dark asphalt, and concrete.

Otsu *et al.* [29] investigate the suitability of different keypoint detectors for visual odometry from ground texture. They evaluate Harris, GFTT, and FAST corner detectors as well as the scale-space detectors SIFT, SURF, and CenSurE. The authors identify that none of the detectors is suited for all situations that occur in the employed desert landscape image datasets. Therefore, they propose to switch between detectors dependent on the terrain.

This survey extends the prior work. We evaluate keypoint detection separately like in [29, 39], but also pair them with varying selectors and descriptors. In addition to the number of correctly matched keypoints [4], we evaluate the repeatability of keypoints and their spatial distribution, the precision of feature matches, and the pose estimation success rate. In comparison to [4], we evaluate a larger variety of methods for detection and description. Furthermore, we evaluate performance on synthetically transformed images as well as on separately taken images. We evaluate sequentially taken image pairs as they occur during incremental localization, where the transformation is close to a pure translation, and image pairs taken at different times from independent views as they occur for absolute localization.

## 4 Experimental setups

Our experimental framework of three setups is summarized in Table 1. The first experiment examines keypoint detection on synthetic transformations, the second one feature matching on synthetically transformed images and the third one pose estimation using both synthetic transformations and separately recorded, partially overlapping ground texture image pairs.

For synthetic transformations, correct feature matches are known and performance can be evaluated in regard to specific types of image modifications. We evaluate geometric and photometric transformations. Typical photometric transformations to consider are Gaussian noise and illumination changes. The noise is independent and identically distributed (i.i.d.) and zero-mean. For illumination changes, we employ gamma correction: pixel values  $g$  are modified as:  $g_{\text{out}} = \text{round}(g_{\text{max}} \cdot (\frac{g_{\text{in}}}{g_{\text{max}}})^\gamma)$ , where  $g_{\text{max}} = 255$ . Additionally, two geometric transformations are relevant when using downward-facing cameras: rotation and translation. Rotated images are computed using bicubic interpolation. In case of translation, an image mask determines a section of the image from which features are extracted. This mask is translated for testing. Accordingly, different image sections are evaluated. For evaluation only keypoints from the intersection between reference mask and test mask are considered.

For separately taken images, it is difficult to obtain sufficiently accurate ground truth in order to determine which feature matches are correct. However, it allows us to examine localization performance with its difficulties that occur during application in the real world. We examine image pairs that are recorded in direct sequence, which represent the challenges of incremental localization, and we examine image pairs that have been recorded at different times and from independent views, which represent the challenges of absolute localization.

## 4.1 Keypoint detection

We use synthetic transformations to examine whether the same keypoint objects are found in reference and test image. Pairs of keypoint objects from reference and test image are considered to match and therefore to represent the same location if their *Intersection over Union* (IoU) in the reference coordinate frame is greater 0.5. As performance metrics, we evaluate keypoint **repeatability** introduced by Mikolajczyk *et al.* [24]. It measures the proportion of keypoints from the test image that were also found in the reference image. Additionally, we introduce two novel performance metrics. **Ambiguity** addresses the problem of repeatability that it does not penalize ambiguous keypoint correspondences. This problem occurs if a keypoint object from the test image has multiple matches in the reference image, which happens if keypoints are clustered. We compute ambiguity as the mean number of matches of test image keypoint objects with at least one match. Therefore, ambiguity  $\geq 1.0$ . An ambiguity greater 1.0 suggests that the repeatability score is inflated by ambiguous keypoint matches. The second new metric, **< N KPs**, shows how often fewer than  $N$  keypoints are found, as having only few keypoints increases the risk of failure for feature-based localization.

## 4.2 Feature matching

In order to evaluate whether the obtained features are suited for the localization pipeline, we examine feature matching performance. We evaluate the **number of correctly matched features** and compute the the matching **precision**:  $\text{precision} = \frac{\#\text{correct matches}}{\#\text{correct matches} + \#\text{incorrect matches}}$ .

## 4.3 Pose estimation

Pose estimates are considered correct if their distance to the *ground truth* is less than 30 pixels (4.8 mm) and if their orientation error is less than  $\pm 1.5$  degrees (these thresholds are adopted from Zhang *et al.* [40]). We evaluate pose estimation performance using the **success rate** metric, which is computed as the ratio of correct pose estimates to incorrect ones.

# 5 Evaluation

For evaluation of our experimental framework, we use the ground texture image database of Zhang *et al.* [40]. We test on all six ground texture types captured by a gray-scale camera (see Figure 1). Images have a size of 1288 by 964 pixels. We select 3 images per texture to be used exclusively for parameter optimization, 100 for synthetic transformations, and 100 image pairs each for incremental and absolute localization tasks. We observed no significant performance variations using more test images. Our strategies for parameter optimization, and the obtained parameter settings can be found in the *supplementary material*.

We make use of OpenCV 4.0 [8] implementations for most of the evaluated detectors and descriptors. Due to bad performance of the ORB implementation of OpenCV, we use its implementation that comes with ORB-SLAM2 [26]. The implementation and the trained network weights of LIFT are provided by the authors, which claim to achieve good generalization performance even without domain specific training samples [57]. We exclude ORB and LIFT from the evaluation on synthetic transformations as they do not allow to define a detection mask. For feature matching, we find most similar reference descriptors and filter them with the ratio test constraint with a threshold of 0.7. Poses are estimated using

RANSAC-based estimation of a Euclidean transformation with 2000 iterations and the error threshold applied in [40] of 3.0. We observed better localization performance estimating not only the three obligatory parameters for position and orientation, but also an additional scale parameter allowing for small variations in height.

Synthetic transformations are parametrized as follows: for rotation, angles between 0 and 180 degrees; for translation, the detection mask of the test image is pushed in direction of the lower right image corner (in respect to the reference mask), the resulting IoUs of reference mask and test mask are between 0.2 and 1.0. Gaussian i.i.d. noise is zero-mean with standard deviation between 0.0 and 40.0; illumination values are changed using a gamma between 0.1 and 3.0. When presenting results from synthetic transformations, metrics are averaged with equal contribution of the results of each transformation type. Transformation and texture dependent results are presented in the *supplementary material*.

## 5.1 Evaluation of selector-detector pairings

We examine the repeatability of keypoint detectors using the keypoint selection methods introduced in Section 2.2 to reduce the number of keypoints to 1000. Respectively, if the keypoint detection method allows to specify the desired number of keypoints, we set this parameter to 1000. For ANMS, we use Suppression via Square Covering (SSC) [8] with a tolerance of 20%. For bucketing, we received good results for non-square buckets, using a grid of 8 rows and 6 columns. For each grid cell 21 keypoints are selected using NMS resulting in a maximum of 1008 keypoints per image. Evaluating a reference image without selection with all our synthetic transformations takes us several days, which is why for this experiment we evaluate a single test image per type of ground texture. In addition to the repeatability scores, Table 2 presents the average number of keypoints before selection. MSER does not provide a keypoint response measure, and is therefore not well suited to be used with a selection method. Together with FAST, AGAST, and BRISK, MSER has significantly better repeatability without selection. In order to select MSER keypoints, we use the order of extracted keypoints as substitution for the response measure. This means that the first found maximally stable extremal regions, which are the ones with lowest intensity values, are considered to have the largest response. With this work-around MSER still achieves surprisingly large repeatability of 51% using NMS and 73% using ANMS. We find MSER to be the only detector that performs best with ANMS. For all other detectors, we use NMS in the following. The repeatability of SURF and especially MSD is increased when using keypoints selected by NMS instead of all available keypoints. This means that keypoints that have been assigned low values of the response measures of these detectors are indeed non-repeatable and are rightly removed by NMS.

In a next step, we evaluate the best performing detector-selector pairings using all 100 test reference images. Results are presented in Table 3. For the  $< N$  KPs metric, we set  $N$  to 100, as we noticed that localization success is low with fewer keypoints. For most detectors, we were able to find parameters that allow to retrieve at least 100 keypoints from almost all images. But, AKAZE, Harris Laplace, and GFTT still find fewer keypoints on at least 4% of the images. This problem occurs almost exclusively on wood texture. Harris Laplace extracts less than 100 keypoints on 49% of wood images, GFTT on 28% and AKAZE on 23%. SIFT, AKAZE, and SURF have with 83% to 84% the best repeatability. However, SIFT has a large ambiguity score of 1.5, as it detects multiple orientations for some keypoints.

Overall, our evaluation suggests that SURF and CenSurE, as well as AKAZE for non-wood texture, are the best detectors on ground texture images. They have among the best

Detector	#KPs before selection	Repeatability w/o selection	Rep. NMS	Rep. ANMS	Rep. Bucketing
AKAZE	10199	0.81	<b>0.82</b>	0.41	0.74
SIFT	755	0.82	<b>0.82</b>	<b>0.82</b>	0.77
SURF	7271	0.80	<b>0.82</b>	0.65	0.74
CenSurE	6434	0.83	<b>0.76</b>	0.39	0.70
MSD	6484	0.59	<b>0.76</b>	0.51	0.68
HarrisLaplace	839	0.76	<b>0.76</b>	<b>0.76</b>	0.68
MSER	15238	0.94	0.51	<b>0.73</b>	0.52
BRISK	58050	0.84	<b>0.71</b>	0.25	0.64
GFTT	894	0.69	<b>0.69</b>	<b>0.69</b>	0.29
AGAST	225361	0.93	<b>0.64</b>	0.22	0.57
FAST	52112	0.78	<b>0.64</b>	0.26	0.56

Table 2: Number of keypoints before selection and repeatability of selector-detector pairings.

Selector	Detector	< 100 KPs	Repeatability	Ambiguity	Computation time (s)
NMS	AKAZE	0.05	0.84	1.00	0.40
NMS	SIFT	0.00	0.83	1.50	3.27
NMS	SURF	0.01	0.83	1.06	0.48
NMS	CenSurE	0.00	0.79	1.00	0.08
NMS	MSD	0.00	0.79	1.04	5.38
NMS	HarrisLaplace	0.08	0.77	1.18	0.62
SSC	MSER	0.00	0.72	1.15	3.91
NMS	BRISK	0.01	0.74	1.44	1.08
NMS	GFTT	0.04	0.73	1.00	0.06
NMS	AGAST	0.00	0.68	2.08	0.14
NMS	FAST	0.01	0.67	1.01	0.04

Table 3: Evaluation of keypoint detectors on synthetically transformed images.

repeatability, and ambiguity scores, and are, unlike SIFT and MSD, fast to compute.

## 5.2 Evaluation of detector-descriptor pairings

We evaluate pose estimation success rate for all working detector-descriptor pairs. The AKAZE descriptor only allows to use AKAZE keypoints. DAISY requires keypoint objects to specify orientation. The ORB descriptor has requirements on the keypoint scaling, which excludes SIFT and LIFT. We evaluate on image pairs from incremental localization (Table 4), as well as on the image pairs from absolute localization (Table 5). The intersection of the sequentially taken image pairs is on average 22.7%. Some almost non-overlapping pairs with intersections as low as 1.7% are particularly challenging. The image pairs from the absolute localization tasks have larger intersections with an average of 43.7%. However, in this case the rotation between the images is with an average of 120 degrees higher as for the pairs from incremental localization with an average rotation of 3 degrees. Again, the number of retrieved keypoints is reduced to 1000 using the best selection method. The best performance for incremental localization of 93% success rate is achieved with ORB on CenSurE or MSD keypoints. BRIEF and LATCH perform similarly well with 90% success rate.

For absolute localization most descriptors can achieve more than 90% success rate if paired with the right detector. SURF and DAISY are not quite as successful as they struggle with images of wooden texture. Detectors that provide orientation information (SIFT, SURF, AKAZE, ORB, BRISK, and LIFT) outperform the other detectors. ORB, BRIEF, LATCH, SIFT, and LIFT descriptors rely on available orientation information and perform poorly if it is missing. In these cases pose estimation success rate drops to about 10% to 15%.

For further analysis of feature descriptor performance, we use synthetic transformations

Detector	Descriptor									
	ORB	BRIEF	LATCH	SURF	SIFT	AKAZE	LIFT	BRISK	FREAK	DAISY
CenSurE	<b>0.93</b>	<b>0.90</b>	<b>0.90</b>	0.24	0.82	NA	<b>0.84</b>	0.80	0.70	NA
MSD	<b>0.93</b>	<b>0.90</b>	<b>0.90</b>	0.68	0.83	NA	<b>0.84</b>	<b>0.81</b>	0.69	NA
HarrisLaplace	0.88	0.86	0.82	0.75	0.60	NA	0.62	0.74	0.67	NA
SURF	0.89	0.86	0.86	<b>0.87</b>	0.73	NA	0.40	0.72	0.73	<b>0.72</b>
AKAZE	0.80	0.85	0.83	0.47	0.73	<b>0.84</b>	0.76	0.80	<b>0.74</b>	0.66
FAST	0.89	0.85	0.85	0.24	0.79	NA	0.77	0.76	0.66	NA
GFTT	0.88	0.84	0.84	0.21	0.80	NA	0.80	0.79	0.68	NA
LIFT	NA	0.84	0.83	0.36	0.69	NA	0.75	0.80	0.72	0.66
MSER	0.83	0.83	0.85	0.59	0.76	NA	0.47	0.62	0.56	NA
SIFT	NA	0.82	0.84	0.59	<b>0.84</b>	NA	0.61	0.70	0.63	0.69
AGAST	0.77	0.76	0.75	0.30	0.66	NA	0.61	0.64	0.55	NA
ORB	0.70	0.71	0.71	0.73	0.58	NA	0.12	0.49	0.62	0.47
BRISK	0.66	0.69	0.70	0.61	0.61	NA	0.38	0.71	0.66	0.48

Table 4: Pose estimation success rate on image pairs from incremental localization tasks, where images are taken in direct sequence.

Detector	Descriptor									
	ORB	BRIEF	LATCH	SURF	SIFT	AKAZE	LIFT	BRISK	FREAK	DAISY
CenSurE	0.14	0.11	0.09	0.50	0.10	NA	0.11	0.88	0.96	NA
MSD	0.14	0.11	0.09	0.72	0.09	NA	0.10	0.86	0.90	NA
HarrisLaplace	0.14	0.10	0.09	0.71	0.10	NA	0.10	0.87	0.89	NA
SURF	0.89	0.98	0.98	<b>0.87</b>	0.94	NA	0.91	<b>0.96</b>	<b>0.98</b>	0.70
AKAZE	<b>0.94</b>	<b>0.99</b>	<b>0.99</b>	0.73	0.96	<b>0.99</b>	<b>0.96</b>	0.92	<b>0.98</b>	<b>0.75</b>
FAST	0.14	0.11	0.09	0.46	0.10	NA	0.11	0.84	0.89	NA
GFTT	0.14	0.11	0.09	0.37	0.10	NA	0.11	0.86	0.89	NA
LIFT	NA	0.71	0.85	0.17	0.70	NA	0.52	0.46	0.73	0.24
MSER	0.13	0.10	0.09	0.76	0.09	NA	0.10	0.90	0.89	NA
SIFT	NA	0.97	<b>0.99</b>	0.75	<b>0.99</b>	NA	<b>0.96</b>	0.93	0.94	0.74
AGAST	0.12	0.10	0.09	0.46	0.08	NA	0.10	0.80	0.86	NA
ORB	0.82	0.86	0.89	0.84	0.90	NA	0.73	0.84	0.87	0.56
BRISK	0.79	0.88	0.89	0.76	0.84	NA	0.84	0.87	0.89	0.53

Table 5: Pose estimation success rate on image pairs from absolute localization tasks, where images are taken at different times and from varying perspectives.

to evaluate the pairings of detectors and descriptors that performed the best on absolute localization. In cases of multiple best performing detectors, we use the faster one. Table 6 presents results for feature matching and pose estimation. Additionally, we provide the results of BRIEF on CenSurE keypoints. We note that the challenges of our synthetic transformations, which include severe rotations and photometric modifications, are more similar to the ones of absolute localization. Accordingly, BRIEF has significantly better performance using AKAZE keypoint objects instead of CenSurE keypoint objects, which lack orientation information. SIFT outperforms the other feature extraction pipelines. We find precision to correlate with the pose estimation success rate, while this relation is not that clear for the number of correct matches. E.g. BRIEF on CenSurE has about 50 more correct matches than DAISY on AKAZE, despite having a significantly lower pose estimation success rate.

Table 7 presents success rates for consecutive image pairs using different numbers of reference image features. Our selection of  $N = 100$  for the  $< 100$  KPs metric is validated, as localization performance is low for 100 or less reference keypoints. On the other hand, pairings like CenSurE-ORB, CenSurE-LATCH, and SIFT-SIFT reach values close to their best performance at 300 features already. Others, like SURF-SURF and SURF-DAISY should not be used with less than 500 reference features. Furthermore, we find further evidence for our observation that the number of correct matches is not a suitable indicator for localization

Detector	Descriptor	Pose estimation success rate	#Correct matches	Precision
SIFT	SIFT	1.00	551	1.00
SURF	BRISK	0.99	559	1.00
AKAZE	AKAZE	0.98	545	0.99
AKAZE	BRIEF	0.98	534	0.99
AKAZE	FREAK	0.98	561	0.99
AKAZE	LATCH	0.97	538	0.98
SURF	SURF	0.97	509	0.99
AKAZE	DAISY	0.87	410	0.98
CenSurE	BRIEF	0.77	460	0.92

Table 6: Evaluation of detector-descriptor pairings on synthetically transformed images.

Detector	Descriptor	10	50	100	200	300	500	750	1000	1500
CenSurE	ORB	0.01	0.54	0.75	0.87	0.89	0.91	0.92	0.92	0.92
CenSurE	BRIEF	0.00	0.47	0.73	0.84	0.87	0.88	0.89	0.89	0.90
CenSurE	LATCH	0.13	0.65	0.77	0.84	0.86	0.88	0.88	0.89	0.89
SURF	SURF	0.01	0.25	0.36	0.56	0.69	0.83	0.84	0.86	0.86
SIFT	SIFT	0.12	0.56	0.71	0.80	0.82	0.83	0.84	0.83	0.84
CenSurE	LIFT	0.04	0.50	0.67	0.76	0.80	0.82	0.82	0.83	0.83
AKAZE	AKAZE	0.03	0.41	0.62	0.73	0.77	0.81	0.83	0.83	0.83
MSD	BRISK	0.03	0.28	0.46	0.69	0.74	0.79	0.81	0.80	0.80
AKAZE	FREAK	0.01	0.33	0.46	0.59	0.65	0.71	0.72	0.73	0.74
SURF	DAISY	0.00	0.16	0.19	0.31	0.45	0.65	0.70	0.71	0.71

Table 7: Pose estimation success rates for varying numbers of reference features.

performance. While the number of RANSAC inliers (and therefore the number of correct matches) increases with the number of reference features, the localization success rates stagnate at some point. For CenSurE-BRIEF, for example, the number of inliers in successful localization attempts increases from about 35 at 300 reference features to 94 for 1500 reference features, even though success rate increases only from 87% to 90%. Once a certain number of correct matches is available, localization performance does not increase further.

## 6 Conclusions

We examined keypoint detectors, selection methods, and feature descriptors on synthetically transformed ground images as well as on pairs of separately taken ground images.

In contrast to Otsu *et al.* [24], we find with SURF and CenSurE keypoint detectors that are well suited for all evaluated ground textures. For image pairs where the transformation consists mainly of a translation, as it is the case for the task of incremental localization, we can confirm the suitability of ORB descriptors on CenSurE keypoints as well as SIFT features, and the weaknesses of FAST and GFTT keypoints, and BRISK and FREAK descriptors, as assessed by Kozak and Alban [17]. This is even though our evaluation has shown that their metric, the number of correctly matched features, is not necessarily a good indicator for localization performance. SURF, on the other hand, has shown good performance for us. Finally, we validated the observation of Zhang *et al.* [10] that SIFT is suited for absolute localization as it is among the best performing methods for the estimation of transformations between image pairs that have been taken at different times and perspectives, and the best feature extractor to deal with even more severe synthetic transformations. However, other pairings like BRIEF, LATCH, and AKAZE descriptors on AKAZE keypoints perform similarly well and are significantly faster to compute. For further research, we are interested in absolute localization performance using different matching and pose estimation methods.

## References

- [1] 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)*, pages 102–115, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
- [2] A. Alahi, R. Ortiz, and P. Vanderghenst. FREAK: Fast Retina Keypoint. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 510–517, June 2012.
- [3] P. F. Alcantarilla, A. Bartoli, and A. J. Davison. KAZE Features. In *IEEE European Conference on Computer Vision (ECCV)*, pages 214–227, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.
- [4] P. F. Alcantarilla, J. Nuevo, and A. Bartoli. Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2013.
- [5] M. O. A. Aqel, M. H. Marhaban, M. I. Saripan, N. Bt. Ismail, and A. Khmag. Optimal Configuration of a Downward-Facing Monocular Camera for Visual Odometry. *Indian Journal of Science and Technology*, 8(32), 2016.
- [6] O. Bailo, F. Rameau, K. Joo, J. Park, O. Bogdan, and I. S. Kweon. Efficient adaptive non-maximal suppression algorithms for homogeneous spatial keypoint distribution. *Pattern Recognition Letters*, 106:53 – 60, 2018.
- [7] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded Up Robust Features. In *IEEE European Conference on Computer Vision (ECCV)*, pages 404–417, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
- [8] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
- [9] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. BRIEF: Binary Robust Independent Elementary Features. In *IEEE European Conference on Computer Vision (ECCV)*, pages 778–792, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- [10] 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)*, pages 1672–1679, Oct 2018.
- [11] M. Cornick, J. Koechling, B. Stanley, and B. Zhang. Localizing Ground Penetrating RADAR: A Step Toward Robust Autonomous Ground Vehicle Localization. *Journal of Field Robotics*, 33(1):82–102, 2016.
- [12] G. N. Desouza and A. C. Kak. Vision for mobile robot navigation: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 24(2):237–267, Feb 2002.
- [13] H. Fang, M. Yang, R. Yang, and C. Wang. Ground-Texture-Based Localization for Intelligent Vehicles. *IEEE Transactions on Intelligent Transportation Systems (ITS)*, 10(3):463–468, Sept 2009.

- [14] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference (AVC)*, volume 15, pages 147–151, 01 1988.
- [15] 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*, 14(3):24–34, Sept 2007.
- [16] B. Kitt, A. Geiger, and H. Lategahn. Visual odometry based on stereo image sequences with RANSAC-based outlier rejection scheme. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 486–492, June 2010.
- [17] 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)*, pages 170–178, April 2016.
- [18] S. Leutenegger, M. Chli, and R. Y. Siegwart. BRISK: Binary Robust invariant scalable keypoints. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2548–2555, Nov 2011.
- [19] G. Levi and T. Hassner. LATCH: Learned arrangements of three patch codes. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9, 2016.
- [20] David G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, Nov 2004.
- [21] E. Mair, G. D. Hager, D. Burschka, M. Suppa, and G. Hirzinger. Adaptive and Generic Corner Detection Based on the Accelerated Segment Test. In *IEEE European Conference on Computer Vision (ECCV)*, pages 183–196, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- [22] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761 – 767, 2004.
- [23] K. Mikolajczyk and C. Schmid. An Affine Invariant Interest Point Detector. In *IEEE European Conference on Computer Vision (ECCV)*, pages 128–142, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.
- [24] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A Comparison of Affine Region Detectors. *International Journal of Computer Vision (IJCV)*, 65(1):43–72, Nov 2005.
- [25] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas. Working hard to know your neighbor’s margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems*, pages 4826–4837, 2017.
- [26] R. Mur-Artal and J. D. Tardós. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, Oct 2017.
- [27] I. Nagai and K. Watanabe. Path tracking by a mobile robot equipped with only a downward facing camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6053–6058, Sept 2015.

- [28] S. Nakashima, T. Morio, and S. Mu. AKAZE-Based Visual Odometry From Floor Images Supported by Acceleration Models. *IEEE Access*, 7:31103–31109, 2019.
- [29] K. Otsu, M. Otsuki, G. Ishigami, and T. Kubota. *An Examination of Feature Detection for Real-Time Visual Odometry in Untextured Natural Terrain*, pages 405–414. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [30] W. K. Pratt. *Digital image processing: PICS Scientific inside*, volume 4. Wiley-interscience Hoboken, New Jersey, 2007.
- [31] E. Rosten and T. Drummond. Machine Learning for High-Speed Corner Detection. In *IEEE European Conference on Computer Vision (ECCV)*, pages 430–443, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.
- [32] 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)*, pages 2564–2571, Nov 2011.
- [33] J. Shi and C. Tomasi. Good Features to Track. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593 – 600, January 1994.
- [34] A. J. Swank. Localization Using Visual Odometry and a Single Downward-Pointing Camera. 2012.
- [35] E. Tola, V. Lepetit, and P. Fua. DAISY: An Efficient Dense Descriptor Applied to Wide-Baseline Stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32(5):815–830, May 2010.
- [36] F. Tombari and L. Di Stefano. Interest Points via Maximal Self-Dissimilarities. In *Asian Conference on Computer Vision (ACCV)*, pages 586–600, Cham, 2014. Springer International Publishing.
- [37] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua. LIFT: Learned Invariant Feature Transform. In *IEEE European Conference on Computer Vision (ECCV)*, pages 467–483, 2016.
- [38] M. Zaman. High Precision Relative Localization Using a Single Camera. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3908–3914, April 2007.
- [39] L. Zhang and S. Rusinkiewicz. Learning to Detect Features in Texture Images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [40] L. Zhang, A. Finkelstein, and S. Rusinkiewicz. High-Precision Localization Using Ground Texture. In *IEEE International Conference on Robotics and Automation (ICRA)*, May 2019.