LaMAR: Benchmarking Localization and Mapping for Augmented Reality

Paul-Edouard Sarlin$^{\star 1}$, Mihai Dusmanu$^{\star 1}$, Johannes L. Schönberger^{2}, Pablo Speciale^{2},
Lukas Gruber^{2}, Viktor Larsson$^{\dagger 1}$, Ondrej Miksik^{2}, and Marc Pollefeys^{1,2}

[1] Department of Computer Science, ETH Zürich, Switzerland
[2] Microsoft Mixed Reality & AI Lab, Zürich, Switzerland

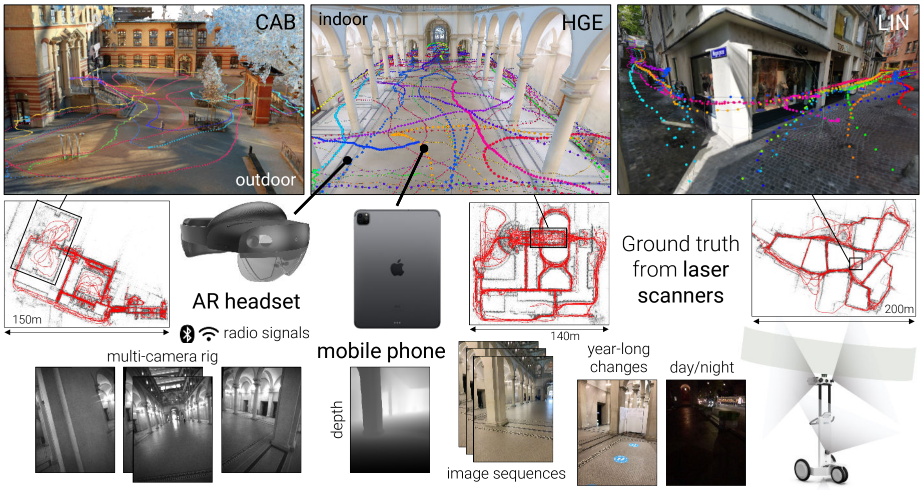

Fig. 1. We revisit localization and mapping in the context of Augmented Reality by introducing LaMAR, a large-scale dataset captured using AR devices (HoloLens2, iPhone) and laser scanners.

Abstract. Localization and mapping is the foundational technology for augmented reality (AR) that enables sharing and persistence of digital content in the real world. While significant progress has been made, researchers are still mostly driven by unrealistic benchmarks not representative of real-world AR scenarios. In particular, benchmarks are often based on small-scale datasets with low scene diversity, captured from stationary cameras, and lacking other sensor inputs like inertial, radio, or depth data. Furthermore, ground-truth (GT) accuracy is mostly insufficient to satisfy AR requirements. To close this gap, we introduce a new benchmark with a comprehensive capture and GT pipeline, which allow us to co-register realistic AR trajectories in diverse scenes and from heterogeneous devices at scale. To establish accurate GT, our pipeline robustly aligns the captured trajectories against laser scans in a fully automatic manner. Based on this pipeline, we publish a benchmark dataset of diverse and large-scale scenes recorded with head-mounted and hand-held AR devices. We extend several state-of-the-art methods to take advantage of the AR specific setup and evaluate them on our benchmark. Based on the results, we present novel insights on current research gaps to provide avenues for future work in the community.

* Equal contribution. † Now at Lund University, Sweden.

1 Introduction

Placing virtual content in the physical 3D world, persisting it over time, and sharing it with other users are typical scenarios for Augmented Reality (AR). In order to reliably overlay virtual content in the real world with pixel-level precision, these scenarios require AR devices to accurately determine their 6-DoF pose at any point in time. While visual localization and mapping is one of the most studied problems in computer vision, its use for AR entails specific challenges and opportunities. First, modern AR devices, such as mobile phones or the Microsoft HoloLens or MagicLeap One, are often equipped with multiple cameras and additional inertial or radio sensors. Second, they exhibit characteristic hand-held or head-mounted motion patterns. The on-device real-time tracking systems provide spatially-posed sensor streams. However, many AR scenarios require positioning beyond local tracking, both indoors and outdoors, and robustness to common temporal changes of appearance and structure. Furthermore, given the plurality of temporal sensor data, the question is often not whether, but how quickly can the device localize at any time to ensure a compelling end-user experience. Finally, as AR adoption grows, crowd-sourced data captured by users with diverse devices can be mined for building large-scale maps without a manual and costly scanning effort. Crowd-sourcing offers great opportunities but poses additional challenges on the robustness of algorithms, e.g., to enable cross-device localization [21], mapping from incomplete data with low accuracy [67,8], privacy-preservation of data [73,25,71,26,23], etc.

However, the academic community is mainly driven by benchmarks that are disconnected from the specifics of AR. They mostly evaluate localization and mapping using single still images and either lack temporal changes [72,56] or accurate ground truth (GT) [65,37,76], are restricted to small scenes [6,72,37,83,70] or landmarks [34,68] with perfect coverage and limited viewpoint variability, or disregard temporal tracking data or additional visual, inertial, or radio sensors [66,65,76,40,12,75].

Our first contribution is to introduce **a large-scale dataset captured using AR devices in diverse environments**, notably a historical building, a multi-story office building, and part of a city center. The initial data release contains both indoor and outdoor images with illumination and semantic changes as well as dynamic objects. Specifically, we collected multi-sensor data streams (images, depth, tracking, IMU, BT, WiFi) totalling more than 100 hours using head-mounted HoloLens 2 and hand-held iPhone / iPad devices covering 45'000 square meters over the span of one year (Figure 1).

Second, we develop **a GT pipeline to automatically and accurately register AR trajectories** against large-scale 3D laser scans. Our pipeline does not require any manual labelling or setup of custom infrastructure (e.g., fiducial markers). Furthermore, the system robustly handles crowd-sourced data from heterogeneous devices captured over longer periods of time and can be easily extended to support future devices.

Finally, we present **a rigorous evaluation of localization and mapping in the context of AR** and provide **novel insights for future research**. Notably, we show that the performance of state-of-the-art methods can be drastically improved by considering additional data streams generally available in AR devices, such as radio signals or sequence odometry. Thus, future algorithms in the field of AR localization and mapping should always consider these sensors in their evaluation to show real-world impact.

dataset	out/indoor	changes	scale	density	camera motion	imaging devices	additional sensors	ground truth	accuracy
Aachen [66,65]	✅❌	🌗🏗️	★★⯪	★★☆	still images	DSLR	❌	SfM	>dm
Phototourism [34]	✅❌	🏃🏗️	⯪☆☆	★★★	still images	DSLR, phone	❌	SfM	∼m
San Francisco [14]	✅❌	🏃🏗️	★★★	★☆☆	still images	DSLR, phone	GNSS	SfM+GNSS	∼m
Cambridge [37]	✅❌	🏃🌧️	⯪☆☆	★★☆	handheld	mobile	❌	SfM	>dm
7Scenes [72]	❌✅	❌	⯪☆☆	★★★	handheld	mobile	depth	RGB-D	∼cm
RIO10 [83]	❌✅	🪑	⯪☆☆	★★★	handheld	Tango tablet	depth	VIO	>dm
InLoc [76]	❌✅	🪑	★⯪☆	⯪☆☆	still images	panoramas, phone	lidar	manual+lidar	>dm
Baidu mall [75]	❌✅	🏃	★⯪☆	★★☆	still images	DSLR, phone	lidar	manual+lidar	∼dm
Naver Labs [40]	❌✅	🏃🪑	★★☆	★★☆	robot-mounted	fisheye, phone	lidar	lidar+SfM	∼dm
NCLT [12]	✅✅	🌧️🪑	★★☆	★★☆	robot-mounted	wide-angle	lidar, IMU, GNSS	lidar+VIO	∼dm
ADVIO [56]	✅✅	🏃	★★☆	⯪☆☆	handheld	phone, Tango	IMU, depth, GNSS	manual+VIO	∼m
ETH3D [70]	✅✅	❌	⯪☆☆	★★☆	handheld	DSLR, wide-angle	lidar	manual+lidar	∼mm
LaMAR (ours)	✅✅	🏃🌧️🌗 🪑🏗️	★★⯪ 3 locations 45'000 m^2	★★★ 100 hours 40 km	handheld head-mounted	phone, headset backpack, trolley	lidar, IMU, 📶 ᛒ depth, infrared	lidar+SfM+VIO automated	∼cm

Table 1. Overview of existing datasets. No dataset, besides ours, exhibits at the same time short-term appearance and structural changes due to moving people 🏃, weather 🌧️, or day-night cycles 🌗, but also long-term changes due to displaced furniture 🪑 or construction work 🏗️.

The LaMAR dataset, benchmark, GT pipeline, and the implementations of baselines integrating additional sensory data are all publicly available at `lamar.ethz.ch`. We hope that this will spark future research addressing the challenges of AR.

2 Related work

Image-based localization is classically tackled by estimating a camera pose from correspondences established between sparse local features [43,7,59,47] and a 3D Structure-from-Motion (SfM) [67] map of the scene [24,42,64]. This pipeline scales to large scenes using image retrieval [2,33,57,78,11,55,79]. Recently, many of these steps or even the end-to-end pipeline have been successfully learned with neural networks [20,62,22,69,3,49,77,61,88,32,63]. Other approaches regress absolute camera pose [37,36,50] or scene coordinates [72,82,46,45,41,9,85,10]. However, all these approaches typically fail whenever there is lack of context (e.g., limited field-of-view) or the map has repetitive elements. Leveraging the sequential ordering of video frames [48,35] or modelling the problem as a generalized camera [53,29,65,73] can improve results.

Radio-based localization: Radio signals, such as WiFi and Bluetooth, are spatially bounded (logarithmic decay) [5,38,28], thus can distinguish similarly looking (spatially distant) locations. Their unique identifiers can be uniquely hashed which makes them computationally attractive (compared with high-dimensional image descriptors). Several methods use the signal strength, angle, direction, or time of arrival [51,13,18] but the most popular is model-free map-based fingerprinting [38,28,39], as it only requires to collect unique identifiers of nearby radio sources and received signal strength. GNSS provides absolute 3-DoF positioning but is not applicable indoors and has insufficient accuracy for AR scenarios, especially in urban environments due to multi-pathing, etc.

Datasets and ground-truth: Many of the existing benchmarks (cf. Tab. 1) are captured in small-scale environments [72,83,19,30], do not contain sequential

device	motion type	cameras					radios	other data	poses
		#	FOV	frequency	resolution	specs			
M6	trolley	6	113°	1-3m	1080p	RGB, sync	🛜ᛒ	lidar points+mesh	lidar SLAM
VLX	backpack	4	90°	1-3m	1080p	RGB, sync	ᛒ	lidar points+mesh	lidar SLAM
HoloLens2	head-mounted	4	83°	30Hz	VGA	gray, GS	🛜ᛒ	ToF depth/IR 1Hz, IMU	head-tracking
iPad/iPhone	hand-held	1	64°	10Hz	1080p	RGB, RS, AF	ᛒ*	lidar depth 10Hz, IMU	ARKit

Table 2. Sensor specifications. Our dataset has visible light images (global shutter GS, rolling shutter RS, auto-focus AF), depth data (ToF, lidar), radio signals (*, if partial), dense lidar point clouds, and poses with intrinsics from on-device tracking.

data [66,34,14,76,75,70,6,68], lack characteristic hand-held/head-mounted motion patterns [65,4,44,86], or their GT is not accurate enough for AR [56,37]. None of these datasets contain WiFi or Bluetooth data (Tab. 1). The closest to our work are Naver Labs [40], NCLT [12] and ETH3D [70]. Both, Naver Labs [40] and NCLT [12] are less accurate than ours and do not contain AR specific trajectories or radio data. The Naver Labs dataset [40] also does not contain any outdoor data. ETH3D [70] is highly accurate, however, it is only small-scale, does not contain significant changes, or any radio data.

To establish ground-truth, many datasets rely on off-the-shelf SfM algorithms [67] for unordered image collections [66,34,37,83,56,75,76,34]. Pure SfM-based GT generation has limited accuracy [8] and completeness, which biases the evaluations to scenarios in which visual localization already works well. Other approaches rely on RGB(-D) tracking [83,72], which usually drifts in larger scenes and cannot produce GT in crowd-sourced, multi-device scenarios. Specialized capture rigs of an AR device with a more accurate sensor (lidar) [40,12] prevent capturing of realistic AR motion patterns. Furthermore, scalability is limited for these approaches, especially if they rely on manual selection of reference images [75], laborious labelling of correspondences [66,76], or placement of fiducial markers [30]. For example, the accuracy of ETH3D [70] is achieved by using single stationary lidar scan, manual cleaning, and aligning very few images captured by tripod-mounted DSLR cameras. Images thus obtained are not representative for AR devices and the process cannot scale or take advantage of crowd-sourced data. In contrast, our fully automatic approach does not require any manual labelling or special capture setups, thus enables light-weight and repeated scanning of large locations.

3 Dataset

We first give an overview of the setup and content of our dataset.

Locations: The initial release of the dataset contains 3 large locations representative of AR use cases: 1) HGE (18'000 m^2) is the ground floor of a historical university building composed of multiple large halls and large esplanades on both sides. 2) CAB (12'000 m^2) is a multi-floor office building composed of multiple small and large offices, a kitchen, storage rooms, and 2 courtyards. 3) LIN (15'000 m^2) is a few blocks of an old town with shops, restaurants, and narrow passages. HGE and CAB contain both indoor and outdoor sections with many symmetric structures. Each location underwent structural changes over the span of a year, e.g., the front of HGE turned into a construction site and the indoor furniture was rearranged. See Figure 2 for a visualization of the locations.

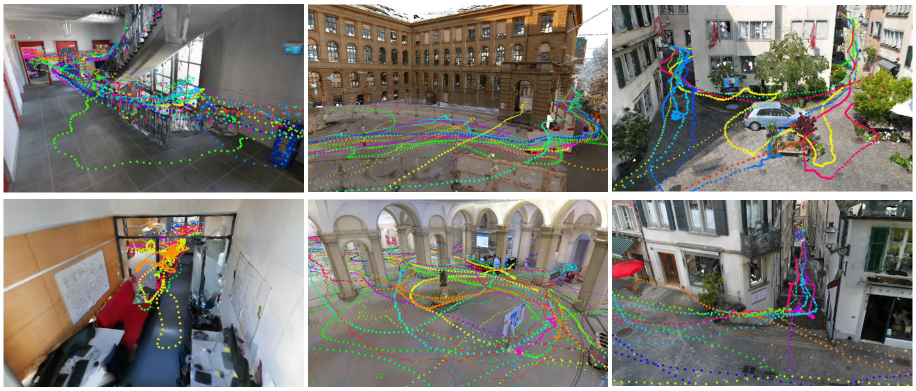

Fig. 2. The locations feature diverse indoor and outdoor spaces. High-quality meshes, obtained from lidar, are registered with numerous AR sequences, each shown here as a different color.

Data collection: We collected data using Microsoft HoloLens 2 and Apple iPad Pro devices with custom raw sensor recording applications. 10 participants were each given one device and asked to walk through a common designated area. They were only given the instructions to freely walk through the environment to visit, inspect, and find their way around. This yielded diverse camera heights and motion patterns. Their trajectories were not planned or restricted in any way. Participants visited each location, both during the day and at night, at different points in time over the course of up to 1 year. In total, each location is covered by more than 100 sessions of 5 minutes. We did not need to prepare the capturing site in any way before recording. This enables easy barrier-free crowd-sourced data collections. Each location was also captured twice by NavVis M6 trolley and VLX backpack mapping platforms, which generate textured dense 3D models of the environment using laser scanners and panoramic cameras.

Privacy: We paid special attention to comply with privacy regulations. Since the dataset is recorded in public spaces, our pipeline anonymizes all visible faces and licence plates.

Sensors: We provide details about the recorded sensors in Table 2. The HoloLens has a specialized large field-of-view (FOV) multi-camera tracking rig (low resolution, global shutter) [81], while the iPad has a single, higher-resolution camera with rolling shutter and more limited FOV. We also recorded outputs of the real-time AR tracking algorithms available on each device, which includes relative camera poses and sensor calibration. All images are undistorted. All sensor data is registered into a common reference frame with accurate absolute GT poses using the pipeline described in the next section.

4 Ground-truth generation

The GT estimation process takes as input the raw data from the different sensors. The entire pipeline is fully automated and does not require any manual alignment or input.

Overview: We start by aligning different sessions of the laser scanner by using the images and the 3D lidar point cloud. When registered together, they form the GT reference map, which accurately captures the structure and appearance of the scene. We then register each AR sequence individually to the reference map using local feature matching and relative poses from the on-device tracker. Finally, all camera poses are refined jointly by optimizing the visual constraints within and across sequences.

Notation: We denote ${}_i\mathbf{T}_j \in \mathrm{SE}(3)$ the 6-DoF pose, encompassing rotation and translation, that transforms a point in frame j to another frame i. Our goal is to compute globally-consistent absolute poses ${}_w\mathbf{T}_i$ for all cameras i of all sequences and scanning sessions into a common reference world frame w.

4.1 Ground-truth reference model

Each capture session $S \in \mathcal{S}$ of the NavVis laser-scanning platform is processed by a proprietary inertial-lidar SLAM that estimates, for each image i, a pose ${}_0\mathbf{T}_i^S$ relative to the beginning of the session. The software filters out noisy lidar measurements, removes dynamic objects, and aggregates the remainder into a globally-consistent colored 3D point cloud with a grid resolution of 1cm. To recover visibility information, we compute a dense mesh using the Advancing Front algorithm [17].

Our first goal is to align the sessions into a common GT reference frame. We assume that the scan trajectories are drift-free and only need to register each with a rigid transformation ${}_w\mathbf{T}_0^S$. Scan sessions can be captured between extensive periods of time and therefore exhibit large structural and appearance changes. We use a combination of image and point cloud information to obtain accurate registrations without any manual initialization. The steps are inspired by the reconstruction pipeline of Choi et al. [15,89].

Pair-wise registration: We first estimate a rigid transformation ${}_A\mathbf{T}_B$ for each pair of scanning sessions $(A, B) \in \mathcal{S}^2$. For each image I_i^A in A, we select the r most similar images $(I_j^B)_{1 \leq j \leq r}$ in B based on global image descriptors [33,3,57], which helps the registration scale to large scenes. We extract sparse local image features and establish 2D-2D correspondences $\{\mathbf{p}_i^A, \mathbf{p}_j^B\}$ for each image pair (i, j). The 2D keypoints $\mathbf{p}_i \in \mathbb{R}^2$ are lifted to 3D, $\mathbf{P}_i \in \mathbb{R}^3$, by tracing rays through the dense mesh of the corresponding session. This yields 3D-3D correspondences $\{\mathbf{P}_i^A, \mathbf{P}_j^B\}$, from which we estimate an initial relative pose [80] using RANSAC [24]. This pose is refined with the point-to-plane Iterative Closest Point (ICP) algorithm [60] applied to the pair of lidar point clouds.

We use state-of-the-art local image features that can match across drastic illumination and viewpoint changes [61,20,58]. Combined with the strong geometric constraints in the registration, our system is robust to long-term temporal changes and does not require manual initialization. Using this approach, we have successfully registered building-scale scans captured at more than a year of interval with large structural changes.

Global alignment: We gather all pairwise constraints and jointly refine all absolute scan poses $\{{}_w\mathbf{T}_0^S\}$ by optimizing a pose graph [27]. The edges are weighted with the covariance matrices of the pair-wise ICP estimates. The images of all scan sessions are finally combined into a unique reference trajectory $\{{}_w\mathbf{T}_i^{\text{ref}}\}$. The point clouds and meshes are aligned according to the same transformations. They define the reference representation of the scene, which we use as a basis to obtain GT for the AR sequences.

Ground-truth visibility: The accurate and dense 3D geometry of the mesh allows us to compute accurate visual overlap between two cameras with known poses and calibration. Inspired by Rau et al. [55], we define the overlap of image i wrt. a reference image j by

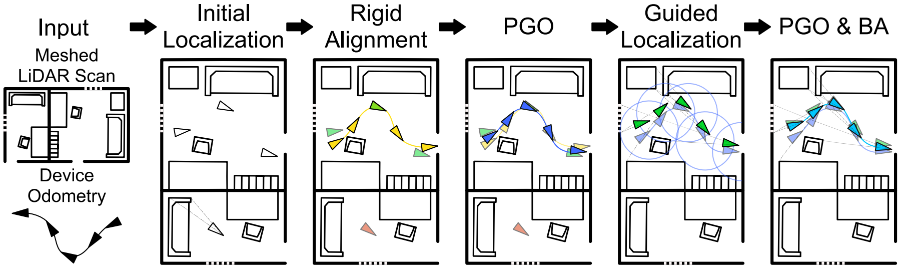

Fig. 3. Sequence-to-scan alignment. We first estimate the absolute pose of each sequence frame using image retrieval and matching. This initial localization prior is used to obtain a single rigid alignment between the input trajectory and the reference 3D model via voting. The alignment is then relaxed by optimizing the individual frame poses in a pose graph based on both relative and absolute pose constraints. We bootstrap this initialization by mining relevant image pairs and re-localizing the queries. Given these improved absolute priors, we optimize the pose graph again and finally include reprojection errors of the visual correspondences, yielding a refined trajectory.

the ratio of pixels in i that are visible in j:

$$O(i \rightarrow j) = \frac{\sum_{k \in (W,H)} \mathbb{1}\left[\Pi_j({}_w\mathbf{T}_j, \Pi_i^{-1}({}_w\mathbf{T}_i, \mathbf{p}_k^i, z_k)) \in (W,H)\right]\ \alpha_k}{W \cdot H}\ , \quad (1)$$

where Π_i projects a 3D point k to camera i, Π_i^{-1} conversely backprojects it using its known depth z_k with (W, H) as the image dimensions. The contribution of each pixel is weighted by the angle $\alpha_k = \cos(\mathbf{n}_{i,k}, \mathbf{n}_{j,k})$ between the two rays. To handle scale changes, it is averaged both ways $i \rightarrow j$ and $j \rightarrow i$. This score is efficiently computed by tracing rays through the mesh and checking for occlusion for robustness.

This score $O \in [0, 1]$ favors images that observe the same scene from similar viewpoints. Unlike sparse co-visibility in an SfM model [54], our formulation is independent of the amount of texture and the density of the feature detections. This score correlates with matchability – we thus use it as GT when evaluating retrieval and to determine an upper bound on the theoretically achievable performance of our benchmark.

4.2 Sequence-to-scan alignment

We now aim to register each AR sequence individually into the dense GT reference model (see Fig. 3). Given a sequence of n frames, we introduce a simple algorithm that estimates the per-frame absolute pose $\{{}_w\mathbf{T}_i\}_{1 \leq i \leq n}$. A frame refers to an image taken at a given time or, when the device is composed of a camera rig with known calibration (e.g., HoloLens), to a collection of simultaneously captured images.

Inputs: We assume given trajectories $\{{}_0\mathbf{T}_i^{\text{track}}\}$ estimated by a visual-inertial tracker – we use ARKit for iPhone/iPad and the on-device tracker for HoloLens. The tracker also outputs per-frame camera intrinsics $\{\mathbf{C}_i\}$, which account for auto-focus or calibration changes and are for now kept fixed.

Initial localization: For each frame of a sequence $\{I_i^{\text{query}}\}$, we retrieve a fixed number r of relevant reference images $(I_j^{\text{ref}})_{1 \leq j \leq r}$ using global image descriptors. We match sparse local features [43,20,58] extracted in the query frame to each retrieved image I_j^{ref}

obtaining a set of 2D-2D correspondences $\{\mathbf{p}^{\text{q}}_{i,k}, \mathbf{p}^{\text{ref}}_{j,k}\}_k$. The 2D reference keypoints are lifted to 3D by tracing rays through the mesh of the reference model, yielding a set of 2D-3D correspondences $\mathcal{M}_{i,j} := \{\mathbf{p}^{\text{q}}_{i,k}, \mathbf{P}^{\text{ref}}_{j,k}\}_k$. We combine all matches per query frame $\mathcal{M}_i = \cup_{j=1}^{r} \mathcal{M}_{i,j}$ and estimate an initial absolute pose ${}_w\mathbf{T}^{\text{loc}}_i$ using the (generalized) P3P algorithm [29] within a LO-RANSAC scheme [16] followed by a non-linear refinement [67]. Because of challenging appearance conditions, structural changes, or lack of texture, some frames cannot be localized in this stage. We discard all poses that are supported by a low number of inlier correspondences.

Rigid alignment: We next recover a coarse initial pose $\{{}_w\mathbf{T}^{\text{init}}_i\}$ for all frames, including those that could not be localized. Using the tracking, which is for now assumed drift-free, we find the rigid alignment ${}_w\mathbf{T}^{\text{init}}_0$ that maximizes the consensus among localization poses. This voting scheme is fast and effectively rejects poses that are incorrect, yet confident, due to visual aliasing and symmetries. Each estimate is a candidate transformation ${}_w\mathbf{T}^{i}_0 = {}_w\mathbf{T}^{\text{loc}}_i \left({}_0\mathbf{T}^{\text{track}}_i\right)^{-1}$, for which other frames can vote, if they are consistent within a threshold τ_{rigid}. We select the candidate with the highest count of inliers:

$$ {}_w\mathbf{T}^{\text{init}}_0 = \underset{\mathbf{T} \in \{{}_w\mathbf{T}^{i}_0\}_{1\le i\le n}}{\arg\max} \sum_{1\le j\le n} \mathbb{1}\left[\text{dist}\left({}_w\mathbf{T}^{\text{loc}}_j, \mathbf{T}\cdot {}_0\mathbf{T}^{\text{track}}_j\right) < \tau_{\text{rigid}}\right] \quad , \tag{2} $$

where $\mathbb{1}\left[\cdot\right]$ is the indicator function and $\text{dist}\left(\cdot,\cdot\right)$ returns the magnitude, in terms of translation and rotation, of the difference between two absolute poses. We then recover the per-frame initial poses as $\{{}_w\mathbf{T}^{\text{init}}_i := {}_w\mathbf{T}^{\text{init}}_0 \cdot {}_0\mathbf{T}^{\text{track}}_i\}_{1\le i\le n}$.

Pose graph optimization: We refine the initial absolute poses by maximizing the consistency of tracking and localization cues within a pose graph. The refined poses $\{{}_w\mathbf{T}^{\text{PGO}}_i\}$ minimize the energy function

$$ E(\{{}_w\mathbf{T}_i\}) = \sum_{i=1}^{n-1} \mathcal{C}_{\text{PGO}}\left({}_w\mathbf{T}^{-1}_{i+1}\, {}_w\mathbf{T}_i,\ {}_{i+1}\mathbf{T}^{\text{track}}_i\right) + \sum_{i=1}^{n} \mathcal{C}_{\text{PGO}}\left({}_w\mathbf{T}_i,\ {}_w\mathbf{T}^{\text{loc}}_i\right) \quad , \tag{3} $$

where $\mathcal{C}_{\text{PGO}}\left(\mathbf{T}_1, \mathbf{T}_2\right) := \left\|\text{Log}\left(\mathbf{T}_1\ \mathbf{T}_2^{-1}\right)\right\|^2_{\Sigma,\gamma}$ is the distance between two absolute or relative poses, weighted by covariance matrix $\Sigma \in \mathbb{R}^{6\times6}$ and loss function γ. Here, Log maps from the Lie group $\text{SE}(3)$ to the corresponding algebra $\mathfrak{se}(3)$.

We robustify the absolute term with the Geman-McClure loss function and anneal its scale via a Graduated Non-Convexity scheme [87]. This ensures convergence in case of poor initialization, e.g., when the tracking exhibits significant drift, while remaining robust to incorrect localization estimates. The covariance of the absolute term is propagated from the preceding non-linear refinement performed during localization. The covariance of the relative term is recovered from the odometry pipeline, or, if not available, approximated as a factor of the motion magnitude.

This step can fill the gaps from the localization stage using the tracking information and conversely correct for tracker drift using localization cues. In rare cases, the resulting poses might still be inaccurate when both the tracking drifts and the localization fails.

Guided localization via visual overlap: To further increase the pose accuracy, we leverage the current pose estimates $\{{}_w\mathbf{T}^{\text{PGO}}_i\}$ to mine for additional localization cues.

Instead of relying on global visual descriptors, which are easily affected by aliasing, we select reference images with a high overlap using the score defined in Section 4.1. For each sequence frame i, we select r reference images with the largest overlap and again match local features and estimate an absolute pose. These new localization priors improve the pose estimates in a second optimization of the pose graph.

Bundle adjustment: For each frame i, we recover the set of 2D-3D correspondences $\mathcal{M}_i$ used by the guided re-localization. We now refine the poses $\{{}_w\mathbf{T}_i^{\text{BA}}\}$ by jointly minimizing a bundle adjustment problem with relative pose graph costs:

$$\begin{aligned} E(\{{}_w\mathbf{T}_i\}) = &\sum_{i=1}^{n-1} \mathcal{C}_{\text{PGO}}\left({}_w\mathbf{T}_{i+1}^{-1}\,{}_w\mathbf{T}_i,\ {}_{i+1}\mathbf{T}_i^{\text{track}}\right) \\ &+ \sum_{i=1}^{n} \sum_{\mathcal{M}_{i,j}\in\mathcal{M}_i} \sum_{(\mathbf{p}_k^{\text{ref}},\mathbf{P}_k^{\text{q}})\in\mathcal{M}_{i,j}} \left\| \Pi({}_w\mathbf{T}_i, \mathbf{P}_{j,k}^{\text{ref}}) - \mathbf{p}_{i,k}^{\text{q}} \right\|_{\sigma^2}^2 \quad , \end{aligned} \tag{4}$$

where the second term evaluates the reprojection error of a 3D point $\mathbf{P}_{j,k}^{\text{ref}}$ for observation k to frame i. The covariance is the noise σ^2 of the keypoint detection algorithm. We pre-filter correspondences that are behind the camera or have an initial reprojection error greater than $\sigma\,\tau_{\text{reproj}}$. As the 3D points are sampled from the lidar, we also optimize them with a prior noise corresponding to the lidar specifications. We use the Ceres [1] solver.

4.3 Joint global refinement

Once all sequences are individually aligned, we refine them jointly by leveraging sequence-to-sequence visual observations. This is helpful when sequences observe parts of the scene not mapped by the LiDAR. We first triangulate a sparse 3D model from scan images, aided by the mesh. We then triangulate additional observations, and finally jointly optimize the whole problem.

Reference triangulation: We estimate image correspondences of the reference scan using pairs selected according to the visual overlap defined in Section 4.2. Since the image poses are deemed accurate and fixed, we filter the correspondences using the known epipolar geometry. We first consider feature tracks consistent with the reference surface mesh before triangulating more noisy observations within LO-RANSAC using COLMAP [67]. The remaining feature detections, which could not be reliably matched or triangulated, are lifted to 3D by tracing through the mesh. This results in an accurate, sparse SfM model with tracks across reference images.

Sequence optimization: We then add each sequence to the sparse model. We first establish correspondences between images of the same and of different sequences. The image pairs are again selected by highest visual overlap computed using the aligned poses $\{{}_w\mathbf{T}_i^{\text{BA}}\}$. The resulting tracks are sequentially triangulated, merged, and added to the sparse model. Finally, all 3D points and poses are jointly optimized by minimizing the joint pose-graph and bundle adjustment (Equation 4). As in COLMAP [67], we alternate optimization and track merging. To scale to large scenes, we subsample keyframes from the full frame-rate captures and only introduce absolute pose and reprojection constraints for keyframes while maintaining all relative pose constraints from tracking.

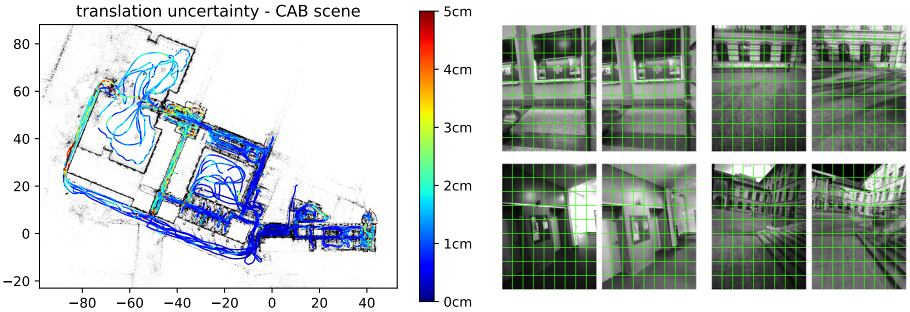

Fig. 4. Uncertainty of the GT poses for the CAB scene. Left: The overhead map shows that the uncertainties are larger in long corridors and outdoor spaces. Right: Pixel-aligned renderings at the estimated camera poses confirm that the poses are sufficiently accurate for our evaluation.

4.4 Ground-truth validation

Potential limits: Brachmann et al. [8] observe that algorithms generating pseudo-GT poses by minimizing either 2D or 3D cost functions alone can yield noticeably different results. We argue that there exists a single underlying, true GT. Reaching it requires fusing large amounts of redundant data with sufficient sensors of sufficiently low noise. Our GT poses optimize complementary constraints from visual and inertial measurements, guided by an accurate lidar-based 3D structure. Careful design and propagation of uncertainties reduces the bias towards one of the sensors. All sensors are factory- and self-calibrated during each recording by the respective commercial, production-grade SLAM algorithms. We do not claim that our GT is perfect but analyzing the optimization uncertainties sheds light on its degree of accuracy.

Pose uncertainty: We estimate the uncertainties of the GT poses by inverting the Hessian of the refinement. To obtain calibrated covariances, we scale them by the empirical keypoint detection noise, estimated as $\sigma{=}1.33$ pixels for the CAB scene. The maximum noise in translation is the size of the major axis of the uncertainty ellipsoids, which is the largest eivenvalue σ_t^2 of the covariance matrices. Fig. 4 shows its distribution for the CAB scene. We retain images whose poses are correct within 10cm with a confidence of 99.7%. For normally distributed errors, this corresponds to a maximum uncertainty $\sigma_t{=}3.33$cm and discards 3.9% of the queries. For visual inspection, we render images at the estimated GT camera poses using the colored mesh. They appear pixel-aligned with the original images, supporting that the poses are accurate.

4.5 Selection of mapping and query sequences

We divide the set of sequences into two disjoint groups for mapping (database) and localization (query). Database sequences are selected such that they have a minimal overlap between each other yet cover the area visited by all remaining sequences. This simulates a scenario of minimal coverage and maximizes the number of query sequences. We cast this as a combinatorial optimization problem solved with a depth-first search guided by some heuristics. We provide more details in the supp. material.

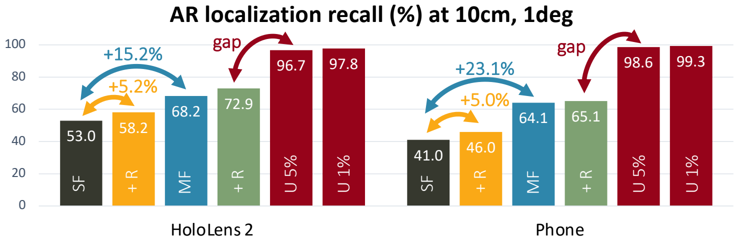

Fig. 5. Main results. We show results for NetVLAD image retrieval with SuperPoint local features and SuperGlue matcher on both HoloLens 2 and phone queries. We consider several tracks: single-frame (SF) localization with / without radios (R) and similarly for multi-frame (MF) localization. In addition, we report a theoretical upper bound (U): the percentage of queries with at least 5% / 1% ground-truth overlap with respect to the best database image.

5 Evaluation

We evaluate state-of-the-art approaches in both single- and multi-frame settings and summarize our results in Figure 5.

Single-frame: We first consider in Sec. 5.1 the classical academic setup of single-frame queries (single image for phones and single rig for HoloLens 2) without additional sensor. We then look at how radio signals can be beneficial. We also analyze the impact of various settings: FOV, type of mapping images, and mapping algorithm.

Multi-frame: Second, by leveraging the real-time AR tracking poses, we consider the problem of multi-frame localization in Sec. 5.2. This corresponds to a real-world AR application retrieving the content attached to a target map using the real-time sensor stream from the device. In this context, we not only care about accuracy and recall but also about the time required to localize accurately, which we call the *time-to-recall*.

5.1 Single-frame localization

We first evaluate several algorithms representative of the state of the art in the classical single-frame academic setup. We consider the hierarchical localization framework with different approaches for image retrieval and matching. Each of them first builds a sparse SfM map from reference images. For each query frame, we then retrieve relevant reference images, match their local features, lift the reference keypoints to 3D using the sparse map, and finally estimate a pose with PnP+RANSAC. We report the recall of the final pose at two thresholds [65]: 1) a fine threshold at $\{1°, 10\text{cm}\}$, which we see as the minimum accuracy required for a good AR user experience in most settings. 2) a coarse threshold at $\{5°, 1\text{m}\}$ to show the room for improvement for current approaches.

We evaluate global descriptors computed by NetVLAD [3] and by a fusion [31] of NetVLAD and APGeM [57], which are representative of the field [52]. We retrieve the 10 most similar images. For matching, we evaluate handcrafted SIFT [43], SOSNet [77] as a learned patch descriptor extracted from DoG [43] keypoints, and a robust deep-learning based joint detector and descriptor R2D2 [58]. Those are matched by exact mutual nearest neighbor search. We also evaluate SuperGlue [62] – a learned matcher based on SuperPoint [20] features. To build the map, we retrieve neighboring images

Hierarchical localization		Query device	
Retrieval	Matching	HL2	Phone
NetVLAD	SIFT	30.3 / 41.4	28.6 / 42.3
	DoG+SOSNet	31.6 / 43.3	29.8 / 45.7
	R2D2	38.9 / 51.3	40.6 / 57.3
	SP+SG	46.3 / 59.8	49.3 / 62.8
Fusion	SIFT	32.8 / 47.0	29.0 / 43.6
	DoG+SOSNet	34.5 / 48.9	30.4 / 46.4
	R2D2	43.0 / 57.8	40.4 / 57.7
	SP+SG	52.4 / 67.3	50.2 / 64.3

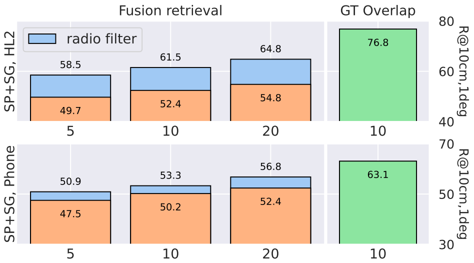

Table 3. Left: single-frame localization. We report the recall at (1°, 10cm)/(5°, 1m) for baselines representative of the state of the art. Our dataset is challenging while most others are saturated. There is a clear progress from SIFT but also large room for improvement. **Right: localization with radio signals.** Increasing the number {5, 10, 20} of retrieved images increases the localization recall at (1°, 10cm). The best-performing visual retrieval (Fusion, orange) is however far worse than the GT overlap. Filtering with radio signals (blue) improves the performance in all settings.

using NetVLAD filtered by frustum intersection from reference poses, match these pairs, and triangulate a sparse SfM model using COLMAP [67].

We report the results in Table 3 (left). Even the best methods have a large gap to perfect scores and much room for improvement. In the remaining ablation, we solely rely on SuperPoint+SuperGlue [20,62] for matching as it clearly performs the best.

Leveraging radio signals: In this experiment, we show that radio signals can be used to constrain the search space for image retrieval. This has two main benefits: 1) it reduces the risk of incorrectly considering visual aliases, and 2) it lowers the compute requirements by reducing that numbers of images that need to be retrieved and matched. We implement this filtering as follows. We first split the scene into a sparse 3D grid considering only voxels containing at least one mapping frame. For each frame, we gather all radio signals in a ±10s window and associate them to the corresponding voxel. If the same endpoint is observed multiple times in a given voxel, we average the received signal strengths (RSSI) in dBm. For a query frame, we similarly aggregate signals over the past 10s and rank voxels by their L2 distance between RSSIs, considering those with at least one common endpoint. We thus restrict image retrieval to 5% of the map.

Tab. 3 (right) shows that radio filtering always improves the localization accuracy over vanilla vision-only retrieval, irrespective of how many images are matches. The upper bound based on the GT overlap (defined in Sec. 4.1) shows that there is still much room for improvement for both image and radio retrieval. As the GT overlap baseline is far from the perfect 100% recall, frame-to-frame matching and pose estimation have also much room to improve.

Varying field-of-view: We study the impact of the FOV of the HoloLens 2 device via two configurations: 1) Each camera in a rig is seen as a single-frame and localized using LO-RANSAC + P3P. 2) We consider all four cameras in a frame and localize them together using the generalized solver GP3P. NetVLAD retrieval with SuperPoint and SuperGlue only achieves 36.6% / 45.8% recall, compared to the results from Tab. 3 (46.3% / 59.8%). Rig localization thus provides much better performance, mainly in hard cases where single cameras face texture-less areas, such as the ground and walls.

Mapping images →		HL2 + Phone			HD 360	Both
Image pairs from →		Retrieval		GT overlap	Retrieval + Poses	Retrieval + Poses
Matching	Device	NetVLAD	+ Poses			
SP + SG	HL2	46.6 / 59.6	46.3 / 59.8	47.4 / 60.2	69.3 / 81.8	68.6 / 80.3
	Phone	48.8 / 63.3	49.3 / 62.8	49.9 / 63.0	48.2 / 63.3	51.1 / 65.4

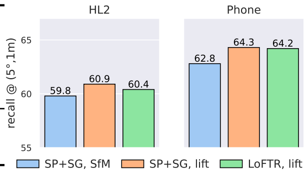

Table 4. Impact of mapping. Left: Scenarios. Building the map with sparse HD 360 images from the NavVis rig, instead of or with dense AR sequences, boosts the localization performance for HL2 as it makes image retrieval easier – NetVLAD tends to incorrectly retrieve same-device HL images over same-location phone images. This does not help phone localization, likely due to the viewpoint sparsity. **Right: Modalities.** Lifting 2D points to 3D using the lidar mesh instead of triangulating with SfM is beneficial. This can also leverage dense matching, e.g. with LoFTR.

Mapping modality: We study whether the high-quality lidar mesh can be used for localization. We consider two approaches to obtain a sparse 3D point cloud: 1) By triangulating sparse visual correspondences across multiple views. 2) By lifting 2D keypoints in reference images to 3D by tracing rays through the mesh. Lifting can leverage dense correspondences, which cannot be efficiently triangulated with conventional multi-view geometry. We thus compare 1) and 2) with SuperGlue to 2) with LoFT [74], a state-of-the-art dense matcher. The results (Tab. 4 right) show that the mesh brings some improvements. Points could also be lifted by dense depth from multi-view stereo. We however did not obtain satisfactory results with a state-of-the-art approach [84] as it cannot handle very sparse mapping images.

Mapping scenario: We study the accuracy of localization against maps built from different types of images: 1) crowd-sourced, dense AR sequences; 2) curated, sparser HD 360 images from the NavVis device; 3) a combination of the two. The results are summarized in Tab. 4 (left), showing that the mapping scenario has a large impact on the final numbers. On the other hand, image pair selection for mapping matters little. Current crowd-sourcing approaches do not yield as good results as capturing a space using a specialized scanning device at high density. Further, crowd-sourcing and manual scans can complement each other. We hope that future work can close the gap between the scenarios to achieve better metrics from crowd-sourced data without curation.

5.2 Multi-frame localization

Inspired by typical AR use cases, we consider the problem of multi-frame localization in this section. The task is to align multiple consecutive frames of varied lengths and aggregated radio signals against the database map. Our baseline for this task is based on the ground-truthing pipeline and has as such relatively high compute requirements. However, we are primarily interested to demonstrate the potential performance gains by leveraging multiple frames. First, we run image retrieval and single-frame localization, followed by a first PGO with tracking and localization poses. Then, we do a second localization with retrieval guided by the poses of the first PGO, followed by a second PGO. Finally, we run a pose refinement by considering reprojections to query frames and tracking cost. Additionally, we can also use radio to restrict image retrieval throughout the pipeline. We keep the same accuracy metric as before, considering only the last frame in each multi-frame query, which is the one that influences the current AR user experience in a real-time scenario.

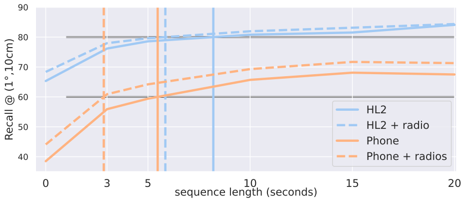

Fig. 6. Multi-frame localization. We report the localization recall of SuperPoint+SuperGlue as we increase the duration of each sequence. The pipeline leverages both on-device tracking and absolute retrieval, as vision-only (solid) or combined with radio signals (dashed). We show the TTR@80% for HL2 (blue) and TTR@60% for phone queries (orange). Using radio signals reduce the TTR from 8s to 5s and from 5s to 3s, respectively.

We evaluate various query sizes and introduce the *time-to-recall* metric as: sequence length (time) until successful localization at X% (recall) for a tight threshold (1°, 10cm) (TTR@X%). Methods should aim to minimize this metric to render retrieved content as quickly as possible after starting an AR experience. We show the results for the CAB scene in Figure 6. While the performance of current methods is not satisfactory yet to achieve a TTR@90% under 20 seconds, using multi-frame localization leads to significant gains of 20-40%. The radio signals improve the performance in particular with shorter sequences and thus effectively reduce time-to-recall.

6 Conclusion

In this paper, we identified several key limitations of current localization and mapping benchmarks that make them unrealistic in the context of AR. To address these limitations, we developed a new GT pipeline to accurately and robustly register realistic AR scenario captures in large and diverse scenes against laser scans without any manual labelling or setup of custom infrastructure. With this new benchmark, initially consisting of 3 large locations (note that we will add more locations over time), we revisited the traditional academic setup and showed a large performance gap for existing state-of-the-art methods when evaluated using our more realistic and challenging dataset. By implementation of simple yet representative baselines to take advantage of the AR specific setup, we present novel insights and pave several avenues of future work. In particular, we showed huge potential for leveraging query sequences instead of single frames as well as leveraging other sensor modalities like radio signals or depth data in the localization and mapping problem. Furthermore, we hope to direct research attention to not only tackle the localization problem in isolation but also improve map representations as well as consider the currently largely ignored time-to-recall metric. The dataset and the source code of the GT pipeline will be available to the community. We will also host an evaluation server to facilitate benchmarking of future work.

Acknowledgements. This paper would not have been possible without the hard work and contributions of Gabriela Evrova, Silvano Galliani, Michael Baumgartner, Cedric Cagniart, Jeffrey Delmerico, Jonas Hein, Dawid Jeczmionek, Mirlan Karimov, Maximilian Mews, Patrick Misteli, Juan Nieto, Sònia Batllori Pallarès, Rémi Pautrat, Songyou Peng, Iago Suarez, Rui Wang, Jeremy Wanner, Silvan Weder and our colleagues in CVG at ETH Zürich and the wider Microsoft Mixed Reality & AI team.

References

1. Agarwal, S., Mierle, K., Others: Ceres solver. http://ceres-solver.org 9
2. Arandjelovic, R.: Three things everyone should know to improve object retrieval. In: CVPR (2012) 3
3. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: NetVLAD: CNN architecture for weakly supervised place recognition. In: Proc. CVPR (2016) 3, 6, 11
4. Badino, H., Huber, D., Kanade, T.: The CMU Visual Localization Data Set. http://3dvis.ri.cmu.edu/data-sets/localization (2011) 4
5. Bahl, P., Padmanabhan, V.N.: Radar: An in-building rf-based user location and tracking system. In: INFOCOM (2000) 3
6. Balntas, V., Lenc, K., Vedaldi, A., Mikolajczyk, K.: HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In: Proc. CVPR (2017) 2, 4
7. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). CVIU (2008) 3
8. Brachmann, E., Humenberger, M., Rother, C., Sattler, T.: On the limits of pseudo ground truth in visual camera re-localisation. In: Proc. ICCV (2021) 2, 4, 10
9. Brachmann, E., Rother, C.: Expert sample consensus applied to camera re-localization. In: ICCV (2019) 3
10. Brachmann, E., Rother, C.: Visual camera re-localization from rgb and rgb-d images using dsac. T-PAMI (2021) 3
11. Cao, B., Araujo, A., Sim, J.: Unifying deep local and global features for image search. In: ECCV (2020) 3
12. Carlevaris-Bianco, N., Ushani, A.K., Eustice, R.M.: University of Michigan North Campus long-term vision and lidar dataset. International Journal of Robotics Research (2015) 2, 3, 4
13. Chan, Y.T., Tsui, W.Y., So, H.C., chung Ching, P.: Time-of-arrival based localization under nlos conditions. IEEE Transactions on Vehicular Technology (2006) 3
14. Chen, D.M., Baatz, G., Köser, K., Tsai, S.S., Vedantham, R., Pylvänäinen, T., Roimela, K., Chen, X., Bach, J., Pollefeys, M., Girod, B., Grzeszczuk, R.: City-scale landmark identification on mobile devices. In: CVPR (2011) 3, 4
15. Choi, S., Zhou, Q.Y., Koltun, V.: Robust reconstruction of indoor scenes. In: CVPR. pp. 5556–5565 (2015) 6
16. Chum, O., Matas, J., Kittler, J.: Locally optimized RANSAC. In: Joint Pattern Recognition Symposium. pp. 236–243 (2003) 8
17. Cohen-Steiner, D., Da, F.: A greedy delaunay-based surface reconstruction algorithm. The visual computer **20**(1), 4–16 (2004) 6
18. Comsa, C.R., Luo, J., Haimovich, A., Schwartz, S.: Wireless localization using time difference of arrival in narrow-band multipath systems. In: 2007 International Symposium on Signals, Circuits and Systems (2007) 3
19. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: CVPR (2017) 3
20. DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperPoint: Self-Supervised Interest Point Detection and Description. In: CVPR workshops (2018) 3, 6, 7, 11, 12
21. Dusmanu, M., Miksik, O., Schönberger, J.L., Pollefeys, M.: Cross-Descriptor Visual Localization and Mapping. In: ICCV (2021) 2
22. Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., Sattler, T.: D2-Net: A Trainable CNN for Joint Detection and Description of Local Features. In: CVPR (2019) 3
23. Dusmanu, M., Schönberger, J.L., Sinha, S., Pollefeys, M.: Privacy-Preserving Image Features via Adversarial Affine Subspace Embeddings. In: CVPR (2021) 2

24. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM (1981) 3, 6
25. Geppert, M., Larsson, V., Speciale, P., Schönberger, J.L., Pollefeys, M.: Privacy preserving structure-from-motion. In: ECCV (2020) 2
26. Geppert, M., Larsson, V., Speciale, P., Schonberger, J.L., Pollefeys, M.: Privacy preserving localization and mapping from uncalibrated cameras. In: CVPR (2021) 2
27. Grisetti, G., Kümmerle, R., Stachniss, C., Burgard, W.: A tutorial on graph-based slam. IEEE Intelligent Transportation Systems Magazine **2**(4), 31–43 (2010) 6
28. He, S., Chan, S.H.G.: Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons. IEEE Communications Surveys Tutorials (2016) 3
29. Hee Lee, G., Li, B., Pollefeys, M., Fraundorfer, F.: Minimal solutions for pose estimation of a multi-camera system. In: The International Journal of Robotics Research, pp. 521–538. Springer (2016) 3, 8
30. Hodan, T., Michel, F., Brachmann, E., Kehl, W., GlentBuch, A., Kraft, D., Drost, B., Vidal, J., Ihrke, S., Zabulis, X., et al.: Bop: Benchmark for 6d object pose estimation. In: ECCV (2018) 3, 4
31. Humenberger, M., Cabon, Y., Guerin, N., Morat, J., Revaud, J., Rerole, P., Pion, N., de Souza, C., Leroy, V., Csurka, G.: Robust image retrieval-based visual localization using kapture. arXiv preprint arXiv:2007.13867 (2020) 11
32. Hyeon, J., Kim, J., Doh, N.: Pose correction for highly accurate visual localization in large-scale indoor spaces. In: ICCV (2021) 3
33. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: CVPR (2010) 3, 6
34. Jin, Y., Mishkin, D., Mishchuk, A., Matas, J., Fua, P., Yi, K.M., Trulls, E.: Image Matching across Wide Baselines: From Paper to Practice. International Journal of Computer Vision (2020) 2, 3, 4
35. Johns, E., Yang, G.Z.: Feature co-occurrence maps: Appearance-based localisation throughout the day. ICRA (2013) 3
36. Kendall, A., Cipolla, R.: Geometric loss functions for camera pose regression with deep learning. In: CVPR (2017) 3
37. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: ICCV (2015) 2, 3, 4
38. Khalajmehrabadi, A., Gatsis, N., Akopian, D.: Modern wlan fingerprinting indoor positioning methods and deployment challenges (2016) 3
39. Laoudias, C., Michaelides, M.P., Panayiotou, C.G.: Fault detection and mitigation in WLAN RSS fingerprint-based positioning. Journal of Location Based Services (2012) 3
40. Lee, D., Ryu, S., Yeon, S., Lee, Y., Kim, D., Han, C., Cabon, Y., Weinzaepfel, P., Guérin, N., Csurka, G., Humenberger, M.: Large-scale localization datasets in crowded indoor spaces. In: CVPR (2021) 2, 3, 4
41. Li, X., Ylioinas, J., Verbeek, J., Kannala, J.: Scene coordinate regression with angle-based reprojection loss for camera relocalization. In: ECCV workshop (2018) 3
42. Li, Y., Snavely, N., Huttenlocher, D., Fua, P.: Worldwide pose estimation using 3d point clouds. In: ECCV (2012) 3
43. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV (2004) 3, 7, 11
44. Maddern, W., Pascoe, G., Linegar, C., Newman, P.: 1 Year, 1000km: The Oxford RobotCar Dataset. IJRR (2017) 4
45. Massiceti, D., Krull, A., Brachmann, E., Rother, C., Torr, P.H.S.: Random forests versus neural networks - what's best for camera localization? In: ICRA (2017) 3
46. Meng, L., Chen, J., Tung, F., Little, J.J., Valentin, J., de Silva, C.W.: Backtracking regression forests for accurate camera relocalization. In: IROS (2017) 3

47. Mikolajczyk, K., Schmid, C.: Scale and affine invariant interest point detectors. IJCV (2004) 3
48. Milford, M.J., Wyeth, G.F.: Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights. In: ICRA (2012) 3
49. Mishchuk, A., Mishkin, D., Radenovic, F., Matas, J.: Working hard to know your neighbor's margins: Local descriptor learning loss. In: NeurIPS (2017) 3
50. Ng, T., Lopez-Rodriguez, A., Balntas, V., Mikolajczyk, K.: Reassessing the limitations of cnn methods for camera pose regression. arXiv (2021) 3
51. Peng, R., Sichitiu, M.L.: Angle of arrival localization for wireless sensor networks. In: 2006 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks (2006) 3
52. Pion, N., Humenberger, M., Csurka, G., Cabon, Y., Sattler, T.: Benchmarking image retrieval for visual localization. In: 3DV (2020) 11
53. Pless, R.: Using many cameras as one. In: CVPR (2003) 3
54. Radenović, F., Tolias, G., Chum, O.: Fine-tuning CNN image retrieval with no human annotation. T-PAMI **41**(7), 1655–1668 (2018) 7
55. Rau, A., Garcia-Hernando, G., Stoyanov, D., Brostow, G.J., Turmukhambetov, D.: Predicting visual overlap of images through interpretable non-metric box embeddings. In: ECCV (2020) 3, 6
56. Reina, S.C., Solin, A., Rahtu, E., Kannala, J.: ADVIO: an authentic dataset for visual-inertial odometry. In: ECCV (2018), http://arxiv.org/abs/1807.09828 2, 3, 4
57. Revaud, J., Almazán, J., de Rezende, R.S., de Souza, C.R.: Learning with average precision: Training image retrieval with a listwise loss. ICCV (2019) 3, 6, 11
58. Revaud, J., Weinzaepfel, P., de Souza, C.R., Humenberger, M.: R2D2: repeatable and reliable detector and descriptor. In: NeurIPS (2019) 6, 7, 11
59. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: an efficient alternative to SIFT or SURF. In: ICCV (2011) 3
60. Rusinkiewicz, S., Levoy, M.: Efficient variants of the icp algorithm. In: 3DIM (2001) 6
61. Sarlin, P.E., Cadena, C., Siegwart, R., Dymczyk, M.: From coarse to fine: Robust hierarchical localization at large scale. In: CVPR (2019) 3, 6
62. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperGlue: Learning feature matching with graph neural networks. In: CVPR (2020) 3, 11, 12
63. Sarlin, P.E., Unagar, A., Larsson, M., Germain, H., Toft, C., Larsson, V., Pollefeys, M., Lepetit, V., Hammarstrand, L., Kahl, F., Sattler, T.: Back to the Feature: Learning Robust Camera Localization from Pixels to Pose. In: CVPR (2021) 3
64. Sattler, T., Leibe, B., Kobbelt, L.: Improving image-based localization by active correspondence search. In: ECCV (2012) 3
65. Sattler, T., Maddern, W., Toft, C., Torii, A., Hammarstrand, L., Stenborg, E., Safari, D., Okutomi, M., Pollefeys, M., Sivic, J., Kahl, F., Pajdla, T.: Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions. In: CVPR (2018) 2, 3, 4, 11
66. Sattler, T., Weyand, T., Leibe, B., Kobbelt, L.P.: Image retrieval for image-based localization revisited. In: BMVC (2012) 2, 3, 4
67. Schönberger, J., Frahm, J.M.: Structure-from-Motion Revisited. In: CVPR (2016) 2, 3, 4, 8, 9, 12
68. Schönberger, J.L., Hardmeier, H., Sattler, T., Pollefeys, M.: Comparative Evaluation of Hand-Crafted and Learned Local Features. In: Proc. CVPR (2017) 2, 4
69. Schönberger, J.L., Pollefeys, M., Geiger, A., Sattler, T.: Semantic Visual Localization. In: CVPR (2018) 3
70. Schops, T., Schönberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: CVPR (2017) 2, 3, 4

71. Shibuya, M., Sumikura, S., Sakurada, K.: Privacy preserving visual slam. In: ECCV (2020) 2
72. Shotton, J., Glocker, B., Zach, C., Izadi, S., Criminisi, A., Fitzgibbon, A.: Scene coordinate regression forests for camera relocalization in rgb-d images. In: CVPR (2013) 2, 3, 4
73. Speciale, P., Schönberger, J.L., Kang, S.B., Sinha, S.N., Pollefeys, M.: Privacy preserving image-based localization. In: CVPR (2019) 2, 3
74. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: LoFTR: Detector-free local feature matching with transformers. CVPR (2021) 13
75. Sun, X., Xie, Y., Luo, P., Wang, L.: A dataset for benchmarking image-based localization. In: CVPR (2017) 2, 3, 4
76. Taira, H., Okutomi, M., Sattler, T., Cimpoi, M., Pollefeys, M., Sivic, J., Pajdla, T., Torii, A.: Inloc: Indoor visual localization with dense matching and view synthesis. In: CVPR (2018) 2, 3, 4
77. Tian, Y., Yu, X., Fan, B., Wu, F., Heijnen, H., Balntas, V.: SOSNet: Second Order Similarity Regularization for Local Descriptor Learning. In: CVPR (2019) 3, 11
78. Tolias, G., Avrithis, Y., Jégou, H.: To aggregate or not to aggregate: Selective match kernels for image search. In: ICCV (2013) 3
79. Torii, A., Arandjelović, R., Sivic, J., Okutomi, M., Pajdla, T.: 24/7 place recognition by view synthesis. In: CVPR (2015) 3
80. Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns. IEEE Transactions on Pattern Analysis & Machine Intelligence **13**(04), 376–380 (1991) 6
81. Ungureanu, D., Bogo, F., Galliani, S., Sama, P., Duan, X., Meekhof, C., Stühmer, J., Cashman, T.J., Tekin, B., Schönberger, J.L., Olszta, P., Pollefeys, M.: Hololens 2 research mode as a tool for computer vision research (2020) 5
82. Valentin, J., Niessner, M., Shotton, J., Fitzgibbon, A., Izadi, S., Torr, P.H.S.: Exploiting uncertainty in regression forests for accurate camera relocalization. In: CVPR (2015) 3
83. Wald, J., Sattler, T., Gododetz, S., Cavallari, T., Tombari, F.: Beyond controlled environments: 3d camera re-localization in changing indoor scenes. In: ECCV (2020) 2, 3, 4
84. Wang, F., Galliani, S., Vogel, C., Speciale, P., Pollefeys, M.: Patchmatchnet: Learned multi-view patchmatch stereo (2021) 13
85. Wang, S., Laskar, Z., Melekhov, I., Li, X., Kannala, J.: Continual learning for image-based camera localization. In: ICCV (2021) 3
86. Wenzel, P., Wang, R., Yang, N., Cheng, Q., Khan, Q., von Stumberg, L., Zeller, N., Cremers, D.: 4Seasons: A cross-season dataset for multi-weather SLAM in autonomous driving. In: GCPR (2020) 4
87. Yang, H., Antonante, P., Tzoumas, V., Carlone, L.: Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection. RA-L **5**(2), 1127–1134 (2020) 8
88. Yi, K.M., Trulls, E., Lepetit, V., Fua, P.: LIFT: Learned Invariant Feature Transform. In: ECCV (2016) 3
89. Zhou, Q.Y., Park, J., Koltun, V.: Open3D: A modern library for 3D data processing. arXiv:1801.09847 (2018) 6