EarthMatch: Iterative Coregistration for Fine-grained Localization of Astronaut Photography

Gabriele Berton[1] Gabriele Goletto[1] Gabriele Trivigno[1]
Alex Stoken[2] Barbara Caputo[1] Carlo Masone[1]
[1]Politecnico di Torino [2]Jacobs Technology, NASA Johnson Space Center

Abstract

Precise, pixel-wise geolocalization of astronaut photography is critical to unlocking the potential of this unique type of remotely sensed Earth data, particularly for its use in disaster management and climate change research. Recent works have established the Astronaut Photography Localization task, but have either proved too costly for mass deployment or generated too coarse a localization. Thus, we present EarthMatch, an iterative homography estimation method that produces fine-grained localization of astronaut photographs while maintaining an emphasis on speed. We refocus the astronaut photography benchmark, AIMS, on the geolocalization task itself, and prove our method's efficacy on this dataset. In addition, we offer a new, fair method for image matcher comparison, and an extensive evaluation of different matching models within our localization pipeline. Our method will enable fast and accurate localization of the 4.5 million and growing collection of astronaut photography of Earth. Code and data are available at https://EarthLoc-and-EarthMatch.github.io/

1. Introduction

Computer vision plays a pivotal role in analyzing remotely sensed Earth observations, demonstrating efficacy across diverse applications including Earth monitoring, atmospheric and climate research, and emergency response strategies [11, 15, 23, 24]. Much of this remotely sensed imagery comes from satellites devoted to Earth observations, but a peculiar, yet valuable complementary source of data is astronaut photography. More than 4.5 million photos of Earth[1] have been taken by astronauts, primarily from the International Space Station (ISS) during its over 20 years of continuous operation in low Earth orbit (roughly 400 km), a vantage point which offers a privileged perspective with a field of view that can span up to thousands of kilometers.

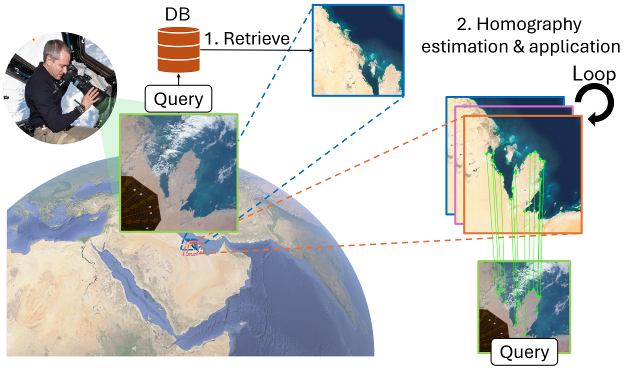

Figure 1. **EarthMatch**: To produce a pixel-wise geolocalization of an astronaut photograph (the *query*), we first retrieve a possible candidate from a worldwide database of satellite images. We then compute image correspondences and coregister the two images iteratively, yielding a precise query localization and confidence value.

Astronaut photography is particularly valuable due to its unique characteristics. For instance, the orbital speed and trajectory of the ISS enables quick response time in case of natural disasters and other emergencies, as astronauts can be promptly alerted and take timely photographs which, after manual localization, are provided to first responders on the ground [51]. Protocols based on this concept have been successfully implemented during the 2013 Haiyan's cyclone, as well as to handle flooding events, wildfires, and several other crises throughout the planet [22, 37, 52]. Geolocated astronaut photographs have also been used for research purposes across a variety of Earth science topics [39, 48, 59, 60]. Furthermore, unlike unmanned satellites, behind each photo is a skilled human (as astronauts undergo photography and geography training) that interprets the observable landscape and is not constrained to a fixed perspective (see Fig. 1). Adjusting parameters like orientation, focal length and viewing angle offers intriguing possibilities for data collection which are difficult or impossible to achieve with conventional satellites. Yet, the very same human-in-the-loop nature of astronaut photographs

[1] https://eol.jsc.nasa.gov/

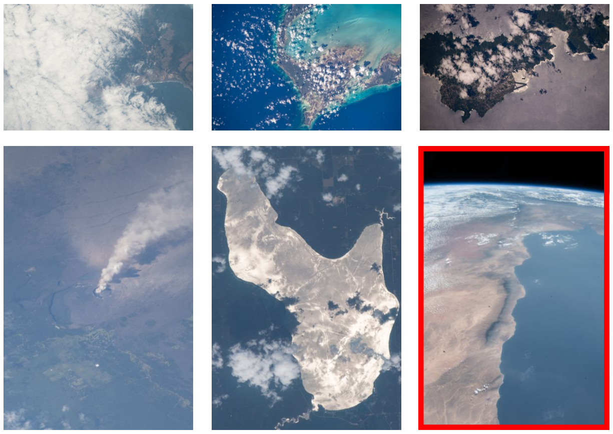

Figure 2. **Examples of astronaut photo queries** from the AIMS dataset [53] which we use in our experiments. The rightmost bottom image is an example of a photo with high tilt/oblique, which we remove prior to our benchmark evaluation. These images are also the least useful/informative for Earth science researchers.

also makes their precise localization challenging, since only a coarse estimate can be inferred from the location of the ISS at the photo acquisition time, and manual verification has to be performed on each image for precise localization. Despite the high manual localization cost, over 300k (*i.e.* still less than 10%) of images have been manually localized, demonstrating the importance and value of geotagged astronaut photographs. In order to unlock the full potential utility of this vast corpus of images, an automated fine-grained geolocalization solution is needed - a task named Astronaut Photo Localization (**APL**) by EarthLoc [8].

To this end, the work of Find My Astronaut Photo (FMAP) [53] proposed the first localization solution for this problem and released a labelled dataset which is of paramount importance for future works aiming to assess the reliability of methods before they are applied to unlabelled images. FMAP uses a brute-force approach based on pairwise matching against a reference obtained from satellite data. The high latency of this extensive matching makes it unsuitable for applications where speed is of the utmost importance, such as assisting in natural disaster response. Recently, EarthLoc [8] showed that retrieval methods can drastically reduce the uncertainty on the area of interest by identifying a handful of candidate regions which are most similar to the query image to be localized. Candidates are retrieved from a worldwide database of cloudless Sentinel-2 satellite images[2], which divide the Earth into regular, rectilinear tiled regions. Despite its robust performance, a pipeline based solely on image retrieval does not provide pixel-wise localization of the query, nor does it provide a confidence measure for its predictions.

In light of these considerations, we advocate for a hierarchical pipeline that combines an EarthLoc-like retrieval step with a coregistration method capable of providing an accurate location estimate as well as a reliable confidence score that allows for the rejection of false positives. To this end, we present EarthMatch, a simple yet efficient fine-grained geolocalization algorithm based on iterative homography refinement, and demonstrate its effectiveness through a comprehensive benchmark. In summary, our main contributions are:

- an APL pipeline made of (i) a retrieval model to obtain possible candidate locations for a given query, and (ii) EarthMatch to confidently obtain a single final prediction through an iterative coregistration algorithm;
- a thorough benchmark in which we implement our algorithm with a large number of existing image matching methods, ranging from sparse detectors, learned matchers, handcrafted methods as well as dense warp estimators;
- an extension to the Astronaut Imagery Matching Subset (AIMS) dataset [53] with 268 astronaut photographs and their the top-10 predictions (satellite images) from the current SOTA retrieval APL model [8], and we release this extended dataset to foster future research.

The code to run the benchmark, as well as scripts to download the data, is available at https://EarthLoc-and-EarthMatch.github.io/. All the images localized with EarthMatch (plus images localized with FMAP) can be explored in a convenient interactive format at https://eol.jsc.nasa.gov/ExplorePhotos/.

2. Related work

Feature detection and description. Finding salient points in an image, and their associated descriptors, is a cornerstone of computer vision. Points and descriptions are employed for a variety of different tasks, such as Visual Localization [43, 45], Structure from Motion (SfM) [46, 47] and Simultaneous Localization and Mapping (SLAM) pipelines [1, 17, 38]. Classical methods rely on a detect-then-describe approach, typically based on handcrafted descriptors computed from local derivatives of the image [32]. The most popular among these methods is SIFT [32] which provides a framework for feature detection and description that is robust to scale and distortions. Follow-up works, such as SURF [5], focused on improving efficiency, while others like ORB [38] combine fast keypoint detection with a robust binary descriptor.

As deep learning gained prominence, learnable approaches for both detection and description were introduced. In particular, [50, 56] are the first to popularize sparse keypoint detection and description with deep neural networks, relying on contrastive strategies to learn patch-level descriptors with a CNN. Later, SuperPoint [16] introduced a self-supervised framework for training on synthetic

[2] https://s2maps.eu/

shapes with arbitrarily defined keypoints. Conversely, D2-Net [18] and R2D2 [42] propose a framework to jointly detect and describe features, modeling keypoints as local maxima of the feature maps. Other subsequent works define salient points as those that maximize the matching probability, again trained through self-supervision [25, 57].

Among the most recent solutions, ALIKE [62] uses a novel detection module based on a patch-wise softmax relaxation and has been further extended in ALIKED [61] by exploiting deformable convolutions that adapt to the keypoints' support. DeDoDe [20] decouples the detection and description steps into two independent models; Steerers [10] further improves upon this approach with a formulation designed to be rotation invariant.

Image matching. Our task of interest, localizing astronaut photography, has challenges akin to the problem of wide baseline image matching [29]. This is mainly due to the substantial appearance shift with respect to our reference set and the minimal overlap that we typically face between a query and its retrieved candidate. Image matching tries to find correspondences (*e.g.*, points or regions) among images by comparing features. Traditionally, matching was performed on handcrafted descriptions with a nearest neighbor search over descriptors followed by Lowe's ratio test [32], or via mutual nearest neighbor in the case of learnable, sparse methods [42]. The landscape evolved with SuperGlue [44], a graph neural network-based approach for matching features, which leverages attention mechanisms to deal with the challenges of significant viewpoint change or occlusion. The introduction of LoFTR [54] represented a departure from discrete feature detection towards a detector-free, semi-dense matching paradigm, using transformers to coarsely match features. These methods stand out for their effectiveness even in texture-sparse scenarios. Following LoFTR, several other methods followed a similar approach [9, 14, 27, 55, 58, 63]. An alternative line of research focuses on dense warp estimation between image pairs, from which keypoints can be sparsely sampled when needed [19]. Recently, RoMA [21] proposed to exploit a universal vision foundation model, DINOv2 [41], for the task of dense feature matching; given the coarse nature of vision transformers, the authors combine it with a specialized CNN for match refinement, achieving state-of-the-art results on a variety of downstream benchmarks [3, 29, 30]. In this work, we are interested in using pairwise matchers to asses the similarity of two images. Typically these methods are benchmarked on downstream applications like homography or pose estimation; however, in our case we match images which may exhibit only partial spatial overlap or remain entirely distinct. Thus, our goal is to understand whether they depict the same place or not. Hence, similar to what is done in [53], we use the keypoints resulting from matching as a similarity measure. Additionally, we also use these salient points to estimate a projective transformation between the images. Iterative homography estimation itself has been studied as a technique to refine the co-registration between two images. Traditionally, these methods were based on the Lucas-Kanade algorithm [2, 34], which relies on photometric error. In [13], the authors propose an end-to-end trainable network for deep iterative homography estimation. Recently, [6, 7] proposed novel handcrafted techniques for match and homography refinement.

Astronaut photography localization. Recent literature has shown a growing interest in the problem of localizing photographs from astronauts. Pioneering works Find My Astronaut Photo (FMAP) [53] and [49] both rely on pairwise comparison between queries and reference data. While the former focuses on daytime pictures with satellite imagery as reference, the latter tries to localize night-time images by generating synthetic data to bridge the domain gap. While showing convincing performance, these methods are hindered by the intrinsic complexity of matching images without prior knowledge of candidate regions. Recently, EarthLoc [8] introduced a retrieval method designed to handle the specific challenges of astronaut photography, namely large orientation, scale and appearance variations. Additionally, astronaut photos were used as a downstream benchmark to demonstrate a rotation equivariant detector proposed by Steerers [10].

In this work, we propose to exploit EarthLoc's candidate predictions to reduce the search area toward a few likely regions. Given EarthLoc top-k predictions, we apply an iterative matching algorithm that is able to correctly and precisely localize the query when a suitable candidate is present. Furthermore, our method provides a confidence metric, enabling the identification of instances where a candidate lacks overlap with the query, thereby establishing a method to reject false positive candidates from retrieval.

3. EarthMatch

Overview. We propose a hierarchical pipeline to solve the APL task and precisely localize astronaut photographs. Our approach is inspired by the re-ranking approaches commonly adopted in Image Retrieval [12, 40] and Visual Place Recognition [4, 26, 33, 64], where a fast retrieval method is used first to obtain a shortlist of candidates, followed by a more computationally demanding matching step to refine the estimate.

An overview of the pipeline is presented in Fig. 3: we rely on an APL image retrieval model (e.g. EarthLoc [8]) which, for a given query, provides a shortlist of candidates. Such candidates are retrieved from a worldwide database of geotagged satellite images (*i.e.* images for which the pixel-wise location, in the form of geographic coordinates, is known). Subsequently, to refine these candidates, we re-

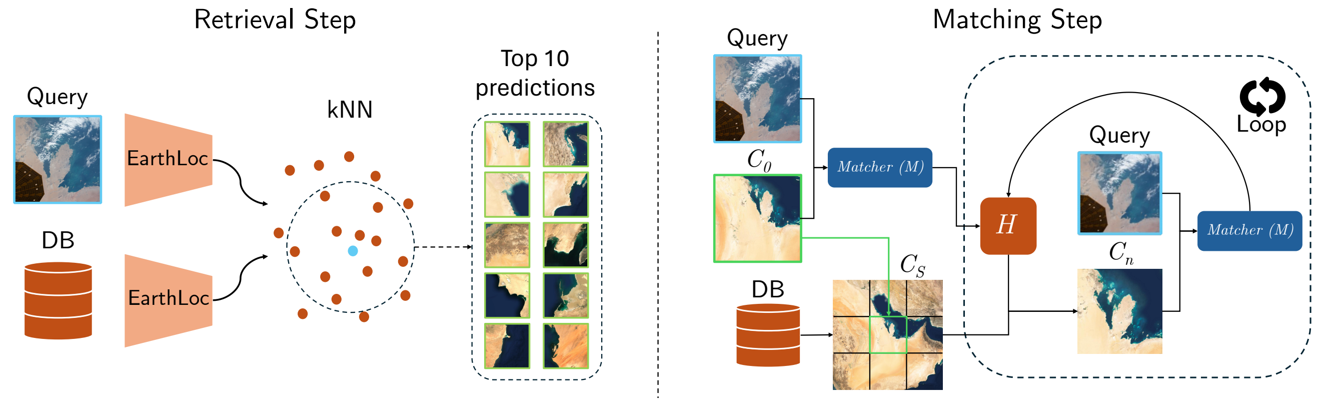

Figure 3. **Left: Overview of retrieval step**, which, for a given query, retrieves candidates/predictions from a worldwide database of geo-tagged images. **Right: Overview of matching step.** The matching pipeline takes as input the query and a retrieved candidate. Surroundings of the candidate are obtained from the database, and then the iterative coregistration (in the form of matching and homographic transformation) is performed.

cast the matching step in the form of an iterative coregistration algorithm, called **EarthMatch**, that is tailored for the task of APL (see Fig. 3). EarthMatch takes as input a query image Q (*i.e.*, the astronaut photograph) and a candidate image C (*i.e.*, a satellite image), where the candidate has been proposed by the retrieval method as potentially overlapping the query. The algorithm's goal is twofold: (1) understand if the two images have any overlap and (2), if there is an overlap, precisely estimate the overlap, in the form of a homography matrix. A precise estimate of the overlap allows extraction of the pixel-wise position of the query, given the known geographic boundaries of the candidate. To achieve this goal, and to deal with the large baselines that retrieval candidates present, we iteratively refine the homography estimate.

Problem setting. The hierarchical pipeline takes as input an RGB query Q, and through EarthLoc [8] we retrieve a set of candidates. Candidates are extracted from a reference database that covers the landmass of the entire planet, as detailed in Sec. 4. We start from a RGB candidate C_0 with its corresponding *footprint* F_C, defined as the latitude and longitude of its four corners, *i.e.*, $F_C = \{(lat_0, lon_0), (lat_1, lon_1), (lat_2, lon_2), (lat_3, lon_3)\}$ and apply our EarthMatch iterative coregistration algorithm. The goal is to compute a homography H_Q that maps the candidate onto the query such that they overlap, thus providing the footprint of the query, F_Q.

3.1. Single-step pipeline

The simplest way to obtain F_Q is via a single homography estimated between the query Q and the initial candidate, C_0. Such a pipeline requires a matcher M which, given Q and C_0, produces an *overlap confidence*, a value that indicates the likelihood of the two images overlapping, and the homography H_0 to map the pixels of C_0 onto Q. Formally, $x_Q = H_0 x_{C_0}$, where x_Q, x_{C_0} are expressed in homogeneous coordinates. The homography H_0 is then simply used to estimate F_Q given the footprint of the candidate F_C.

Although this single-step pipeline would be sufficient in the ideal case - with a perfect matcher - in practice matchers have a hard time producing enough well-distrubuted keypoint correspondences to estimate a good homography. This is especially true when the overlap is limited, or when the transformation is significant, for example those that include large degrees of rotation and re-scaling. Further, aligning Earth observations imagery can have added difficulty, as imagery often depict barren, featureless landscapes which are notoriously hard to match [35]. In cases when there are many correspondences, but they are poorly distributed or, worse yet, highly localized to a single region of Q, H_0 is poorly constrained, resulting in a transform that does not well model the entire image area.

To overcome these issues, we implement a multi-step, iterative pipeline, which reduces the burden on the matcher by iteratively producing candidates that have increasingly more overlap with the query.

3.2. Iterative pipeline

The iterative pipeline picks up from the end of the single-step version, with a homography H_0 estimated from Q and C_0. From here, we apply H_0 to C_0 in order to obtain a new image C_1, which by construction will have a larger overlap with the query. Then, we use our new C_1 as a matching target for Q and obtain a second, improved estimate of the homography H_1.

Applying the homography directly to C_0 can lead C_1 to have some "empty" areas from locations depicted in Q but not in C_0. These areas require filling, often handled with black padding in computer vision libraries. To minimize this issue, instead of applying the homography directly to

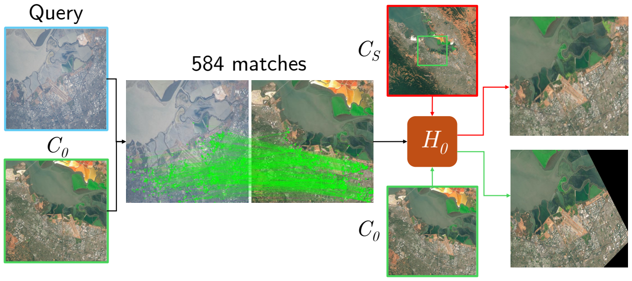

Figure 4. **Coregistration example.** Directly applying the homography to the candidate image results in empty areas (see bottom-right image). Transforming the image along with its surroundings solves this issue (top-right image).

C_0, we apply it on a "wider" satellite image (an image of a larger area), comprising C_0 and its 8 adjacent tiles, which we call C_S, where the S stands for *surroundings*. This reduces the chance that C_1 will have "empty" areas, unless a very strong homography is applied, which in practice we found to happen in a negligible number of cases. Examples of this process, by applying the homography to C_0 versus applying it to C_S, is shown in Fig. 4.

Once C_1 is generated from $H_0 * C_S$, we can obtain a new homography estimate H_1, which maps C_1 onto Q. Since we seek to geolocate with respect to the original candidate, we must track the estimated transformation between C_0 to Q. This can be computed as $H_Q^1 = H_1 * H_0$, as linear transformations can be equivalently applied sequentially by multiplying them. We repeat this process for a number of iterations, and at each iteration i we have $H_Q^i = \prod_{j=0}^{j=i} H_j$. Examples of Q, C_S, C_0, C_1, C_2 are shown in Fig. 5.

This iterative coregistration process is designed to stop if the matcher M predicts no overlap between the query and the candidate at any iteration of the pipeline. In such cases, EarthMatch will restart the algorithm with the next retrieved image from EarthLoc. In practice, we run EarthMatch for at most four iterations. At any iteration, the loop stops if the prediction is deemed invalid due to any of the following criteria: (i) the number of matches is less than 4 (too few to compute a homography), (ii) the predicted footprint is non-convex, indicating a failure in matching, or (iii) the predicted footprint has area bigger than C_S, *i.e.* 9 times larger than C_0, again indicating significant misalignment from matching. These stopping criteria ensure that almost no false positives are predicted, *i.e.* when the iterative process has successfully finished for a query-candidate pair, it is very unlikely that the final prediction is a false positive. In practice we found that the majority of models (11 of 16, see Sec. 5) do not generate *any* false positives when these stopping criteria are applied. However, in cases where false positives are obtained, it is necessary to find a technique to identify and remove them.

To this end, we propose computing a threshold T_{inl} on the number of inliers, and using this threshold to discard predictions with less inliers than T_{inl}. To obtain T_{inl}, we fit a logistic regression using the number of inliers for false and true positive astronaut photo/candidate image pairs. We set the threshold to a value that guarantees a 99.9% probability that any prediction with number of inliers above T_{inl} is in fact a true positive (*i.e.* a precision of 99.9%). This is in line with the requirements of APL [53], where having a high precision is more important than high recall.

4. Dataset

We evaluate our method on images from the Astronaut Imagery Matching Subset (AIMS) dataset [53]. These images have been manually located by NASA scientists, which have labelled each image's centerpoint in the form of geographic coordinates. These images are a representative subset of the over 4.5 million photographs of Earth taken by astronauts on the ISS. The full collection can be accessed at the Gateway to Astronaut Photography of Earth [3]. From the 323 images within AIMS, we removed 55 photographs with high tilt/oblique, as these often contain Earth limb and cannot be fit with a homography. This results in a subset of 268 images for our benchmark, of which some examples are shown in Fig. 2.

For each of these, we use an enhanced version of EarthLoc [8], representing the state-of-the-art model for Astronaut Photography Localization through image retrieval, to obtain the top-10 most similar images from a worldwide database of satellite geo-tagged images. This database contains over 13 million images, at different zooms (*i.e.* different meter-per-pixel resolutions), covering the entire landmass of the planet between latitude 60° and -60° (*i.e.* within the boundaries of the ISS's orbit). This covers all areas that could be depicted in an astronaut photo. Following EarthLoc [8] we apply $4x$ test-time augmentation on the database, by creating four copies of each database image, one each from a rotation of the original image of 0°, 90°, 180° and 270° respectively. This not only improves the results of the retrieval stage, but also embeds the top-10 candidates with a coarse estimate of the rotation, with respect to North, of the query: as an example, an ideally perfect retrieval method would find as first prediction a candidate that is no further than 45° rotated from the query.

Even with state-of-the-art retrieval, some queries do not have any positive within the top-10 candidates. Yet, we want our benchmark to be as realistic as possible (reflecting the deployment-time situation of localizing astronaut photos), so we still include these images within the test set. Of the 268 queries, 244 (91%) have at least one positive pre-

[3] https://eol.jsc.nasa.gov/

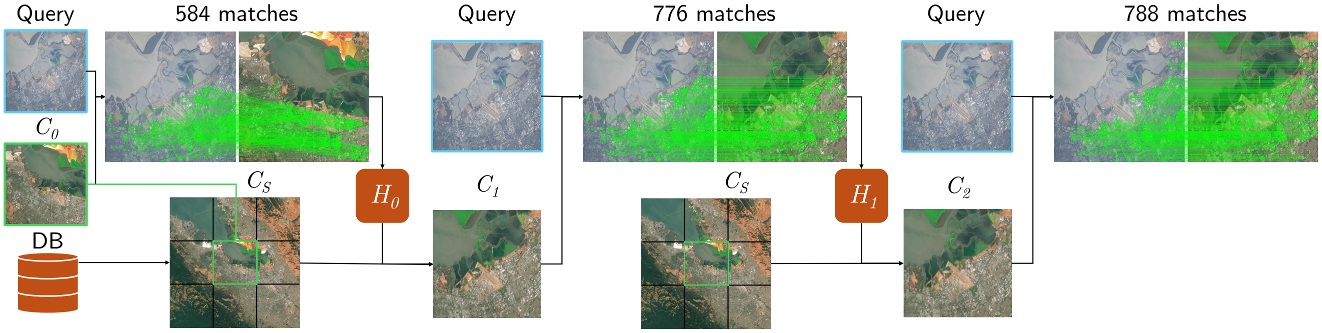

Figure 5. **Examples of images during the EarthMatch procedure.** The procedure starts with matching the query Q and the candidate C_0 to produce a first homography H_0. The candidate surroundings C_S is generated and H_0 applied to produce C_1. C_1 is then matched with Q, producing H_1 which applied to C_S yields C_2. This iterative process continues for a fixed number of iterations (4 iterations in our experiments).

diction within the top-10 candidates. For EarthMatch, this is the maximum number that can be precisely localized.

5. Experiments

5.1. Experimental setup

EarthMatch is performed on the top-10 retrieval predictions, and can be executed with virtually any image matching technique. We run the process (matching+homography) for 4 iterations per retrieval candidate, and at each iteration check the stopping criteria to reject invalid predictions. If the process reaches the end of the 4 iterations, a prediction is obtained, otherwise the process is performed on the next candidate. For some queries, no prediction is obtained at the end of the iterative process over the top-10 candidates: this characteristic reflects the requirement of APL, for which a prediction should be generated only if it's almost certainly correct, otherwise no prediction should be created. We compute the homography with a vanilla RANSAC, with the default parameters from OpenCV [28].

For our benchmark, we define a simple metric below, and we perform a large number of experiments with different matchers, image resolutions and number of keypoints, as described in the following section.

Metric. Each query image is labelled with a manually annotated centerpoint, so we deem an image to be correctly localized if the estimated footprint contains the centerpoint. This is a different metric from other astronaut photography works [8, 10, 53], but is more closely aligned with the localization task (see Sec. 7 for more details). Note that while in theory, this metric leaves open the possibility that a prediction could be considered correct in the case when an inaccurate footprint happens to contain the centerpoint, we find (after visually analyzing hundreds of predictions) that in practice this does not happen: *i.e.* a prediction which contains the centerpoint has almost perfect overlap with the query, as visually shown in Fig. 5.

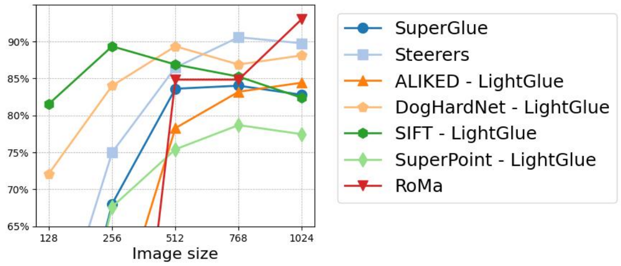

Figure 6. **Changing resolution.** Although RoMa is best overall, SIFT-LightGlue is best with low-resolution images, followed by DogHardNet-LightGlue. All learnable descriptors rapidly fail as images' resolution decreases.

Image Size. We repeat the main experiment with input image sizes ranging from 64 to 1024 px width/height (images are square), resized with a naive resize operation (no padding/cropping). The same size is applied to both the query and candidates.

Number of Keypoints. When a matcher is parameterized by the maximum number of keypoints, we analyze how matching performance changes as we adjust this value. We test max keypoints in (1024, 2048, 4096, 8192).

Matchers. We conduct a comprehensive benchmark of matchers, inserting matching models from different families of methods. We divide models into three main families: (i) detector-based local feature descriptors, with either nearest neighbor (NN) matching or learned matchers (*e.g.* SuperGlue [44] or LightGlue [31]); (ii) detector-free matchers, and finally (iii) dense matchers, which estimate a dense warp between two images. Specifically, we test:

- **Detector-based**: we consider handcrafted methods (SIFT [32], ORB [38]) as well as more recent

Method	All	Focal Length *f* (mm)				Camera Tilt		Cloud cover		Time	Threshold T_{inl}
	(268 / 244)	$f \leq 200$	$200 < f \leq 400$	$400 < f \leq 800$	$f > 800$	$< 40°$	$\geq 40°$	< 40 %	≥ 40 %	(*sec*)	(*#inliers*)
		(90 / 82)	(62 / 55)	(61 / 56)	(55 / 51)	(201 / 184)	(67 / 60)	(177 / 167)	(91 / 77)		
Detector-based											
SIFT - NN	70.5	64.6	74.5	64.3	82.4	74.5	58.3	76.6	57.1	9	16
ORB - NN	23.4	23.2	23.6	17.9	29.4	26.1	15.0	30.5	7.8	6	-
D2-Net	62.7	67.1	58.2	55.4	68.6	63.0	61.7	65.9	55.8	9	16
R2D2	30.7	32.9	30.9	26.8	31.4	28.3	38.3	31.7	28.6	9	48
SuperPoint - SuperGlue	84.0	79.3	89.1	82.1	88.2	84.8	81.7	86.8	77.9	6	-
DeDoDe	58.2	53.7	52.7	55.4	74.5	61.4	48.3	62.9	48.1	6	-
Steerers	90.6	86.6	90.9	87.5	**100.0**	**93.5**	81.7	94.6	81.8	7	-
ALIKED - LightGlue	84.4	81.7	90.9	75.0	92.2	85.3	81.7	88.0	76.6	4	-
DISK - LightGlue	55.3	58.5	52.7	44.6	64.7	57.1	50.0	61.1	42.9	4	-
DogHardNet - LightGlue	89.3	84.1	**96.4**	87.5	92.2	90.2	86.7	93.4	80.5	12	23
SIFT - LightGlue	89.3	85.4	89.1	**92.9**	92.2	90.8	85.0	94.0	79.2	11	-
SuperPoint - LightGlue	78.7	73.2	81.8	78.6	84.3	79.3	76.7	81.4	72.7	4	-
Detector-free											
Patch2Pix	68.0	65.9	74.5	60.7	72.5	69.0	65.0	69.5	64.9	7	74
Patch2Pix - SuperGlue	83.6	79.3	89.1	82.1	86.3	84.8	80.0	86.8	76.6	7	-
LoFTR	72.1	72.0	78.2	60.7	78.4	71.7	73.3	71.9	72.7	5	305
Dense matcher											
RoMa	**93.0**	**90.2**	90.9	**92.9**	**100.0**	92.4	**95.0**	**95.2**	**88.3**	11	128
Average	70.9	68.6	72.7	66.5	77.3	72.0	67.4	74.4	63.22	7.31	-

Table 1. **Percent of images correctly localized by each model, for different subsets of AIMS.** Each column indicates a different subset of AIMS, with the subset depending on the image acquisition conditions - camera focal length, camera tilt (*i.e.* the obliqueness of the camera w.r.t. Earth's surface) and the cloud coverage in the image. In each column header, the two numbers indicate the number of queries within the subset and the number of localizable queries, *i.e.* those for which at least one of the top-10 retrieval predictions is correct. Results show the **percentage of queries correctly localized among the "localizable" ones**, so that the upper bound is 100%. Methods with threshold $T_{inl} = -$ do not produce any false positives - negative candidates are rejected before 4 iterations - so there is no threshold to compute. Best **bolded**, second best underlined. Time is seconds per astronaut photo query for evaluation of all 10 candidates per query. Each model's score is shown for best *image size, number of keypoints* configuration (see Tab. 2).

Model	SIFT - NN	ORB - NN	D2-Net	R2D2	SuperPoint - SG	DeDoDe	Steerers	ALIKED - LG	DISK - LG	DogHardNet - LG	SIFT - LG	SuperPoint - LG	Patch2Pix	Patch2Pix - SG	LoFTR	RoMa
Best # Keypoints	8192	8192	-	8192	2048	4096	4096	4096	2048	8192	4096	1024	-	2048	-	1024
Best Image Size	768	1024	768	1024	768	768	768	1024	768	512	256	768	768	768	768	1024

Table 2. **Best image size and number of keypoints for each model**, resulting from a grid search for each method with exponentially growing image sizes between 64×64 and 1024×1024, and number of keypoints of 1024, 2048 and 4096. These are the hyperparameters used in Tab. 1. D2-Net, Patch2Pix and LoFTR do not take as input the number of keypoints. SP stands for SuperGlue, LG for LightGlue.

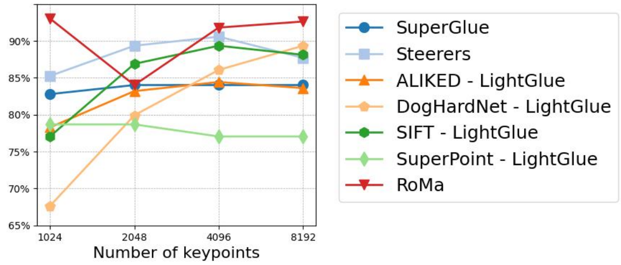

Figure 7. **Changing the number of keypoints** for the best-performing models. A flat line (only LoFTR) indicates that the model does not take as input the number of keypoints.

learned features (Dog-AffNet-HardNet [36], D2Net [18], R2D2 [42], DISK [57], ALIKED [61], DeDoDe [20]). Among these, R2D2 and D2Net require multiple forward passes with increasing resolution up to a user specified threshold. Steerers [10] was recently proposed to be rotation invariant. In addition to NN matching, we combine them with LightGlue [31] where possible;

- **Detector-free**: among these, LoFTR [54] is the most popular. We also experiment with Patch2Pix [63], which can also be utilized as a match refiner on top of SuperGlue [44];
- **Dense matching**: we use RoMa [21], which recently generated community interest, showing convincing results on wide baseline matching benchmarks. Although it does not directly output keypoints, they can be sampled in an arbitrary quantity, and thus we treat number of keypoints as a hyperparameter, similar to detector-based methods.

5.2. Results

Image Size. In Fig. 6 we observe varying performance depending on image size. Most models see performance degradation as image size decreases below 512 px, but the highest performing model at higher resolution, RoMA, sees the steepest dropoff, with no successful matches at 256 px sizes and lower. Dense models like RoMa appear to have a minimum size for matchability, whereas other sparse models have a more gradual performance dropoff. In the increasing size direction, many models begin to plateau at 512 px. This suggests that for these models, matching on larger images does not improve performance, only increases runtime/compute.

Number of Keypoints. There are two patterns that emerge from the (maximum) number of keypoints experiments (Fig. 7). For most models, performance increases with number of keypoints, plateauing somewhere between 2048 and 4096. This generally matches expectations, as more keypoints in an image increase the chances the same keypoint is selected in both images, increasing the change of forming a good correspondence.

AIMS Subsets. In Tab. 1 we examine matching performance on multiple subsets of AIMS, particularly with different focal lengths, low/high cloud cover (occlusion) and low/high tilt (*i.e.* the angle at which the camera was held, inducing larger perspective change in query images). Generally, RoMa is the best performing model across scenarios, showcasing robustness under strong visual changes. Due to its dense matching nature it requires a larger number of inliers for identifying good predictions. Among detector-based matchers, Steerers is the clear winner, achieving good performance with a much lower number of inliers, and at almost twice the speed of RoMa. In general, the methods of ALIKED + LightGlue, Steerers and RoMa produce optimal speed-accuracy trade-offs.

While the majority of models do not produce any false positives (*i.e.* for wrong candidates, EarthMatch stops before reaching the final iteration, thus rejecting the candidate), a few models do admit them, requiring computation of an inlier threshold T_{inl} to further validate correctness. When a threshold is computed, candidates must exceed this value to be considered a confident prediction. T_{inl} can be strongly affected by even a single highly confident false positive. This can increase T_{inl} such that a number of true positives are discarded as their inlier count does not exceed this inflated T_{inl}. This behaviour, seen in the rightmost column of Tab. 1, explains the uneven behavior of the RoMa curve in Fig. 7 and Fig. 6, where just a single high-confident false positive causes a strong dip in results.

In analyzing different subsets of AIMS, the most challenging setting is high cloud cover, which creates a natural occlusion to the matching process. Most methods also benefit from low camera tilt, which leads to high similarity between the query and the candidate. RoMa presents a notable exception to this trend, achieving better performances with high camera tilt.

5.3. Limitations

Limited hyperparamer tuning. One of the goals of this benchmark is the fair evaluation of existing image matching models on the out-of-distribution (w.r.t. their training sets) domain of Astronaut Photography Localization. To this end, in our evaluation we tuned only the two hyperparameters that are recurrent across multiple models, namely the input image size and the number of keypoints. For fair evaluations, we decided not to tune any model-specific hyperparameters, given that (1) this would give an advantage to methods with more hyperparameters and (2) it would lead to an explosion in the number of experiments. We therefore note that despite EarthMatch showing the potential of each model on the domain of APL, these results could potentially still be improved with per-model hyperparameter optimization, including the threshold used for RANSAC, which could be an interesting future development.

Map Projection. Finally, in this work we used a map projected representation of Earth. In particular, we used the Mercator projection, which preserves angles but not areas. While this can be problematic when viewing large areas (*i.e.* visually, areas closer to the poles appear larger than they are), we found this to have no noticeable impact on matchability or the estimated footprint of considered images, due to the images having a relatively small area with respect to the Earth's surface. It's possible other projections could yield improved matching in some, likely more polar, regions where the Mercator projection is visually more dissimilar to the true view of the Earth. This would have a stronger impact when registering Earth limb photos, which are however not considered within our benchmark.

6. Conclusion

In this work we present a pipeline to reliably and confidently estimate the footprint of astronaut photographs. The pipeline is made of a pre-existing retrieval method [8] and our newly introduced EarthMatch, an iterative coregistration algorithm that takes advantage of image matching models to output a predicted footprint and confidence. We run a large number of experiments with many matchers using different images sizes and number of keypoints, thoroughly evaluating their usability in the domain of Astronaut Photography Localization.

To foster future research and simplify reproducibility, we release the post-retrieval dataset of astronaut photo query and top 10 candidates used for our experiments, relieving researchers to having to compute the time-consuming large-scale retrieval step. We also release the code to replicate all experiments within this paper. This code is trivially extensible to future matching methods or other domains.

Finally, our pipeline offers an efficient and robust method for astronaut photography localization that can immediately be deployed on the existing 4.5 million and growing database of astronaut photos of Earth.

References

[1] Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping (slam): Part ii. *IEEE robotics & automation magazine*, 13(3):108–117, 2006. 2

[2] Simon Baker and Iain Matthews. Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56:221–255, 2004. 3

[3] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5173–5182, 2017. 3

[4] Giovanni Barbarani, Mohamad Mostafa, Hajali Bayramov, Gabriele Trivigno, Gabriele Berton, Carlo Masone, and Barbara Caputo. Are local features all you need for cross-domain visual place recognition? In *CVPRW*, pages 6155–6165, 2023. 3

[5] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110:346–359, 2008. 2

[6] Fabio Bellavia. Image matching by bare homography. *IEEE Transactions on Image Processing*, 33:696–708, 2024. 3

[7] Fabio Bellavia, Luca Morelli, Carlo Colombo, and Fabio Remondino. Progressive keypoint localization and refinement in image matching. pages 322–334, 2023. 3

[8] Gabriele Berton, Alex Stoken, Barbara Caputo, and Carlo Masone. Earthloc: Astronaut photography localization by indexing earth from space. In *CVPR*, 2024. 2, 3, 4, 5, 6, 8, 1

[9] Georg Bökman and Fredrik Kahl. A case for using rotation invariant features in state of the art feature matchers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5110–5119, 2022. 3

[10] Georg Bökman, Johan Edstedt, Michael Felsberg, and Fredrik Kahl. Steerers: A framework for rotation equivariant keypoint descriptors. *arXiv preprint arXiv:2312.02152*, 2023. 3, 6, 7, 1

[11] Gustau Camps-Valls, Devis Tuia, Xiao Xiang Zhu, and Markus Reichstein. *Deep learning for the Earth Sciences: A comprehensive approach to remote sensing, climate science and geosciences*. John Wiley & Sons, 2021. 1

[12] B. Cao, A. Araujo, and J. Sim. Unifying deep local and global features for image search. In *ECCV*, pages 726–743. Springer Int. Publishing, 2020. 3

[13] Si-Yuan Cao, Jianxin Hu, Zehua Sheng, and Hui-Liang Shen. Iterative deep homography estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1879–1888, 2022. 3

[14] Hongkai Chen, Zixin Luo, Lei Zhou, Yurun Tian, Mingmin Zhen, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. Aspanformer: Detector-free image matching with adaptive span transformer. *European Conference on Computer Vision (ECCV)*, 2022. 3

[15] Jonathan P Dandois and Erle C Ellis. Remote sensing of vegetation structure using computer vision. *Remote sensing*, 2(4):1157–1176, 2010. 1

[16] Tomasz Malisiewicz Daniel DeTone and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPRW*, 2018. 2

[17] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: part i. *IEEE robotics & automation magazine*, 13(2):99–110, 2006. 2

[18] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-Net: A Trainable CNN for Joint Detection and Description of Local Features. In *CVPR*, 2019. 3, 7

[19] Johan Edstedt, Ioannis Athanasiadis, Mårten Wadenbäck, and Michael Felsberg. Dkm: Dense kernelized feature matching for geometry estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17765–17775, 2023. 3

[20] Johan Edstedt, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Dedode: Detect, don't describe–describe, don't detect for local feature matching. *arXiv preprint arXiv:2308.08479*, 2023. 3, 7

[21] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Roma: Revisiting robust losses for dense feature matching. *arXiv preprint arXiv:2305.15404*, 2023. 3, 7

[22] JR Elliott. Earth observation for the assessment of earthquake hazard, risk and disaster management. *Surveys in geophysics*, 41(6):1323–1354, 2020. 1

[23] Kenton Fisher, Sara Schmidt, and Alex Stoken. Crew earth observations: New tools to support your research. In *12th Annual International Space Station Research and Development Conference*. Center for the Advancement of Science in Space, Inc., 2023. 1

[24] Saman Ghaffarian, João Valente, Mariska Van Der Voort, and Bedir Tekinerdogan. Effect of attention mechanism in deep learning-based remote sensing image processing: A systematic literature review. *Remote Sensing*, 13(15):2965, 2021. 1

[25] Pierre Gleize, Weiyao Wang, and Matt Feiszli. Silk: Simple learned keypoints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22499–22508, 2023. 3

[26] Stephen Hausler, Sourav Garg, Ming Xu, Michael Milford, and Tobias Fischer. Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition. In *CVPR*, pages 14141–14152, 2021. 3

[27] Dihe Huang, Ying Chen, Yong Liu, Jianlin Liu, Shang Xu, Wenlong Wu, Yikang Ding, Fan Tang, and Chengjie Wang. Adaptive assignment for geometry aware local feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5425–5434, 2023. 3

[28] Itseez. Open source computer vision library. https://github.com/itseez/opencv, 2015. 6

[29] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *International Journal of Computer Vision*, 129(2):517–547, 2021. 3

[30] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018. 3

[31] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. LightGlue: Local Feature Matching at Light Speed. In *ICCV*, 2023. 6, 7

[32] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004. 2, 3, 6

[33] Feng Lu, Lijun Zhang, Xiangyuan Lan, Shuting Dong, Yaowei Wang, and Chun Yuan. Towards seamless adaptation of pre-trained models for visual place recognition. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[34] Bruce Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision (ijcai). 1981. 3

[35] Jiayi Ma, Xingyu Jiang, Aoxiang Fan, Junjun Jiang, and Junchi Yan. Image matching from handcrafted to deep features: A survey. *International Journal of Computer Vision*, 129:23–79, 2020. 4

[36] Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Repeatability is not enough: Learning affine regions via discriminability. In *Proceedings of the European conference on computer vision (ECCV)*, pages 284–300, 2018. 7

[37] Hiroyuki Miyazaki, Masahiko Nagai, and Ryosuke Shibasaki. Reviews of geospatial information technology and collaborative data delivery for disaster risk management. *ISPRS international journal of geo-information*, 4(4): 1936–1964, 2015. 1

[38] Raúl Mur-Artal, J. M. M. Montiel, and Juan D. Tardós. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015. 2, 6

[39] Aadil Nathani, Rishi Iyer, Annabelle Wang, Aarnav Chitari, Adele Wilson, and Hannah Norris. Observing Earth From Space: Using Astronaut Photography to Analyze Geographical Climate Patterns. In *AGU Fall Meeting Abstracts*, pages ED44C–06, 2022. 1

[40] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *ICCV*, pages 3476–3485, 2017. 3

[41] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 3

[42] Jerome Revaud, Philippe Weinzaepfel, César Roberto de Souza, and Martin Humenberger. R2D2: repeatable and reliable detector and descriptor. In *NeurIPS*, 2019. 3, 7

[43] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12716–12725, 2019. 2

[44] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*, 2020. 3, 6, 7

[45] T. Sattler, W. Maddern, C. Toft, A. Torii, L. Hammarstrand, E. Stenborg, D. Safari, M. Okutomi, M. Pollefeys, J. Sivic, F. Kahl, and T. Pajdla. Benchmarking 6DOF outdoor visual localization in changing conditions. In *CVPR*, pages 8601–8610, 2018. 2

[46] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 2

[47] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 2

[48] Johannes A. Schultz, Maik Hartmann, Sascha Heinemann, Jens Janke, Carsten Jürgens, Dieter Oertel, Gernot Rücker, Frank Thonfeld, and Andreas Rienow. Diego: A multispectral thermal mission for earth observation on the international space station. *European Journal of Remote Sensing*, 53(sup2):28–38, 2020. 1

[49] Peter Schwind and Tobias Storch. Georeferencing urban nighttime lights imagery using street network maps. *Remote Sensing*, 14(11), 2022. 3

[50] Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, Pascal Fua, and Francesc Moreno-Noguer. Discriminative learning of deep convolutional feature point descriptors. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 118–126, 2015. 2

[51] W. L. Stefanov and C. A. Evans. Data collection for disaster response from the international space station. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-7/W3:851–855, 2015. 1

[52] W. L. Stefanov and C. A. Evans. Data collection for disaster response from the international space station. *International Symposium on Remote Sensing of Environment*, 2015. 1

[53] Alex Stoken and Kenton Fisher. Find my astronaut photo: Automated localization and georectification of astronaut photography. In *CVPRW*, pages 6196–6205, 2023. 2, 3, 5, 6, 1

[54] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. LoFTR: Detector-free local feature matching with transformers. *CVPR*, 2021. 3, 7

[55] Shitao Tang, Jiahui Zhang, Siyu Zhu, and Ping Tan. Quadtree attention for vision transformers. *ICLR*, 2022. 3

[56] Yurun Tian, Bin Fan, and Fuchao Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6128–6136, 2017. 2

[57] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient. *Advances in Neural Information Processing Systems*, 33:14254–14265, 2020. 3, 7

[58] Qing Wang, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. Matchformer: Interleaving attention in transformers for feature matching. In *Asian Conference on Computer Vision*, 2022. 3

[59] M. Justin Wilkinson and Yanni Gunnell. *Fluvial Megafans on Earth and Mars*. Cambridge University Press, 2023. 1

[60] Yoav Yair, Melody Korman, Colin Price, and Eytan Stibbe. Observing lightning and transient luminous events from the international space station during ilan-es: An astronaut's perspective. *Acta Astronautica*, 211:592–599, 2023. 1

[61] Xiaoming Zhao, Xingming Wu, Weihai Chen, Peter CY Chen, Qingsong Xu, and Zhengguo Li. Aliked: A lighter keypoint and descriptor extraction network via deformable transformation. *IEEE Transactions on Instrumentation and Measurement*, 2023. 3, 7

[62] Xiaoming Zhao, Xingming Wu, Jinyu Miao, Weihai Chen, Peter C. Y. Chen, and Zhengguo Li. Alike: Accurate and lightweight keypoint detection and descriptor extraction. *IEEE Transactions on Multimedia*, 25:3101–3112, 2023. 3

[63] Qunjie Zhou, Torsten Sattler, and Laura Leal-Taixe. Patch2pix: Epipolar-guided pixel-level correspondences. In *CVPR*, 2021. 3, 7

[64] Sijie Zhu, Linjie Yang, Chen Chen, Mubarak Shah, Xiaohui Shen, and Heng Wang. R2former: Unified retrieval and reranking transformer for place recognition. In *CVPR*, pages 19370–19380, 2023. 3