Geo-Localization of Sequential Aerial Data

Project overview

This project focuses on the visual geo-localization task using sequential aerial imagery. The main objective is to determine the precise location of a series of small images captured by an Unmanned Aerial Vehicle (UAV) within a larger, geo-referenced satellite image.

Given:

- A sequence of small UAV-captured images,
- One large satellite image with known geographic coordinates,

Goal: Match each UAV image to its corresponding location on the satellite map.

Data

Our dataset contains 6,742 drone images and 11 satellite maps. We took it from this paper. These are the examples of images which can be found in this dataset.

This is an example of the satellite image, these images were of different shapes, but on average they are of shape 25000 by 10000.
Also here is an example of picture from UAV:

These pictures are usually of shape 3000 by 2000.

Literature review

We reviewed several existing approaches, but most focused on a different objective (photo geolocation) whereas our goal is the geo-localization of sequential aerial data. As a result, we only adopted a few common applicable techniques from those methods, such as dividing large satellite images into smaller cropped sections.

PlaNet

The PlaNet paper presents a deep learning approach to photo geolocation by framing the problem as image classification rather than image retrieval.

- The Earth is divided into thousands of multi-scale geographic cells.
- A CNN is trained on millions of geotagged images to predict which cell an image belongs to.
- Unlike traditional methods that rely only on matching landmarks or global image features, PlaNet learns from diverse visual cues (architecture, climate).
- The model is extended to handle photo sequences by incorporating an LSTM.

Wide-Area Image Geolocalization

This research paper addresses the challenge of cross-view image geolocalization, which involves determining the location of a ground-level image by matching it with aerial (satellite) imagery.

- Uses CNNs to learn a shared feature space between ground-level and aerial images.
- Introduces a cross-view training approach to align semantic features across different viewpoints.
- Builds a large dataset of paired aerial and ground images across the U.S. to support training and evaluation.

PIGEON

The PIGEON paper introduces a geolocalization system that predicts the geographic location of an image using the following techniques:

- Semantic geocells
  Instead of using fixed geographic grids, the authors create geocells based on administrative boundaries and metadata, then apply Voronoi tessellation and clustering for semantic grouping.

- Haversine smoothing
  Continuous labels are generated for training by smoothing geocell labels based on haversine distance.

- Multi-task CLIP pre-training with synthetic geographic captions

- Two-stage Inference
    1. Initial prediction → The image embedding is passed through a linear layer to predict top-K geocells.
    2. Refinement → The image embedding is compared to location cluster representations using L2-distance to refine the prediction within the top geocells.

Methods overview

Brute force

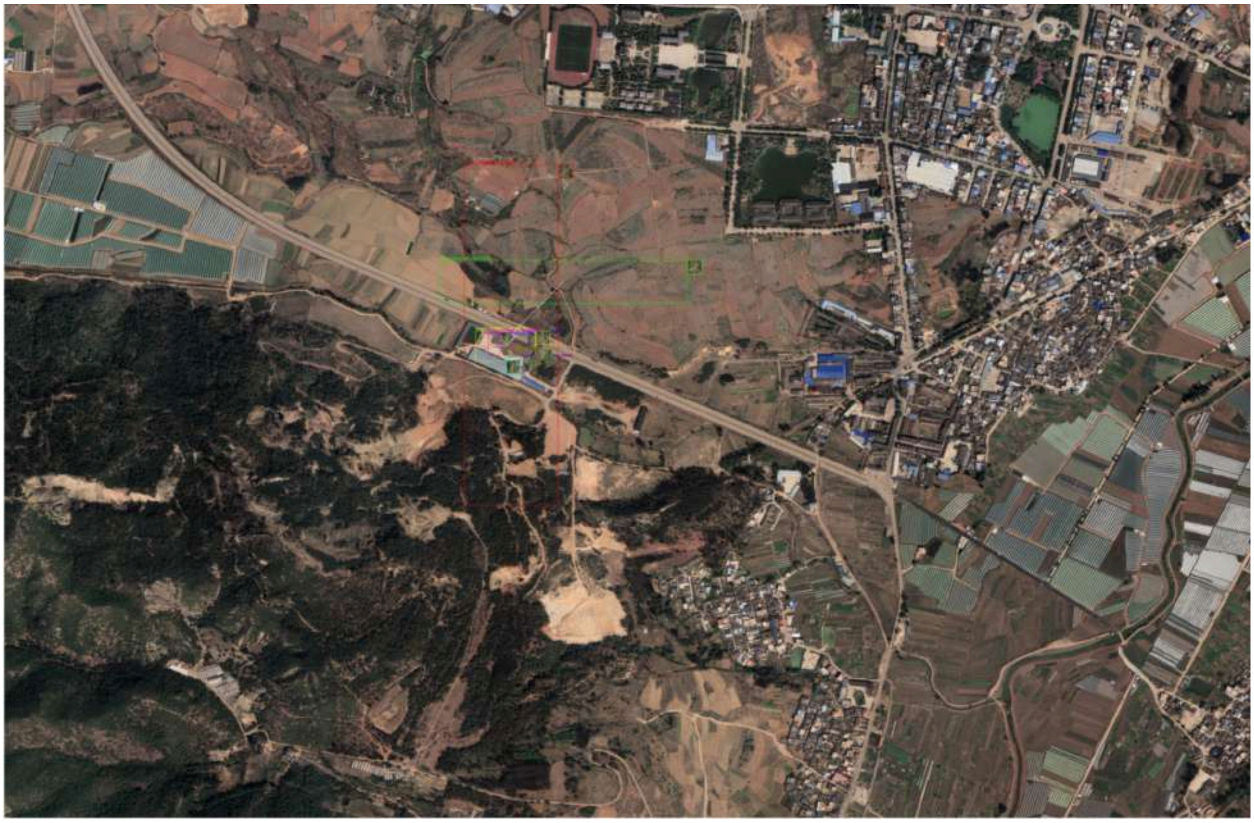
So, as the starting point we wanted to do the straightforward template matching: just simply take a small drone picture and do the template matching with a satellite one. The problem here was that firstly, drone images which are 3k x 2k on the satellite image may take only 100 by 100 pixels or even less. Also, these images might be taken from another perspective. So we tried to resize the image to different shapes, like 100 x 100, 200 x 200, 500 x 500, and also rotated them from 0 to 360 degrees with step 15 degrees. This was computationally expensive, but we had to try this. Of course, it was not a success at all. So we had to move forward.

Something more advanced

After we understood that brute force is a disaster in both computational complexity and results, we decided to do **something more advanced.** At the beginning we had 3 main ideas: try some CNN architecture for features extraction, try a ViT for feature extraction and matching, try some model as SIFT for feature detection and then use FLANN for matching. We actually implemented all 3 of them.

CNN

We tried to make some feature extraction with a resnet50 architecture, and then, using FLANN matcher, match it with some area on satellite image.
Here is a result of what we got:



The real way of the UAV should have been from the left top corner and along the round to the center. But we can see that it started somewhere in the center and finished also there. Actually, we then understood that it didn’t actually have a lot of sense, as the resnet was trained on ImageNet, and is not suitable for such tasks.

ViT

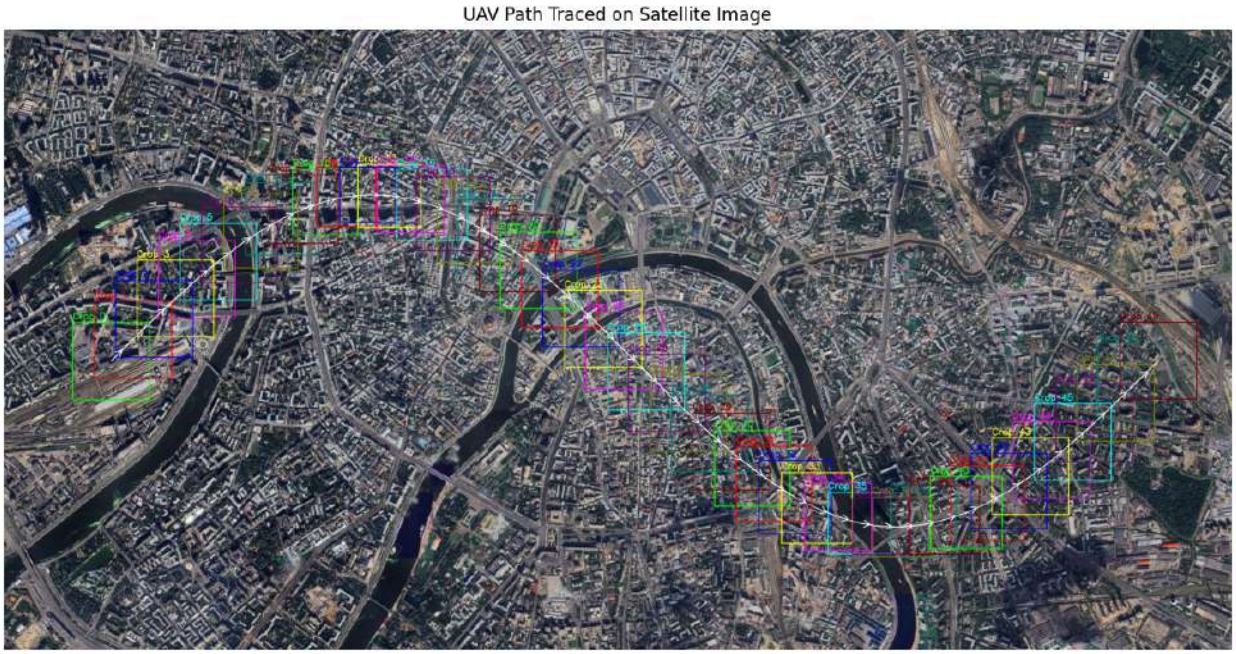
Actually this approach was the most promising for us, as visual ViTs, unlike traditional CNN-based models that primarily focus on local spatial features, are capable of capturing global contextual information through self-attention mechanisms. But, very unfortunately, ViT returned results which were a disaster and for a sequence of 9 images it returned the same 9 coordinates.

SIFT + FLANN

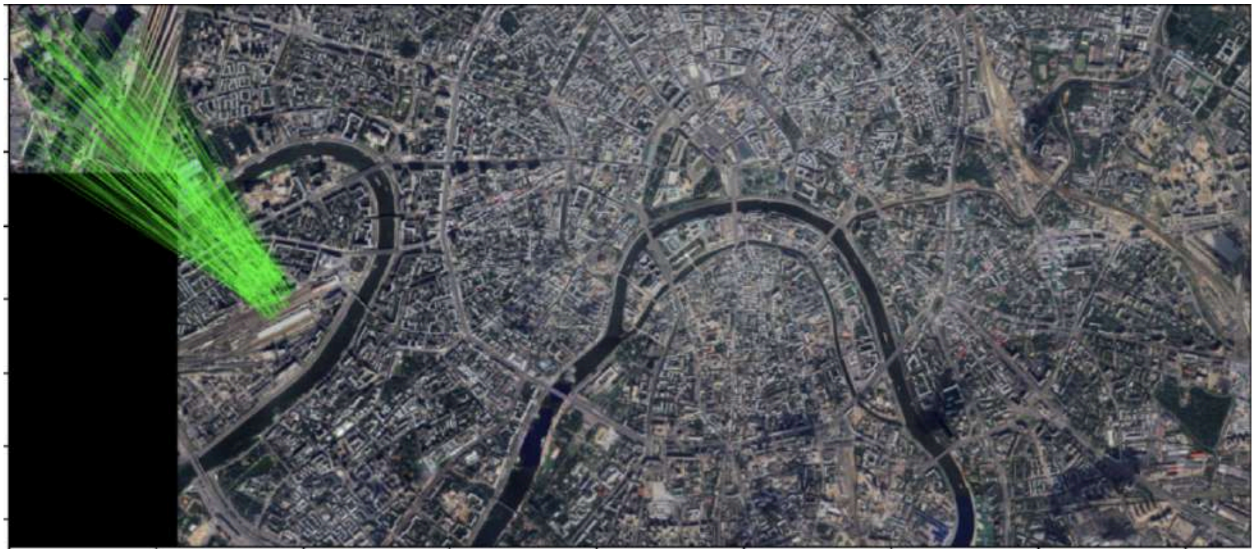
The idea behind this approach is pretty simple: for the initial UAV image, we extract keypoints and descriptors using SIFT from both the UAV crop and the entire satellite image. We then use FLANN-based matching to identify correspondences and estimate the Homography matrix using RANSAC, which allows for robust geometric alignment and visualization.

Since the UAV captures images in sequence, we assume the next image is close to the last one in location. Based on the location detected in the satellite image for the first UAV crop, we define a Region of Interest (ROI) — a square window centered at the previous matched location, extended ±100 pixels in all directions (up, down, left, and right). This significantly reduces the satellite search space for future crops, improving both efficiency and accuracy.

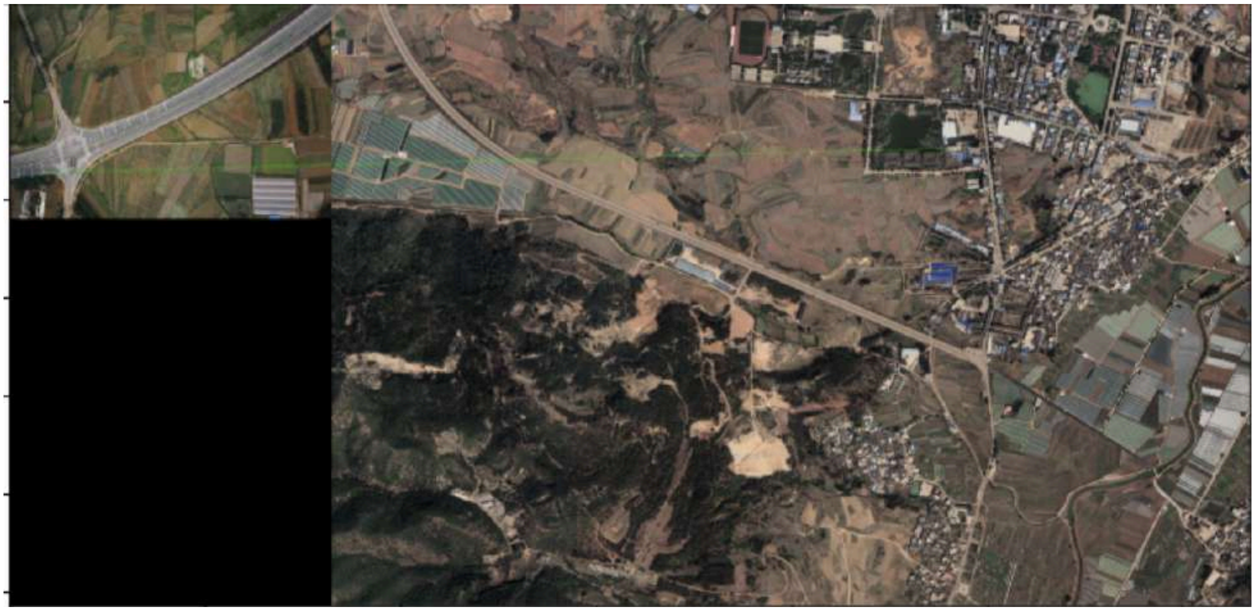
This approach performed pretty well on some smaller satellite images, here is an example:


UAV Path Traced on Satellite Image

So we chose this approach as our baseline and decided to move forward and do some experiments on it to make it work on the larger images. What was the basic problem we faced? Well, it is easier to demonstrate on pictures.



This is an example of keypoints detected and matched on a smaller satellite map.

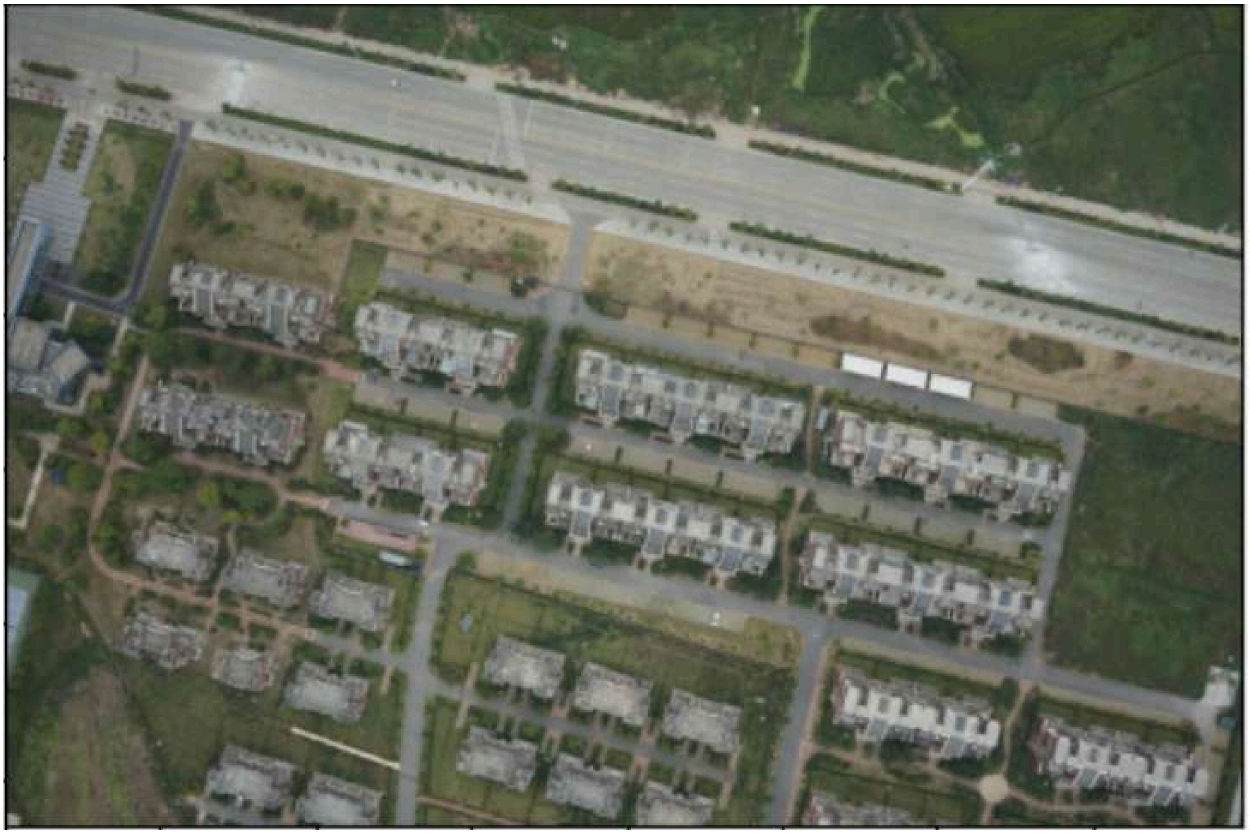


And this is the number of matched keypoints on a larger satellite image. We can clearly see that this is very bad. We tried different thresholds,from 0.6 to 0.3, but still SIFT couldn't detect enough keypoints for good matching.

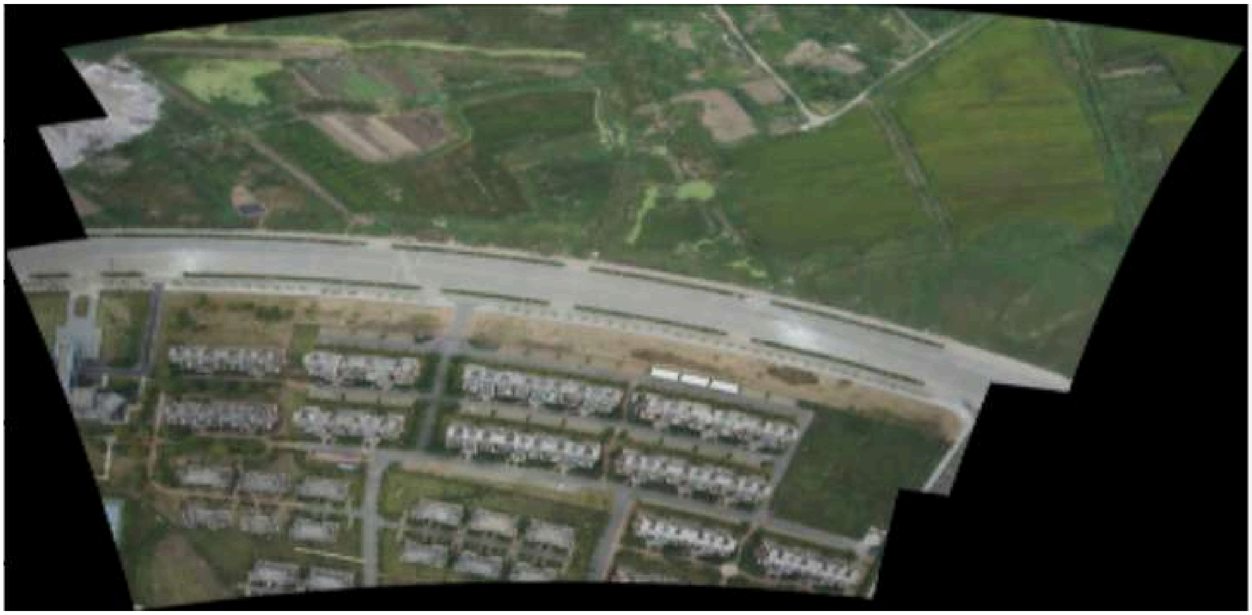
How did we tri to fix this?

So first what we tried was just simply change the approach by extracting keypoints with ALIKED and match them with LightGlue. This, unfortunately, didn't fix the problem. So we moved forward. Next we decided to play with some resizing. We were taking the 500 x 500 blocks from the satellite image, resizing them to 1000 x 1000, and then, using the idea from this paper. So we basically tried to match the original 3000 x 2000 UAV image with a 1000 x 1000 block, also UAV image reshaped to ½ from the original and reshaped to ¼ from the original. Now our task was to correctly classify the block. But the method we implemented also couldn’t fix the initial problem.

The last thing that we tried was actually pretty interesting and very promising. One of the main tasks of this project was to correctly identify the first crop, and then based on this it would be easier to determine the next crops. What we thought was that maybe the first image alone did not contain enough key points for reliable matching. However, if there was a way to combine the first three images into a single panoramic image, we could then detect keypoints on this larger field of view. This approach could potentially increase the number and diversity of keypoints, leading to more robust matches with the satellite image.
Hopefully, we know such way. This is possible again with RANSAC. By estimating the transformations between consecutive UAV images using feature matching and RANSAC-based homography estimation, we can align and stitch the images into a single composite view.
This is how we did this:



Here is how the first image looked like. And this is the panoramic image we got:



Results

All in all, we experimented with various methods, but none significantly outperformed our basic pipeline based on SIFT and FLANN. For evaluation, we used the haversine distance, and on large satellite images, our approach achieved an average error of around 0.7 km. On smaller satellite images, the localization was nearly perfect, if not entirely accurate.

If we had more time and resources, the next steps we would consider include:

- Incorporating deep learning-based feature extractors, such as SuperPoint or D2-Net, to obtain more robust and invariant descriptors.
- Fine-tuning feature matching strategies, perhaps by integrating geometric verification beyond RANSAC or combining global and local features.

Resources

- UAV-VisLoc Data https://arxiv.org/abs/2405.11936
- PlaNet https://arxiv.org/abs/1602.05314
- PIGEON https://arxiv.org/abs/2307.05845
- Wide-Area Image Geolocalization https://ieeexplore.ieee.org/document/7410808
- Feature Pyramid Networks for Object Detection https://arxiv.org/abs/1612.03144