CrowdDriven: A New Challenging Dataset for Outdoor Visual Localization

Ara Jafarzadeh1 Manuel López Antequera2 Pau Gargallo2 Yubin Kuang2 Carl Toft1 Fredrik Kahl1 Torsten Sattler3
1Chalmers University of Technology 2Facebook 3Czech Technical University in Prague

Abstract

Visual localization is the problem of estimating the position and orientation from which an image (or a sequence of images) is taken in a known scene. It is an important part of a wide range of computer vision and robotics applications, from self-driving cars to augmented/virtual reality systems. Visual localization techniques should work reliably and robustly under a wide range of conditions, including seasonal, weather, illumination and man-made changes. Recent benchmarking efforts model this by providing images under different conditions, and the community has made rapid progress on these datasets since their inception. However, they are limited to a few geographical regions and often recorded with a single device. We propose a new benchmark for visual localization in outdoor scenes, using crowd-sourced data to cover a wide range of geographical regions and camera devices with a focus on the failure cases of current algorithms. Experiments with state-of-the-art localization approaches show that our dataset is very challenging, with all evaluated methods failing on its hardest parts. As part of the dataset release, we provide the tooling used to generate it, enabling efficient and effective 2D correspondence annotation to obtain reference poses.

1. Introduction

Visual localization is the problem of estimating the position and orientation from which an image was taken, i.e., its camera pose, with respect to the scene. Visual localization is a vital part of many computer vision and robotics applications such as self-driving cars, service robots such as gardening robots, and augmented/mixed/virtual reality.

Most visual localization methods rely heavily on local descriptors for pose estimation, and finding 2D-3D matches between images is a fundamental part of it. However, the trade-off between the local descriptors’ discriminative power and their invariance limits their performance under changing conditions. On the other hand, in practice, changes in the scene are unavoidable, and there is a need for visual localization methods to be robust to them. Traditionally, ground truth poses for localization have been obtained via Structure-from-Motion (SfM) [35, 62, 66]. Yet, SfM itself relies on local descriptors and matching. This makes it extremely difficult to generate localization benchmarks where feature matching using traditional feature descriptors does not work, e.g., in the presence of day/night and strong viewpoint changes. Benchmark datasets for localization under changing conditions such as Aachen, CMU Seasons, and Robotcar [58] rely on manual annotations to be able to provide ground truth poses. While these datasets provide interesting challenges, they have mostly been captured in controlled conditions. However, there is little control over the capture in many applications such as Autonomous Driving, collaborative AR/MR, or crowd-sourced mapping.

In this paper, we have constructed a dataset revisiting the common challenges that could be seen in different environments. We first actively mined a crowd-sourced database for image sequences where classic SfM approaches fail. To generate reliable poses, we have relied on human annotations and have created 40 sets of image sequences with diverse visual changes. The dataset and tools used for its creation are available at mapillary.com.

This paper makes the following contributions: (1) a workflow to mine and annotate challenging image sequences for benchmarking visual localization. Our approach explicitly takes pose uncertainty into account during the annotation process. (2) the CrowdDriven dataset, a geographically diverse and challenging dataset with reliable poses that covers various scenarios in illumination, weather, seasonal, and viewpoint changes. (3) experiments with state-of-the-art baselines showing CrowdDriven presents challenges that existing methods cannot handle.

2. Related Work

Visual localization methods aim to estimate the full camera pose and can be categorized based on their map representation: (1) Image-based representations encode each image as a feature vector [3, 17, 29, 54, 72, 73]. Using image retrieval and the known poses of the database images, the pose of a test image can either be approximated via the poses of the top-retrieved database images [61, 83] or computed precisely based on relative poses [59, 85]. (2) Local feature-based representations rely on classical or learned local feature descriptors and multiple view ge-
Figure 1. Easy (top), Medium (left) and Hard (right) subsets of our CrowdDriven dataset. Our dataset includes reference poses derived from hand-annotated control points, enabling the benchmarking of localization algorithms in new scenarios such as extreme viewpoint (180°) changes. CrowdDriven includes sequences from over the globe captured with a diversity of cameras.

ometry [43, 44, 55–57, 64, 68, 84]. These approaches build a 3D model of the scene and then estimate the camera pose based on 2D-3D matches between local features in a test image and the 3D model [23, 32, 41]. Such approaches, especially using learned features, constitute the current state-of-the-art in terms of visual localization in changing conditions [22, 30, 51, 52, 63] and we evaluate and analyze their performance on our benchmark. (3) Learning-based representations typically represent the scene using convolutional neural networks (CNNs). Pose regression techniques directly regress the camera pose for a given input image [10, 38, 39, 79]. These methods have been shown not to perform better than image retrieval methods [61]. Learning-based approaches that do not learn the full localization pipeline but only the 2D-3D matching part [6–9, 14, 15, 21, 53, 65] achieve state-of-the-art performance in small scenes [5, 80]. While most of these approaches struggle to handle conditions not seen during training, [53] generalizes well.

While localization approaches use metric maps, visual place recognition methods typically rely on topological scene representations [19, 20, 27, 28, 49, 50, 78], and thus do not directly provide a camera pose estimate. Using image retrieval and related techniques, they identify the places depicted in test images, where places are defined as sets of database images. Place recognition can be used to guide visual localization by identifying which parts of a scene are visible in a test image [37, 51, 60]. For an overview over place recognition techniques, we refer the reader to [26, 47].

Tab. 1 provides an overview over datasets commonly used to measure localization and place recognition performance under changing conditions. Coarse-scale location information for place recognition datasets such as Nordland [67], Pittsburgh [74], Tokyo 24/7 [72], and Mapillary Street-Level Sequences [81] can be obtained relatively easily via GPS measurements. In contrast, obtaining 6DOF poses for localization requires considerable manual effort as the classical approaches used to automatically obtain ground truth fail under challenging conditions. RIO10 [80] focuses on changes in indoor scenes. Long-term outdoor localization datasets such as Aachen Day-Night [58, 60], CMU Seasons [4, 58], and RobotCar Seasons [48, 58] only cover a few geographical locations and using a small number of cameras. In contrast, even though our new benchmark dataset is not the largest, it has much more geographical diversity. While the other datasets were captured using only a few cameras, our image sequences are taken by a large number of different camera types and photographers.

Other localization datasets include the indoor 7 Scenes [65], 12 Scenes [76], and InLoc [70] datasets and the outdoor Dubrovnik [43], Rome [25], Vienna [37], San Francisco [16, 44], and Cambridge Landmarks [39] datasets. None of these datasets is designed to measure the impact of changing conditions on localization performance [58].

The focus of recent datasets from self-driving car companies such as Lyft [40], Waymo [2], Aptiv (nuScenes) [11], and Baidu (Apolloscape) [36] is benchmarking 2D/3D object detection based on multimodal sensor data in limited geographical areas, semantic segmentation, and depth estimation. An exception is Baidu’s Apolloscape which provides a localization benchmark track in small urban areas.

Another shortcoming of the datasets mentioned above is that the changes in viewpoints are limited. The query images are typically taken from similar vantage points as the reference images. Our dataset contains image sequences that are, for instance, taken in opposite directions with different cameras. This case commonly happens during multi-session or collaborative capture of man-made environments, where paths can be traversed in two directions.

Based on the shortcomings of current localization datasets, we generate a diverse dataset for long-term visual localization (1) containing different challenging scenarios such as day-night, seasonal, daylight illumination, strong
viewpoint and man-made changes, (2) with reliable 6DOF camera poses based on human annotations, (3) using crowdsourced data to imitate real driving situations, (4) with reliable camera calibration and geographical information.

3. CrowdDriven: Dataset Creation

Historically, visual localization algorithms have been evaluated on crowd-sourced datasets collected from internet photo collection websites such as Flickr, e.g., the Dubrovnik [43], Rome [43], and Landmarks 1k [44] datasets. This way to acquire datasets scales very well to the use of different camera types and geographical locations. However, since the images have been taken by handheld cameras around famous landmarks, these datasets are not suitable for measuring visual localization performance in an autonomous driving scenario, which is our main focus (even though a small part of our dataset was captured by pedestrians and bicycles). In contrast, recently released datasets tailored for this task, e.g., the RobotCar Seasons [48,58] and the (extended) CMU Seasons [4,58] datasets, contain only one or a few larger locations, use only a few cameras, and have been taken by experts. Our new dataset, CrowdDriven, is designed to cover a wide range of locations, visual conditions (e.g., seasonal, day-night changes or changes in facades, presence/absence of humans (man-made changes)), and camera types by tapping into a large database of crowdsourced images.

3.1. Data Source

To maximize the diversity and geographical coverage, we use Mapillary, a collaborative street-level imagery platform that hosts more than 1 billion images collected by members of their community while driving or walking on public spaces and roads. It covers most countries with hundreds of camera models in varying times of day and weather conditions. Its data is thus well-suited to evaluate problems in conditions similar to those faced in self-driving scenarios, since most images are captured with consumer-grade devices such as smartphones, action cameras, and dashcams.

3.2. Sequence Selection

Images in Mapillary are grouped into ordered sequences based on the photographer ID and the capture time. Each image’s GPS position and compass are available, and the database can be queried to find images near a particular position. To generate a dataset that is challenging for current state-of-the-art localization algorithms, we attempt to find pairs of neighboring sequences to be localized with respect to each other. We categorize each sequence pair into a preliminary difficulty level based on the success of traditional SfM algorithms on jointly reconstructing the pair.

We begin by querying the Mapillary database to find pairs of sequences covering a wide range of geographical locations and appearances. The pairs are selected to satisfy the following criteria: 1. Sequence length between 40 and 60 images. 2. Minimum sequence density of 0.2 images/m. 3. Maximum sequence-to-sequence distance of 3m. 4. Each sequence can be reconstructed individually using SfM.

Note that we deliberately chose to focus on small scenes rather than attempting to collect data for a larger spatial area. In practice, pose priors such as GPS are used to limit the search space during localization. Rather than artificially making the problem harder by ignoring such priors, we are interested in finding realistic hard examples with small scenes. Our dataset might not be as large as previous benchmarks. However, a large fraction of test images in previous benchmarks can be localized accurately. In contrast, our dataset contains many very challenging scenarios that current state-of-the-art methods cannot handle.

After collecting the sequence pairs, we run SfM on each of them. We assign a difficulty rating to each pair depending on the result of SfM and the relative orientation of the sequences: Sequence pairs whose joint reconstruction succeeds¹ are categorized as easy and are shown in Fig. 1 (top). If the joint reconstruction fails, we look at the average view direction of the sequences to categorize them into medium (Fig. 1 - left) if they have similar orientations (less than a 45° difference), or hard (Fig. 1 - right) if they have different orientations (more than a 45° difference).

This preliminary categorization is based on the following observations: sequences with similar orientations are difficult to match mostly due to changes in appearance (e.g., illumination or seasonal change) and state-of-the-art methods

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Some Type</th>
<th># Images</th>
<th>condition changes</th>
<th>6DOF query poses</th>
<th># locations</th>
</tr>
</thead>
<tbody>
<tr>
<td>CrowdDriven</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>26</td>
</tr>
</tbody>
</table>

Table 1. Comparing localization datasets: CrowdDriven is the most diverse in term of scene types and changes in viewing conditions.

1 All images are included in the reconstruction with a sufficiently large number of well-distributed inliers.
such as [51–53] are able to handle such changes quite well. In contrast, sequence pairs with different view directions are harder to match due to the addition of dramatic appearance changes of scene parts due the large viewpoint change and low visual overlap [63]. As our experiments in Sec. 5 show, this preliminary classification aligns well with how current state-of-the-art methods perform on our dataset.

Tab. 2 shows statistics over the sequence pairs in CrowdDriven. Many capture conditions that are challenging for current localization algorithms are included in CrowdDriven. The medium category consists of different scene types shown under conditions such as day-night changes, rain, snow and other seasonal variations that impact the scene geometry. The main focus of the hard category is the combination of large viewpoint changes with less extreme variations in illumination, seasons, and weather conditions.

**On the dataset size.** Similar to existing datasets [58,70,80], CrowdDriven only provides reference images taken under a single condition for each scene with test images. As shown in Tab. 1, CrowdDriven contains fewer images than most such datasets, e.g., the CMU and RobotCar Seasons datasets from [58]. However, this does not imply that our dataset is too small for learning-based methods such as camera pose [10,38,39,79] or scene coordinate [6–9,14,15,65] regression. These approaches regress the camera pose from a given image and a 3D point coordinate from an image patch, respectively. Both regression tasks are instance-level problems. As such, the number of images in local scene parts is more important for their performance in these parts than the absolute number of images in a dataset. Extending CrowdDriven in terms of the absolute number of images by adding additional scenes is easy. However, doing so is unlikely to improve the performance in existing scenes. Extending CrowdDriven by adding more images to existing scenes is hard as such images are simply not available: while crowd-sourced image capture allows us to obtain images from a diverse set of scenes from multiple continents, there is no control over how much data is acquired per scene. Still, the number of reference images per scene should be sufficient for learning-based techniques.

### 3.3. Reference Pose Generation

To use the sequences for benchmarking visual localization algorithms, two processing steps are required: 1. estimating intrinsic camera calibrations and reference poses for all images in a common coordinate system, such that distances can be measured in meters. 2. subdividing the datasets into reference (training) images and test images. For (2), we use a simple strategy: for each pair of sequences, the larger sequence defines the reference images, and the images from the smaller sequence are used for testing.

**Easy Datasets** are those where both sequences can be reconstructed in a common coordinate frame by off-the-shelf SfM using SIFT [46] features. We follow common practice [25,39,44,58] and use the camera poses and intrinsics estimated during the reconstruction process to define our reference poses. After running SfM on the sequences, we visually inspect the 3D models. If the two sequences are *aligned*, *i.e.*, there are no duplicate 3D points, and the camera poses look visually correct, we scale the SfM poses based on available GPS data to (approximately) recover the scale of the scene. For this category, we expect state-of-the-art localization approaches to perform well given that SIFT features are sufficiently powerful to register the sequences.

We used OpenSfM [1] as our SfM pipeline. Experiments with COLMAP [62] on our data showed similar results.

Tab. 2 provides statistics over the 13 sequence pairs in the easy category. As can be seen, diversity in scenes such as weather and illumination changes is covered in different environments such as roads, suburban and urban areas. Fig. 1 (top) shows sample images from this category.

**Medium and Hard Datasets** are those where SfM fails to register the sequence pairs due to large changes in appearance (medium) and/or viewpoint (hard). Both medium and hard datasets are thus significantly more challenging for existing visual localization algorithms. Thus, these datasets will be most interesting to the community.

To obtain reference poses, we first reconstruct each sequence individually using SfM and recover the scale of the models using known GPS data from Mapillary. Next, we manually annotate corresponding pixel positions between images from the two sequences (Figs. 2 & 3). These annotations define manual tracks within each sequence and between sequences, which we will call control points (CPs). On average, 12 different control points have been annotated for each dataset (sequence pair), where each such point has been observed on average in 10 images.

To align the sequences, we first obtain the 3D position of the CPs in the reference frame of each reconstruction by triangulating the 2D annotations. The corresponding 3D points (as per the annotation) provide us with 3D-3D matches that we use to compute an initial registration as a similarity transformation, bringing both reconstructions into a common reference frame. Given this initial alignment, we perform bundle adjustment [75] over all camera poses and intrinsics, the 3D points triangulated from SIFT features, and the 3D points triangulated from annotations.

Fig. 2 shows the quality of the alignment when using only GPS constraints (middle) and when using manual annotations (right). For clarity of visualization, we show the scenes as dense Multi-View Stereo point clouds, although the registration is performed using only sparse points.

We do not use dense point clouds for alignment as they cannot always be obtained solely from images, *e.g.*, due to...
3.4. Quality Control

To verify the reference poses, we use the following criteria and workflow. If SfM succeeds and there are no noticeable artefacts and errors, then it typically produces highly accurate estimates for the camera poses and the 3D structure of the scene. Thus, we directly trust the reference poses we obtained for the easy datasets.

For the medium and hard datasets, we rely on manually-annotated matches, which are susceptible to human error and might be distributed in a sub-optimal way in the scene, under-constraining the alignment. Previous work on building benchmark datasets that also uses manual annotations [58, 70] often relied on visual inspection and did not rigorously measure the uncertainty of their generated poses. In the case of the Aachen Day-Night dataset [58, 60], [86] recently showed that inaccurate poses passed visual inspection. We thus made a conscious effort to quantitatively measure the accuracy of our poses.

We use two metrics to verify that the annotations are correct and sufficient. After a few control points have been annotated, an alignment is generated by estimating a similarity transform followed by bundle adjustment (as described above). To detect potential issues, we begin by estimating the **reprojection error** of every annotation. This metric
identifies wrongly annotated points such as wrong correspondences between different objects with similar appearance or simply misclicks during annotation. All annotations with an error of more than one pixel (at VGA resolution) are flagged as incorrect and must be refined.

After all the reprojection errors are sufficiently small, we compute the camera position covariance for each image using bundle adjustment [24]. To fix the gauge ambiguity, we fix the poses of one of the sequences, run the bundle adjustment problem, and compute the covariance of the poses of the other sequence. We repeat this while fixing the poses of the other sequence. From the covariances, we compute the standard deviation of the camera positions in meters and seek for small values when annotating. This metric is used to understand if the calculated position of an image is under-constrained: all annotations can be correct with very small reprojection errors, but the poses might still be under-constrained, e.g., if the annotations were only performed on far-away points, resulting in large standard deviations.

**Annotation Tools.** These two metrics are integrated with the annotation UI that we have developed and will release as part of this work (c.f. Fig. 3). Annotators can run bundle adjustment and get feedback directly in the UI, visualize wrongly annotated points and focus their efforts on annotating those frames with higher positional uncertainty, simplifying and accelerating the annotation process. Using our tool, the manual annotation (and QA) of a sequence pair reconstruction takes, on average, only 30 minutes.

After all positional std. deviations are confirmed to be under 30 cm, we perform a final check by visually inspecting the combined point cloud for the aligned models (Fig. 2). Tab. 2 shows some statistics about the annotation process, including the number of annotated points, the median reprojection error and median positional std. deviation for each dataset. For the whole set of Medium and Hard datasets, the median annotation reprojection error is 0.28 px and the median positional std. deviation is 5.4 cm.

### 4. Baselines

In order to show that our benchmark introduces new challenges for localization algorithms, we use a set of state-of-the-art localization methods as baselines. We focus on methods that have been shown to work well under changing conditions (based on their performance on the benchmark from [58]) and provide source code and trained models:

- **HLoc** [51,52] uses learned SuperPoint [18] features and SuperGlue [52] to establish 2D-3D matches with a SfM model, which are then used for camera pose estimation.
- **D2-Net** [22] uses a single CNN for feature detection and description. Pose estimation is implemented in COLMAP [62]. In contrast to HLoc, which runs (close to) real-time, D2-Net requires multiple seconds per test image.
- **Rectified SIFT** [71] uses a depth estimation network [45] to detect planar regions in the image. Warping them to remove perspective foreshortening leads to features that can be matched under strong viewpoint changes.
- **S2DHM** [30] uses an asymmetric matching approach: reference images are represented by sparse features corresponding to the 3D points in the SfM model, while the test image is represented by densely extracted descriptors.

The above-mentioned baselines are based on matching local features to establish 2D-3D matches between the test image and an SfM model of the scene. In contrast, **PixLoc** [53] does not rely on feature matching but refines an initial pose estimate by minimizing a feature-metric cost function.

We do not evaluate camera pose and scene coordinate regression methods. Pose regressors have been shown to be significantly less accurate than other localization approaches [61], even on scenes without changing conditions. We thus see no reason why they should perform well on our more challenging benchmark. Current scene coordinate regressors seem to struggle with strong condition changes between the training and test sets: on the Aachen Day-Night dataset [58, 60], where all training images are taken at day, ESAC [8] performs significantly worse for nighttime test images compared to daytime queries and is less accurate than HLoc by a wide margin. CrowdDriven only provides references images taken under a single condition for each scene. Given the strong condition changes between training and test images, current scene coordinate regressors are unlikely to perform well on our dataset.

In our experiments, for a given test image, we consider only reference images from the same scene, e.g., Boston1, and not from different scenes. Since our scenes are rather small (c.f. Tab. 2), using place recognition / image retrieval is not necessary. Instead, we exhaustively match the test image against all reference images in the scene.

The supp. material shows results for **sequence-based localization** methods that simultaneously estimate the poses of all images in a sequence [12, 41, 42, 69, 77]. While this improves performance, it is still not sufficient to obtain reasonable performance on our more challenging scenes.

### 5. Experimental Evaluation

This section shows how our dataset introduces new challenging scenarios that most baselines cannot cope with. After introducing our evaluation measures, we analyze the impact of different types of changes on the performance.

**Evaluation Measures.** We follow common evaluation protocols [25,39,58,70] and report median position (in meters) and orientation errors (in degrees), as well as the percentage of test images with poses that differ within certain error bounds from their reference poses. The position error is
Figure 4. Inlier plots for selected scenes. None of the evaluated methods is able to robustly localize the medium and hard datasets are color-coded in light, standard, and dark gray, respectively. We also provide information about the type of change between mated rotation matrix as made apparent by the fraction of localized images under the 2\textsuperscript{nd} rotation error.

Table 3. Localization performance on our CrowdDriven benchmark. We report the median position (in meters) and orientation (in degrees), and the percentage of test images localized within certain error bounds on the position and orientation errors. Easy, medium, and hard datasets are color-coded in light, standard, and dark gray, respectively. We also provide information about the type of change between the training and test sequences: illumination: \(\uparrow\), overcast: \(\downarrow\), foliage: \(\oslash\), snow: \(\oslash\), seasonal: \(\oslash\), day-night: \(\oslash\), small viewpoint: \(\oslash\), man-made changes: \(\oslash\), and \(\mathbf{F}\) stands for failure to localize any image within the coarsest precision regime.

measured as the Euclidean distance between the reference and estimated position. To measure the orientation error, we compute the minimum rotation angle \(\alpha\) that aligns the estimated rotation matrix \(R_{est}\) with the reference rotation \(R_{ref}\) as \(2\cos(|\alpha|) = \text{trace}(R_{ref}^T R_{est}) - 1\) [33]. Inspired by [58], we use three error bounds: high precision (images localized within 0.5m and 2\(^{\circ}\) of their reference poses), medium precision (1m, 5\(^{\circ}\)), and coarse precision (5m, 10\(^{\circ}\)). In addition, we introduce a very coarse precision regime (10m, 20\(^{\circ}\)).

We also report the changes in conditions between the test
and training images: (slight) illumination changes (il.), e.g., between a cloudy and a sunny day, foliage / no foliage on the vegetation (fo.), snow / no snow on the ground (sn.), other seasonal changes (se.) (summer/autumn), day-night changes (ng.), rain / no rain (rn.), small (sm. v.) and strong viewpoint (st. v.), and man-made (e.g., appearance / disappearance of cars) changes. Tab. 3 summarizes the results obtained by evaluating the baselines. In the following, we focus our analysis on three types of changes: slight illumination, day-night, and strong viewpoint changes, which are the dominant types of changes.

**Slight illumination changes.** As shown in Tab. 3, datasets such as Muehlhausen, Tsuru, Poing, Bayern, Savannah, Curitiba, Melbourne, Sydney, and Clermont-Ferrand are categorized as being affected solely by illumination changes seen during the day. Almost all the test images are localized with high precision by most of the methods. In this scenario, SIFT is already robust enough to slight changes in illumination. However, rectified SIFT struggles to deal with the heavy vegetation present in Poing, Sydney, and Subcarpathia. We observe that for some datasets, PixLoc performs significantly worse than the other baselines. We attribute this to different viewing condition between the training set of PixLoc and our dataset, e.g., PixLoc was not trained on images showing as much snow as the Subcarpathia dataset, and the challenges of identifying the best pose from all available pose estimates (c.f. supp. mat.) and of pose refinement in complex scenes such as Melbourne and Sydney. Still, we conclude that slight illumination changes are not challenging for current algorithms.

**Day-night changes.** The significant appearance changes that occur between day and night have considerable effects on the performance of most of the methods, leading to a noticeable increase in the errors. D2-Net’s robustness to illumination changes makes the model perform well in some of the datasets such as Massachusetts4 and Boston4, allowing localization with medium precision. On the other hand, the effect of artificial lights during the night could cover the previously-seen features or considerably change them, making one part of the image sparser, and the pose estimation less accurate. In Massachusetts2, Massachusetts3, Boston2, and Boston5, the errors are larger than those of consumer-grade GPS. HLoc and S2DHM, as seen in Tab. 3, produce large errors in all of the night cases except for Cambridge, with the median errors of these methods starting at more than 30 meters. We can conclude that for night-time test images, due to the considerable changes in feature appearances, approaches such as S2DHM do not seem to improve the results as long as the extracted Hypercolumn descriptors themselves are not robust enough to the day-night changes (c.f. Fig. 4). Rectified SIFT inherits the limitations of SIFT features and also struggles significantly on these scenes.

**Strong viewpoint changes.** By far the most challenging condition present in our dataset is strong viewpoint change, as seen in Tab. 3. The rotation errors obtained for all the methods are over 160°, indicating complete failures. Strong viewpoint changes significantly alter the appearance of objects, resulting in substantially different feature descriptors depending on the viewing angle. In theory, removing perspective distortion via rectification should help to better handle larger viewpoint changes [71]. Yet, the rectified SIFT baseline still mostly fails under strong viewpoint changes. This can be attributed to the limited visual overlap as well as inaccurate depth predictions on the distant overlapping image regions (c.f. Fig. 4). Unsurprisingly, PixLoc always fails under strong viewpoint changes: PixLoc refines an initial pose estimate, which is obtained from the poses of the database images. In the case of opposite viewpoints, these initial estimates are simply too different from the test poses for PixLoc to converge to a reasonable pose estimate.

**Difficulty levels revisited.** Sec. 3 provided a preliminary classification based on our understanding of what is challenging for current localization algorithms. Based on the measured performance of our baselines, we revisit this classification: a dataset should be considered as easy if (nearly) all baselines perform well, as medium if some baselines work well (at least for the coarsest threshold), and as hard if all methods fail. We color-coded the datasets in Tab. 3 accordingly. Comparing the color-codings in Tabs. 2 and 3 shows that the classification based on performance is consistent with our preliminary categorization for most scenes.

### 6. Conclusion

We have introduced the CrowdDriven dataset for visual localization, with a focus on exposing the failure cases of state-of-the-art pipelines. The dataset covers three difficulty tiers: (1) slight illumination changes, (2) severe illumination changes, (3) severe viewpoint changes.

We have evaluated the performance of the best visual localization methods available on our dataset and analyzed the failure modes for the different methods. We conclude that there is a large performance gap to be covered by future work and that can be evaluated with our dataset. Finally, we release the annotation tool we developed to create this dataset, enabling easy extension to more scenarios.

As shown concurrently to our work [5], using a feature-based approach (SfM) to generate reference poses can create a bias towards feature-based methods. We consider the problem of eliminating such a bias from the evaluation as an important direction for future work.

**Acknowledgements.** This work has been funded by the Chalmers AI Research Centre (VisLocLearn), the Swedish Foundation for Strategic Research (Semantic Mapping and Visual Navigation for Smart Robots), the EU Horizon 2020 project RICAIP (grant agreement No 857306), and the European Regional Development Fund under IMPACT No. CZ.02.1.01/0.0/0.0/15_003/0000468.
References


[29] Yixiao Ge, Haibo Wang, Feng Zhu, Rui Zhao, and Hong-sheng Li. Self-supervising fine-grained region similarities.


[41] Zuzana Kukelova, Jan Heller, and Andrew Fitzgibbon. Efficient Intersection of Two Quadrics and Applications in Computer Vision. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.


[67] Niko Sünderhauf, Peer Neubert, and Peter Protzel. Are we there yet? challenging seqslam on a 3000 km journey across all four seasons. 2, 3


[72] Akihiko Torii, Relja Arandjelović, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 Place Recognition by View Synthesis. In CVPR, 2015. 1, 2, 3


