The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLVIII-2/W8-2024
8th International ISPRS Workshop LowCost 3D - Sensors, Algorithms, Applications, 12–13 December 2024, Brescia, Italy

Investigating Visual Localization Using Geospatial Meshes

Francesco Vultaggio[1, 3], Phillipp Fanta-Jende[1], Matthias Schörghuber[1], Alexander Kern[2], Markus Gerke[3]

[1] Austrian Institute of Technology - Center for Vision, Automation and Control, Unit Assistive and Autonomous Systems
[2] Technische Universität Braunschweig - Institute of Flight Guidance
[3] Technische Universität Braunschweig - Institute of Geodesy and Photogrammetry

Emails: (francesco.vultaggio, phillipp.fanta-jende, matthias.schoerghuber)@ait.ac.at
(a.kern, m.gerke)@tu-braunschweig.de

Keywords: Visual Localization, Image Orientation, Mesh, Aerial Image, Smartphone Image, Image descriptors, GNSS, Navigation

Abstract

This paper investigates the use of geospatial mesh data for visual localization, focusing on city-scale aerial meshes as map representations for locating ground-level query images captured by smartphones. Visual localization, essential for applications such as robotics and augmented reality, traditionally relies on Structure-from-Motion (SfM) reconstructions or image collections as maps. However, mesh-based approaches offer dense spatial representation, memory efficiency, and real-time rendering capabilities. In this work, we evaluate initialization strategies, image matching techniques, and pose refinement methods for mesh-based localization pipelines, comparing the performance of both traditional and deep-learning-based techniques in image matching between real and synthetic views. We created a dataset from nadir and oblique aerial imagery and accurately georeferenced smartphone images to test cross-modal localization. Our findings demonstrate that combining global feature retrieval with GNSS-based spatial filtering yields significant improvements in accuracy and efficiency, achieving submeter positional and subdegree rotational errors. This study advances scalable visual localization using meshes and highlights the potential of integrating smartphone GNSS data for improved performance in urban environments.

1. Introduction

Visual localization is crucial for numerous applications, including robotics, augmented reality (AR), and autonomous navigation, where accurate and reliable localization is necessary for interaction with the physical world. Its power lies in the ability to obtain a position and orientation estimate without the need for GNSS sensors, which are increasingly becoming targets of attacks (Radoš et al., 2024, Figuet et al., 2022), and in indoor settings where no GNSS signal is available (Taira et al., 2018).

Visual localization has been a prominent research area in the computer vision community for more than a decade (Li et al., 2010). Visual localization can be defined in general as the process of finding the pose, P_q, at which a query image, $\mathcal{I}_\mathrm{q}$, was captured, given a preexisting *map* of the environment in which the image was taken. The term *map*, as used in the visual localization community, is rather broad encompassing 3D reconstructions obtained from Structure from Motion (SfM) software (Sarlin et al., 2019), collections of geo-located images (Berton et al., 2024), and even the weights of a neural network (Brachmann et al., 2023).

Most visual localization methods rely on a Structure from Motion (SfM) solution as *map* (Sarlin et al., 2019), where each 3D point is linked to the reference images through a local descriptor. Localization is achieved by establishing 2D-3D correspondences by matching $\mathcal{I}_\mathrm{q}$ with those descriptors. However, this approach has several limitations. First, descriptors are inherently tied to the viewing angles of reference images, which poses challenges in scenarios with domain gaps, such as matching aerial images with ground-level images (Fanta-Jende et al., 2019). Second, testing new descriptors requires recomputing the entire map and, finally, storing these descriptors becomes increasingly impractical when working with large environments (Mera-Trujillo et al., 2020).

An emerging research avenue involves the use of mesh-based representations (Panek et al., 2022). Mesh-based representations offer multiple benefits: (i) dense spatial representation, (ii) greater memory efficiency compared to SfM methods, and (iii) real-time rendering from arbitrary views, thanks to modern renderers and hardware.

In this work, we investigate the use of city-scale aerial meshes, see Figure 2, to locate images captured at ground level with a smartphone. Smartphones are particularly suited for visual localization due to their portability and widespread use, although the techniques presented are applicable to other devices as well. We conduct an extensive series of tests on state-of-the-art image matching techniques and pose initialization strategies, including how to leverage the smartphone internal GNSS. We evaluate both the final accuracy and the execution time of these strategies.

To the best of our knowledge, there is currently no public dataset for cross-modal (aerial-to-ground) visual localization with full 6 degrees of freedom reference poses for the ground-level query images and nadir-oblique reference aerial images. To address this gap, we created a custom dataset, and we describe the generation process in detail in Section 3.

An overview of our proposed visual localization method can be seen in Figure 1. In this work, we implement a two-stage pose estimation strategy, the *Pose Initialization* step aims at providing an initial coarse pose estimate, P_r^i, in the subsequent *Pose Refinement* step we render an image, $\mathcal{I}_\mathrm{r}$ and associated depth buffer D_r, to be used for relative pose estimation. In this paper,

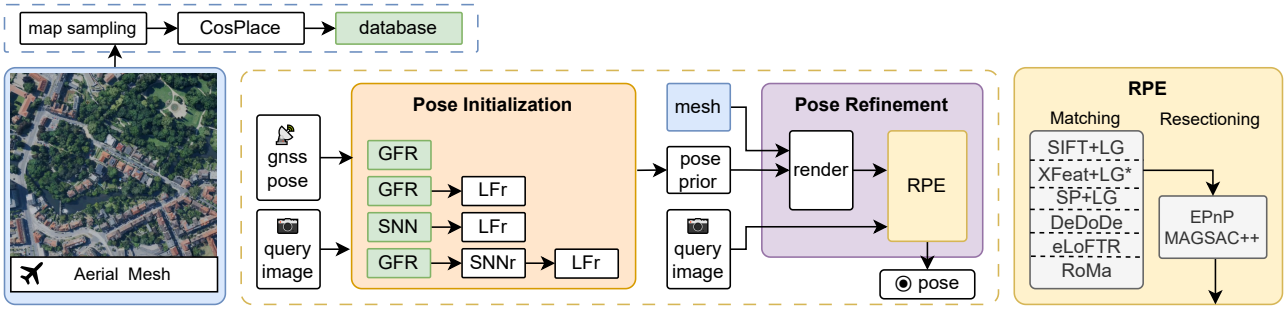

Figure 1. **Overview of proposed approach**: In an offline step, a typical photogrammetric pipeline generates a mesh out of a series of reference images. Additionally, we compute a database of geo-tagged image embeddings, *global descriptors*, to aid in the initialization phase. At testing time, the query image, and optionally a pose prior, is used to render the image corresponding to the best embedding. Using 2D-3D correspondences between the query and reference image we recover the pose at which the query image was captured was captured.

we investigate three main components of a mesh-based visual localization pipeline:

i **Dataset Generation:** Propose new best practices to generate a dataset for aerial-to-ground Visual Localization

ii **Pose Initialization:** Strategies to select a pose to render views from. We explore strategies which rely exclusively on image data and also strategies which instead take advantage of the smartphone GNSS pose estimate, we show how this increases both the speed and accuracy of the initialization.

iii **Pose Refinement:** Image-to-Image matching techniques to match the rendered image with the query image to find correspondences for pose resectioning. We test a wide range of approaches spanning from those which claim state of the art in accuracy to those which claim the state of the art when it comes to inference speed.

Meshes can serve as an effective map representation, enabling Visual Localization techniques to scale to city-wide maps while maintaining high accuracy and robustness, with a median translation error below 1 meter and median rotation error below 1 degree. This level of accuracy is achieved through the use of aerial oblique and nadir images as references, with a ground sampling distance of $7.5\,\mathrm{cm}$ and independently sampled Ground Control Points. To the best of the authors' knowledge this is the most accurate result for cross-view localization at city scale (Sarlin et al., 2023b, Berton et al., 2024, Sarlin et al., 2023a). It is important to note that using reference images captured from a lower flying drone or ground vehicle would further improve the quality of the map.

2. Related Works

Visual localization methods are often categorized based on the type of map representation they use; see (Miao et al., 2024) for a recent comprehensive review. Early approaches constructed maps from a set of reference images $\mathcal{I}_{\text{ref}}$, typically numbering in the thousands, to create a Structure-from-Motion (SfM) reconstruction of the environment. A subset of these images, along with their descriptors, was withheld to form the query test set $\mathcal{I}_{\text{test}}$, usually consisting of hundreds of images, enabling evaluation against the constructed map (Sattler et al., 2018). Such maps often contain millions of 3D points, each representing a track built from multiple 2D descriptors, leading to the community focus on accelerating search over these large descriptor sets (Sattler et al., 2012a).

Hierarchical methods (Sarlin et al., 2019) use global descriptors to quickly retrieve the most similar reference images to a query image $\mathcal{I}_{\text{q}}$. The retrieved reference images are grouped based on co-visibility, then matches are established between local descriptors from $\mathcal{I}_{\text{q}}$ and the reference images' mapped local descriptors. These correspondences are then used to estimate the pose of $\mathcal{I}_{\text{q}}$. By leveraging ever more powerful local descriptors and matchers(DeTone et al., 2018, Edstedt et al., 2024b, Potje et al., 2024, Wang et al., 2024, Edstedt et al., 2024a) these methods have achieved remarkable accuracy and robustness. However, these methods are insufficient to deal with cross-view localization as the appearance change between aerial and ground point of view is too extreme for retrieval and matching techniques to overcome. Moreover, since they require to store a full SfM model, including the descriptors associated to each 3D point, these approaches end up being large to store.

Fully learned models aim to learn how to regress the pose of $\mathcal{I}_{\text{q}}$ by training on a set of reference images and their poses, $\mathcal{I}_{\text{ref}}$ and P_{ref} from the same scene. Of these Scene Coordinate Regressors (SCRs) (Brachmann et al., 2023), predict per-pixel scene coordinates, followed by pose estimation using robust PnP algorithms. These models are extremely fast, accurate, and robust but tend to scale poorly to larger, building-sized scenes. Conversely, Visual Place Recognition (VPR)(Keetha et al., 2023) networks are trained to create distinctive image embeddings, also called global descriptors, and are used to build extensive databases of geo-tagged global descriptors. The same networks then generate the global descriptor for $\mathcal{I}_{\text{q}}$ and its pose is approximated to that of the closest global descriptor. VPR networks are extremely scalable but offer only limited accuracy. (Sarlin et al., 2023b) introduces SNAP, an approach to ground-to-aerial localization by using a neural network to generate a *neural map* from orthorectified aerial images. The same neural map is generated from ground query images, and by aligning the two, it is possible to determine the pose of $\mathcal{I}_{\text{q}}$. However, the pose is only determined in SE(2) and with meter-level accuracy.

Recent works have explored the use of meshes for visual localization. Under this paradigm, query images are matched against rendered views, $\mathcal{I}_{\text{r}}$, to establish 2D-2D correspondences. Local 2D-2D matches are then lifted to 3D using the rendered depth map, D_r. (Panek et al., 2022) demonstrated the feasibility of this approach by generating a mesh of the popular dataset from

Figure 2. In the center, the reference trajectory of the camera in blue, the smartphone's internal GNSS pose estimate in red, and the sampled views in orange, over imposed to a satellite image of the area. On the left and right side we show four examples of the query images and the view rendered from its reference pose

(Sattler et al., 2018), rendering views from the original reference poses $\mathcal{P}_{\text{ref}}$, and using those synthetic views to localize $\mathcal{I}_{\text{q}}$ following a standard hierarchical approach. In their tests they observed a drop in accuracy when using meshes and synthetic views rendered from them, when compared to the original data. Building on this, (Berton et al., 2024) leverages meshes, both aerial and ground ones, obtained from the web and ground imagery to train neural networks to produce similar embeddings for real and synthetic views of the same scene. By sampling views around a city, they create a database of geo-tagged embeddings, which are used to predict the position $\in$ SE(2) of query images captured from the ground. We use the smallest of their models (Berton et al., 2022) to compute the global descriptors used in our initialization phase, we chose this model due to its inference speed. The work of (Yan et al., 2023) is closest in goal and data to ours, as they also perform Visual Localization in SE(3) using an aerial mesh and a ground-based smartphone. However, in their case, the aerial platform was a low-flying drone, enabling more accurate 3D reconstruction but limiting scalability. Their approach uses an iterative method to converge to the correct view, trading speed for accuracy.

3. Method

In this work, we propose a comprehensive analysis of the design space for mesh-based visual localization approaches, along with a set of best practices for dataset generation.

3.1 Dataset Generation

Visual localization datasets typically consist of two sets, one of reference images $\mathcal{I}_{\text{ref}}$ and one of query images $\mathcal{I}_{\text{q}}$. Dataset creation typically involves co-registering both sets in a single Structure-from-Motion (SfM) solution, with the resulting poses considered as ground truth. Since SfM models are non-metric, scale information is recovered from additional data sources, such as by manually aligning the 3D model to online maps (Sattler et al., 2012b). However, this method is not without limitations; prior studies (Brachmann et al., 2021) have observed that the reference pose generation procedure can significantly influence the error for different sets of algorithms .

In this study, we separate the map-building process from ground truth pose estimation, using independently measured Ground Control Points (GCPs) to co-register and scale both components. The aerial mesh was generated from nadir and oblique images captured by a Vexel Osprey, resulting in an average ground sampling distance (GSD) of $7.5\,\text{cm}$ with $80\,\%$ along-track and $70\,\%$ across-track overlap. The Skyline software was used to create the mesh, with an example provided in Figure 2.

To test our mesh-based localization pipeline, we captured image data using a handheld smartphone rigidly attached to a tactical-grade inertial navigation system (INS), see Figure 3. The images were resampled to a resolution of 960x1280, yielding an average GSD of $4\,\text{cm}$. Post-processed kinematics (PPK) were applied to estimate the smartphone trajectory, achieving a standard deviation of a few centimeters under favorable conditions. The PPK solution was computed using Inertial Waypoint Explorer by NovAtel (Novatel, 2024). In post-processing, the dataset was subjected to a structure-from-motion and bundle adjustment pipeline, incorporating GCPs and Check Points (CPs), the first measured in the field with a survey-grade GNSS receiver, the latter sourced from the Cyclomedia Explorer[1]. This adjustment achieved a Root Mean Squared Error (RMSE) at GCPs of $(6, 7, 2)\ \text{cm}$ in X, Y, and Z directions, respectively, with mean errors at CPs of $(7, 4, 1)\ \text{cm}$. Image exterior orientation (EO) parameters were obtained with a mean error of $1.5\,\text{cm}$ in X and Y, $0.7\,\text{cm}$ in Z, and standard deviations of $6\,\text{mm}$ and $2\,\text{mm}$. Rotational components exhibited a mean error of $0.02°$ with a standard deviation of $0.04°$.

3.2 Visual localization

Our mesh-based visual localization pipeline is presented in Figure 1 and consists primarily of two stages: Pose Initialization and Pose Refinement. Although our test data was acquired sequentially, we compute the pose estimation for each query image independently. While leveraging information from the previous pose estimate could improve the initialization of the current pose estimation, we deliberately choose not to rely on temporal information. This decision allows us to test the visual

[1] https://www.cyclomedia.com/en/street-smart

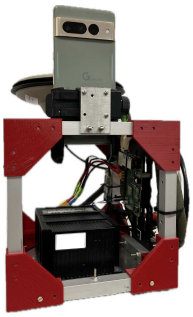

Figure 3. Smartphone rigidly mounted to INS system to capture query data.

localization module in isolation, ensuring that its performance is evaluated without the influence of sequential data dependencies.

3.2.1 Pose Initialization The Pose Initialization goal is to estimate a pose from which to render an image that visually overlaps the query image. In the offline stage, we generate reference data comprising of two main components: an aerial mesh of the area and a database of geo-tagged global descriptors. These descriptors are computed using a neural network (Berton et al., 2022), fine-tuned specifically for real and synthetic image pairs (Berton et al., 2024). To populate the database, we create synthetic images by sampling along the road network, which is obtained from publicly available OpenStreetMap data. Our sampling strategy involves extracting latitude-longitude coordinates at regular intervals of 5 meters along the road network. At each sampled coordinate, we render eight synthetic views, equally spaced around the Z-axis of the map. The sampled nodes can be seen in Figure 2 overlayed on the map.

We implement and test four distinct initialization strategies inspired by different approaches attempted in the literature. Many visual localization methods assume that no pose prior is available and thus use a Global-Feature-Retrieval (GFR) approach to acquire visually similar images from the set of reference images. Global feature retrieval is often not sufficient to disambiguate between visually similar images and thus often GFR is followed by Local Feature re-ranking (LFr) (Cao et al., 2020).

Subsequently we test how to leverage the smartphone's GNSS pose estimate, in the first case using a simple retrieval of all the images within a specified radius from the pose estimate - starting from 5m, expanding to 10m and then 20m, if no images are retrieved by the earlier threshold - we indicate this as retrieving the Spatial Nearest Neighbors (SNN), we then find the best image among the retrieved ones using LFr. Finally we combine image data and pose prior in our fourth test where we first retrieve the top 50 closest embeddings from the database (GFR), we then select among these the top 5 closest ones to our GNSS pose estimate, Spatial Nearest Neighbors re-ranking (SNNr) and finally pick the best one based on the number of matches, LFr. In summary, we explore and compare the following four initialization strategies:

(i) GFR

(ii) GFR followed by LFr

(iii) SNN followed by LFr

(iv) GFR followed by SNNr and finally LFr

Descriptor	Matcher	Class	Reference
SIFT	LightGlue	Sparse	(Lowe, 2004)
XFeat	LightGlue	Sparse	(Potje et al., 2024)
SuperPoint	LightGlue	Sparse	(DeTone et al., 2018)
DeDoDe	DSM	Sparse	(Edstedt et al., 2024a)
eLoFTR	N.A.	Dense	(Wang et al., 2024)
RoMa	N.A.	Dense	(Edstedt et al., 2024b)

Table 1. Image matching techniques

3.2.2 Pose Refinement Central to pose refinement is accurate and robust image matching between $\mathcal{I}_q$ and $\mathcal{I}_r$ as reliable 2D-2D matches are fundamental to obtain a precise pose estimates. Image matching is a highly active area of research, with numerous approaches being proposed (Bonilla et al., 2024). In recent years, learning-based methods have largely replaced hand-crafted techniques (Fourth Workshop on Image Matching: Local Features & Beyond, 2022). While modern learning-based methods are typically trained on real image pairs, the challenge of matching synthetic and real image pairs remains under explored. To gain a comprehensive understanding of how different approaches are influenced by real and synthetic image pairs, we evaluate both sparse and dense learning-based local descriptors. For a detailed list of the methods see Tab. 1.

We include SIFT-based matching (Lowe, 2004) as a representative of hand-crafted methods in our tests. However, instead of relying on hand-crafted methods for matching we use LightGlue (Lindenberger et al., n.d.). LightGlue is descriptor-agnostic in its architecture but requires descriptor-specific training for each local descriptor. Once trained it takes as input the key-points image coordinates and descriptors for an image pairs and outputs a matching matrix between the two key-points and descriptor sets. We used the same model for SuperPoint (DeTone et al., 2018) and a smaller version of the same architecture for XFeat (Potje et al., 2024), as recommended by the authors.

All matches will then follow the same relative pose estimation strategy, which we implement using EPnP (Lepetit et al., 2009) together with MAGSAC++ (Baráth et al., 2019) for the outlier removal step.

4. Experiments

The experimental results are presented in Table 2. We report the percentage of images localized within three pose thresholds — $(0.5\,\mathrm{m}, 2^\circ)$, $(2\,\mathrm{m}, 5^\circ)$, and $(5\,\mathrm{m}, 10^\circ)$. Poses that are beyond these thresholds are categorized as outliers. We also report the median (ME) and root mean squared error (RMSE). The ME is computed from all images. The RMSE from images within our pose thresholds. Lastly, we report the execution times for the pose initialization (t_{in}) and refinement step (t_{ref}). Note that all the experiments have been conducted on a machine equipped with a Intel Xeon W-2245 and a Nvidia RTX 3090 Ti. All images have been downscaled to 1000 pixels along the largest dimension due to GPU memory constraints. The image coordinates of the 2D-2D matches are then upscaled to the original resolution to sample the 3D coordinate from the depth maps of the rendered images.

Looking at the result of the GFR initialization method, we can notice that this approach results in the fastest initialization times

Initialization	**Refinement**	**0.5m,2°/2m,5°/5m,10°**[1] ↑ % / % / %	**Outliers**[2] ↓ %	**ME**[3] ↓ m / °	**RMSE**[4] ↓ m / °	t_{in}[5] ↓ s	t_{ref}[6] ↓ s
GFR	SIFT+LG	4.82 / 28.67 / 41.93	15.15	1.95 / 1.82	2.05 / 2.26		0.34
	XFeat+LG*	10.50 / 46.10 / 60.35	17.61	1.55 / 1.43	1.74 / 1.81		**0.08**
	SP+LG	13.30 / 52.48 / 64.70	**11.28**	1.17 / 1.13	1.65 / 1.64	**0.11**	0.10
	DeDoDe	16.19 / 55.27 / 66.25	32.85	1.43 / 1.33	1.56 / 1.46		0.46
	eLoFTR	18.04 / 58.63 / 64.40	35.56	1.28 / 1.13	1.32 / 1.27		0.15
	RoMa	23.07 / 66.38 / 69.65	30.31	1.05 / 0.93	1.12 / 1.00		0.65
GFR-LFr	SIFT+LG	5.94 / 34.31 / 51.83	18.17	2.04 / 1.84	2.11 / 2.18	1.28	0.37
	XFeat+LG*	12.40 / 52.22 / 67.76	26.30	1.66 / 1.47	1.75 / 1.73	0.40	**0.08**
	SP+LG	16.19 / 57.43 / 70.86	19.59	1.22 / 1.25	1.60 / 1.54	0.44	0.09
	DeDoDe	18.73 / 63.19 / 72.58	27.38	1.21 / 1.15	1.44 / 1.33	1.43	0.47
	eLoFTR	18.34 / 63.62 / 69.48	30.48	1.18 / 1.05	1.28 / 1.21	0.78	0.16
	RoMa	24.02 / 67.76 / 72.15	27.81	1.03 / 0.95	1.15/1.05	2.95	0.60
SNN-LFr	SIFT+LG	4.82 / 24.80 / 41.11	31.21	3.66 / 3.37	2.27 / 2.66	5.70	0.36
	XFeat+LG*	10.33 / 49.07 / 67.59	31.21	2.00 / 2.01	1.92 / 2.30	1.63	**0.08**
	SP+LG	15.58 / 58.98 / 76.93	20.79	1.34 / 1.57	1.83 / 2.06	1.77	0.09
	DeDoDe	**23.46 / 68.36 / 81.70**	18.25	1.03 / 1.18	1.55 / 1.63	6.42	0.42
	eLoFTR	20.75 / 67.03 / 76.67	23.29	1.12 / 1.08	1.43 / 1.57	3.27	0.14
	RoMa	21.39 / 50.71 / 52.78	47.18	1.69 / 1.73	**0.96 / 1.18**	15.94	0.61
GFR-SNNr-LFr	SIFT+LG	5.81 / 31.51 / 49.42	22.26	2.44 / 2.25	2.20 / 2.47	1.21	0.36
	XFeat+LG*	11.67 / 52.95 / 70.30	25.14	1.63 / 1.61	1.82 / 1.97	0.39	**0.08**
	SP+LG	17.26 / 62.76 / 76.50	16.32	1.14 / 1.28	1.60 / 1.64	0.43	0.09
	DeDoDe	22.69 / 67.67 / 79.68	20.28	1.07 / 1.14	1.52 / 1.44	1.23	0.41
	eLoFTR	21.05 / 66.34 / 75.08	24.88	1.11 / 1.06	1.38 / 1.38	0.70	0.14
	RoMa	29.14 / 73.44 / 79.08	20.08	**0.84 / 0.93**	1.15 / 1.16	3.08	0.61

Table 2. Quantitative comparison of various pose initialization and refinement methods. The table presents: [1] the percentage of query images successfully localized within three pose thresholds — 0.5m, 2°, 2m, 5°, and 5m, 10°; [2] percentage of images incorrectly localized beyond the last threshold; [3] the Median Error (ME) across all images, and [4] Root Mean Square Error (RMSE) among accurately localized images, calculated for both positional and rotational errors; and [5] the execution times for the initialization (t_{in}) and [6] refinement (t_{ref}) stages. underline: best per initialization method **bold**: overall best

as it only consists of a forward pass through a small neural network and a retrieval in a vector database. In total, this process takes about 0.1 seconds. We note that the retrieval time scales with the size of the database. In our case, the database consists of 7300 image embeddings and takes 15MB, which can fit into computer memory. How to deal with databases consisting of billion of elements is beyond the scope of this work and we refer the reader to the seminal analysis from (Johnson et al., 2019).

The second initialization strategy we tested, GFR-LFr, shows improvements in the overall accuracy across all local descriptors. However, it is considerably slower because LFr is a rather time consuming operation, since it requires computing matches between the $\mathcal{I}_q$ and all retrieved reference images in order to select the best image. This process is particularly slow for DeDoDe and RoMa. Both achieve the best accuracy in this initialization setup, but this precision comes at the cost of slower matching times.

The same can be seen at an even more pronounced scale for the SNN-LFr setup. Here, the same trend across local descriptors can be observed but, since the number of reference images to be re-ranked is only limited by the radius of the spatial search, the re-ranking step is even more time consuming. Interestingly, this initialization setup shows strong improvements in accuracy for some but not all the image matching techniques, surprisingly RoMa ends up suffering from a sharp decrease in ME performance due to a high number of outliers. This indicates that while RoMa demonstrates a strong capability to identify correspondences between positive image pairs, it also identifies correspondences between negative image pairs, i.e., images with no visual overlap.

Our last experiment (GFR-SNNr-LFr) offers the best compromise between robustness, accuracy, and execution time. It leverages the speed of the fast GFR to retrieve a large amount of visually similar images and uses the coarse GNSS pose estimate to filter only the plausible views. This reduces the initialization times when compared to SNNr while maintaining high accuracy, and reducing the number of outliers. In particular RoMa increases the percentage of images localized within $5\,\mathrm{m}$, 10° from 52.78% to 79.08% when initializing with GFR-SNNr-LFr.

Focusing on the performances of the image matching techniques in more detail we can observe how the SIFT descriptor, while still competitive for real images (Jin et al., 2021), proves ineffective when applied to real-synthetic image pairs. XFeat is the fastest model we tested, even while running on the CPU, it offers much better performance compared to SIFT but falls short to the larger models we tested. SuperPoint is still competitive to more modern architectures and offers a compromise between inference speed and accuracy for platforms that have access to GPU acceleration. The best sparse descriptor we tested is DeDoDe which leverages the DINOv2 (Oquab et al., 2023) foundation model to improve the local features. eLoFTR offers higher accuracy when compared with sparse descriptors and it does so considerably faster than the other dense matching technique we tested, RoMa.

Figure 4. Comparison of feature matches between the query image $\mathcal{I}_q$ and an image rendered from the best pose (top) as determined by SNN-LFr, showing successful matching, and an image rendered from the reference pose (bottom), only spurious matches were found. Both examples use XFeat and Lighter Glue for image matching.

In all our tests we never observed more than 30% of images being localized within $0.5\,\mathrm{m}$, $2°$. Across rendering-based re-localization techniques iterated rendering is often seen as being instrumental in unlocking greater precision. To validate this hypothesis we design a test where we initialize our rendering from the ground truth pose and match $\mathcal{I}_q$ to its rendered counterpart. Here, we notice an improvement in accuracy for most image matching techniques, in particular RoMa achieves 38.2% of localized frames below $0.5\,\mathrm{m}$, $2°$. SuperPoint achieves the lowest ME $0.72\,\mathrm{m}$, $0.9°$, due to the low rate of images localized with an error above $5\,\mathrm{m}$, $10°$. However, we have a high rate of images either wrongly localized or not localized at all. This is because the assumption underlying iterative re-rendering, that the rendered images will be easier to match the closer they are to the ground truth image, does not always hold true. Figure 2 (B) and (C) shows how vegetation can degrade the mesh leading to poor localization outcomes. In fact, DeDoDe has better performance initializing with GFR-SNNr-LFr than when rendering from the ground truth pose. Similarly, RoMa has lower median error, this is because these can identify poses which have better visual overlap with $\mathcal{I}_q$, see Figure 4.

Refinement	**0.5m,2°/2m,5°/5m,10°** % / % / %	**Outliers** %	**ME** m / °
SIFT+LG	8.78 / 44.04 / 53.55	11.75	1.59 / 1.86
XFeat+LG*	19.03 / 65.30 / 76.11	10.68	0.95 / 1.07
SP+LG	26.52 / 75.08 / 82.52	**4.56**	**0.72 / 0.90**
DeDoDe	22.34 / 69.26 / 81.27	16.57	1.03 / 0.97
eLoFTR	28.58 / 73.91 / 82.95	16.79	0.90 / 0.91
RoMa	**38.18 / 80.89 / 84.46**	15.50	0.94 / 1.08

Table 3. Localization outcomes obtained when initializing from the reference pose. In **bold** the best method

5. Conclusions

Visual localization plays a crucial role in applications requiring precise, real-time positioning, from pedestrian navigation to autonomous exploration. This paper demonstrates the potential of consumer-grade imaging and positioning sensors, such as smartphones, used in combination with standard data products like aerial imagery and meshes, to enhance localization accuracy and scalability. Through examining the utility of mesh data as a georeferenced information source, we assess the impact of aerial meshes on localization accuracy, focusing on the initialization and pose refinement stages. Our findings indicate that even imperfect smartphone GNSS data can effectively improve image retrieval, and we introduce a novel methodology for generating test data to rigorously evaluate visual localization performance.

In future work, we plan to validate our approach with more challenging query images collected under nighttime or adverse weather conditions. We also aim to investigate the degree to which higher-quality meshes can further improve localization accuracy and robustness. Finally, while this study evaluated images independently, we see an opportunity to explore sequential localization techniques to leverage temporal coherence across frames, potentially enhancing real-time performance in applications such as autonomous navigation.

Acknowledgements

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the EUSPA. Neither the European Union nor the EUSPA can be held responsible for them.

References

Baráth, D., Noskova, J., Ivashechkin, M., Matas, J., 2019. MAGSAC++, a Fast, Reliable and Accurate Robust Estimator. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13–19. doi: 10.1109/CVPR42600.2020.00138.

Berton, G., Junglas, L., Riccardo Zaccone, Thomas Pollok, B. C., Masone, C., 2024. Meshvpr: Citywide visual place recognition using 3d meshes. *European Conference on Computer Vision (ECCV)*.

Berton, G., Masone, C., Caputo, B., 2022. Rethinking visual geo-localization for large-scale applications. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*, 4878–4888. doi: 10.1109/CVPR52688.2022.00483.

Bonilla, S., Di Vece, C., Daher, R., Ju, X., Stoyanov, D., Vasconcelos, F., Bano, S., 2024. Mismatched: Evaluating the Limits of Image Matching Approaches and Benchmarks. *arXiv*. doi: 10.48550/arXiv.2408.16445.

Brachmann, E., Cavallari, T., Prisacariu, V. A., 2023. Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*. doi: 10.1109/CVPR52729.2023.00488.

Brachmann, E., Humenberger, M., Rother, C., Sattler, T., 2021. On the limits of pseudo ground truth in visual camera re-localisation. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 6218–6228. doi: 10.1109/ICCV48922.2021.00616.

Cao, B., Araujo, A., Sim, J., 2020. Unifying deep local and global features for image search. *European Conference on Computer Vision (ECCV)*, Springer, 726–743. doi: 10.1007/978-3-030-58565-5_43.

DeTone, D., Malisiewicz, T., Rabinovich, A., 2018. Superpoint: Self-supervised interest point detection and description. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*, 224–236. doi: 10.1109/CVPRW.2018.00060.

Edstedt, J., Bökman, G., Wadenbäck, M., Felsberg, M., 2024a. DeDoDe: Detect, Don't Describe — Describe, Don't Detect for Local Feature Matching. *2024 International Conference on 3D Vision (3DV)*, IEEE. doi: 10.1109/3DV62453.2024.00035.

Edstedt, J., Sun, Q., Bökman, G., Wadenbäck, M., Felsberg, M., 2024b. RoMa: Robust Dense Feature Matching. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*, 19790-19800. doi: 10.1109/CVPR52733.2024.01871.

Fanta-Jende, P., Nex, F., Vosselman, G., Gerke, M., 2019. Co-registration of panoramic mobile mapping images and oblique aerial images. *The Photogrammetric Record 34 (166)*.

Figuet, B., Waltert, M., Felux, M., Olive, X., 2022. GNSS Jamming and Its Effect on Air Traffic in Eastern Europe. *Engineering Proceedings*, 28(1), 12. doi: 10.3390/engproc2022028012.

Fourth Workshop on Image Matching: Local Features & Beyond, 2022. [Online; accessed 15. Nov. 2024].

Jin, Y., Mishkin, D., Mishchuk, A., Matas, J., Fua, P., Yi, K. M., Trulls, E., 2021. Image Matching Across Wide Baselines: From Paper to Practice. *International Journal of Computer Vision*, 129(2), 517–547. doi: 10.1007/s11263-020-01385-0.

Johnson, J., Douze, M., Jégou, H., 2019. Billion-Scale Similarity Search with GPUs. *IEEE Transactions on Big Data*, 7(3), 535–547. doi: 10.1109/TBDATA.2019.2921572.

Keetha, N., Mishra, A., Karhade, J., Jatavallabhula, K. M., Scherer, S., Krishna, M., Garg, S., 2023. Anyloc: Towards universal visual place recognition. *IEEE Robotics and Automation Letters*. doi: 10.1109/LRA.2023.3343602.

Lepetit, V., Moreno-Noguer, F., Fua, P., 2009. EPnP: An Accurate O(n) Solution to the PnP Problem. *International Journal of Computer Vision*, 81(2), 155–166. doi: 10.1007/s11263-008-0152-6.

Li, Y., Snavely, N., Huttenlocher, D., 2010. Location Recognition Using Prioritized Feature Matching. *European Conference on Computer Vision ECCV*. doi: 10.1007/978-3-642-15552-9_57.

Lindenberger, P., Sarlin, P.-E., Pollefeys, M., n.d. LightGlue: Local Feature Matching at Light Speed. *IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, 01–06. doi: 10.1109/ICCV51070.2023.01616.

Lowe, D. G., 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2), 91–110. doi: 10.1023/B:VISI.0000029664.99615.94.

Mera-Trujillo, M., Smith, B., Fragoso, V., 2020. Efficient Scene Compression for Visual-based Localization. *2020 International Conference on 3D Vision (3DV)*, IEEE, 25–28, doi: 10.1109/3DV50981.2020.00111.

Miao, J., Jiang, K., Wen, T., Wang, Y., Jia, P., Wijaya, B., 2024. A Survey on Monocular Re-Localization: From the Perspective of Scene Map Representation. *IEEE Transaction on Intelligent Vehicles*, 1–33. doi: 10.1109/TIV.2024.3378716.

Novatel, 2024. Inertial Waypoint Explorer. `https://novatel.com/products/waypoint-post-processing-software/inertial-explorer`.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H. V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Howes, R., Huang, P.-Y., Xu, H., Sharma, V., Li, S.-W., Galuba, W., Rabbat, M., Assran, M., Ballas, N., Synnaeve, G., Misra, I., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P., 2023. Dinov2: Learning robust visual features without supervision.

Panek, V., Kukelova, Z., Sattler, T., 2022. MeshLoc: Mesh-Based Visual Localization. *European Conference on Computer Vision ECCV*, 589–609, doi: 10.1007/978-3-031-20047-2_34.

Potje, G., Cadar, F., Araujo, A., Martins, R., Nascimento, E. R., 2024. Xfeat: Accelerated features for lightweight image matching. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*. doi: 10.1109/CVPR52733.2024.00259.

Radoš, K., Brkić, M., Begušić, D., 2024. Recent Advances on Jamming and Spoofing Detection in GNSS. *Sensors*, 24(13), 4210. doi: 10.3390/s24134210.

Sarlin, P.-E., Cadena, C., Siegwart, R., Dymczyk, M., 2019. From coarse to fine: Robust hierarchical localization at large scale. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*. doi: 10.1109/CVPR.2019.01300.

Sarlin, P.-E., DeTone, D., Yang, T.-Y., Avetisyan, A., Straub, J., Malisiewicz, T., Bulo, S. R., Newcombe, R., Kontschieder, P., Balntas, V., 2023a. OrienterNet: Visual Localization in 2D Public Maps with Neural Matching. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*. doi: 10.1109/CVPR52729.2023.02072.

Sarlin, P.-E., Trulls, E., Pollefeys, M., Hosang, J., Lynen, S., 2023b. SNAP: Self-Supervised Neural Maps for Visual Positioning and Semantic Understanding. *Advances in Neural Information Processing Systems*, 36, 7697–7729.

Sattler, T., Leibe, B., Kobbelt, L., 2012a. Improving Image-Based Localization by Active Correspondence Search. *European Conference on Computer Vision (ECCV)*, 752–765, doi: 10.1007/978-3-642-33718-5_54.

Sattler, T., Maddern, W., Toft, C., Torii, A., Hammarstrand, L., Stenborg, E., Safari, D., Okutomi, M., Pollefeys, M., Sivic, J. et al., 2018. Benchmarking 6dof outdoor visual localization in changing conditions. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*, 8601–8610. doi: 10.1109/CVPR.2018.00897.

Sattler, T., Weyand, T., Leibe, B., Kobbelt, L., 2012b. Image retrieval for image-based localization revisited. *British Machine Vision Conference*.

Taira, H., Okutomi, M., Sattler, T., Cimpoi, M., Pollefeys, M., Sivic, J., Pajdla, T., Torii, A., 2018. InLoc: Indoor visual localization with dense matching and view synthesis. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*. doi: 10.1109/TPAMI.2019.2952114.

Wang, Y., He, X., Peng, S., Tan, D., Zhou, X., 2024. Efficient LoFTR: Semi-dense local feature matching with sparse-like speed. *IEEE/CVF Conference on Computer Vision and Pattern Recognition CVPR*. doi: 10.1109/CVPR52733.2024.02047.

Yan, S., Cheng, X., Liu, Y., Zhu, J., Wu, R., Liu, Y., Zhang, M., 2023. Render-and-compare: Cross-view 6-dof localization from noisy prior. *IEEE International Conference on Multimedia and Expo (ICME)*, 2171–2176. doi: 10.1109/ICME55011.2023.00371.