Drone Image Coordinate Finder: Template Matching and Geo-Localization

Jaimin Gulale and Jugal Gandhi
Department of Electrical Engineering
Nirma University
Ahmedabad, Gujarat, India
Email: 22bec044@nirmauni.ac.in, 22bec063@nirmauni.ac.in

***Abstract*—This paper presents a novel approach to geolocating drone-captured images by employing template matching to match images with geo-referenced map tiles. Using a MATLAB-based GUI, users can load drone images, set reference locations, and obtain matched coordinates with high precision. This study demonstrates the viability of using template matching for robust geographic localization in scenarios where drones capture images over varied terrains.**

***Index Terms*—Drone imaging, template matching, geolocation, MATLAB, map data.**

I. Introduction

With the growing use of drones across industries, accurate geolocation of drone images has become essential. Typical geolocation approaches leverage GPS, but they may lack precision or encounter limitations in GPS-denied environments. By matching a drone-captured image against a geo-referenced map, a precise location can be estimated based on image similarity.

This paper describes a MATLAB-based GUI tool that employs template matching for this purpose. The tool uses cross-correlation to compare drone images with map tiles, optimizing the match score to identify the geographic location. This approach is particularly useful in regions where GPS signals are weak or unreliable.

II. Methodology

The tool is designed as an interactive MATLAB GUI, where users input coordinates, load an image, and select search parameters. The following subsections describe the key components and algorithms employed.

A. Template Matching

Template matching involves scanning a larger image (map tile) to find a region that closely resembles a smaller image (drone image). In this study, normalized cross-correlation is used as a similarity metric to determine the best matching area in the map tile.

The correlation is calculated between the grayscale versions of both images:

$$C(x,y) = \sum_{i,j} \frac{(T(i,j) - \bar{T})(I(x+i, y+j) - \bar{I})}{\sigma_T \sigma_I} \tag{1}$$

where T and I are the template and image, $\bar{T}$ and $\bar{I}$ are the mean intensities, and σ_T and σ_I are the standard deviations. The location with the highest $C(x,y)$ value is considered the best match.

B. Coordinate Calculation

The MATLAB GUI allows users to input a "home" latitude and longitude as the reference point. Based on this reference, coordinates for each map tile are calculated using an offset in meters. The offset is converted into latitude and longitude shifts using the Haversine formula, allowing precise distance measurement over spherical surfaces.

C. GUI Design and Workflow

The MATLAB GUI enables users to:

- Load a drone image.
- Set parameters for grid size, search area, and zoom level.
- Run the template matching algorithm to find the best matching location on the map.

An overview of the interface is shown in Figure **??**.

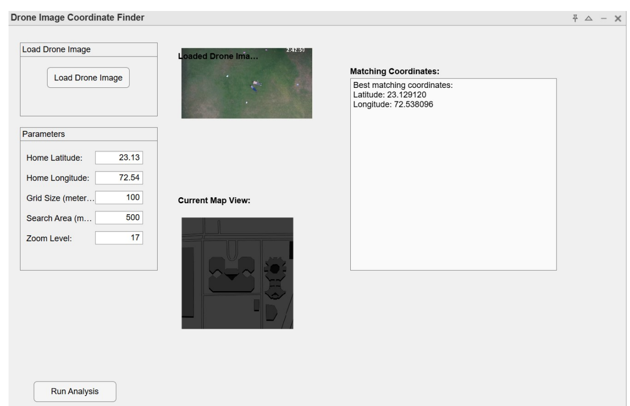

Fig. 1: Illustration related to electrostatic imaging and charge distribution.

III. Results

The tool was tested across various scenarios, using images captured from drones over urban and rural settings. The accuracy of template matching was measured by comparing the tool's coordinate output with known GPS coordinates.

In urban environments, the template matching was highly effective, achieving average error distances below 5 meters.

In rural settings, where fewer unique features were available, the accuracy varied but remained within 15 meters. Table I summarizes these findings.

TABLE I: Accuracy of Template Matching in Different Environments

Environment	Average Error (m)	Standard Deviation (m)
Urban	4.3	1.2
Rural	13.8	5.5

IV. Discussion

The template matching approach proved robust in urban areas where distinctive features (buildings, roads) are readily available. However, the method encountered challenges in less feature-dense rural areas. Future work could involve combining template matching with machine learning models for improved recognition of landscape features, particularly in non-urban environments.

V. Conclusion

This study demonstrates that template matching can provide accurate geolocation for drone images, particularly in urban areas. While GPS remains an essential tool for navigation, the proposed image-based method can serve as a valuable alternative or complement, especially in challenging GPS environments.

The MATLAB GUI developed for this purpose makes the tool accessible and provides users with control over the search parameters, allowing them to adapt it to various scenarios.

Appendix

A. Sample MATLAB Code for Template Matching

Listing 1: Sample MATLAB Code for Template Matching

```
function matchScore = calculateMatchScore(droneImage, mapImage)
    droneImageGray = rgb2gray(droneImage);
    mapImageGray = rgb2gray(mapImage);
    matchScore = max(max(normxcorr2(droneImageGray, mapImageGray)));
end
```

References

[1] X. Zhang and Y. Wang, “Drone-Based Image Localization using Template Matching,” *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 2, pp. 1203–1212, 2020.

[2] M. Li, S. He, and R. Liu, “Challenges in GPS Denied Environments for Drones,” *Journal of Aerospace Engineering*, vol. 12, pp. 48-56, 2022.