Pixel-wise Geo-registration of Drone Images

Qingyang Liu* David G. Shatwell* Parth Parag Kulkarni Mubarak Shah

Institute of Artificial Intelligence, University of Central Florida, USA

Abstract. Cross-view geo-registration is the task of aligning a query image to a geodetically accurate reference (e.g., satellite image), so that each query pixel maps to a real-world GPS coordinate. Most existing work addresses the related problem of cross-view geo-localization, where the goal is typically to estimate the camera center using retrieval, classification, matching, or regression. Because these approaches do not provide dense supervision, they are poorly suited for learning and evaluating pixel-wise alignment. We introduce *SkyReg*, a geometry-aware geo-registration model that estimates the transformation between the query and reference images by explicitly modeling the 3D scene geometry. Applying this transformation to warp the query into the reference frame yields pixel-wise geo-localization, without relying on 2D point matches, and remains robust to occlusions and large viewpoint changes. To enable training and standardized evaluation, we release (i) *SkyReg-Train*, a large-scale dataset of drone–satellite images annotated with per-pixel GPS coordinates, depth maps, and camera parameters derived from LiDAR and structure-from-motion, and (ii) *SkyReg-Bench*, a held-out benchmark of unseen Urban and Suburban scenes with the same dense annotations. SkyReg achieves state-of-the-art performance against strong retrieval and homography baselines, demonstrating the value of geometry-aware models and dense geodetic benchmarks for cross-view geo-registration. Dataset available at https://parthpk.github.io/skyreg-webpage.

Keywords: Cross-view, geo-localization, geo-registration, remote sensing, 3D reconstruction

1 Introduction

Cross-view geolocalization is the task of estimating the GPS coordinates of a query image (typically captured from a ground or drone viewpoint) by associating it with a geo-referenced satellite view. Existing methods generally follow two paradigms: *retrieval*, where given a query image the goal is to identify the best-matching satellite image from a large gallery of geo-tagged images [49, 59, 60], and *3-DOF pose regression*, where the reference is a single satellite image and the objective is to predict the query's location and compass orientation in the satellite image plane [49, 50]. Both paradigms, however, ultimately reduce the problem to estimating the camera geo-location.

* These authors contributed equally to this work.

This image-level formulation does not capture the spatial structure within a scene. In many practical settings, knowing the camera position alone is insufficient: a small localization error can translate into much larger spatial displacement for distant structures, especially in urban environments with significant depth variation and oblique viewpoints. Applications such as infrastructure inspection, urban planning, and disaster assessment require reasoning about the geographic location of *visible scene content*, not just the camera center. Related tasks, including map updating and defense-oriented geospatial analysis often demand precise geo-referencing of specific targets within the image. These requirements motivate a shift from image-level localization to *pixel-level geo-registration*, where each query pixel is mapped to real-world latitude and longitude.

Geo-registration has been studied extensively in remote sensing, but most existing approaches are tailored to small viewpoint changes, assume approximately planar geometry, and rely on homography-based alignment [4, 23, 57]. Evaluation is also often limited to sparse correspondences rather than dense, per-pixel geodetic consistency. As a result, these methods do not adequately address satellite–drone geo-registration, where large viewpoint changes and heterogeneous sensing modalities (orthographic satellite imagery versus perspective drone imagery) make the problem substantially more challenging. Moreover, the field lacks standardized training data and benchmark protocols for assessing fine-grained cross-view geo-registration across methods.

In this paper, we propose *SkyReg*, a drone geo-registration method that leverages state-of-the-art 3D reconstruction backbones to model scene geometry and predict pixel-wise GPS coordinates, even under occlusions and large viewpoint differences. Unlike previous approaches that rely on homography estimation or assume known satellite DEMs, SkyReg uses camera parameters and dense point maps for the query and reference images. It then computes a query-to-reference transformation by composing the geometric mappings that lift each image plane into a shared 3D reference frame. The resulting warp projects query pixels into the reference image plane, where geodetic coordinates are assigned by interpolating the reference latitude–longitude map.

We also introduce *SkyReg-Train*, a diverse training set of drone images and geo-referenced satellite views, covering multiple scene types and acquisition conditions. Each image is annotated with dense per-pixel latitude–longitude labels, along with depth maps and camera parameters used as training signals, which are derived from LiDAR and structure-from-motion (SfM). For quantitative evaluation, we propose *SkyReg-Bench*, a held-out benchmark of unseen Urban and Suburban scenes with the same pixel-level GPS annotations, designed to evaluate models under challenging distribution shifts. We evaluate a broad suite of baselines, including feature matching and homography-based alignment. Across all settings, we observe a substantial gap between image-level localization and dense geo-registration, underscoring the need for geometry-aware cross-view methods tailored to pixel-wise alignment.

To summarize, our contributions are the following:

- We introduce *SkyReg*, the first 3D-aware drone–satellite geo-registration method that explicitly models the 3D scene geometry to infer pixel-wise GPS coordinates, remaining robust under occlusions and large viewpoint changes.
- We release *SkyReg-Train*, a large-scale training dataset for drone geo-registration, providing dense per-pixel latitude–longitude labels together with depth maps and camera parameters across diverse scenes.
- We propose *SkyReg-Bench*, a held-out benchmark for evaluating generalization to unseen Urban and Suburban environments, multiple reference modalities (orthorectified and perspective), and varied scene layouts.
- We formalize pixel-wise geo-registration with standardized evaluation protocols, metrics, and benchmark a broad set of retrieval- and homography-based baselines, showing consistent improvements in geodetic error and recall.

2 Background and Related Work

2.1 Geolocalization

Geo-localization is the task of inferring *where* an image was captured by estimating its GPS coordinates. Existing approaches can be broadly grouped into four directions. (i) **Global Level Classification** methods [9, 17, 22, 24, 30, 43, 48] divide the Earth into discrete geo-cells and train a model to predict the corresponding location class. These methods require only a single forward pass at inference time, but their accuracy is inherently limited by the geo-cell resolution. (ii) **City Level Classification** methods [40, 52, 53], similar to Global Level Classification, try to classify input into discrete geo-cells, except the geo-cell gallery are only at a city level instead of global level, thus allowing a more fine grained classification. (iii) **Retrieval** methods predict location by matching a query image to a gallery of geo-tagged references and transferring the retrieved reference coordinates to the query. Galleries may consist of GPS embeddings [15, 16, 31, 32, 42], ground view images (same-view) [1, 2, 5, 6, 13, 14, 41], or top-down satellite images (cross-view) [19, 25, 35–37, 39, 55, 58–60]. Cross-view methods like TransGeo [59] learn a shared embedding space for ground-level and satellite imagery with reranking for refining nearest-neighbor retrieval, while University-1652 [58] jointly embeds ground, drone, and satellite views into a unified feature space. (iv) **Fine-grained cross-view localization** methods [46, 49–51, 54, 60] go a step further by estimating the camera location within a geodetically accurate reference image (e.g., satellite), often leveraging an approximate GPS prior to restrict the search region.

Although these approaches can accurately recover the camera's geo-location and sometimes the 3-DoF camera orientation, they operate at the image level in the 2D domain and therefore do not provide pixel-wise geographic correspondence. Estimating the camera center reveals the observer's position, but it does not determine where a visible object (e.g., a building facade or vehicle) lies in real-world GPS coordinates. Unlike geo-localization, which assigns a single coordinate pair to the entire image, geo-registration estimates coordinates for each pixel, enabling dense geographic alignment.

2.2 Georegistration

Geo-registration aligns a query image to an Earth-fixed coordinate system (e.g., WGS84/UTM) so that image pixels can be mapped to geodetic coordinates. Given a query image $I_q(u, v)$ and a geodetically accurate reference $I_{\text{ref}}(x, y)$, the goal is to estimate a mapping $T : (u, v) \mapsto (x, y)$ such that $I_q(u, v) \approx I_{\text{ref}}(T(u, v))$.

Early work primarily treated geo-registration as pose refinement: starting from a noisy initialization, methods iteratively improved alignment using DEM-based rendering and template/feature matching [33,34], or stabilized aerial video by remapping frames with planar homographies [12], which can be brittle in non-planar scenes with strong 3D structure. Later approaches include joint correction across multiple satellite images via probabilistic 3D surface reconstruction [23] and learned cross-view descriptors followed by feature matching [57], but many formulations still rely on 2D homography estimation and therefore struggle under large parallax and viewpoint changes.

Recent progress in learned correspondence has improved robustness in challenging regimes: SuperPoint/SuperGlue [10, 27] and RoMa [11] provide stronger sparse and dense matches, and have been adopted in modern homography-based registration pipelines [4]. However, many systems still default to planar warps (e.g., homographies) when aligning cross-view imagery, and recent 3D rendering-based pipelines [7] typically require multiple drone views to build consistent geometry. In contrast, we target geo-registration from a *single* drone image to a geodetically accurate satellite reference, motivating geometry-aware, feed-forward inference beyond 2D alignment.

2.3 3D Reconstruction and Multi-View Geometry

Models utilizing 3D have become increasingly prominent [21]. Recent feed-forward reconstruction models such as DUSt3R [47], MASt3R [18], and VGGT [45] advance multi-view geometry by predicting dense *point maps* and *camera parameters* from a sparse set of images. They recover 3D structure in a shared coordinate frame (up to a global similarity transform) without running a full SfM pipeline. However, these models often degrade under the extreme appearance and viewpoint gaps of satellite–drone alignment: satellite imagery differs in scale, projection (near-orthographic), and radiometry, and the baseline to oblique drone views is typically outside the regime covered by standard reconstruction benchmarks. Consequently, predicted point maps and camera poses can become unreliable, limiting direct use for satellite-conditioned geo-registration.

Aerial MegaDepth [44] partially bridges this gap by providing aerial scenes with calibrated intrinsics, poses, and depth in a unified frame. While valuable for cross-view reconstruction, it does not target the satellite setting or explicitly benchmark pixel-level geo-registration.

Overall, existing benchmarks and methods largely operate at the image-level (camera-center localization) or rely on planar homographies and/or multi-view inputs, which are brittle under the parallax, occlusions, and modality gap of satellite–drone pairs. Meanwhile, feed-forward 3D reconstruction is promising

but lacks satellite-conditioned training data and standardized pixel-level evaluation. This motivates our geometry-aware, single-image geo-registration model and the accompanying dataset and benchmark with dense geodetic supervision.

3 Method

SkyReg operates in two stages. First, a feed-forward 3D reconstruction backbone predicts dense point maps, camera intrinsics, and the relative pose between the query and reference views. Second, we compose these predictions into a query-to-reference warping function that lifts query pixels into 3D and reprojects them onto the reference image plane. We then obtain a GPS coordinate for each query pixel by interpolating the reference's geodetic map at the warped locations.

3.1 Inputs

The inputs to our geo-registration framework consist of a georeferenced reference image $I_r(\mathbf{x}_r)$ and an uncalibrated query image $I_q(\mathbf{x}_q)$, where $\mathbf{x} = [u, v]^\top$ corresponds to pixel coordinates. The query image $I_q(\mathbf{x}_q)$ is captured from a low altitude, often at an oblique angle, and lacks any associated metadata. Its only available information is the pixel intensity values. The reference image, typically a satellite view, has known intrinsics and pose $(R^{r\to w}, \mathbf{t}^{r\to w})$ relative to an Earth-Centered, Earth-Fixed (ECEF) world coordinate system. This image may optionally include a Digital Elevation Map (DEM) $D(\mathbf{x}_r)$ that provides height information for each pixel. In addition, we assume that each pixel in the reference image and DEM can be mapped to a 3D location on the surface of the WGS-84 ellipsoid in the ECEF coordinate system, such that $\mathbf{X}_{\text{WGS84}} = \mathcal{E}(\mathbf{x}_r) = [X_{\text{WGS84}}, Y_{\text{WGS84}}, Z_{\text{WGS84}}]^\top$. Each 3D point $\mathbf{X}_{\text{WGS84}}$ corresponds to latitude-longitude pairs (φ, λ) described by:

$$
\begin{aligned}
e^2 &= 1 - \frac{b^2}{a^2}, \quad N(\varphi) = \frac{a}{\sqrt{1 - e^2 \sin^2 \varphi}}, \\
X &= N(\varphi) \cos\varphi \cos\lambda, \\
Y &= N(\varphi) \cos\varphi \sin\lambda, \\
Z &= (1 - e^2) N(\varphi) \sin\varphi,
\end{aligned}
\tag{1}
$$

where a and b are the WGS84 ellipsoid semi-major and semi-minor axes.

We aim to **georegister** the query image by finding the warping function $\mathbf{x}'_q = \text{warp}(\mathbf{x}_q)$ that maps pixel coordinates between the two image planes. Unlike traditional planar homography-based approaches, our formulation explicitly models 3D geometry and does not assume a flat ground plane, making it suitable for low-altitude images with noticeable parallax and terrain relief. Once the warping function is estimated, we can assign a geographic coordinate (latitude and longitude, or equivalently $\mathbf{X}_{\text{WGS84}}$) to every query pixel based on their warped location in the reference image. Additionally, we can project the query image onto the geodetically reference image and create a mosaic.

3.2 Feed-Forward 3D Reconstruction

Our method begins by estimating the geometric relationship between the query and reference images using a feed-forward multi-view 3D reconstruction network (i.e., MASt3R [18]). Given I_q and I_r, the model predicts two sets of point-maps in two forward passes: $(X^{r,r}, X^{q,r})$ and $(X^{q,q}, X^{r,q})$, where $X^{i,j}$ corresponds to the point-map of image i in camera's j reference frame, and each pixel in point-map $X^{i,j}$ is a 3D vector.

After predicting the point-maps, we can estimate the intrinsics of the query image K_q and reference image K_r, as well as the relative rotation $R^{q\to r}$, translation $\mathbf{t}^{q\to r}$, and scale $\sigma^{q\to r}$. Following DUSt3R [47], the intrinsics are recovered via solving an optimization problem. We assume that the principal point $\mathbf{c}$ is centered and solve for the query focal length f_q^* as follows:

$$\arg\min_{f_q} \sum_{\mathbf{x}_q} \left|\left| (\mathbf{x}_q - \mathbf{c}) - f_q \frac{X^{q,q}_{0:1}(\mathbf{x}_q)}{X^{q,q}_{2}(\mathbf{x}_q)} \right|\right|, \tag{2}$$

where $X^{q,q}_k(\mathbf{x}_q)$ represents the kth channel of the point map, *i.e.*, the X, Y, or Z coordinates of the 3D point at pixel location $\mathbf{x}_q$. This process can also be performed similarly for the reference image to obtain f_r^* and K_r. Similarly, we can recover the relative transformation $\hat{P}^{q\to r}$ through Procrustes alignment:

$$\arg\min_{\hat{P}^{q\to r}} \sum_{\mathbf{x}_q} \left|\left| h^{-1}(\hat{P}^{q\to r} h(X^{q,q}(\mathbf{x}_q)) - X^{q,r}(\mathbf{x}_q) \right|\right|^2, \tag{3}$$

$$\hat{P}^{q\to r} = \begin{bmatrix} \sigma^{q\to r} R^{q\to r} & \mathbf{t}^{q\to r} \\ \mathbf{0}^\top & 1 \end{bmatrix}, \tag{4}$$

where $h(\cdot)$ transforms the 3D point into homogeneous coordinates.

3.3 Pixel-Wise Geo-Registration

In order to geo-register the query image, we need to find the warping function that maps pixel coordinates between the query and reference image planes, such that $\mathbf{x}'_q = \text{warp}(\mathbf{x}_q)$. We can easily create this function from the predicted query point-map. Since $X^{q,r}$ is already in the reference coordinate frame, we simply project the 3D points back into the reference image plane using K_r. Mathematically, we define it as:

$$\mathbf{x}'_q = \text{warp}(\mathbf{x}_q) = h^{-1}(K_r X^{q,r}(\mathbf{x}_q)), \tag{5}$$

Using this warping function, we can easily find the global 3D location (or alternatively, the latitude and longitude) of each query pixel simply by finding its location in the reference image plane passing it to the ellipsoid mapping function $\mathcal{E}(\cdot)$:

$$\mathbf{X}_{\text{WGS84}} = \mathcal{E}(\mathbf{x}'_q) = \mathcal{E}(\text{warp}(\mathbf{x}_q)) = \mathcal{E}(h^{-1}(K_r X^{q,r}(\mathbf{x}_q))). \tag{6}$$

Alternatively, we can create a mosaic of the query image overlaid on top of the reference by assigning the RGB values of the query to the warped pixel coordinates in the reference image space.

3.4 Training Details

We instantiate our feed-forward geo-registration model from the MASt3R [18] architecture and train it on the SkyReg-Train dataset, initializing from the checkpoint pretrained on AerialMegaDepth [44]. During training, we feed the query image and the geo-referenced reference image as a pair into the 3D backbone. For each forward pass, the network predicts two dense point maps, both expressed in the reference camera coordinate frame, together with per-pixel confidence maps. As supervision, we construct ground-truth point maps from the depth maps and camera parameters. Specifically, for a pixel (i, j) in view n, its 3D point expressed in the coordinate frame of view m is computed as

$$X_{i,j}^{n,m} = P_m P_n^{-1}\, h\big(K^{-1}[iD_{i,j},\, jD_{i,j},\, D_{i,j}]^\top\big)\,, \tag{7}$$

where K is the intrinsic matrix, $D_{i,j}$ is the depth at pixel (i, j), P_n and P_m denote the camera extrinsic matrices, and $h(\cdot)$ is the homogeneous lifting operator.

We supervise the predicted point maps using the confidence-weighted regression loss introduced in DUSt3R [47]. The loss is defined as

$$\mathcal{L}_{\text{conf}} = \sum_{v\in\{1,2\}} \sum_{i\in\mathcal{D}^v} C_i^{v,1}\, \ell_{\text{regr}}(v, i) \;-\; \alpha \log C_i^{v,1}, \tag{8}$$

where

$$\ell_{\text{regr}}(v, i) = \left\| \frac{1}{z} X_i^{v,1} - \frac{1}{z} \bar{X}_i^{\bar{v},1} \right\|. \tag{9}$$

Here, $X_i^{v,1}$ denotes the predicted 3D point, $\bar{X}_i^{\bar{v},1}$ is the corresponding ground-truth point, $C_i^{v,1}$ is the predicted confidence, $\mathcal{D}^v$ is the set of valid supervised pixels in view v, and α controls the confidence regularization term.

We train the model for 20 epochs. To preserve the strong geometric priors of the 3D backbone while adapting it to the cross-view geo-registration setting, each epoch is formed by a balanced mixture of training pairs from multiple sources. In particular, every epoch contains 20k randomly sampled pairs from Urban scenes, 20k pairs from Landmarks scenes, and 20k pairs from traditional 3D reconstruction datasets, including ARKitScenes [3], ScanNet++ [56], and CO3D [26].

We optimize the network with AdamW [20] using a base learning rate of 1e-4, weight decay of 0.05, and momentum parameters $(\beta_1, \beta_2) = (0.9, 0.95)$. We use an effective batch size of 48 image pairs, train with input resolutions between 512×160 and 512×384. The learning rate is scheduled using a cosine decay with one warm-up iteration. Training is performed on three GPUs for approximately 20 hours. The models are trained on NVIDIA H100 GPUs, and inference is conducted on a single NVIDIA RTX 3090 GPU.

4 Dataset

We introduce *SkyReg*[1], a dataset for drone–satellite geo-registration, with *SkyReg-Train* and *SkyReg-Bench* as its training and benchmarking components respec-

[1] https://github.com/dshatwell23/skyreg-dataset

tively. SkyReg spans diverse locations, scene types, and camera configurations, and provides dense supervision, including per-pixel GPS coordinates, metric depth, and full 6-DoF camera poses, which enable evaluation beyond image-level geo-localization.

Each sample contains a geodetically accurate reference image $I_r(\mathbf{x}_r)$ with a per-pixel latitude–longitude array. We support two reference modalities: (i) orthorectified satellite tiles paired with a digital elevation map $DEM(\mathbf{x}_r)$, and (ii) perspective-projection satellite views paired with metric depth $D_r(\mathbf{x}_r)$ and camera parameters $(K_r, R_r^{w\to r}, \mathbf{t}_r^{w\to r})$ defined in an ECEF world frame. The query is always a perspective-projection drone image $I_q(\mathbf{x}_q)$ with metric depth $D_q(\mathbf{x}_q)$ and camera parameters $(K_q, R_q^{w\to q}, \mathbf{t}_q^{w\to q})$, also in ECEF.

We aggregate data from multiple sources into three subsets: **Urban**, **Landmarks**, and **Suburban**. (i) **Urban** is built from three U.S. cities (Chicago, San Francisco, and Seattle) by sampling GPS locations to match the coordinate distribution of VIGOR [60]. (ii) **Landmarks** comprises 83 scenes centered on well-known landmarks worldwide, leveraging drone imagery from AerialMegaDepth [44]. (iii) **Suburban** follows the GPS distribution of drone images in the WRIVA Public dataset [8]. Urban satellite references are orthorectified Bing Maps tiles, while the drone images are rendered from Google Earth. The Landmarks and Suburban subsets use perspective-projection drone and satellite views also rendered in Google Earth. We derive metric depth maps for Urban scenes from USGS LiDAR, and for Landmarks/Suburban we run COLMAP with fixed camera parameters, optimizing only for metric depth. Figures 1 and 2 show the data collection pipeline and sample images from the Urban subset, respectively. Qualitative examples for other subsets are presented in the supplementary material.

4.1 Train and Test Splits

SkyReg-Train. Our training split combines two cities from the Urban subset (Seattle and San Francisco) with all scenes from the Landmarks subset. Urban scenes span a wide range of buildings, streets, and landscapes across both cities. Data acquisition in this subset follows a relatively regular sampling pattern, pairing orthorectified satellite tiles with drone images captured at fixed altitudes and yaw angles. This split is designed to expose models to a large number of distinct environments and to encourage robustness to the substantial viewpoint and scale gap between orthographic satellite imagery and perspective drone views captured closer to the ground.

In contrast, the Landmarks subset contains fewer scenes but exhibits greater viewpoint diversity. For each landmark scene, we randomize both drone and satellite camera intrinsics and extrinsics, producing a broad distribution of camera configurations and relative poses. Together, the Urban and Landmarks subsets are complementary: the former provides breadth across many scenes under controlled sensor assumptions, while the latter provides challenging geometric variation, enabling models trained on SkyReg-Train to generalize across scene layouts, acquisition conditions, and camera models.

Table 1: SkyReg dataset statistics. Number of drone queries and satellite references for each split and subset, along with the satellite projection type and the depth source used to derive dense geodetic supervision (LiDAR for Urban; SfM for Landmarks/Suburban).

Split	Subset	Sat. Proj.	Depth	#Drone	#Sat.
Train	Urban (SEA & SF)	Orthographic	LiDAR	27,836	152
	Landmarks	Perspective	SfM	87,778	7,766
Test	Urban (CHI)	Orthographic	LiDAR	13,191	238
	Suburban	Perspective	SfM	950	5

SkyReg-Bench. SkyReg-Bench is created to evaluate geo-registration under strict geographic generalization, using images from entirely unseen locations and covering a broad range of environments, sensor models, and scene layouts. It combines a held-out city from the Urban subset (Chicago) with all scenes from the Suburban subset, yielding a challenging test set that differs substantially from the training distribution. The Urban subset is designed to evaluate robustness to new city-scale appearance statistics and scene semantics under the same orthorectified satellite–to–drone setting, while the Suburban subset evaluates generalization across less structured environments and more diverse acquisition conditions. Together, these components complement one another: the Urban split measures transfer across dense man-made scenes at scale, whereas the Suburban split emphasizes robustness to distribution shifts in viewpoint, layout, and scene type, providing a comprehensive benchmark for pixel-level cross-view geo-registration.

Table 1 summarizes the dataset composition. We report the number of *images* for both drone and satellite for each subset, how the *depth* is derived, and the type of *satellite projection*.

4.2 Data Collection Details

Urban Scenes. We sample 390 orthorectified satellite tiles over Chicago, Seattle, and San Francisco. Reference images are taken from Bing Maps RGB tiles with side lengths of 380–750 m. Query drone views are rendered in Google Earth with fixed intrinsics and extrinsics: cameras are placed 100 m above ground, pitched at 45°, and assigned a random yaw. Viewpoints are selected by sampling GPS coordinates from VIGOR [60] and setting the camera pose so that the principal ray intersects the ground plane at the sampled location.

For depth supervision, we download geo-referenced LiDAR point clouds for each tile from USGS 3DEP [38], expressed in the global ECEF frame. We form $DEM(\mathbf{x}_r)$ for each reference tile via interpolation from the LiDAR elevations. To obtain dense drone depth-maps, we first note that a world point $\tilde{X} = [X^\top, 1]^\top$ projects to the query as $\tilde{\mathbf{x}}_q \sim K_q P^{w\to q}\tilde{X}$, where $P^{w\to q} = [(R^{q\to w})^\top \mid -(R^{q\to w})^\top \mathbf{t}^{q\to w}]$ and $\tilde{\mathbf{x}}_q = [u, v, 1]^\top$. Because direct projection

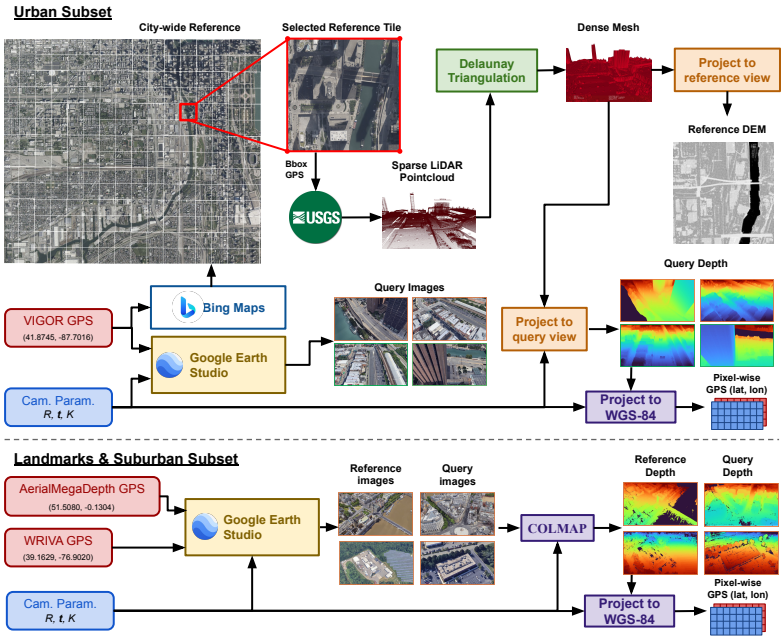


Fig. 1: SkyReg data construction for SkyReg-Train and SkyReg-Bench. Top (Urban). We use VIGOR [60] GPS coordinates from three cities (San Francisco, Seattle, Chicago) to download orthorectified Bing Maps tiles and render perspective drone views in Google Earth Studio. LiDAR point clouds from USGS are densified via Delaunay triangulation and projected into the known cameras to produce DEMs, depth maps, and per-pixel latitude–longitude labels through projection to the WGS-84 ellipsoid. **Bottom (Landmarks & Suburban).** Landmark drone images come from AerialMegaDepth [44], while landmark references and all Suburban imagery are rendered in Google Earth Studio. Depth is obtained with a structure-from-motion pipeline using fixed camera parameters, and pixel-wise GPS is computed using the same 3D-to-WGS-84 projection as in Urban.

yields sparse samples, we densify the LiDAR into a triangular mesh $\mathcal{M}$ and compute per-pixel intersections by ray tracing:

$$\mathbf{r}_q(s) = \mathbf{t}^{q\to w} + s\, R^{q\to w} K_q^{-1} \tilde{\mathbf{x}}_q, \qquad X_q(\mathbf{x}_q) = \text{Intersect}\big(\mathbf{r}_q(\cdot), \mathcal{M}\big). \tag{10}$$

The query depth map $D(\mathbf{x}_q)$ is then obtained from the intersection point (its depth along the camera ray).

Landmark Scenes. We build the Landmarks subset from AerialMegaDepth scenes [44], which provide drone images with depth and camera parameters, and augment each scene with perspective satellite views rendered in Google Earth Studio. For each scene, we compute the convex hull of drone camera centers projected onto the ground plane, then sample 100 ground-plane anchors $\mathcal{A}$ uniformly inside the hull and up to 100 satellite camera centers $\mathcal{B}$ by sampling



Fig. 2: Representative samples from SkyReg-Train (Urban). The center panel shows a large orthorectified satellite image of San Francisco. We create reference satellite images from local reference tiles. The left/right panels show example drone queries associated with each reference tile.

horizontal locations within the hull and altitudes up to 2500 m. We pair points in $\mathcal{A}$ and $\mathcal{B}$, place a virtual satellite camera at each $\mathbf{b} \in \mathcal{B}$, and orient it toward its paired anchor in $\mathcal{A}$. We further randomize the field of view in $[10°, 45°]$ to obtain diverse intrinsics and viewpoints.

Since these scenes span worldwide locations where LiDAR is often unavailable, we generate depth via a COLMAP SfM/MVS pipeline [28,29]: SfM recovers metric-scale point clouds (with bundle adjustment), and MVS produces dense per-view metric depth maps. In total, Landmarks contains 7.7k satellite and 87k drone perspective images with depth, 6-DoF poses, and intrinsics.

Suburban Scenes. The Suburban scenes consist of 5 scenes from the WRIVA Public Dataset [8] with 950 drone images. This subset is dominated by non-urban environments and areas affected by natural disasters, yielding visual characteristics and scene structure that differ substantially from the training distribution. As a result, it serves as a challenging testbed for evaluating robustness and generalization to previously unseen settings. Both satellite and drone imagery in the Suburban setting use perspective projection and are collected similarly to the Landmarks subset. Specifically, we adopt the GPS spatial distribution provided by the WRIVA Public Dataset [8] to select acquisition locations, and then collect corresponding views from Google Earth with full camera parameter annotations. Finally, we run an SfM pipeline to reconstruct scene geometry and generate dense depth maps for the satellite and drone views.

5 Benchmarking and Evaluation

5.1 Evaluation Protocol

At test time, we feed the query and reference images into the SkyReg model and obtain the predicted scene geometry, including dense 3D point maps and the

camera parameters required to define the cross-view warping function from Eq. 5. Using this function, we map each valid query pixel into the reference image plane and transfer the GPS coordinates to the query view by interpolating the geodetic map of the reference at the warped query pixel locations, yielding a dense set of predicted GPS coordinates for the query image.

We propose two complementary ways to quantify the model's performance. The first is a *dense* metric that evaluates pixel-level geo-registration accuracy. For each query, after warping the query into the reference frame, we compare the predicted GPS coordinate of each valid query pixel against its ground-truth coordinate and find the *geodetic error* (GE). Specifically, we compute the ground-truth latitude and longitude of each valid query pixel from the known depth maps and camera parameters. This yields, for each image, a set of pixels with predicted and ground-truth GPS coordinates. We then compute the per-pixel error using the Haversine distance between the predicted and ground-truth latitude–longitude pairs, from which we assign the median error as the geodetic error. Given two coordinates (ϕ_1, λ_1) and (ϕ_2, λ_2), the Haversine distance is

$$d_{\text{hav}} = 2R \arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right), \quad (11)$$

where R is the Earth's radius, and all angles are expressed in radians.

The second metric measures geo-localization performance. For each query, we compute a *single* GPS coordinate to facilitate comparison with image-level cross-view geo-localization methods. Concretely, we compute an image-level GPS estimate by finding the median of the predicted pixel-wise coordinates over all valid pixels, and do the same for the ground-truth coordinates. We then report the geodetic distance between these two image-level estimates. This protocol mirrors retrieval baselines, where a query location is typically taken from a single point (often the center) of the retrieved reference.

Finally, some baselines may fail catastrophically on a subset of samples and produce no valid registration. To measure robustness, we also report *recall*—the fraction of query images that are successfully geo-registered under multiple distance thresholds. An image is considered correctly registered at threshold τ if its geodetic error is below τ m.

5.2 Results and Discussion

Tables 2 and 3 show that our method consistently outperforms retrieval-based and homography-based baselines by a large margin in both the pixel-wise and median-GPS mean geodetic error and recall. Relative to the strongest competing baseline (RoMa), which uses dense point correspondence, we reduce the pixel-wise mean geodetic error by 88.18 m and 90.76 m in the Urban and Suburban splits, respectively. On the median-GPS evaluation we observe a similar trend, reporting 68.30 m and 76.33 m lower mean geodetic error than RoMa. RoMa Dense, which uses the dense point correspondence from RoMa, has similar performance. Recall exhibits the same trend: SkyReg registers a substantially

Table 2: Pixel-wise georegistration performance on the Urban and Suburban benchmarks. We report the mean geodetic error (GE) in meters, and the recall of images registered correctly under τ as a percentage.

Method	Urban					Suburban				
	GE ↓	Recall@τ (m) ↑				GE ↓	Recall@τ (m) ↑			
		20	30	40	50		20	30	40	50
SP+SG [10, 27]	209.40	1.40	1.94	2.34	2.82	223.69	36.42	37.15	37.47	37.47
RoMa [11]	133.06	25.69	27.18	28.35	29.33	112.10	60.00	61.26	62.25	63.15
RoMa Dense [11]	131.45	17.62	20.08	22.43	24.72	111.25	51.89	52.52	53.57	54.42
Ours	**44.88**	**78.78**	**80.03**	**80.53**	**80.94**	**21.34**	**74.94**	**93.57**	**97.78**	**98.52**

Table 3: Median-GPS geo-localization performance on Urban and Suburban benchmarks. We report the mean geodetic error (GE) in meters, and the recall of images geolocated correctly under τ as a percentage.

Method	Urban					Suburban				
	GE ↓	Recall@τ (m) ↑				GE ↓	Recall@τ (m) ↑			
		20	30	40	50		20	30	40	50
TransGeo [59]	160.98	1.50	3.47	5.98	9.06	272.53	0.31	1.47	1.99	2.94
University-1652 [58]	121.17	4.25	9.06	15.01	21.48	209.60	1.05	2.42	3.36	4.84
EarthMatch [4]	163.60	1.80	3.83	6.57	9.13	302.34	0.42	0.74	1.26	1.79
SP+SG [10, 27]	134.09	1.97	3.60	5.76	8.07	122.47	32.41	35.37	38.33	40.76
RoMa [11]	104.70	21.75	26.42	29.34	31.77	97.78	48.05	53.62	57.29	59.28
RoMa Dense [11]	98.38	18.66	24.58	29.41	34.52	86.91	40.29	45.54	48.68	52.25
Ours	**36.40**	**66.72**	**78.16**	**80.47**	**81.57**	**21.45**	**75.32**	**88.55**	**93.17**	**95.27**

larger fraction of images across all distance thresholds, indicating robust behavior rather than gains concentrated on a small subset of easy examples.

The improvements are most pronounced on the Urban subset, which is particularly challenging due to tall structures and strong parallax. In these scenes, planar homography assumptions often break down and can lead to catastrophic misregistrations, while our 3D-aware formulation remains stable. Retrieval methods perform worst overall, as they inherit the discretization and viewpoint mismatch of the gallery; notably, even strong matchers such as SP+SG can underperform simpler cross-view retrieval baselines (e.g., University-1652) in the Urban setting. RoMa is generally the overall best-performing non-3D baseline, but it still falls well short of SkyReg, underscoring the benefit of explicitly modeling scene geometry for dense cross-view geo-registration.

5.3 Qualitative Results

To provide additional intuition, Figure 3 shows three examples in which we warp the query drone image into the reference frame and overlay it (green) on the georeferenced satellite image (RGB). We compare against RoMa. In this cross-view setting, RoMa often fails to estimate a plausible homography, resulting in

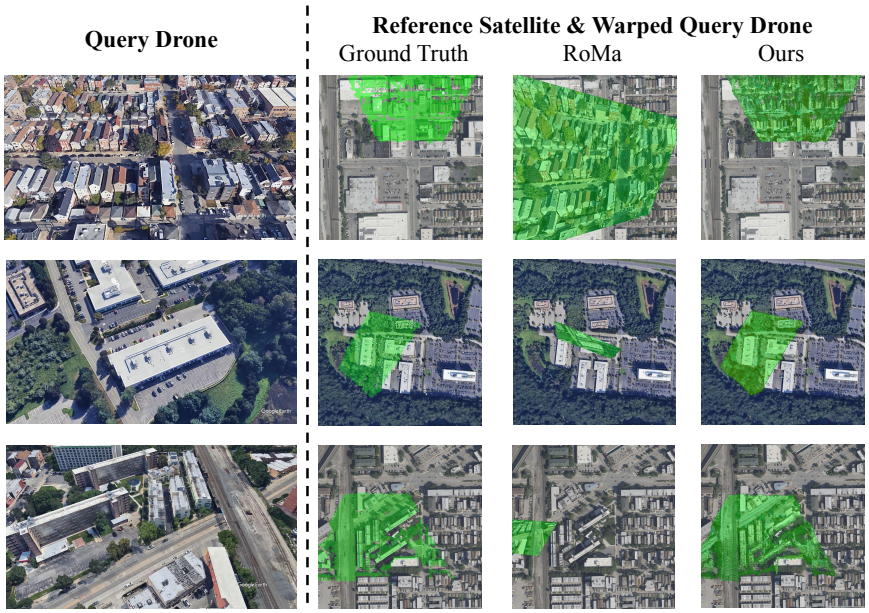


Fig. 3: Qualitative geo-registration on SkyReg-Bench. For each example, we warp the query drone image into the geodetically accurate satellite reference frame and visualize the overlay in green. From left to right: (1) query drone image, (2) ground-truth warp obtained using the ground-truth depth map and camera parameters, (3) RoMa result (homography-based), and (4) SkyReg (ours). RoMa often fails under large cross-view viewpoint/scale changes; even when successful, a single planar homography cannot model parallax, leading to stretching or shearing of elevated structures. In contrast, SkyReg's 3D-aware warping more closely matches the ground-truth alignment and naturally respects occlusions, producing sharper, geometrically consistent overlays.

large geodetic errors and visibly misaligned overlays. Even when RoMa produces a reasonable registration, the planar warp can introduce noticeable distortions—most prominently stretching or shearing of elevated structures—because a single homography cannot explain parallax induced by height variations. In contrast, our 3D-aware model yields accurate alignments under extreme scale changes, large viewpoint differences, and partial occlusions.

5.4 Training Ablation

Here we perform an additional ablation by varying the training data used to train the SkyReg model. Specifically, we evaluate three variants:

- **Landmark Subset** – SkyReg trained on Landmarks subset and 3D datasets.
- **Urban Subset** – SkyReg trained on Urban subset and 3D datasets.
- **Landmark+Urban** – SkyReg trained on combined Urban, Landmark and 3D datasets as described in Section 3.4.

Table 4: Median-GPS geo-localization performance on Urban and Suburban benchmarks. We report the mean geodetic error (GE) in meters, and the recall of images geolocated correctly under τ as a percentage.

Method	Urban					Suburban				
	GE ↓	Recall@τ (m) ↑				GE ↓	Recall@τ (m) ↑			
		20	30	40	50		20	30	40	50
Landmark Subset	48.70	59.89	69.83	72.03	73.18	27.23	73.66	85.20	89.93	91.40
Urban Subset	50.38	58.47	66.19	68.26	69.37	24.79	72.30	86.46	91.92	93.07
Landmark+Urban	**36.40**	**66.72**	**78.16**	**80.47**	**81.57**	**21.45**	**75.32**	**88.55**	**93.17**	**95.27**

All models are evaluated on the SkyReg-Bench benchmark using the same evaluation protocol for median-GPS geo-localization described in Section 5.1. These results (Table 4) highlight the importance of diverse training data for robust cross-view geo-registration. Training on a single subset reduces generalization performance, while combining Urban and Landmark scenes provides the best overall accuracy.

6 Conclusion

We present *SkyReg*, a geometry-aware approach to pixel-level drone–satellite geo-registration that leverages feed-forward 3D reconstruction to estimate the cross-view warping between a perspective drone query and a geodetically accurate satellite reference, enabling dense pixel-wise GPS assignment without relying on 2D point matches and remaining robust to occlusions and large viewpoint changes. To support training and rigorous evaluation, we introduced SkyReg-Train and SkyReg-Bench, providing dense per-pixel latitude–longitude supervision with depth and camera parameters derived from LiDAR and SfM, and testing generalization on unseen Urban and Suburban scenes. Across both dense and image-level protocols, SkyReg consistently outperforms strong retrieval and homography baselines, substantially improving geodetic error and recall (e.g., large gains over RoMa on unseen Urban/Suburban splits). These findings establish SkyReg as a reliable baseline for structure-level geodetic alignment and provide a foundation for future research on dense cross-view geo-registration.

Acknowledgments

Supported by Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DOI/IBC) contract number 140D0423C0074. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/IBC, or the U.S. Government.

References

1. Ali-Bey, A., Chaib-Draa, B., Giguere, P.: Mixvpr: Feature mixing for visual place recognition. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 2998–3007 (2023)
2. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: Netvlad: Cnn architecture for weakly supervised place recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5297–5307 (2016)
3. Baruch, G., Chen, Z., Dehghan, A., Dimry, T., Feigin, Y., Fu, P., Gebauer, T., Joffe, B., Kurz, D., Schwartz, A., et al.: Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. arXiv preprint arXiv:2111.08897 (2021)
4. Berton, G., Goletto, G., Trivigno, G., Stoken, A., Caputo, B., Masone, C.: Earthmatch: Iterative coregistration for fine-grained localization of astronaut photography. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (June 2024)
5. Berton, G., Masone, C.: Megaloc: One retrieval to place them all. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 2886–2892 (June 2025)
6. Berton, G., Trivigno, G., Caputo, B., Masone, C.: Eigenplaces: Training viewpoint robust models for visual place recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11080–11090 (2023)
7. Bredvik, A., Richardson, S., Crispell, D.: Metadata-free georegistration of ground and airborne imagery (2025), https://arxiv.org/abs/2503.04927
8. Brown, M., Chan, M., Twardowski, M.: Wriva public data (2024). https://doi.org/10.21227/cjk5-gf33, https://dx.doi.org/10.21227/cjk5-gf33
9. Clark, B., Kerrigan, A., Kulkarni, P.P., Cepeda, V.V., Shah, M.: Where we are and what we're looking at: Query based worldwide image geo-localization using hierarchies and scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23182–23190 (2023)
10. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 224–236 (2018)
11. Edstedt, J., Sun, Q., Bökman, G., Wadenbäck, M., Felsberg, M.: Roma: Robust dense feature matching. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19790–19800 (2024)
12. Hafiane, A., Palaniappan, K., Seetharaman, G.: Uav-video registration using block-based features. In: IGARSS 2008-2008 IEEE International Geoscience and Remote Sensing Symposium. vol. 2, pp. II–1104. IEEE (2008)
13. Izquierdo, S., Civera, J.: Optimal transport aggregation for visual place recognition. In: Proceedings of the ieee/cvf conference on computer vision and pattern recognition. pp. 17658–17668 (2024)
14. Keetha, N., Mishra, A., Karhade, J., Jatavallabhula, K.M., Scherer, S., Krishna, M., Garg, S.: Anyloc: Towards universal visual place recognition. IEEE Robotics and Automation Letters **9**(2), 1286–1293 (2023)
15. Klemmer, K., Rolf, E., Robinson, C., Mackey, L., Rußwurm, M.: Satclip: Global, general-purpose location embeddings with satellite imagery. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 4347–4355 (2025)
16. Kulkarni, P.P., Gupta, R., Chhipa, P.C., Shah, M.: Vidtag: Temporally aligned video to gps geolocalization with denoising sequence prediction at a global scale.

In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23977–23987 (2026)
17. Kulkarni, P.P., Nayak, G.K., Shah, M.: Cityguessr: City-level video geo-localization on a global scale. In: European Conference on Computer Vision. pp. 293–311. Springer (2024)
18. Leroy, V., Cabon, Y., Revaud, J.: Grounding image matching in 3d with mast3r (2024), https://arxiv.org/abs/2406.09756
19. Li, G., Qian, M., Xia, G.S.: Unleashing unlabeled data: A paradigm for cross-view geo-localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16719–16729 (2024)
20. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
21. Meng, Z., Xia, X., Ma, J.: Toward foundation models for inclusive object detection: Geometry- and category-aware feature extraction across road user categories. IEEE Transactions on Systems, Man, and Cybernetics: Systems **54**(11), 6570–6580 (2024). https://doi.org/10.1109/TSMC.2024.3385711
22. Muller-Budack, E., Pustu-Iren, K., Ewerth, R.: Geolocation estimation of photos using a hierarchical model and scene classification. In: Proceedings of the European conference on computer vision (ECCV). pp. 563–579 (2018)
23. Ozcanli, O.C., Dong, Y., Mundy, J.L., Webb, H., Hammoud, R., Victor, T.: Automatic geo-location correction of satellite imagery. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 307–314 (2014)
24. Pramanick, S., Nowara, E.M., Gleason, J., Castillo, C.D., Chellappa, R.: Where in the world is this image? transformer-based geo-localization in the wild. In: European Conference on Computer Vision. pp. 196–215. Springer (2022)
25. Regmi, K., Shah, M.: Bridging the domain gap for ground-to-aerial image matching. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 470–479 (2019)
26. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., Novotny, D.: Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10901–10911 (2021)
27. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4938–4947 (2020)
28. Schönberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
29. Schönberger, J.L., Zheng, E., Pollefeys, M., Frahm, J.M.: Pixelwise view selection for unstructured multi-view stereo. In: European Conference on Computer Vision (ECCV) (2016)
30. Seo, P.H., Weyand, T., Sim, J., Han, B.: Cplanet: Enhancing image geolocalization by combinatorial partitioning of maps. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 536–551 (2018)
31. Shatwell, D.G., Dave, I.R., Swetha, S., Shah, M.: Gt-loc: Unifying when and where in images through a joint embedding space. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1–11 (2025)
32. Shatwell, D.G., Swetha, S., Shah, M.: Tiger: A unified framework for time, images and geo-location retrieval. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23955–23965 (2026)

33. Sheikh, Y., Khan, S., Shah, M.: Feature-based georegistration of aerial images. GeoSensor Networks **4**, 2 (2004)
34. Sheikh, Y., Khan, S., Shah, M., Cannata, R.W.: Geodetic alignment of aerial video frames (2003), https://api.semanticscholar.org/CorpusID:6230258
35. Shi, Y., Liu, L., Yu, X., Li, H.: Spatial-aware feature aggregation for image based cross-view geo-localization. Advances in Neural Information Processing Systems **32** (2019)
36. Shi, Y., Yu, X., Campbell, D., Li, H.: Where am i looking at? joint location and orientation estimation by cross-view matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4064–4072 (2020)
37. Shugaev, M., Semenov, I., Ashley, K., Klaczynski, M., Cuntoor, N., Lee, M.W., Jacobs, N.: Arcgeo: Localizing limited field-of-view images using cross-view matching. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 209–218 (2024)
38. Sugarbaker, L.J., Constance, E.W., Heidemann, H.K., Jason, A.L., Lukas, V., Saghy, D.L., Stoker, J.M.: The 3d elevation program initiative: a call for action. Tech. rep., US Geological Survey (2014)
39. Toker, A., Zhou, Q., Maximov, M., Leal-Taixé, L.: Coming down to earth: Satellite-to-street view synthesis for geo-localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6488–6497 (2021)
40. Trivigno, G., Berton, G., Aragon, J., Caputo, B., Masone, C.: Divide&classify: Fine-grained classification for city-wide visual geo-localization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 11142–11152 (October 2023)
41. Tzachor, I., Lerner, B., Levy, M., Green, M., Shalev, T.B., Habib, G., Samuel, D., Zailer, N.K., Shimshi, O., Darshan, N., et al.: Effovpr: Effective foundation model utilization for visual place recognition. arXiv preprint arXiv:2405.18065 (2024)
42. Vivanco Cepeda, V., Nayak, G.K., Shah, M.: Geoclip: Clip-inspired alignment between locations and images for effective worldwide geo-localization. Advances in Neural Information Processing Systems **36**, 8690–8701 (2023)
43. Vo, N., Jacobs, N., Hays, J.: Revisiting im2gps in the deep learning era. In: Proceedings of the IEEE international conference on computer vision. pp. 2621–2630 (2017)
44. Vuong, K., Ghosh, A., Ramanan, D., Narasimhan, S., Tulsiani, S.: Aerialmegadepth: Learning aerial-ground reconstruction and view synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21674–21684 (2025)
45. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer (2025), https://arxiv.org/abs/2503.11651
46. Wang, S., Nguyen, C., Liu, J., Zhang, Y., Muthu, S., Maken, F.A., Zhang, K., Li, H.: View from above: Orthogonal-view aware cross-view localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14843–14852 (2024)
47. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy (2024), https://arxiv.org/abs/2312.14132
48. Weyand, T., Kostrikov, I., Philbin, J.: Planet-photo geolocation with convolutional neural networks. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14. pp. 37–55. Springer (2016)

49. Xia, Z., Alahi, A.: Fg^2: Fine-grained cross-view localization by fine-grained feature matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6362–6372 (June 2025)
50. Xia, Z., Booij, O., Kooij, J.F.P.: Convolutional cross-view pose estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence **46**(5), 3813–3831 (2024). https://doi.org/10.1109/TPAMI.2023.3346924
51. Xia, Z., Booij, O., Manfredi, M., Kooij, J.F.P.: Visual cross-view metric localization with dense uncertainty estimates. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. pp. 90–106. Springer Nature Switzerland, Cham (2022)
52. Xu, S., Zhang, C., Fan, L., Meng, G., Xiang, S., Ye, J.: Addressclip: Empowering vision-language models for city-wide image address localization. In: European Conference on Computer Vision (ECCV) (2024)
53. Xu, S., Zhang, C., Fan, L., Zhou, Y., Fan, B., Xiang, S., Meng, G., Ye, J.: Addressvlm: Cross-view alignment tuning for image address localization using large vision-language models (2025), https://arxiv.org/abs/2508.10667
54. Ye, J., Lin, H., Ou, L., Chen, D., Wang, Z., Zhu, Q., He, C., Li, W.: Where am i? cross-view geo-localization with natural language descriptions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5890–5900 (2025)
55. Ye, J., Lv, Z., Li, W., Yu, J., Yang, H., Zhong, H., He, C.: Cross-view image geo-localization with panorama-bev co-retrieval network. In: European Conference on Computer Vision. pp. 74–90. Springer (2024)
56. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12–22 (2023)
57. Yuan, Y., Huang, W., Wang, X., Xu, H., Zuo, H., Su, R.: Automated accurate registration method between uav image and google satellite map. Multimedia Tools and Applications **79**(23), 16573–16591 (2020)
58. Zheng, Z., Wei, Y., Yang, Y.: University-1652: A multi-view multi-source benchmark for drone-based geo-localization. In: Proceedings of the 28th ACM international conference on Multimedia. pp. 1395–1403 (2020)
59. Zhu, S., Shah, M., Chen, C.: Transgeo: Transformer is all you need for cross-view image geo-localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1162–1171 (2022)
60. Zhu, S., Yang, T., Chen, C.: Vigor: Cross-view image geo-localization beyond one-to-one retrieval. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3640–3649 (2021)

Pixel-wise Geo-registration of Drone Images (Supplementary Material)

Qingyang Liu* David G. Shatwell* Parth Parag Kulkarni Mubarak Shah

Institute of Artificial Intelligence, University of Central Florida

In this supplementary material, we provide additional details on our method and dataset. We organize our supplementary material into the following sections:

A. Additional Data Gathering Details

A.1. Urban Scenes

A.2. Landmarks and Suburban Scenes

B. Additional Dataset Information

B.1. SkyReg Urban

B.2. SkyReg Landmark

B.3. SkyReg Suburban

C. Additional Qualitative Results

A Additional Data Gathering Details

A.1 Urban Scenes

Ground Truth Pixel-wise GPS For the Urban subset of SkyReg, the ground-truth geodetic data for each reference tile is obtained from USGS 3DEP dataset with the official TNM Download V2 tool in the form of LiDAR files, each containing approximately 18 million 3D points in each tile's native projected coordinate system. The planar coordinates (x, y) are transformed into GPS coordinates using a coordinate reference system conversion to WGS–84, yielding latitude and longitude values. The elevation coordinate z, after unit conversion when necessary, is used as the altitude. This process produces a georeferenced point map in which each location is represented by its corresponding $(\text{lat}, \text{lon}, \text{alt})$ values, which serve as the ground truth for the satellite reference tiles.

* These authors contributed equally to this work.

Reference Satellite Images To obtain the corresponding RGB reference satellite images, we use the GPS coordinates of the four corners of each LiDAR tile to retrieve satellite imagery from Bing Maps. We choose Bing Maps because its satellite images are orthorectified, which allows us to use the LiDAR point clouds to project and estimate depth at each pixel without requiring the ground-truth satellite pose. The images are collected at zoom level 17 for Chicago and 16 for Seattle and San Francisco since the latter two encompass a significantly larger area, providing high spatial resolution. Since the Bing Maps imagery is orthorectified and the GPS coordinates of the four tile corners are known, we can interpolate the geographic coordinates of all remaining pixels, thereby obtaining a complete georeferencing of the reference satellite images.

Query Drone Images For the queries, we define synthetic RGB images along with camera intrinsics and poses expressed in the ECEF coordinate system. To compute ground-truth geodetic coordinates for each valid pixel, we use LiDAR points from the corresponding reference tile and its neighboring tiles, transform them to ECEF coordinates, and project them into the query camera frame. The camera intrinsics are then used to determine the correspondence between projected LiDAR points and query pixels, so that each matched pixel can be assigned an ECEF coordinate and subsequently converted to geodetic coordinates.

Since the query images are captured much closer to the ground, they exhibit higher spatial resolution than the reference data. Consequently, this direct projection process yields only a sparse ground-truth point map: large portions of the image remain unlabeled, and some pixel assignments may be incorrect due to occlusions or projections onto geometry behind foreground structures. To overcome this limitation, we construct a surface mesh using 2D Delaunay triangulation and then perform ray tracing over the mesh to obtain a dense depth map in metric scale (Eq. 10 in main paper). Using the known camera pose and intrinsics, we convert this depth map into a dense point map in ECEF coordinates, which is then transformed into geodetic coordinates. This procedure significantly densifies the ground-truth map and allows nearly every pixel in the query image to be assigned an accurate geodetic coordinate. To keep mesh triangulation computationally tractable, we sample 300k points for mesh construction and enforce a maximum mesh edge length of 20 meters to prevent the mesh from spanning large empty regions.

A.2 Landmarks and Suburban Scenes

For landmarks and suburban scenes, we first define a set of camera poses and intrinsics over a predefined area, following the procedure described in Section 4.2 of the main paper. Once the images, camera poses, and intrinsics are available, we keep the camera parameters fixed and run triangulation to recover the 3D scene structure. In this process, pairs of spatially nearby images are selected, local features are extracted and matched across those pairs, and the resulting correspondences are used together with the known camera geometry to estimate

3D points in the scene. In other words, triangulation computes the 3D location of scene points by intersecting the viewing rays associated with matching image observations from multiple cameras. Since the camera poses and intrinsics are fixed, this step focuses on reconstructing the scene geometry rather than estimating the camera parameters themselves. The reconstructed 3D points can then be projected back into each image to obtain per-image scene geometry, and the final output is a set of metric depth maps for each image in the scene.

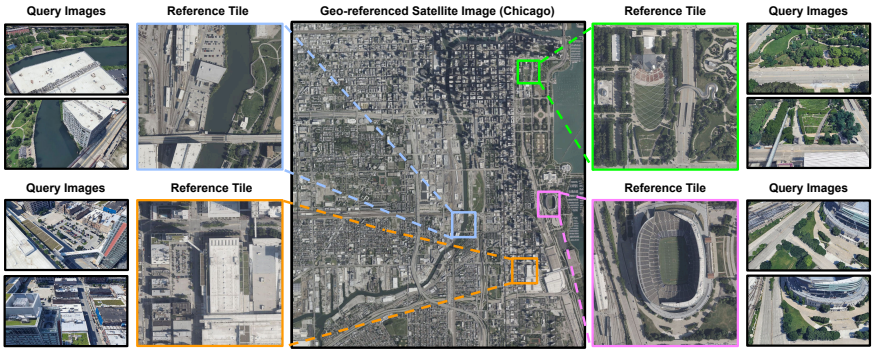


Fig. 1: Representative samples from SkyReg Urban Chicago. The center panel shows a large orthorectified satellite image of Chicago. We create reference satellite images from local reference tiles. The left/right panels show example drone queries associated with each reference tile.

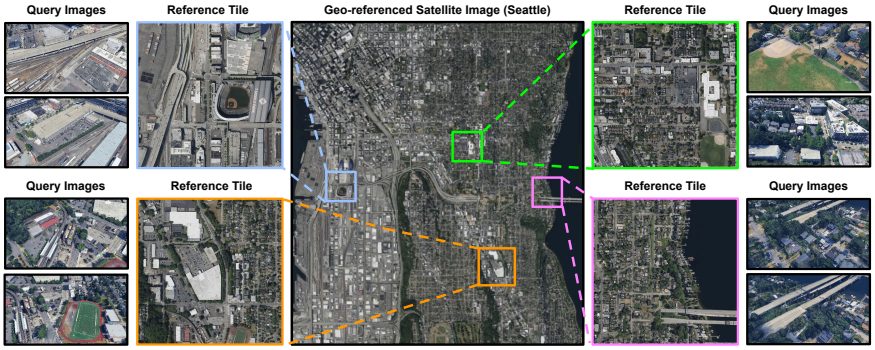


Fig. 2: Representative samples from SkyReg Urban Seattle. The center panel shows a large orthorectified satellite image of Seattle. We create reference satellite images from local reference tiles. The left/right panels show example drone queries associated with each reference tile.

B Additional Dataset Information

We provide additional statistics describing the distribution of the dataset introduced in Section 4 of the main paper.

B.1 SkyReg Urban

SkyReg Urban consists of three cities– Seattle, San Francisco, and Chicago (distribution shown in Table 1). Combined, the three cities consist of 41,027 drone images and 390 satellite images. Figures 1 and 2 show sample reference and drone images for the cities of Chicago and Seattle respectively.

Table 1: Number of drone and satellite images for each city in SkyReg Urban subset.

City	# Drone	# Sat.
Chicago	13,191	238
San Francisco	14,346	71
Seattle	13,490	81

B.2 SkyReg Landmark

SkyReg Landmark consists of 83 scenes, each centered around landmarks located around the world. We show a histogram of the number of drone images per scene in Figure 3. Figure 4 shows representative samples from SkyReg Landmark.

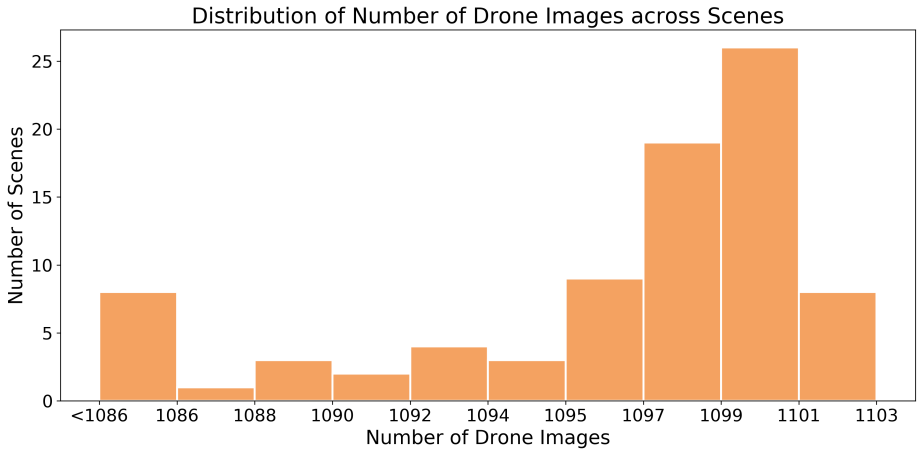


Fig. 3: Histogram showing the distribution of number of drone images across scenes for SkyReg Landmark Most scenes contain between approximately 1.0k and 1.1k drone images. This indicates that the dataset provides dense image coverage for majority of scenes. Freedman–Diaconis rule was used to compute the number of bins.

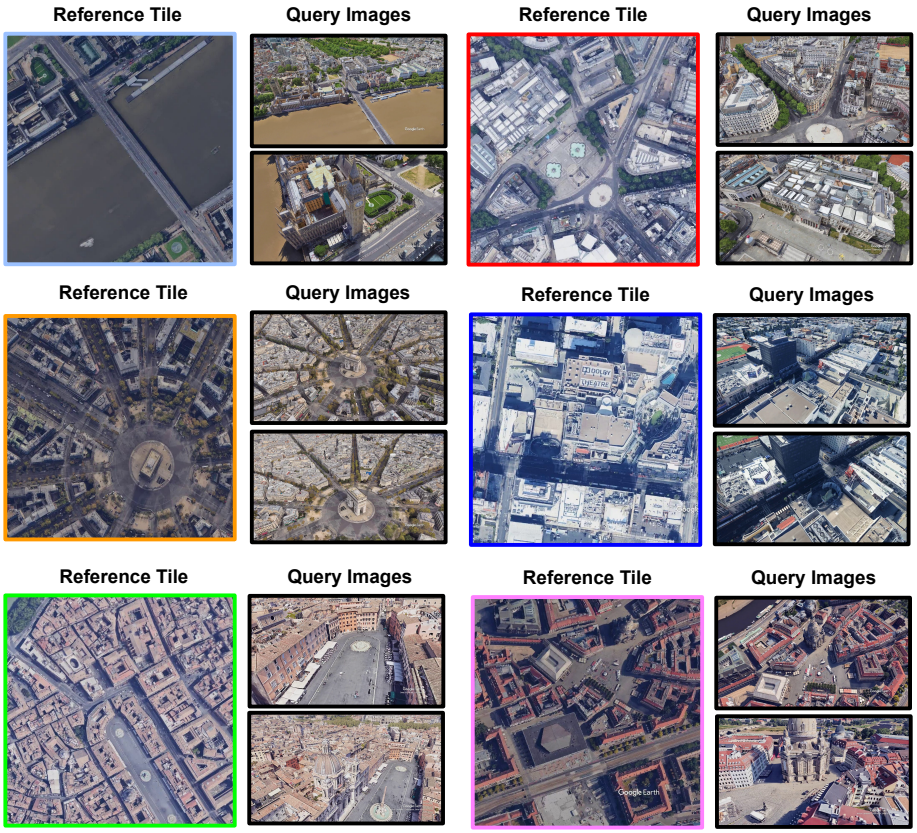


Fig. 4: Representative samples from SkyReg Landmark Each pair consists of a drone query image and the reference tile consists of a satellite image centered around a significant landmark.

B.3 SkyReg Suburban

The Suburban split of SkyReg-Bench comprises five scenes across three U.S. states: one scene from Baltimore, MD; one from Beverly, MA; and three from Pittsburgh, PA. Each scene contains between 100 and 250 drone query images, each associated with a georeferenced satellite image and corresponding ground-truth GPS coordinates. The exact data distribution is shown in Table 2. Figure 5 shows sample reference images for each of the suburban scenes.

Table 2: SkyReg Suburban number of drone images for each scene.

Scene	# Images
Baltimore	100
Beverly	100
Pittsburgh-1	250
Pittsburgh-2	250
Pittsburgh-3	250

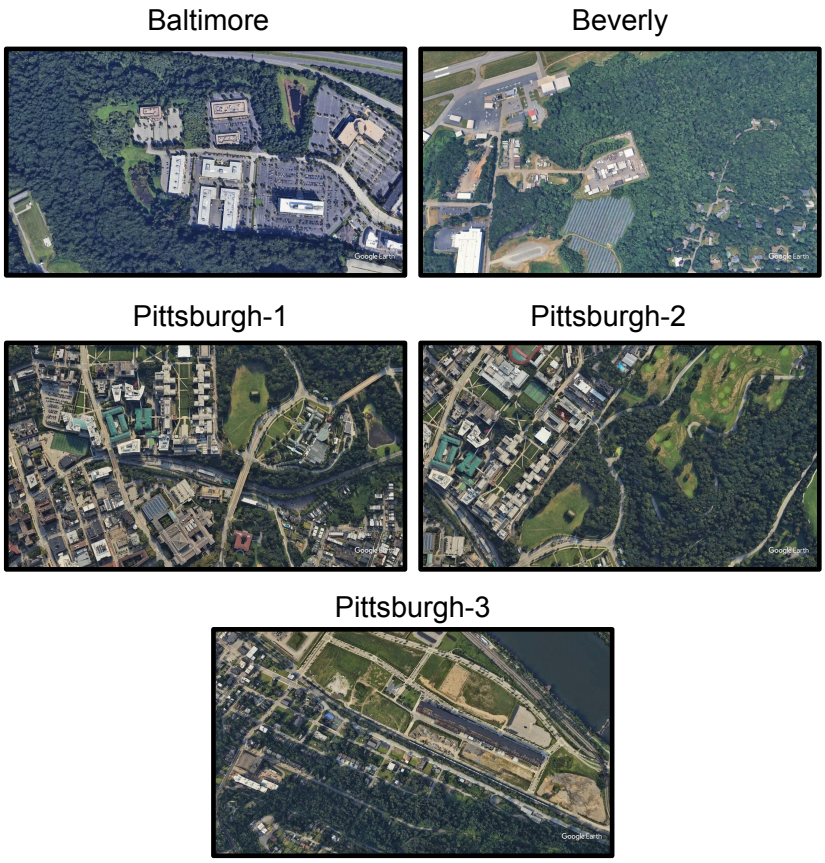


Fig. 5: Satellite Reference Tile from SkyReg Suburban SkyReg Suburban consists of 5 scenes, each one consisting of one satellite image encompassing the entire scene.

C Additional Qualitative Results

Figures 6 and 7 provide additional qualitative results on the SkyReg-Bench dataset. Similar to Figure 3 in the main paper, we visualize the warped query

image overlaid on the georeferenced satellite reference image. The overlays are shown in green to highlight the alignment.

These examples further demonstrate that the proposed geometry-aware registration produces accurate alignments even under large viewpoint differences and significant scene depth variation.

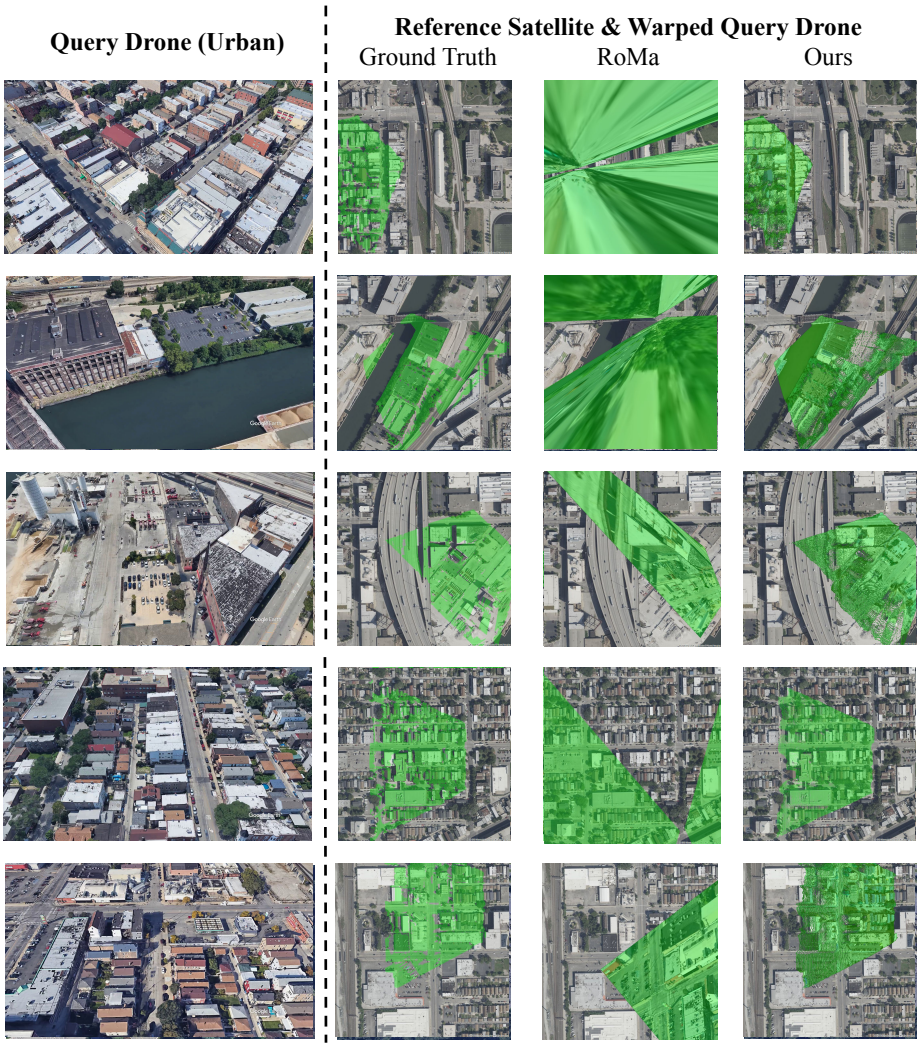


Fig. 6: Additional qualitative examples of geo-registration on the Urban subset of SkyReg-Bench. From left to right: (1) query drone image, (2) ground-truth warp, (3) RoMa result, and (4) SkyReg (ours). The warped query image is visualized in green on top of the georeferenced satellite reference.



Fig. 7: Additional qualitative examples of geo-registration on the Suburban subset of SkyReg-Bench. From left to right: (1) query drone image, (2) ground-truth warp, (3) RoMa result, and (4) SkyReg (ours). The warped query image is visualized in green on top of the georeferenced satellite reference.