



UNIVERSITEIT•STELLENBOSCH•UNIVERSITY
jou kennisvennoot • your knowledge partner

Synthesis and Evaluation of an Image-Based Redundancy System for UAV Navigation

Sameer Shaboodien

25002783

Report submitted in partial fulfilment of the requirements of the module Project (E) 448 for the degree Baccalaureus in Engineering in the Department of Electrical and Electronic Engineering at Stellenbosch University.

Supervisor: Dr R. P. Theart

October 2024



UNIVERSITEIT•STELLENBOSCH•UNIVERSITY
jou kennisvennoot • your knowledge partner

Plagiaatverklaring / Plagiarism Declaration

1. Plagiaat is die oorneem en gebruik van die idees, materiaal en ander intellektuele eiendom van ander persone asof dit jou eie werk is.

Plagiarism is the use of ideas, material and other intellectual property of another's work and to present it as my own.

2. Ek erken dat die pleeg van plagiaat 'n strafbare oortreding is aangesien dit 'n vorm van diefstal is.

I agree that plagiarism is a punishable offence because it constitutes theft.

3. Ek verstaan ook dat direkte vertalings plagiaat is.

I also understand that direct translations are plagiarism.

4. Dienooreenkomsdig is alle aanhalings en bydraes vanuit enige bron (ingesluit die internet) volledig verwys (erken). Ek erken dat die woordelikse aanhaal van teks sonder aanhalingstekens (selfs al word die bron volledig erken) plagiaat is.

Accordingly all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism

5. Ek verklaar dat die werk in hierdie skryfstuk vervat, behalwe waar anders aangedui, my eie oorspronklike werk is en dat ek dit nie vantevore in die geheel of gedeeltelik ingehandig het vir bepunting in hierdie module/werkstuk of 'n ander module/werkstuk nie.

I declare that the work contained in this assignment, except where otherwise stated, is my original work and that I have not previously (in its entirety or in part) submitted it for grading in this module/assignment or another module/assignment.

Studentenommer / 25002783	Handtekening / SamS
Voorletters en van / S Shaboodien	Datum / 27/10/24

Abstract

Unmanned Aerial Vehicles (UAVs) rely on the Global Positioning System (GPS) for precise navigation, but vulnerabilities such as jamming and spoofing can jeopardize mission success. This project developed an image-based GPS localization system to enable reliable UAV navigation in GPS-denied environments. The system optimizes the localization pipeline by integrating advanced feature extraction, matching, and planar transformation techniques, achieving radial localization errors below 10% of the UAV's displacement from a reference image. Designed for real-time operation, the system responds within two seconds after GPS signal loss, allowing pilots to maintain effective control. Comprehensive testing across diverse environments demonstrated the system's high accuracy, robustness, and generalizability without requiring environment-specific tuning. The system excelled in low-light and low-overlap conditions, ensuring dependable navigation data delivery in various scenarios. This image-based localization provides a practical and reliable alternative to GPS, enhancing UAV operational safety and effectiveness in both military and civilian applications. With further enhancements, the system is poised for real-world deployment, promising safer and more reliable UAV missions.

Contents

Declaration	i
Abstract	ii
List of Figures	vi
List of Tables	viii
Nomenclature	ix
1. Introduction	1
1.1. Background	1
1.2. Problem Statement	2
1.3. Aims and Objectives	2
1.3.1. Aim	2
1.3.2. Objectives	2
1.4. Scope	3
1.5. Summary of Work	4
1.6. Structure of the Report	4
2. Literature Review	6
2.1. Global Navigation Satellite Systems (GNSS) Vulnerabilities	6
2.2. Alternative Navigation Approaches	7
2.2.1. Quantum Sensing Navigation	7
2.2.2. Odometry-Based Solutions	7
2.2.3. RF Communication and Signals-Based Navigation	7
2.2.4. LIDAR and Radar for Ground Mapping	8
2.2.5. Image-Based Homography Navigation	8
2.3. Background	9
2.3.1. Feature Detectors	9
2.3.2. Feature Matching	11
2.3.3. Image Similarity Computation	12
2.3.4. Planar Transformation Estimators	14
2.3.5. Optimization Techniques	15

3. System Design	18
3.1. System Pipeline	18
3.1.1. Pipeline Stages	18
3.2. Dynamic Methods and Techniques	21
3.3. Rotational Normalization Strategies	22
3.4. Testing Shortlist	23
3.5. Conclusion	24
4. Comparison of Methods	25
4.1. Testing Setup	25
4.1.1. Datasets	25
4.1.2. Testing Structure	27
4.1.3. Parameters	27
4.2. Feature Detectors	27
4.2.1. Accuracy and Runtime	28
4.2.2. Final Selection of Feature Detectors	28
4.3. Local Feature Matchers	29
4.3.1. Accuracy and Runtime Evaluation	29
4.3.2. Robustness Under Varying Detector Thresholds	29
4.3.3. Final Selection of Local Feature Matcher	30
4.4. Planar Transform Estimators	30
4.4.1. Accuracy and Runtime Evaluation	30
4.4.2. Robustness Testing	31
4.4.3. Final Selection of Transformational Estimator	31
4.5. Image Similarity Estimators	32
4.5.1. Accuracy Evaluation	32
4.5.2. Accuracy and Runtime Evaluation	32
4.5.3. Robustness to Rotational Offsets	33
4.5.4. Final Selection of Global Matching Technique	33
4.6. Optimization Techniques	33
4.6.1. Planar Transform Outlier Rejection Methods	33
4.6.2. Lowe's Ratio Test	34
4.6.3. Standard Deviation Filtering	34
4.6.4. N-Match or Absolute Thresholding	35
4.7. Summary	35
5. Results	36
5.1. Detailed Performance Analysis	36
5.1.1. Key Performance Metrics	36
5.1.2. Flight Path Analysis	37

5.1.3. Detailed Performance Metrics	37
5.1.4. Dataset Performance Analysis	38
5.1.5. Error Distribution Map	38
5.1.6. An Account of Variance Between and Error in Datasets	39
5.2. Stress Testing	41
5.2.1. Mutual Information Results	41
5.2.2. Low-Light Testing	42
5.3. Results Conclusion	44
6. Summary and Conclusion	45
7. Future Work	47
7.0.1. Multi-Image Weighted Inference System	47
7.0.2. Non-Planar Stereo Matching	47
7.0.3. Integration with Mapping and Reference Images	47
7.0.4. Distortion Corrections	48
7.0.5. Implementation on a Single Board Computer (SBC)	48
7.0.6. Higher Resolution Imaging	48
Bibliography	49
A. Project Planning Schedule	52
B. Outcomes Compliance	53

List of Figures

2.1.	Feature Detection and Matching	9
2.2.	Structural Similarity Index (SSIM)	13
2.3.	With Lowe's Ratio Test	16
3.1.	High-Level Flow of the System	18
3.2.	Loss comparison between pre-normalization and post-normalization methods.	23
4.1.	Examples of the CITY1 and CITY2 Datasets	26
4.2.	Example of the ROCKY Dataset	26
4.3.	Example of the DESERT Dataset	26
4.4.	Example of the AMAZON Dataset	26
4.5.	Overview of Datasets	26
4.6.	RMSE for Various Local Detectors	28
4.7.	Runtime for Various Local Detectors	28
4.8.	RMSE and Runtime Comparison for ORB, AKAZE, and SuperPoint Across Datasets	28
4.9.	RMSE GPS Accuracy for BFMatcher and FLANN	29
4.10.	Runtime Comparison for BFMatcher and FLANN	29
4.11.	Divergence in RMSE GPS Error Between FLANN and BFMatcher Across Keypoint Targets	30
4.12.	RMSE Comparison Across Datasets for Planar Transform Estimators	31
4.13.	Runtime Comparison Across Datasets for Planar Transform Estimators	31
4.14.	RMSE and Runtime Comparison Across Datasets for Planar Transform Estimators	31
4.15.	RMSE Comparison Across Datasets for Global Matching Techniques	32
4.16.	Runtime Comparison Across Datasets for Global Matching Techniques	32
4.17.	RMSE GPS Error Comparison Across Datasets for Various Methods	33
4.18.	Percentage Change in GPS Error with 10-degree Rotational Offset	33
4.19.	Radial GPS RMSE Comparison Across Datasets for LMEDS and RANSAC	34
4.20.	Runtime Comparison Across Datasets for LMEDS and RANSAC	34
5.1.	Key Accuracy And Runtime Performance Metrics	36
5.2.	Flight Path of UAV in poorest performing Dataset	37
5.3.	Flight Path of UAV in best performing Dataset	37

5.4. Accuracy Metrics for Each Dataset	37
5.5. Runtime Metrics for Each Stage of the Pipeline	37
5.6. Plot of distribution of radial errors in each dataset	38
5.7. Heatmap of Pixel Deviations in X and Y Directions	39
5.8. Accuracy vs Mutual Information	42
5.9. Daytime Image of Cape Town	43
5.10. Evening Image of Cape Town	43
5.11. Night Image of Cape Town	43
5.12. Lighting Conditions in Cape Town	43
5.13. Night and Evening RMSE GPS for Various Datasets	44

List of Tables

Nomenclature

Variables and Functions

$RMSE$	Root Mean Square Error, representing the average magnitude of errors. In this study, the usage of this metric specifically, and frequently, uses this term to refer to the radial error in GPS estimate, in metres, averaged across the dataset.
d_{radial}	Radial error distance, measuring the distance between estimated and true positions in meters.
x, y	Coordinates of a point in an image or on the ground plane.
θ	Rotation angle between consecutive images in degrees or radians, used for orientation alignment.
T	Transformation matrix representing the planar transformation between images.
H	Homography matrix, specifically mapping points from one image plane to another.
GPS	Global Positioning System, providing geolocation and time information for navigation.
$GNSS$	Global Navigation Satellite System, a generic term for satellite-based navigation systems.
MAE	Mean Absolute Error, measuring the average localization error without directionality.
MI	Mutual Information, measuring the similarity or overlap between images, often used in image matching.

Key Terms and Concepts

Features	Unique keypoints in an image along with their descriptors, used for identifying and uniquely matching points across different images for localization.
Affine Transformation	A planar transformation that preserves points, straight lines, and planes, including translation, scaling, rotation, and shearing.
Descriptors	Numerical values describing the characteristics of keypoints in an image, allowing for effective matching between different images.
Feature Detectors (or Extractors)	Processes for identifying distinct features in an image, which are invariant to changes in scale, rotation, and illumination.
Global Matching	Process of finding similarities and determining pose between entire images, considering full-image context rather than isolated keypoints.
GPS (Global Positioning System)	A satellite navigation system providing geolocation and time information. Vulnerable to jamming and spoofing, posing reliability issues for UAV navigation.
Homography	A planar transformation mapping points from one image plane to another in 2D space. Used in image processing to describe transformations like rotation, translation, scaling, and shearing.
Homography Estimation	The process of calculating the homography matrix that defines the transformation between two images for alignment and reliable localization.
Keypoints	Specific points in an image used to identify features. Typically areas of strong contrast or distinct patterns, enabling reliable matching across different images.
Local Matching	Matching keypoints between images by comparing descriptors, focusing on individual feature descriptors to establish precise localization.
Mutual Information	A metric quantifying the information shared between two images, aiding in assessing the quality of feature matching and overlap similarity.
Planar Transforms	General transformations applied to a plane in 2D space, such as affine transformations, homography, scaling, and shearing, which are crucial for image alignment.
Scaling	A transformation that changes the size of an image or its features without altering its shape, normalizing feature sizes across different

Acronyms and Abbreviations

GPS	Global Positioning System
RMSE	Root Mean Square Error
MAE	Mean Absolute Error
MI	Mutual Information
UAV	Unmanned Aerial Vehicle

Chapter 1

Introduction

1.1. Background

Unmanned Aerial Vehicles (UAVs) have become critical in modern military operations, providing capabilities in surveillance, reconnaissance, and intelligence gathering. In South Africa, UAVs are increasingly used to monitor borders and support military missions by enabling persistent aerial observation [1]. Their ability to operate in dangerous or inaccessible areas makes them invaluable for various operations.

However, UAVs heavily rely on the Global Positioning System (GPS) for navigation, which poses a significant vulnerability. GPS signals are easily disrupted by jamming, spoofing, or interference in hostile environments, leading to navigation failure. Adversaries increasingly exploit these vulnerabilities, resulting in UAVs becoming disoriented or lost, jeopardizing both the mission and the UAV.

The increasing prevalence of GPS disruptions highlights a critical need for alternative navigation solutions that can operate independently of external signals like GPS. These systems must be robust, accurate, and adaptable to a wide range of environments. Some alternatives include inertial navigation systems, radio frequency-based localization, and quantum positioning systems. However, these suffer from high costs, limited accuracy, susceptibility to detection, Earth curvature issues, or drift, making them impractical for widespread UAV deployment. One promising approach that does not fall prey to these issues is image-based navigation, where UAVs use onboard cameras to capture and match images with previously stored reference images to estimate their position. This method allows for autonomous navigation in GPS-denied environments without requiring continuous GPS access.

However, image-based systems face uncertainty regarding their ability to offer real-time operation, achieve accurate results, and handle various environments. This project aims to test the viability of such a system, addressing the need for reliable UAV navigation without GPS dependency.

While conditions such as poor lighting and dynamic environments can pose practical challenges, advancements in technology and image processing algorithms have significantly improved the robustness of computer vision tasks to these challenges.

By leveraging these advancements, this project will develop and evaluate a system that employs the latest image processing techniques to deliver accurate and real-time localization for UAVs in GPS-denied environments. The solution will be rigorously tested against stringent performance metrics to assess its practicality and reliability in real-world scenarios.

xxx The latter method is employed in this study and allows return navigation to base along the outbound path. - MUST BE SOMEWHERE scope should include the IMUs for path finding

1.2. Problem Statement

GPS-based UAV navigation is increasingly vulnerable to jamming and spoofing, risking loss of control. Current alternatives are either too expensive or unreliable, creating a need for a more robust, GPS-independent solution.

1.3. Aims and Objectives

1.3.1. Aim

The primary aim of this project is to develop an accurate and generalizable image-based GPS localization system to facilitate UAV navigation in GPS-denied environments.

1.3.2. Objectives

To achieve the stated aim, the following objectives have been established:

1. **Optimize the Localization Pipeline:** Identify and implement the most effective pipeline, including parameter selection and methodological approaches, to achieve the highest possible localization accuracy while maintaining real-time performance and generalizability. This includes the selection, integration and optimization of feature extraction, matching, and planar transformation estimation techniques.
2. **Accurate Localization Estimation:** Estimate the GPS location, using only prior telemetry data and image features, with a radial error below 10% of the UAV's radial displacement from the reference image. This accuracy is sufficient for manual control of the UAV in GPS-denied environments, given that the system provides continuous correction.
3. **Real-Time Operation:** Ensure the localization system operates in real-time, with a response time of less than 2 seconds following GPS signal loss. This is sufficient as

the relative motion of the UAV to the ground is sufficiently slow that a 2-second delay will not significantly affect the pilots ability to navigate.

4. **Environmental Generalizability:** Validate that the system maintains its performance metrics across diverse environments without the need for environment-specific parameter tuning.

1.4. Scope

The scope of this project is defined to ensure feasibility and manageability within the allocated timeframe. The limitations are considered reasonable in practical scenarios or can be addressed in future, more detailed studies. These constraints do not hinder the ability to assess the solution's viability. The following assumptions and restrictions outline the project's boundaries:

- **No Perspective Distortion:** The UAV's downward-facing camera is assumed to remain perfectly aligned downward throughout all operations, so no pitch or roll occurs. Additionally, the ground is assumed to be predominantly planar (2-dimensional), simplifying the model by removing the need for perspective distortion correction.
- **No Scale Distortion:** The UAV maintains a constant altitude, negating the requirement for scale estimation in localization calculations.
- **Sufficient Image Quality:** It is assumed that the captured images have adequate resolution and full visibility without significant distortion.
- **No Dynamic Movements:** The UAV is expected to operate over primarily static land surfaces with minimal dynamic objects, ensuring a stable environment for image-based localization.
- **Path Overlap:** The UAV leverages complementary odometry-based systems to determine its path. This system assumes that the UAV has located the path, and the overlap of reference image information is greater than 60%.
- **Method and Parameter Set Testing:** The study aims to assess the system's viability given sufficiently accurate methods and parameter sets, focusing on the overall performance of the system rather than on exhaustive testing of optimal methods or fine-tuning of parameters.
- **Training Requirements:** Machine-learning-based methods are excluded due to the extensive training data required, which is outside the project's scope.

- **Data Provisioning:** The project used data from Google Earth due to not receiving the prerequisite data, acknowledging a minor accuracy impact due to the use of estimated ground truth heading data.
- **System Output:** The system provides an estimated GPS location and heading, intended for manual UAV control by pilots rather than for automated control systems.

These scope limitations streamline the project while allowing for future enhancements, such as incorporating scale and perspective correction or improving robustness to dynamic environmental conditions.

1.5. Summary of Work

This project successfully developed and implemented an image-based GPS localization system for UAV navigation in GPS-denied environments. The system met the outlined objectives, demonstrating high accuracy, real-time performance, and generalizability across multiple environments. Through comprehensive research and implementation of feature extraction, matching, and homographic techniques, the system effectively estimated the UAV's location with minimal error. The system was tested and deemed to be practically robust under low-light and limited overlap conditions, with the ability to handle various environments. The outcomes indicate significant potential for enhancing UAV navigation reliability in both military and civilian applications.

1.6. Structure of the Report

This report is structured to provide a comprehensive overview of the research undertaken to develop the image-based GPS localization system for UAVs. The chapters are organized as follows:

- **Chapter 2: Literature Review**

This chapter reviews the current state of GPS-alternative UAV navigation systems, emphasizing their drawbacks and associated vulnerabilities. It provides the reader with the prerequisite knowledge to understand the techniques employed in this study, including feature detection and matching techniques, and planar transformation estimation methodologies.

- **Chapter 3: Methodology**

This chapter details the system pipeline designed to achieve accurate, efficient, and robust image-based GPS localization. It covers the flow and integration of feature extraction, matching, and homographic estimation, as well as the associated considerations made to optimize the system's performance.

- **Chapter 4: Testing**

This chapter describes the datasets utilized, the testing environment setup, and the comparative analysis of different methods employed at each stage of the pipeline. It culminates in the selection of the final methods integrated into the system.

- **Chapter 5: Results**

This chapter presents the evaluation of the developed pipeline against the established objectives. It includes stress testing results and assesses the system's performance in various scenarios to demonstrate its effectiveness and reliability.

- **Chapter 6: Conclusion**

The final chapter summarizes the project's outcomes, and evaluates the viability of the developed system.

- **Chapter 7: Future Work**

This section outlines recommendations for future work to address the project's limitations and explore additional enhancements.

Chapter 2

Literature Review

This chapter provides a comprehensive review of the existing literature on UAV navigation systems, focusing on image-based approaches and alternative navigation methodologies. The discussion encompasses feature extraction, matching, and planar transformations, elucidating the theoretical underpinnings essential for the proposed image-based navigation system. By exploring the strengths and limitations of various navigation techniques, this review sets the stage for the subsequent system design and implementation, highlighting the critical role of feature-based methods in UAV navigation.

2.1. Global Navigation Satellite Systems (GNSS)

Vulnerabilities

Global Navigation Satellite Systems (GNSS), including the U.S. Global Positioning System (GPS), Russia's GLONASS, the European Union's Galileo, and China's BeiDou, provide essential positioning, navigation, and timing (PNT) information globally. GNSS operates by deploying a constellation of satellites that transmit precise time-stamped signals to Earth-based receivers. Each receiver calculates its position by measuring the time delay of signals from multiple satellites, triangulating data from at least four satellites to determine its three-dimensional location. While GNSS is critical across sectors such as aviation, maritime, and land navigation, its reliance on weak, line-of-sight satellite signals introduces significant vulnerabilities [2].

The reliance on GPS for UAV navigation has grown extensively, yet vulnerabilities to jamming and spoofing have become increasingly pronounced. With the advent of more accessible jamming and spoofing technology, intentional GPS interference is no longer a complex undertaking, raising serious concerns for security-sensitive applications. Instances of GPS jamming and spoofing are particularly prevalent in conflict zones, where adversaries exploit GPS weaknesses to disrupt or mislead UAV operations. The issue is far from hypothetical; in fact, more than 1,100 incidents of GPS spoofing alone were reported worldwide in August 2024, underscoring the urgency of addressing these vulnerabilities [3].

Such interference has direct implications for national security, as border surveillance and reconnaissance missions are heavily dependent on accurate GPS data [1]. Consequently,

alternative redundancy navigation methods that are not susceptible to such interference are critical in ensuring operational continuity and resilience in high-stakes scenarios.

2.2. Alternative Navigation Approaches

In the realm of UAV navigation, various redundancy measures have been considered and implemented to reduce reliance on GNSS. This section reviews these techniques, highlighting their strengths and limitations in the context of UAV applications.

2.2.1. Quantum Sensing Navigation

Quantum sensing navigation employs cold atom inertial sensors to achieve superior precision compared to traditional inertial measurement units (IMUs). In this approach, cold atoms are trapped in a vacuum and manipulated using lasers to measure motion through interference patterns, offering exceptionally accurate acceleration and rotation data that theoretically minimize navigation drift [4]. However, the **high cost and bulky nature** of cold atom sensors make them impractical for many UAV applications. Operational limitations also persist, as aviation-grade systems incorporating cold atom sensors can reduce drift by approximately half but still suffer from significant drift over extended periods [4].

2.2.2. Odometry-Based Solutions

Odometry-based navigation estimates UAV displacement by tracking wheel rotation or utilizing accelerometer readings, providing a direct measurement of movement for real-time position tracking. However, odometry systems are inherently susceptible to cumulative **drift**, where minor errors accumulate over time and distance, leading to significant inaccuracies in long-range navigation [5]. Small discrepancies in sensor measurements can result in substantial positional errors over extended flights, and environmental factors such as variations in terrain, wheel slippage, or unexpected accelerations can further degrade odometry accuracy. These limitations render odometry-based methods unsuitable as a primary navigation solution for long-distance missions. Nevertheless, odometry can be effectively utilized for short-range navigation tasks, such as pathfinding and supplementary guidance in conjunction with more reliable systems like image-based navigation.

2.2.3. RF Communication and Signals-Based Navigation

RF communication systems enable navigation by triangulating the UAV's position using signal sources like ground beacons, cellular towers, or Wi-Fi networks. By analyzing signal strength, frequency, and timing, RF-based navigation offers an independent alternative

to GPS, particularly viable in urban areas with abundant signals and enhanced security through encryption. However, reliance on existing infrastructure limits its use in remote areas, and the Earth's curvature further restricts line-of-sight, limiting applications to short distances, around 300km, reducing signal strength and accuracy [6].

2.2.4. LIDAR and Radar for Ground Mapping

LIDAR and radar are effective for UAV ground mapping through active wave emission, measuring distances to surfaces and generating detailed 3D environmental maps. These sensors provide precise altitude measurements and mapping capabilities, allowing for accurate location inference. However, they emit **detectable signals**, which may be undesirable for applications requiring minimal detectability. Additionally, LIDAR and radar systems can be resource-intensive and face challenges in dense environments where interference and signal scattering may impair accuracy and operational range [7].

2.2.5. Image-Based Homography Navigation

Image-based homography navigation leverages homography estimation techniques to enhance UAV navigation by aligning images and inferring planar transformations. This approach uses feature extraction and matching from reference images to determine the UAV's orientation and position relative to its environment. By creating a reliable alignment between the UAV's current view and previously captured data, the system can accurately infer location and heading, enabling effective navigation back to base, particularly in GPS-denied or GPS-compromised scenarios. Reference images are captured during flight before any GPS loss and are later used to guide the UAV's return to base. Future applications could extend this method by utilizing a database of recent reference images, allowing broader deployment across various terrains and operational contexts.

This method addresses several limitations associated with alternative navigation solutions. Specifically, it is cost-effective, space-efficient, does not emit detectable signals, and does not drift.

Much research exists in the field of homography, including its application to aerial imagery, as demonstrated in [8], where its effectiveness across various applications is well established. However, existing studies primarily focus on high-level overviews or isolated improvements within the homography algorithm. This study distinguishes itself by not only evaluating the effectiveness of homography estimation for UAV navigation but also by providing a comprehensive implementation pipeline, encompassing feature detection through transformation estimation, offering a practical framework that can be readily adapted for diverse applications.

Image-based navigation systems are often overlooked due to perceived limitations in achieving real-time performance, robustness to distortions, generalization across various en-

vironments, and high accuracy. However, as computer vision, machine learning techniques, and camera technology continue to advance, these challenges are becoming increasingly surmountable. Their main practical limitation is related to extreme visibility issues, which are relatively infrequent. This study aims to develop an optimized model pipeline and evaluate the viability of the image-based solution in real-world scenarios, determining its potential to be accurate, efficient, and adaptable across diverse environments.

2.3. Background

This section provides a comprehensive overview of the fundamental concepts and techniques that underpin the proposed image-based UAV navigation system. By delving into feature extraction, matching, and planar transformations, it establishes the theoretical foundation essential for the subsequent system design and implementation. The discussion emphasizes the critical role of feature-based methods over direct approaches, setting the stage for understanding the chosen methodologies.

2.3.1. Feature Detectors

Feature extraction is a cornerstone of image-based UAV navigation, enabling the estimation of transformations such as rotation and translation between consecutive images. Feature detectors identify **keypoints**—distinct, repeatable points within an image—and generate **descriptors** that encapsulate information about the local image region surrounding each keypoint. These keypoints and descriptors are essential for accurate matching across multiple frames, allowing the system to track movement while maintaining invariance to changes in scale, rotation, and illumination. This precision is critical for accurately inferring both rotational and translational shifts between images. An example of feature detection and matching, to be explained in the subsequent section, is illustrated in Figure 2.1 as outputted by the navigation system. This image shows the top 50 matches between two images after rotational alignment.

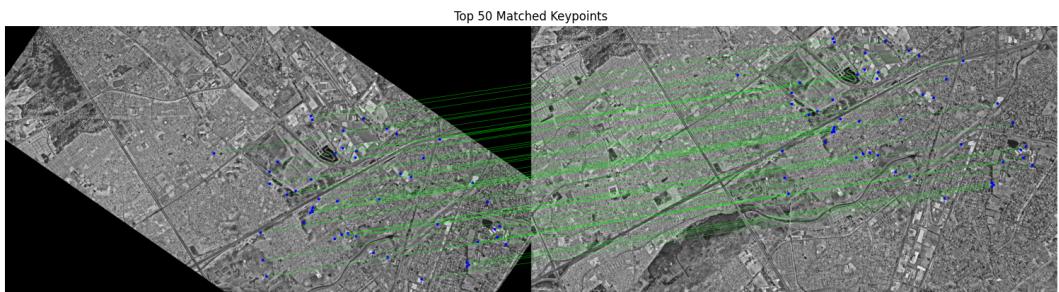


Figure 2.1: Feature Detection and Matching

In this context, *features* refer to both keypoints and descriptors, as each feature extraction method provides both components to facilitate effective image matching. The selection of feature extractors is guided by criteria such as accuracy, computational efficiency, robustness to diverse datasets, and the necessity for freely available tools, thereby excluding proprietary methods like SURF due to licensing fees. Additionally, chosen methods must hold credibility within the field, given their application in safety-critical UAV navigation systems.

The feature detectors selected for this system—**ORB**, **AKAZE**, and **SuperPoint with LightGlue**—offer a range of performance characteristics, balancing accuracy, computational efficiency, and machine learning integration. These extractors are further elaborated in the subsequent sections.

ORB (Oriented FAST and Rotated BRIEF)

ORB combines the **FAST** keypoint detector with the **BRIEF** descriptor, enhanced for rotation invariance. **FAST** rapidly identifies keypoints by analyzing pixel intensity differences in a circular region around each candidate point. Once detected, **BRIEF** encodes the local image patch into a binary string through intensity comparisons. **ORB** introduces rotational invariance by aligning keypoints based on their dominant orientation before descriptor computation. This enhancement makes **ORB** both fast and robust to scale and in-plane rotation, although it may struggle with repetitive textures or complex lighting variations [9].

AKAZE (Accelerated-KAZE)

AKAZE constructs a nonlinear scale space using diffusion-based filtering, capturing finer image details more effectively than linear methods. It detects keypoints by assessing local contrast with a specialized adaptive filter, enabling the identification of subtle features that simpler detectors might miss. The **Modified Local Difference Binary (MLDB)** descriptor encodes the neighborhood of each keypoint into a binary vector based on pixel intensity differences. While **AKAZE** is both fast and compact, its performance can be sensitive to detection thresholds across different environments, potentially affecting its robustness in varied operational contexts [10].

SuperPoint with LightGlue

SuperPoint is a deep learning-based keypoint detector and descriptor that leverages convolutional neural networks (CNNs) to identify and describe keypoints in a single forward pass. Pre-trained on extensive image datasets, **SuperPoint** excels at recognizing stable and distinctive keypoints under varied conditions [11]. However, its performance may degrade on datasets significantly different from its training data. Pairing **SuperPoint**

with **LightGlue**, a machine-learning-based matcher, enhances matching accuracy through advanced graph-based techniques [12]. Despite their high accuracy, SuperPoint and LightGlue are computationally intensive, necessitating GPU acceleration for real-time applications.

2.3.2. Feature Matching

Feature matching establishes correspondences between keypoints in different images based on descriptor similarity. After identifying these correspondences, ambiguities and low-quality matches are removed to retain only the most reliable matches, which are then used to estimate transformations such as translation and rotation. This filtering ensures that transformation estimations are based solely on mutual information between images, enhancing the accuracy and reliability of the navigation system.

Each matcher generates a list of potential matches along with their similarity scores, quantified using a descriptor-space distance metric. These scores are instrumental in determining the quality of the matches and play a crucial role in the subsequent filtering and transformation estimation processes.

Feature matching involves two primary components: the choice of matching technique to acquire potential matches and the search technique that determines which of these matches to retain.

Types of Feature Matching Techniques

The following match acquisition techniques are commonly employed in feature-based navigation systems:

Brute-Force Matcher (BFMatcher): The Brute-Force Matcher compares each feature in one image with every feature in the second image, ensuring the best possible match based on descriptor similarity. While this guarantees high accuracy, it is computationally expensive, especially with large numbers of keypoints, making it less suitable for real-time applications without optimization [13].

Fast Library for Approximate Nearest Neighbours (FLANN): FLANN accelerates the nearest neighbour search in high-dimensional descriptor spaces using algorithms such as KD-trees or hierarchical clustering. This approximate matching approach offers significant speed improvements with minimal loss in accuracy, making it ideal for real-time applications with extensive datasets [14].

LightGlue: Leveraging deep learning, LightGlue improves matching accuracy by employing advanced graph-based techniques to establish more reliable correspondences. Although highly effective, its structure necessitates the use of other neural network-based feature extractors like SuperPoint to realize its full potential [12]. The enhanced accuracy comes at the cost of increased computational demands, requiring GPU acceleration for

optimal performance.

The following search techniques are commonly used to filter matches and retain only the most reliable correspondences:

Radius Search: This method retains matches within a specified distance in descriptor space, effectively filtering out weaker matches. However, it does not guarantee a fixed number of matches per keypoint, leading to inconsistent results [15].

K-Nearest Neighbours (KNN) Matching: KNN matching retains the top K matches for each keypoint, allowing the application of post-filtering techniques such as Lowe's ratio test to eliminate ambiguous matches [15].

Vanilla Matching: Vanilla matching returns the single best match for each keypoint based on the closest descriptor distance. It is a subset of KNN matching with K=1, offering simplicity and ease of implementation [15].

Further, these matches are filtered using multiple techniques to ensure accuracy; this is discussed in the optimization techniques section.

2.3.3. Image Similarity Computation

Image similarity computation is a pivotal component of UAV navigation systems that rely on reference images for accurate localization and pose estimation. Effective similarity measures ensure efficient processing of extensive image datasets and facilitate precise transformation estimations, which are essential for reliable navigation.

Proximity-Based Techniques

To achieve real-time performance, the search space is reduced to images within the proximity of UAV's last known location, filtering images within a static or dynamic radius. While this method is highly efficient, it does not account for potential deviations from the expected flight path or the presence of poor-quality reference images. This limitation implicates that this measure cannot be used as the sole basis for image similarity computation, necessitating the integration of additional techniques for comprehensive assessment.

Global Matching Techniques

Direct methods estimate planar transformations by comparing entire image pixel intensities and minimizing differences through optimization techniques like gradient descent. This single-layer processing allows for efficient analysis of multiple images and a comprehensive understanding of image similarity within the global context, making direct methods suitable for scenarios with minor viewpoint changes. However, relying on full pixel usage reduces accuracy in cases of significant transformations due to increased sensitivity to noise and the assumption of smooth, incremental UAV movements. Direct methods are best suited for applications that aim to estimate similarity across large image sets. These methods

necessitate rotational alignment of images to eliminate bias introduced by orientation discrepancies [16]. The following subsections detail the primary global matching techniques employed in this system to compute similarity scores.

Cross-Correlation

Cross-correlation measures similarity by sliding one image over another and computing the sum of pixel-wise multiplications at each position. The peak value signifies the best alignment, and its magnitude indicates the confidence level of the similarity. Higher confidence values reflect greater similarity between the images. While straightforward to implement, cross-correlation is sensitive to noise and illumination changes, which can compromise the reliability of the similarity measure [17].

Histograms

Histogram comparison assesses similarity by analyzing the distribution of pixel intensities within each image. Typically, each image's histogram is divided into 256 intensity bins, and similarity is quantified using metrics such as Chi-Square or Bhattacharyya distance. This method emphasizes global color and brightness distributions but neglects spatial information, making it less effective for nuanced structural differences [18].

Structural Similarity Index (SSIM)

SSIM evaluates similarity by decomposing images into luminance, contrast, and structure components. It computes local statistics within small windows and integrates them into a single similarity score that mirrors perceived image quality. SSIM effectively captures structural information like edges and textures, aligning closely with human visual perception. Although slightly more computationally expensive than the former methods, it is robust to varied conditions [19]. An illustration of SSIM is shown in Figure 2.2 from [19].



Figure 2.2: Structural Similarity Index (SSIM)

Local Detectors Conversion

Although not inherently a global matching technique, local feature matching can be adapted to achieve a global understanding of image similarity. This involves identifying and matching keypoints in both images and assessing the overall number of good matches. However, this approach alone does not ensure an even distribution of matches across the

entire image, potentially leading to biased, localized similarity assessments. To mitigate this, a grid matching technique is employed, dividing the image into grids and limiting the number of matches per grid. Although the most computationally intensive, this method enhances robustness against distortions and rotations by ensuring a uniform distribution of matches across the image.

2.3.4. Planar Transformation Estimators

Feature-based methods extract and match keypoints from both reference and real-time images, often incorporating outlier removal stages [16]. By focusing on distinctive features rather than every pixel, these methods excel in handling large viewpoint changes and rotations, enabling more precise transformation inference. This targeted approach enhances computational efficiency and robustness to environmental distortions, though it requires careful management to avoid performance degradation from variation in the number of extracted feature points. Feature-based methods are preferable for applications demanding high accuracy and adaptability in complex environments [20]. These feature-based transformations enable precise updates to the UAV's pose and location, facilitating accurate navigation and control. The following subsections outline the primary planar transformations employed in this system.

Affine Transformation

Affine transformation captures translation, rotation, scaling, and shear, providing six degrees of freedom. It is represented by a 2×3 matrix that maps points from one plane to another while preserving lines and parallelism. Affine transformations are computed by estimating the affine transformation matrix between two sets of corresponding points using OpenCV's `estimateAffine2D` function [21]. While versatile, the inclusion of scaling and shear can introduce unnecessary complexity for scenarios where only rotation and translation are relevant, potentially impacting the accuracy of transformation estimations.

Rigid Transformation Estimation (SVD)

Rigid transformation preserves the shape and size of objects by estimating only rotation and translation, excluding scaling and shear. Represented by a 2×3 matrix, rigid transformation ensures orthogonality in the rotation component. Utilizing Singular Value Decomposition (SVD), this method minimizes the least-squares error between two point sets. The process involves: computing the weighted centroids of both point sets, centering the points by subtracting their respective centroids, calculating the covariance matrix of the centered points, performing SVD on the covariance matrix to derive the rotation matrix, and determining the translation vector based on the centroids. The resulting 2×2 rotation matrix and 2×1 translation vector are combined to form the rigid transformation

matrix. Rigid transformation is computationally efficient and well-suited for applications requiring only rotation and translation [22].

Partial Affine Transformation

Partial affine transformation simplifies the full affine model by focusing solely on translation, rotation, and uniform scaling, offering four degrees of freedom. This transformation is also represented by a 2×3 matrix, similar to the affine transformation but without shearing and with reduced, uniform scaling. By limiting scaling to be uniform, partial affine transformation reduces potential distortions and maintains simplicity, making it ideal for scenarios where only rotation and translation are significant [21].

Homography Transformation

Homography transformation accounts for translation, rotation, scaling, shear, and perspective distortion, providing eight degrees of freedom. It is represented by a 2×3 matrix and is estimated using OpenCV's `findHomography` function, typically with RANSAC for outlier rejection [23]. While homography offers greater flexibility in modeling complex transformations, its additional degrees of freedom can introduce unnecessary errors and computational overhead for simpler applications that only require rotation and translation. Consequently, homography is reserved for scenarios necessitating comprehensive transformation modeling beyond rotation and translation.

2.3.5. Optimization Techniques

Optimizing parameter sets is essential for enhancing the performance and robustness of the UAV navigation system. These optimization techniques aim to refine the accuracy and reliability of the point cloud used for transformation estimation by effectively filtering out erroneous matches and improving transformation accuracy. The following subsections detail the primary optimization methods employed in this system.

Random Sample Consensus (RANSAC) for Planar Transformation

RANSAC is a robust estimation technique used to estimate planar transformations by iteratively selecting random subsets of point correspondences to fit a model and identify inliers [24]. The process involves randomly selecting a minimal subset of point pairs, estimating the transformation model (e.g., affine or homography) based on the selected subset, determining the number of inliers that fit the estimated model within a predefined threshold, and repeating the process for a set number of iterations or until a sufficient inlier ratio is achieved. This approach is highly effective in datasets with significant outliers, focusing on finding a model that best fits the largest subset of inliers. However, due to

its iterative nature and the need to sample repeatedly, RANSAC can result in increased runtime, particularly in larger datasets or when dealing with numerous outliers [24].

Local Maxima Extrema Density Selection (LMEDS) for Planar Transformation

LMEDS is a keypoint selection method designed to prioritize areas of high feature density by identifying local maxima as keypoints for matching [25]. This technique involves analyzing the image to identify regions with high feature density, selecting keypoints located at local maxima within these dense regions, and filtering out less significant keypoints to reduce redundancy and improve match quality. By concentrating on areas with high feature concentration, LMEDS ensures that keypoints represent the most distinctive and informative regions of the image, enhancing both accuracy and performance by reducing redundant keypoints and improving match quality, particularly in areas with high feature variability.

Lowe's Ratio Test

Lowe's ratio test is a filtering technique used to eliminate ambiguous or false keypoint matches by comparing the distance of the best match to the second-best match [26]. For each keypoint match, the ratio of the distance of the best match to that of the second-best match is calculated, and the match is retained if this ratio is below a predefined threshold. A lower ratio indicates that the best match is significantly better than the alternatives, thereby increasing the likelihood of the match being correct. An example of Lowe's ratio test filtration are showed in figure 2.3a and 2.3b from [26].

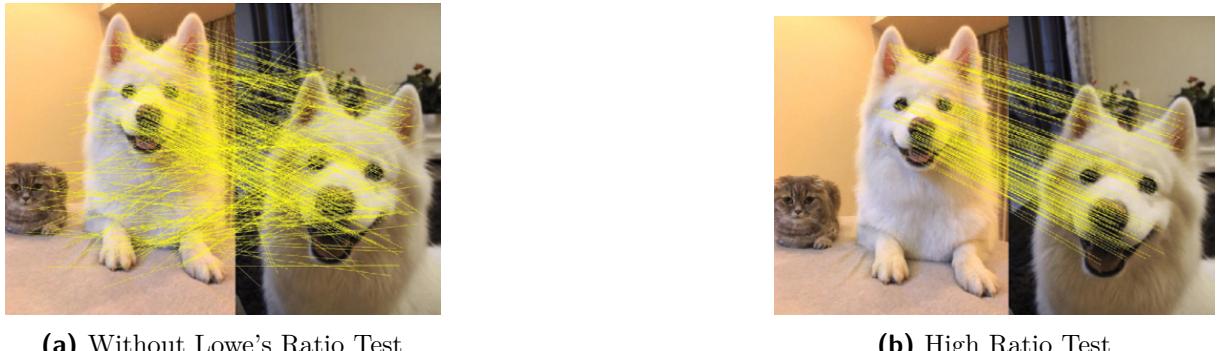


Figure 2.3: With Lowe's Ratio Test

Standard Deviation Filtering

Standard deviation filtering involves computing an average for the dataset, typically the mean or median, and removing matches that deviate significantly from this average. This technique is simple to implement but has the ability to generalize well across datasets due to its relative nature.

N-Match or Absolute Thresholding

N-match thresholding involves setting a threshold that allows only a specific number of matches with the smallest descriptor distances to be retained. Absolute thresholding filters matches based on a fixed distance in descriptor space. Only matches that meet or fall below this predefined distance threshold are retained, ensuring that only sufficiently similar matches are used in the transformation estimation process. These methods primarily suffer from difficulty in setting the threshold, which is often extremely sensitive to dataset variations and method parameters.

Chapter 3

System Design

This chapter outlines the chosen methodology for the UAV navigation system, detailing the pipeline and its various components. The system is designed to accurately estimate the UAV's position and heading, even in scenarios where GPS data is unreliable or unavailable. The high-level flow of the system is illustrated in Figure 3.1.

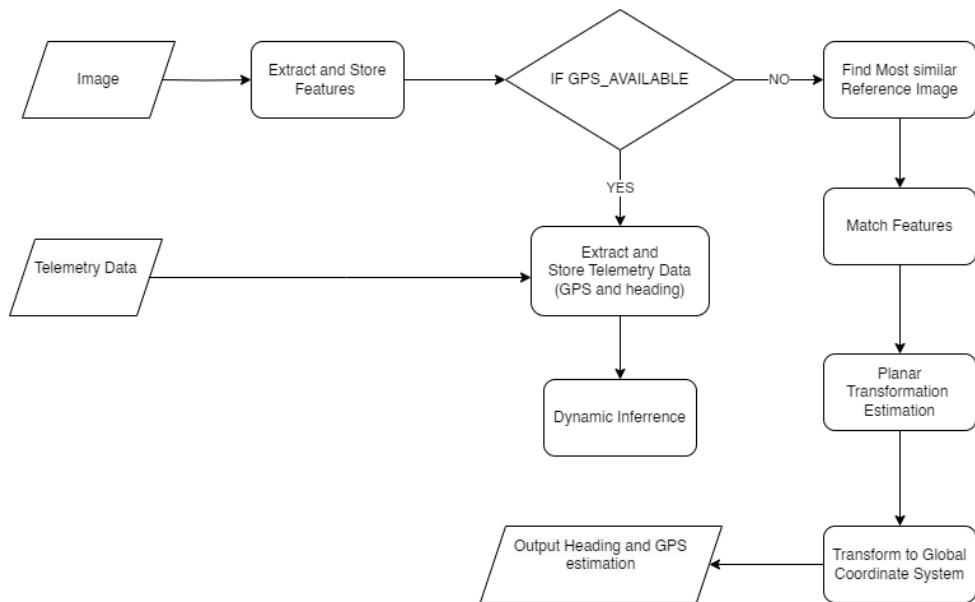


Figure 3.1: High-Level Flow of the System

3.1. System Pipeline

The UAV image-processing pipeline comprises several stages, each addressing specific goals in the larger aim to estimate the new GPS and heading of the UAV. These stages require varying levels of precision and efficiency to ensure robust performance.

3.1.1. Pipeline Stages

1. **Image Input:** The process begins with capturing a live image from the UAV's downward-facing camera. This real-time visual input provides essential information about the UAV's environment, forming the basis for position estimation.

2. **Feature Extraction:** Keypoints and descriptors are extracted from the current image to aid in the matching process. This extraction occurs in two layers:

- (a) **Coarse Layer:** Used for proximity search space reduction.
- (b) **Dense Layer:** Used for precise matching and transformation estimation.

By also performing feature extraction when GPS is available, the system reduces computational load during the critical phase when GPS is lost.

3. **Storage (Telemetry and Features):** Extracted features (keypoints and descriptors) along with telemetry data (GPS position and heading) are stored for future reference. Stored features facilitate relative transformation inference when GPS is unavailable, while telemetry data assists in converting relative transformations to real-world coordinates and headings.

4. **Match Features:** Features between the current image and reference images are matched to ensure that comparisons are based on mutual features. This involves two matching stages:

- (a) **Coarse Matching:** Matches the coarse layer of features to reduce the search space.
- (b) **Dense Matching:** Matches the dense layer for precise position estimation.

The matching process is optimized to ensure robustness against noise and outliers, with a focus on computational efficiency. The techniques are as follows:

- (a) **Feature Matching:** Utilizes FLANN and BruteForce matchers to identify potential matches between the current and reference images. The matchers are optimized to ensure robustness against noise and outliers, with a focus on computational efficiency.
 - (b) **Optimization:** Refines the matches using techniques such as Lowe's ratio test to ensure that only high-quality matches are considered for further processing. This refinement step is crucial for accurate transformation estimation.
5. **Similarity Comparison:** The system compares the input image with reference images to identify the most similar one. This comparison is conducted using global matching techniques on aligned images to ensure efficiency and accuracy.

- (a) **Proximity Search Space Reduction:** Reduces the search space based on proximity from the last known or estimated GPS location. A fixed radius was initially used, but later updated to return a specific number of the closest matches.

- (b) **Rotational Alignment:** Estimates the rotation between the input and reference images using the coarse layer of features. The images are aligned using the coarse estimate as global matchers tolerate minor rotational inaccuracies well.
 - (c) **Best Match Identification:** Computes similarity scores between the input image and each candidate image using global matching techniques. The image with the highest similarity score is selected as the best match, serving as the reference for position estimation.
6. **Planar Transformation Estimation:** After identifying the best match, the system performs a precise estimation of both rotation and translation between the input and reference images using the dense match layer. This involves several sub-steps to enhance accuracy:
- (a) **Angle Estimation:** Utilizes the chosen transformation method to obtain a precise estimate of the rotation between the input and reference images, forming the basis for heading estimation.
 - (b) **Image Alignment & Recomputation of Dense Layer:** Aligns the input image with the reference image based on the estimated rotation, thereby implicitly removing non-mutual information by rotating it off the canvas. The dense layer of features is recomputed on the aligned images to ensure accurate translation estimates.
 - (c) **Translation Estimate:** Performs a precise estimation of translation between the two images using the refined dense layer, providing the basis for GPS inference.
7. **Update to Global Coordinate System:** The estimated translation, initially in the internal image coordinate system (pixels), must be converted to real-world coordinates (metres) and then to the global coordinate system (longitude and latitude). This conversion involves several stages:
- (a) **Heading Update:** Adds the estimated rotation to the reference image's heading to determine the UAV's new heading.
 - (b) **Translation Rotation from Internal to Global Coordinate System:** Rotates the translation vector by the UAV's estimated global heading to align it with the global coordinate system without altering its magnitude.
 - (c) **Pixel to Metres (Relative):** Converts the estimated change in pixels to metres using the dynamically inferred conversion factor.
Translates the metre-based changes to relative changes in longitude and latitude using the following equations:

$$\Delta\text{Longitude} = \frac{\Delta\text{Metres}}{111320 \times \cos(\text{Latitude})}, \Delta\text{Latitude} = \frac{\Delta\text{Metres}}{111320}$$

These calculations account for the Earth's oblate spheroid shape, ensuring accurate positional data by adjusting for the decreasing distance between longitudes as one moves towards the poles.

- (d) **Conversion to Absolute GPS:** Adds the relative changes in longitude and latitude to the GPS coordinates of the reference image, resulting in the UAV's new GPS position.
- 8. **Output New GPS and Heading:** The calculated translation and rotation values are used to estimate the UAV's new GPS coordinates and heading. This enables pilots to navigate the UAV back to its base along the original path, even in the absence of GPS.
- 9. **Dynamic Parameter Inference:** This stage is performed while GPS is available. It estimates the GPS position using a temporary pixel-to-metre conversion factor of 1, utilizing the pipeline's stages to compute this estimate. The estimated GPS is compared to the ground truth GPS, and the difference is used to adjust the pixel-to-metre conversion factor. This ensures accurate conversion across datasets with varying altitudes. The first five images are used to infer this scaling factor.

3.2. Dynamic Methods and Techniques

Upon rigorous testing, it was seen that many parameters in the pipeline simply cannot generalize without tuning. This included the parameters of some feature detectors, optimization techniques, and search space reduction methods. To address this, dynamic methods were implemented to adjust these parameters based on the current scenario. The following dynamic methods were implemented:

- **Dynamic Feature Detector Selection:** AKAZE, the only detector without a dedicated keypoint target parameter (such as ORB's nFeatures), was highly accurate when tuned. However, the leniency threshold of AKAZE was found to be inconsistent across datasets. To address this, a dynamic method was implemented to adjust the leniency threshold based on the number of keypoints detected. This ensures that the detector is neither too lenient nor too strict, providing a balance between accuracy and efficiency.
- **Lowe's Ratio:** Lowe's ratio test is a critical step in the matching process, ensuring that only high-quality matches are considered for further processing. However, the optimal threshold for this test varies highly across datasets and scenarios. To address this, a dynamic method was implemented to adjust the Lowe's ratio threshold,

starting at a strict threshold, and gradually relaxing it until a sufficient number of matches are found. This ensures that the system remains robust against noise and outliers while maintaining computational efficiency.

The above dynamic methods ensure that the system remains stable in different scenarios, adapting to varying conditions to provide accurate and efficient performance. However, these methods still, in part, fail to understand the nuances of the dataset perfectly. In the future, in flight error to ground truth estimates can be used to iteratively solve for the most optimal static parameters for the given environment, assuming it does not significantly change during flight. Alternatively, a dedicated machine learning model can be trained to predict and constantly adjust the optimal parameters based on the dataset's characteristics in-flight, providing a more robust and efficient solution.

3.3. Rotational Normalization Strategies

Rotational normalization is crucial for accurately aligning images internally and to the global heading space to estimate the UAVs relative and absolute positions. However, this inherently discards information that is rotated off of the canvas. Two primary strategies were evaluated: pre-normalization, which aligns images to a global North-East (NE) reference before keypoint detection, and post-normalization, which aligns images relative to each other before applying normalization after translation estimation.

In the pre-normalization approach, information loss scales sinusoidally with the absolute values of the difference in rotations between images. Further, this loss contains predominantly losses in non-mutual pixels due to its internal alignment. Conversely, post-normalization scales with both angle losses independently, increasing the likelihood of compounded loss.

Given that the UAV's typical return flight will be 180 degrees to the outbound path, with a maximum path deviation of ± 20 degrees, the loss functions for the two methods are illustrated in Figure 3.2. This function shows losses exceeding 2.0 due to the compounded effect of independent image normalization, whereas the post-normalization method maintains losses around 0.35 by preserving keypoint information more effectively. Therefore, and in accordance with empirical testing results, post-normalization was adopted for its increased mutual information retention rates.

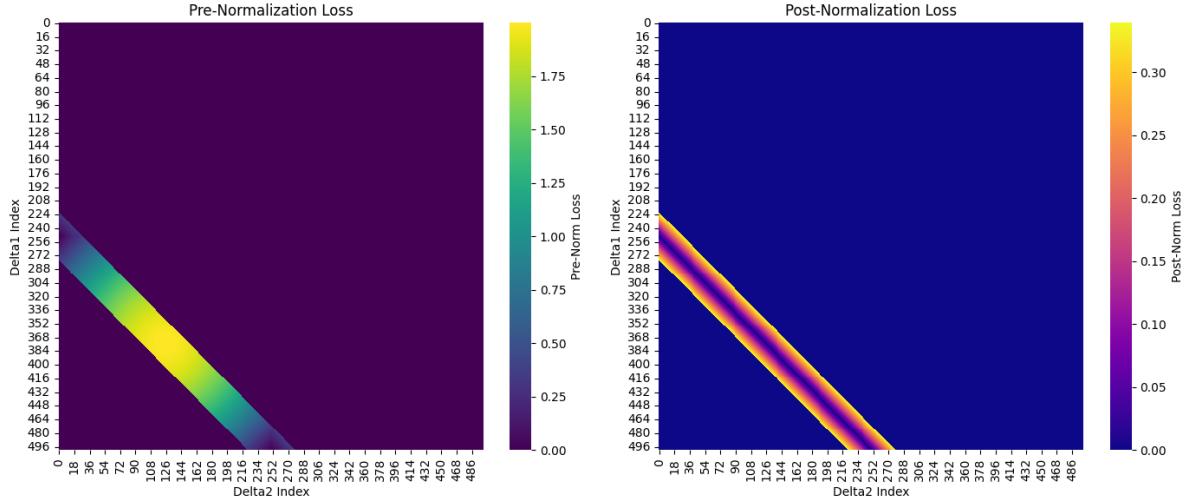


Figure 3.2: Loss comparison between pre-normalization and post-normalization methods.

3.4. Testing Shortlist

The system design integrates multiple features to ensure robustness, accuracy, and efficiency in UAV navigation tasks. The following components have been selected and tested across various scenarios:

- **Feature Detectors:** AKAZE, SuperPoint (with LightGlue matcher), and ORB. Note, the SuperPoint detector is used in conjunction with the LightGlue matcher to enhance performance.
- **Feature Matchers:** FLANN and BruteForce. KNN
- **Search Techniques:** KNN with $K = 2$. Empirical tests showed a value of 1 to be ineffective without Lowe's ratio test, while values above 2 introduced excessive computational overheads. Further, radius search was seen to be unreliable due to the varying density of keypoints across datasets.
- **Planar Transformation Estimation:** Homography, Affine, Partial Affine (Rigid) transformations using OpenCV, and SVD-based rigid transformations for rotation and translation estimations. Rotational and Translational estimates were initially tested separately, due to the possibility of different responses to different prior stages and methods. However, it was seen that responses were comparatively equivalent; the methods were tested using a combined transform.
- **Global Image Similarity Measures:** SSIM, histogram matching, local retrofit, and cross-correlation.

- **Optimization Techniques:** Standard Deviation Filtering, LMEDS, RANSAC, Lowe's ratio test, n-Match thresholding, and absolute thresholding for match refinement. Empirical tests indicated that cross-checking was not applicable due to the excessive computational costs involved. KNN

3.5. Conclusion

The system design integrates feature extraction, matching, and transformation estimation to enable precise UAV navigation, even in the absence of reliable GPS data. By employing a combination of diverse feature detectors and optimizing various pipeline stages, the system ensures robustness, accuracy, and computational efficiency across different operational scenarios.

Chapter 4

Comparison of Methods

This chapter compares the performance of various methods used in the UAV navigation system, focusing on accuracy, runtime, and robustness. The evaluation includes feature detectors, local feature matchers, rotational and translational estimators, and optimization techniques. Each method is tested across multiple datasets to assess generalization and suitability for real-world UAV applications.

4.1. Testing Setup

This section outlines the framework used to evaluate the performance of the proposed UAV navigation methods. The evaluation focuses on three primary metrics: accuracy, runtime, and robustness. These metrics are critical to ensure that the navigation system can reliably operate under diverse and challenging conditions, reflecting real-world scenarios where the UAV may encounter varying environmental factors.

4.1.1. Datasets

Five distinct datasets were selected to rigorously evaluate the methods' generalization and performance across diverse environments. These datasets were captured using Google Earth and involved both translational and rotational movements, simulating typical UAV navigation tasks. Although the primary transformations were translation and rotation, perspective distortions and other subtle deformations were present due to the uneven terrain. Multiple methods with varying degrees of freedom were tested to implicitly estimate which of these transforms were significantly present. Each of the properties and choices were done to ensure the task was challenging but not impossible.

General Properties of the Datasets:

- **Altitude:** Ranging from 5.5 to 6.5 km, constant within each dataset. This is standard for UAV navigation applications and ensures the ground is sufficiently planar.
- **Resolution:** 1920x972 pixels per image. This is the standard resolution in most applications. Stress tests were performed at lower resolutions.

- **Radial Movement Between Frames:** Approximately 600 pixels. This ensured sufficient overlap for a challenging but feasible task.
- **Number of Images per Dataset:** 15 images per dataset. This balances testing time and is sufficient to evaluate the methods' performance.

The datasets used for testing were as follows:

- **CITY1 and CITY2 (Cape Town):** Both datasets were captured in Cape Town. **CITY1** includes both rotational and translational changes between frames, while **CITY2** focuses solely on translational movements. This distinction allows for isolated testing of performance under rotational stress.
- **ROCKY:** This dataset covers rugged terrain with varying features, testing the methods' robustness in handling complex, uneven topographies.
- **DESERT and AMAZON:** These datasets are characterized by sparse, repetitive patterns, making it challenging even for human observers to distinguish differences between frames. They present significant difficulty for feature extraction and matching.

Examples of the datasets are shown below:



Figure 4.1: Examples of the CITY1 and CITY2 Datasets

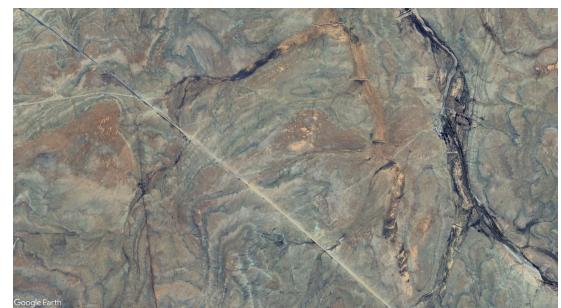


Figure 4.2: Example of the ROCKY Dataset

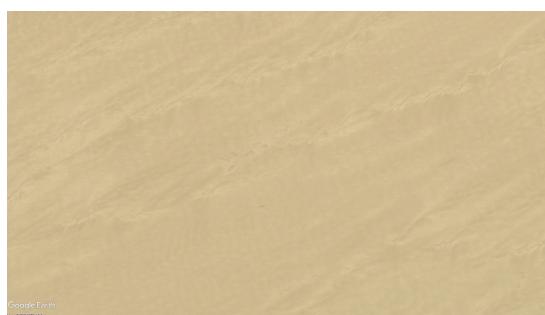


Figure 4.3: Example of the DESERT Dataset



Figure 4.4: Example of the AMAZON Dataset

Figure 4.5: Overview of Datasets

4.1.2. Testing Structure

Each method is subjected to rigorous testing based on the following criteria:

- **Accuracy:** Evaluated using the Root Mean Square Error (RMSE) of GPS estimations, accuracy assessments focus on the end error. This is justified because outputs—whether images or degrees—propagate through the pipeline, meaning any errors or poor choices are ultimately reflected in the final GPS error.
- **Runtime:** The entire system runtime per dataset is assessed to evaluate the computational efficiency of each method. Efficient runtimes are crucial for real-time UAV applications, as delays—combined with pilot and UAV response times—can result in consistently missing the target and failing to follow the intended path. As before, runtime may propagate through the system, and therefore the runtime per line is not necessarily indicative of better performance; Runtime per line is not tested.
- **Robustness:** Tested to verify each method’s stability under parameter variations and challenging conditions. Robust methods maintain consistent performance despite environmental or parameter changes, ensuring reliable UAV navigation across different scenarios.

This structured testing approach ensures that each component of the UAV navigation system is thoroughly evaluated, facilitating the selection of methods that deliver optimal performance across all critical metrics.

4.1.3. Parameters

Parameters were held constant across tests and chosen to ensure optimal performance across methods. The focus was not on selecting the absolute best parameter set, as this was not relevant for inter-method testing; rather, the goal was to minimize bias while allowing each method to perform effectively. For instance, multiple methods were tested within a single runtime to enhance testing speed without compromising the integrity of the inter-method comparisons. The takeaway here is to avoid interpreting results objectively, as they do not reflect the realistic and overall performance of the system.

4.2. Feature Detectors

This section presents the evaluation results of three feature detectors: ORB, AKAZE, and SuperPoint with LightGlue. The results from these tests informed the selection of the most appropriate detector and threshold parameters for the crude and dense layer of keypoints.

4.2.1. Accuracy and Runtime

Figure 4.6 and Figure 4.7 present the RMSE and runtime values for each feature detector across different datasets. AKAZE, utilizing dynamic keypoint targeting, demonstrated the highest accuracy across all datasets while maintaining reasonable runtime. SuperPoint recorded the highest RMSE values, particularly in challenging datasets such as ROCKY and AMAZON, indicating its limited generalizability across diverse environments. Meanwhile, ORB proved to be the most efficient detector, making it suitable for applications requiring fast processing. SuperPoint demonstrated the longest runtimes across all datasets, highlighting its limited applicability for time-sensitive applications unless optimized with GPU acceleration.

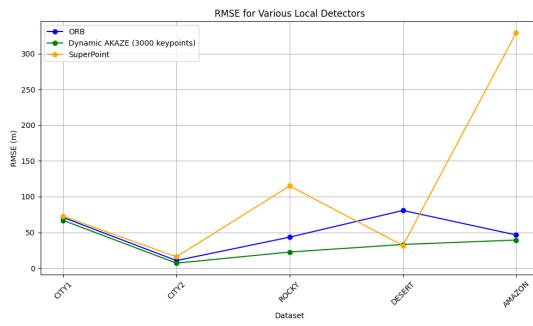


Figure 4.6: RMSE for Various Local Detectors

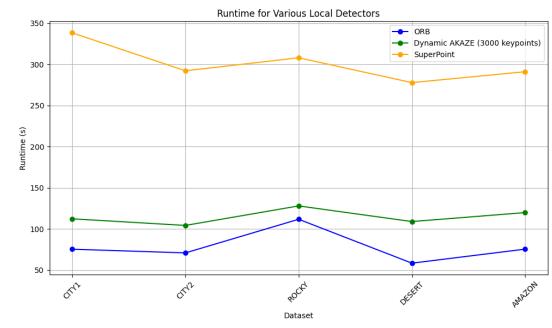


Figure 4.7: Runtime for Various Local Detectors

Figure 4.8: RMSE and Runtime Comparison for ORB, AKAZE, and SuperPoint Across Datasets

Robustness

When paired with keypoint targeting, all features obtained the same number of keypoints across datasets. However, the quality of the keypoints varied significantly. However, AKAZE managed to consistently deliver less noisy keypoints, making it the most robust detector across all datasets.

4.2.2. Final Selection of Feature Detectors

Based on the comprehensive evaluation, the following detectors were selected for the respective stages of the UAV navigation system:

- **Crude Layer (Initial Detection):** ORB was chosen for the crude layer due to its balance of accuracy and efficiency with 2500 keypoints.
- **Dense Layer (Refined Detection):** Dynamic AKAZE with 3000 keypoints was selected for the dense layer due to its consistent performance and robustness. This

detector complements the crude layer by refining feature extraction and enhancing the system's accuracy.

4.3. Local Feature Matchers

This section evaluates two prominent local matchers, BFMatcher and FLANN, within the context of a UAV navigation system.

4.3.1. Accuracy and Runtime Evaluation

Figure 4.9 presents the Root Mean Squared Error (RMSE) in GPS values for BFMatcher and FLANN across different datasets. The results indicate that while BFMatcher achieves slightly better accuracy in certain cases, FLANN remains highly competitive with only marginally higher RMSE values.

Figure 4.10 shows the runtime comparison for BFMatcher and FLANN across different datasets. FLANN consistently outperforms BFMatcher in terms of speed, with significantly lower execution times across all datasets.

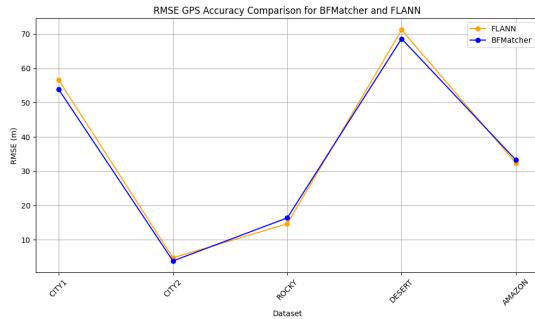


Figure 4.9: RMSE GPS Accuracy for BFMatcher and FLANN

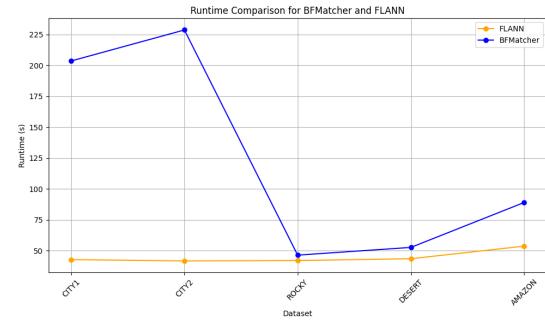


Figure 4.10: Runtime Comparison for BFMatcher and FLANN

4.3.2. Robustness Under Varying Detector Thresholds

Figure 4.11 depicts the divergence in RMSE GPS error between FLANN and BFMatcher as the number of keypoints increases. Positive values indicate that BFMatcher outperforms FLANN, while negative values suggest FLANN performs better. The plot demonstrates the convergence of both matchers' accuracy as the number of keypoints increases, with FLANN generally maintaining slightly higher RMSE values than BFMatcher. Note that outliers are present, indicating instances where FLANN achieves better accuracy than BFMatcher. This is attributed to FLANN's approximate matching, paired with a static Lowe's ratio threshold, which may preserve valid matches that BFMatcher might discard.

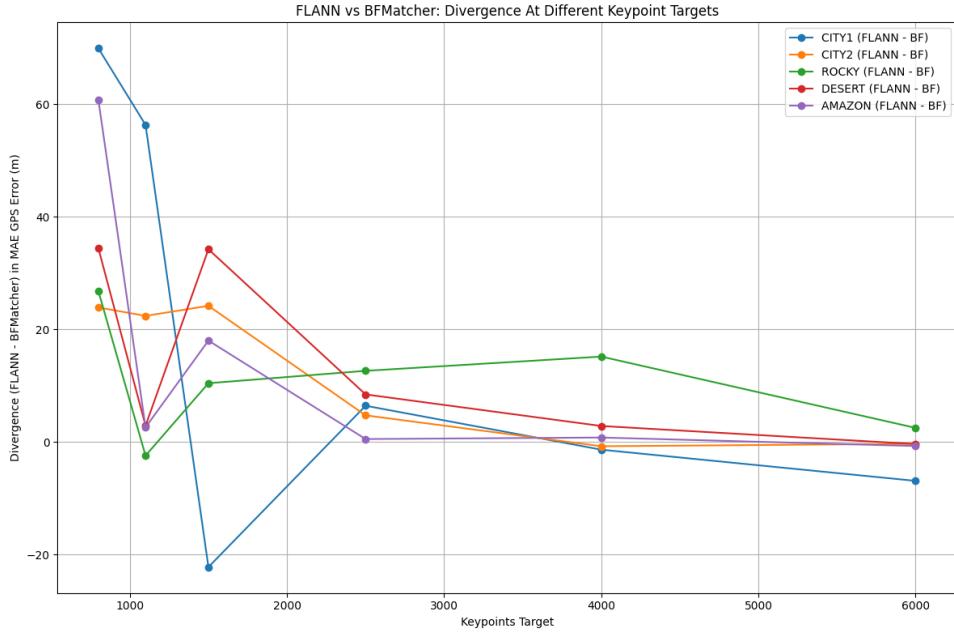


Figure 4.11: Divergence in RMSE GPS Error Between FLANN and BFMatcher Across Keypoint Targets

4.3.3. Final Selection of Local Feature Matcher

Based on the comprehensive evaluation of accuracy, runtime, and robustness, FLANN emerges as the optimal choice for the UAV navigation system. FLANN offers significantly faster runtimes and better scalability while maintaining comparable accuracy to BFMatcher. This makes FLANN highly suitable for real-time applications where computational efficiency is paramount.

4.4. Planar Transform Estimators

This section evaluates the performance of four planar transformation estimation methods: Partial Affine 2D (Rigid Transform plus minor Scaling), Affine 2D, Homography, and Rigid Transform Via SVD. As noted in section 3.4, the accuracy of the method when used for rotational and translational estimates were seen to be comparatively equivalent. Thus, methods were tested for their combined influence on the accuracy on the transformation estimates, as viewed through the GPS error. For these tests, RANSAC was used on all methods, except the Rigid Transform Via SVD, to filter outliers.

4.4.1. Accuracy and Runtime Evaluation

Figure 4.14 summarizes the RMSE and runtime values across datasets for each rotational estimator method. From the results, it is clear that each method exhibits strengths in different datasets. However, the Rigid SVD method and Partial Affine 2D consistently

emerge as the most accurate methods. This implies that the Rigid SVD transform provides the most stable and accurate estimate of the rotational and translational components of the UAV's movement. The Rigid SVD method slightly outperforms Partial Affine 2D in most datasets, attributed to its strict rigid transformation alignment with dataset characteristics.

The runtime results, shown in 4.14, are influenced by the degrees of freedom in each method, with rigid-based transforms demonstrating the best performance across datasets. SVD proves the fastest due to its lack of iterative optimization.

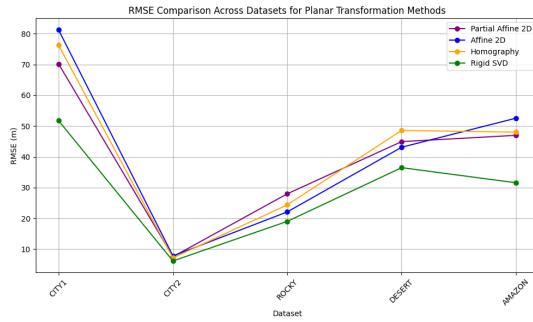


Figure 4.12: RMSE Comparison Across Datasets for Planar Transform Estimators

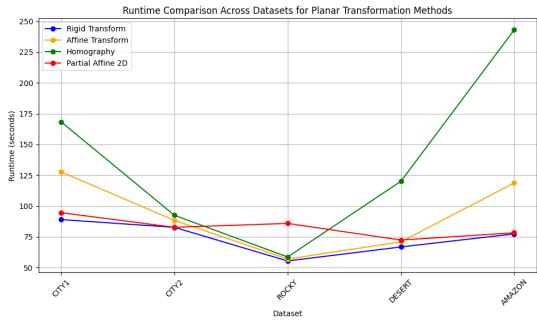


Figure 4.13: Runtime Comparison Across Datasets for Planar Transform Estimators

Figure 4.14: RMSE and Runtime Comparison Across Datasets for Planar Transform Estimators

4.4.2. Robustness Testing

Robustness testing assessed each method's sensitivity to parameter variations, including Lowe's ratio and keypoint detectors and their corresponding detection thresholds. This evaluation was essential to determine the performance consistency of each method under different challenging conditions. For the sake of brevity, the results are omitted; The recorded observations are summarized below.

Observations:

- **Parameter Sensitivity:** All methods were highly robust to parameter variations, maintaining consistent performance across different settings.
- **Best Robustness:** The Rigid SVD Transform exhibited the highest robustness across all parameters, especially at low keypoint counts.

4.4.3. Final Selection of Transformational Estimator

Based on the comprehensive evaluation of accuracy, runtime, and robustness, the rigid transform by SVD emerged as the most suitable rotational estimator for the UAV navigation

system. It demonstrated the lowest combined radial RMSE in GPS across all datasets, the fastest runtime, and high robustness due to its precise match to the degrees of freedom in the datasets. Further, it required no parameter tuning, making it highly suitable for real-time UAV applications.

4.5. Image Similarity Estimators

Accurate image similarity estimation, or global matching, is essential for UAV navigation systems to choose reasonable images to compare to. They should ensure accuracy and efficiency while maintaining robustness against small rotational offsets.

4.5.1. Accuracy Evaluation

4.5.2. Accuracy and Runtime Evaluation

Naturally, the appropriateness of the choice of best match is implicitly passed through to subsequent stages and realized as an error in GPS estimations. Figure 4.15 summarizes the RMSE values (in meters) for Local Retrofit, Cross Correlation, Histogram, and SSIM. The results indicate that the Histogram technique consistently achieved the lowest RMSE across most datasets, followed by Cross Correlation and SSIM. The Local Retrofit method recorded the highest RMSE values, especially in the DESERT dataset, indicating poor generalizability and higher complexity; It was excluded from further analysis.

Figure 4.16 compares the computational efficiency of each global matching technique across the five datasets. The Histogram technique demonstrated the fastest runtimes, followed closely by Cross Correlation. SSIM exhibited the longest runtimes, particularly in the AMAZON and DESERT datasets, rendering it less suitable for real-time applications.

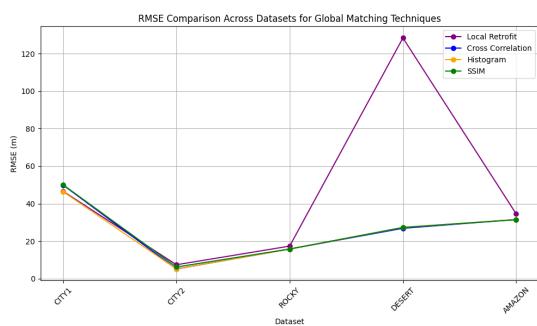


Figure 4.15: RMSE Comparison Across Datasets for Global Matching Techniques

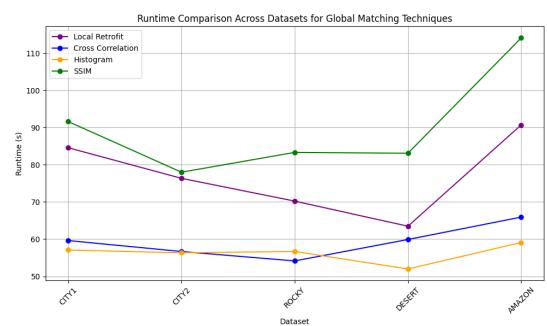


Figure 4.16: Runtime Comparison Across Datasets for Global Matching Techniques

4.5.3. Robustness to Rotational Offsets

Robustness testing evaluated each global matching technique's sensitivity to a 5- and 10-degree rotational offset. All matchers that made it this far saw no change in choice combination below a 5-degree offset. The resultant percentage deviation under a 10-degree offset is shown in Figure 4.18. Cross-correlation was the most robust, with the lowest percentage change in GPS error, from its prior estimate, across all datasets. Histogram and SSIM exhibited moderate robustness, while SSIM demonstrated the highest maximum error and variability. However, the ability of all matchers to maintain equivalent performance under offsets of up to 5 degrees suggests that they are all extremely robust to small rotational misalignments; This allows for prioritization of efficiency in the sections outputting the rotational estimate to this stage.

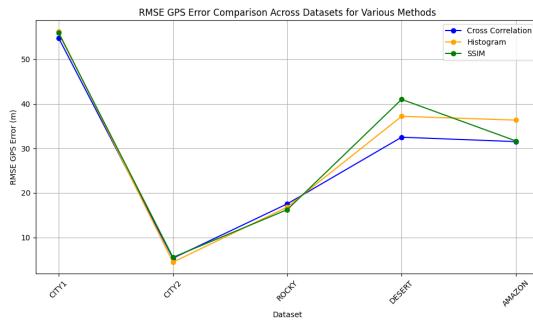


Figure 4.17: RMSE GPS Error Comparison Across Datasets for Various Methods

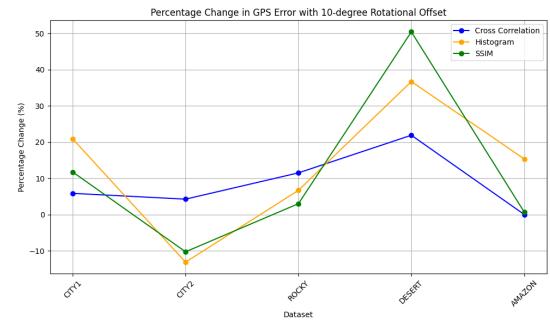


Figure 4.18: Percentage Change in GPS Error with 10-degree Rotational Offset

4.5.4. Final Selection of Global Matching Technique

Based on the comprehensive evaluation of accuracy, runtime, and robustness, the **Histogram** technique is identified and chosen as the most suitable global matching method for the system. Histogram consistently provided superior performance in terms of both RMSE and runtime, whilst maintaining sufficient robustness to rotational error.

4.6. Optimization Techniques

This section outlines the selected optimization methods employed to improve the performance of the UAV navigation system. The optimization techniques focus on filtering matches between images to balance noise and stability in the point sets.

4.6.1. Planar Transform Outlier Rejection Methods

Two outlier rejection methods, LMEDS (Least Median of Squares) and RANSAC (Random Sample Consensus), were evaluated for match filtration. Both methods performed near

equivalently, with LMEDS displaying a slightly lower mean deviation. LMEDS was also significantly faster than RANSAC, making it the preferred choice for planar transform outlier rejection. The results are summarized in Figure 4.19 and Figure 4.20.

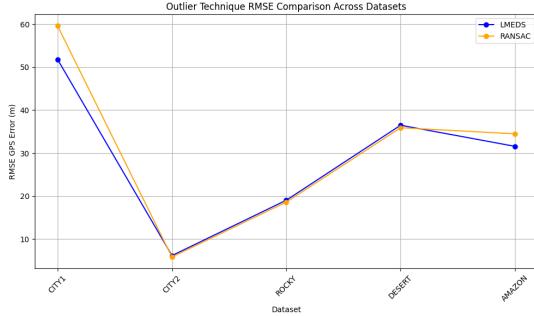


Figure 4.19: Radial GPS RMSE Comparison Across Datasets for LMEDS and RANSAC

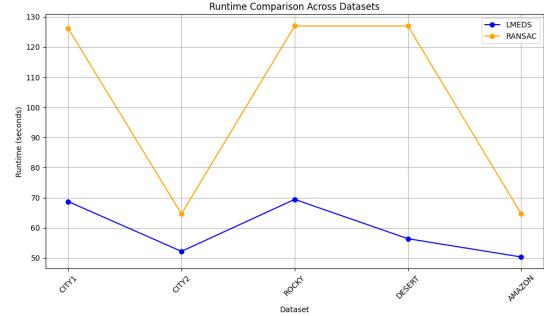


Figure 4.20: Runtime Comparison Across Datasets for LMEDS and RANSAC

4.6.2. Lowe's Ratio Test

Lowe's ratio test was employed to filter keypoint matches by comparing the distance of the best match to the second-best match. A static threshold proved inadequate for generalizing across diverse datasets, leading to inconsistent accuracy. To enhance robustness, a dynamic thresholding approach was implemented. The initial Lowe's ratio was set low and incrementally increased until a predetermined number of matches was achieved. The parameters chosen were 0.5 to 0.7 for the initial Lowe's ratio, 0.025 to 0.1 for the step size, 10 to 30 for the maximum iterations, and 2000 to 3000 for the minimum matches. These parameters were selected to balance generalization and match quality, ensuring a sufficient number of reliable matches without compromising accuracy. The dynamic Lowe's ratio filtering method was found to significantly enhance the quality of matches, improving the overall accuracy of the system.

4.6.3. Standard Deviation Filtering

Standard Deviation Filtering was applied to both the magnitudes and angles of the match points. It was effective when coupled with the median, for robust outlier rejection, and applied to the vector using a standard deviation in the range of 1 and 3. This balanced stability and the exclusion of false positives. Standard Deviation Filtering was found to significantly improve the quality and consistency of matches, enhancing the overall accuracy of the system.

4.6.4. N-Match or Absolute Thresholding

These approaches leveraged fixed number or descriptor distance thresholds to filter matches. However, without relativity to the density and quality of keypoints, empirical tests revealed that these methods added instability to the system. They either had no meaningful effect or removed too many matches, leading to loss of accuracy. The thresholds were deemed too sensitive for generalized applications.

4.7. Summary

The following methods were selected based on the comprehensive evaluation of accuracy, runtime, and robustness across diverse datasets:

- **Feature Detectors:** ORB for the crude layer and Dynamic AKAZE for the dense layer.
- **Local Feature Matcher:** FLANN for its superior runtime and scalability.
- **Transformational Estimator:** Rigid Transform SVD for its high accuracy, fast runtime, and robustness.
- **Image Similarity Estimator:** Histogram for its superior accuracy and runtime.
- **Optimization Techniques:** Dynamic Lowe's filtering and Standard Deviation Filtering for enhanced match quality and consistency; LMEDS was not included due to prior choices of methods not requiring it.

Chapter 5

Results

5.1. Detailed Performance Analysis

This section presents the results achieved by the navigation system under optimal conditions across various challenging datasets. The analysis focuses on the system's performance in terms of accuracy, efficiency, and generalizability, providing insights into its applicability in real-world UAV navigation scenarios.

5.1.1. Key Performance Metrics

The system's generalized performance across five diverse datasets—CITY1, CITY2, ROCKY, DESERT, and AMAZON—is summarized with two key metrics: Localization radial error in RMSE m and percentage, and localization inference time in seconds across all datasets. The results are presented in Figure 5.1. Importantly, the system meets the real-time requirement of 2 seconds and the accuracy requirement of 10%.

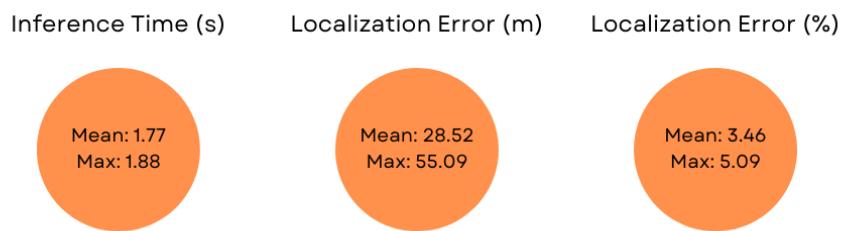


Figure 5.1: Key Accuracy And Runtime Performance Metrics

5.1.2. Flight Path Analysis

Figures 5.2 and 5.3 show the actual vs estimated flight of the UAV across the worst and best performing dataset, CITY1 and CITY2 respectively. Even in the worst performing dataset, the system maintains a relatively accurate flight path, relative to its movement size, with the estimated path closely following the actual path. This indicates the system's ability to generalize and gives an idea of the scope of the error in the system.

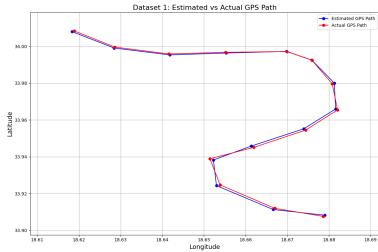


Figure 5.2: Flight Path of UAV in poorest performing Dataset

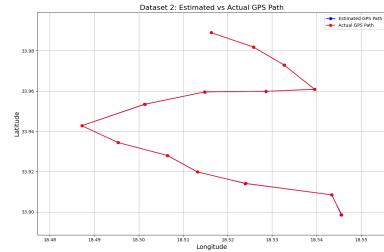


Figure 5.3: Flight Path of UAV in best performing Dataset

5.1.3. Detailed Performance Metrics

The detailed performance metrics include per dataset accuracy and time metrics. Namely, the radial RMSE error in GPS in meters and percentage, and the time taken for each stage of the pipeline, including the extraction and parameter inference time per while GPS is available, and the location inference time per image when GPS is lost. The results are summarized in Figures 5.4 and 5.4.

The results show that the system achieves consistent accuracy and runtime performance across each dataset. Further, its noted that both pipelines are able to maintain real-time performance, seen by the sum of the parameter inference and add times being below 2 seconds, as well as the location inference time being below 2 seconds. The accuracy of the system is also maintained, with the RMSE error being below 10% for all datasets.

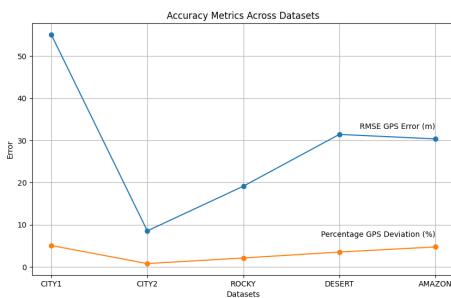


Figure 5.4: Accuracy Metrics for Each Dataset

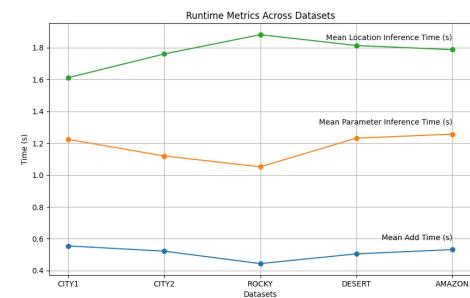


Figure 5.5: Runtime Metrics for Each Stage of the Pipeline

5.1.4. Dataset Performance Analysis

This section utilizes the violin plot, in Figure 5.6, to visualize the statistical distribution of radial errors across each dataset. This plots details the mean, median, and variance of the radial errors, as well as the distribution of errors across the datasets. The results provide insights into the system's accuracy and robustness in diverse environments.

The plot shows relatively consistent performance across datasets, with CITY2, being consistent with the global heading space, having the most accurate and reliable inter-dataset performance. The CITY1 and AMAZON datasets show comparatively high variation between their minimum and maximum errors, indicating a higher variance in direction.

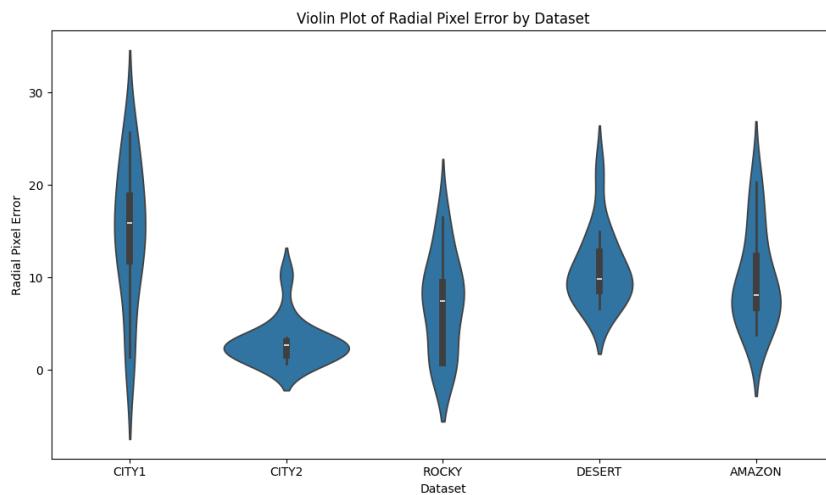


Figure 5.6: Plot of distribution of radial errors in each dataset

5.1.5. Error Distribution Map

The heatmap in Figure 5.7 visualizes the distribution of pixel deviations in the X and Y directions across the datasets. The results provide insights into the system's error patterns and highlight areas of potential improvement. The heatmap shows that the system has a relatively consistent error distribution across the datasets, with the majority of errors being within the 10-20 pixel range. However, some datasets, such as DESERT and AMAZON, exhibit larger errors of up to 30 pixels, indicating potential areas of significant error.

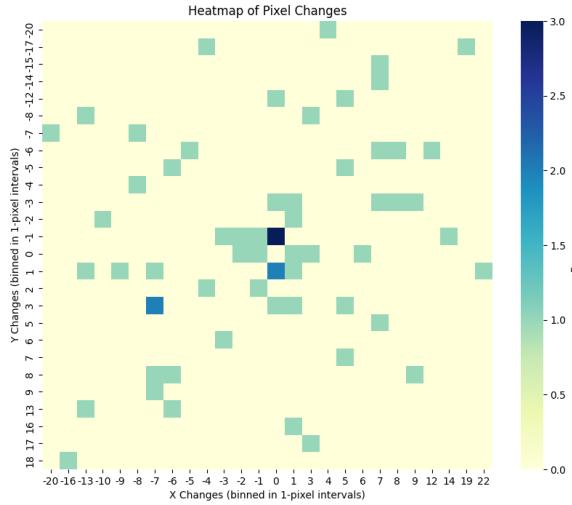


Figure 5.7: Heatmap of Pixel Deviations in X and Y Directions

5.1.6. An Account of Variance Between and Error in Datasets

Given the large variation in error across datasets, as seen in both the violin plot in figure 5.6 and the heatmap in figure 5.7, it is important to understand the sources of this variance. This section will detail the analysis conducted to identify the sources of this variance and the impact of these sources on the system's performance.

Internal Uncertainty and Variance

Correlations between error sizes and measures of uncertainty in the pipeline, specifically the standard deviation and mean-median difference in match vector magnitudes, were assessed. The variability in these vectors was used as a proxy for the uncertainty of the method. If the method was more uncertain when it was wrong, then the error likely originated from there. The match vectors, tested at various stages in the filtration process imply the accuracy of the transformational estimate. The results, in terms of correlation coefficients, as well as visual inspection, showed no relationship between uncertainty in the pipeline and error, indicating that there was not an increased likelihood of the variances in performance propagating from anywhere including or before the transformational estimate.

Ground Truth Heading error and Variance

Upon observing a moderate correlation between error magnitude and image sequence in the dataset, the accuracy of the estimated headings from Google Earth data was reconsidered. Ground truth headings, as explained in 1.4, were estimated. These heading estimates were generated early in the project, prior to the development of an optimized rotational estimation technique. Additionally, this technique involved sequential estimation,

introducing cumulative errors. To evaluate the impact of these inaccuracies, a newer, optimized pipeline for heading estimation was employed, which would significantly reduce, but not eliminate, ground truth heading errors on the AMAZON dataset; The dataset was retested. The results demonstrated a significant reduction in mean radial error from 30.37 m to 17.91 m. This improvement underscores that inaccurate ground truth heading data was a significant contributor to the system's variance in performance across datasets.

Impact of Heading Error on Accuracy

Given these findings, and the factor improvements, it is estimated that the true absolute maximum error for an individual image in the given datasets, with the ground truth data available, is in the range of 7.5-15 pixels radially; this corresponds to around 3.4%-6.8% of the magnitude of the translation vector for that image, remaining within acceptable limits for the system's design objectives. Similar impacts are expected on the accuracy results of the other datasets, excluding the CITY2 dataset which has a fixed North heading for all images. The absence of true ground truth heading data constituted the primary factor in the variance in accuracy across datasets.

Additional Sources of Error

While heading estimation errors were identified as the main contributor to the overall inaccuracies, several other factors also contribute to the errors observed:

Variations in Terrain and Environmental Conditions: Differences in terrain, lighting conditions, and feature density affect keypoint detection and matching accuracy. The system, even with dynamic parameter adjustments, cannot perfectly predict the expected quantity or quality of features without specific dataset tuning. Consequently, it is due to inherently miss some high-quality features or incorrectly match less reliable ones. Incorporating a machine learning model to predict expected feature density could enhance the system's performance.

Inaccuracies in Camera Parameter Estimations: While the system dynamically infers camera parameters, any inaccuracies in these estimations accumulate, leading to scaled errors during subsequent localization steps that rely on these parameters.

Image Noise and Resolution Limitations: The Google Earth images used are at a fixed resolution of 1920x1080 pixels, which inherently contain some noise. This noise can adversely affect feature detection and matching accuracy. **Algorithmic Constraints:** The selected image processing and matching algorithms have inherent limitations, particularly when handling extreme conditions or highly repetitive features. Additionally, being optimized for portability, they may not capture and represent every feature within an image perfectly.

Despite these factors, the achieved optimal radial error range, roughly 1-20 pixels

across datasets, underscores the robustness and effectiveness of the image-based navigation system, affirming its suitability for practical UAV applications.

5.2. Stress Testing

This section involves the analysis of the system's robustness under various stressful conditions, including low-light environments and reduced mutual information. These tests were chosen practical and current relevance. The results aim to evaluate the system's performance under challenging scenarios and identify potential areas for improvement.

5.2.1. Mutual Information Results

In UAV navigation, accurate path maintenance is essential, especially when GPS signals are lost. Under these conditions, mutual information, or overlap—defined as the overlapping pixel information between current and reference images—plays a crucial role in guiding the UAV back on course. Specifically, when the UAV loses its path, there will be much less overlapping pixels than once the path is found. This study assesses whether reduced mutual information can still provide reliable navigational guidance and examines the trade-off between accuracy and computational efficiency.

The first objective was to identify the threshold of mutual information required to maintain an RMSE below 10%, ensuring navigational accuracy without inducing additional drift. Secondly, it was to evaluate if reducing mutual information by cropping image sizes can significantly decrease runtime while preserving acceptable accuracy levels.

Methodology

The CITY2 dataset is utilized, focusing exclusively on translational changes to simplify the mutual pixel calculation to the difference between the image size and translation vector. Images are progressively cropped from the default resolution of 1920x972 pixels to reduce mutual information. The analysis uses the mutual information as that of the image pair with the lowest mutual information, since it is the systems bottleneck. This ensures visual brevity, noting that mutual information of that of the image pair with the highest mutual information remains within 30% of the minimum across the dataset.

Results

Figure 5.8 shows that navigational accuracy remains stable until mutual information decreases to approximately 250,000 pixels (15.67% of a 1920x1080 image), below which accuracy sharply declines, indicating unreliable estimates. To maintain an RMSE below 10%, mutual information must exceed 300,000 pixels (18.81% of a 1920x1080 image).

Reducing mutual information by cropping images significantly decreases runtime. While the identified mutual information thresholds are specific to this dataset and pipeline, the results demonstrate the feasibility of balancing mutual information and runtime for effective navigation. Lastly, it is noted that the system achieved largely varying levels of accuracy at the same mutual information level, indicating different amounts of information is stored in different axes. It is therefore recommended, if this method of decreasing runtime is considered, that the image is cropped towards an aspect ratio of one to maintain stability as the UAV rotates.

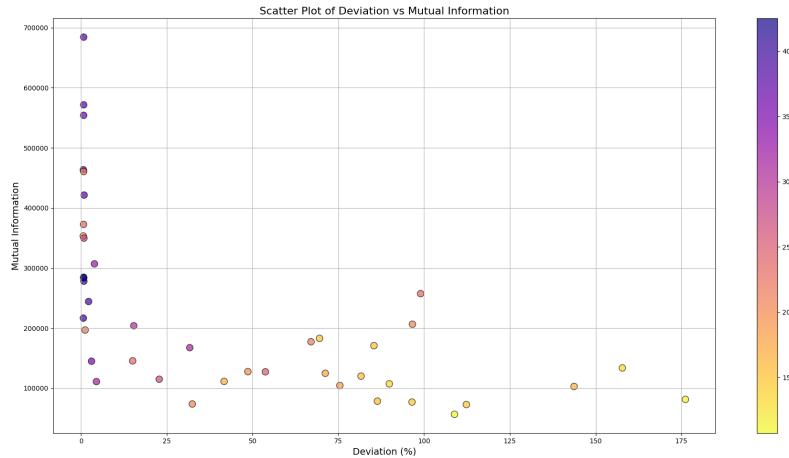


Figure 5.8: Accuracy vs Mutual Information

Conclusions

This study demonstrates that UAV navigational accuracy is maintained with mutual information levels up to approximately 20% of the original image size. Above this threshold, additional mutual information yields diminishing returns in accuracy, indicating an inelastic performance region. Runtime performance improves proportionally with reduced mutual information, presenting a trade-off between accuracy and computational efficiency. For instance, cropping the image resolution to 1100x635, subtending 300,000 mutual pixels achieves a 1.8-fold runtime reduction with only a 0.1% increase in error. These findings emphasize the importance of calibrating resolution levels to balance navigational accuracy and real-time performance. Further, when applying this method, cropping the image towards an aspect ratio of one is recommended to maintain stability as the UAV rotates. Future work should explore adaptive methods that start with reduced mutual information and increase it only when necessary, such as when the UAV is turning around to find its path.

5.2.2. Low-Light Testing

UAV navigation performance can be significantly impacted by low-light conditions, where keypoint detection and matching accuracy decline due to reduced visibility and increased

noise. This study evaluates the robustness of the navigation method under simulated evening and nighttime conditions across five datasets: CITY1, CITY2, ROCKY, DESERT, and AMAZON. Conditions are simulated through adaptions in image brightness, contrast, tint, and noise levels to mimic real-world low-light environments. Evening conditions involve moderate applications of these adjustments, while nighttime conditions feature severe reductions in visibility and increased noise levels.

The objectives of this study are twofold: firstly, to assess the robustness of the UAV navigation system under low-light conditions by evaluating how evening and nighttime lighting affect the accuracy of navigational estimates; and secondly, to compare the system's performance across diverse environments with varying feature densities under these low-light scenarios.

Representative examples from the CITY1 dataset illustrate these conditions:



Figure 5.9: Daytime Image of Cape Town



Figure 5.10: Evening Image of Cape Town



Figure 5.11: Night Image of Cape Town

Figure 5.12: Lighting Conditions in Cape Town

Results

Figure 5.13 summarizes the accuracy of the navigation method under evening and nighttime conditions for each dataset. Nighttime performance shows that the CITY and ROCKY datasets maintain relatively low RMSE values, indicating resilience to severe low-light environments, while the DESERT and AMAZON datasets exhibit substantial increases in RMSE due to sparse and repetitive features hindering effective keypoint matching. Evening conditions result in minimal degradation of navigational accuracy for most datasets, with the DESERT dataset experiencing a moderate error increase from its initial RMSE of 31.44 m to 156.34 m, corresponding to a 17.51% RMSE error, which remains usable.

The performance disparity between datasets underscores the importance of feature density in low-light navigation, with environments rich in distinct landmarks (e.g., CITY and ROCKY) demonstrating greater robustness compared to feature-poor environments (e.g., DESERT and AMAZON).

Overall, the navigation method demonstrates profound resilience in well-featured environments under low-light conditions. However, challenges still remain in feature-sparse environments during nighttime. The usage of additional sensors, such as infrared cameras, could enhance navigational accuracy in these challenging scenarios.

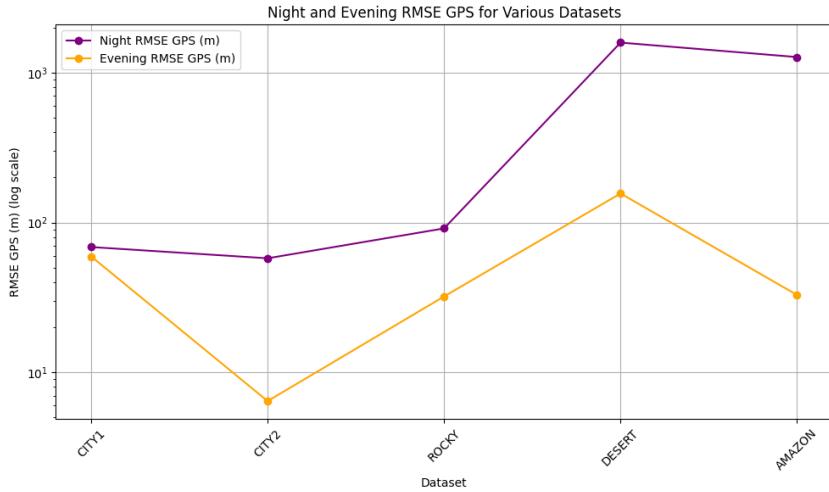


Figure 5.13: Night and Evening RMSE GPS for Various Datasets

5.3. Results Conclusion

The results presented in this chapter confirm that the system meets the accuracy and time requirements established in the objectives. The system demonstrated reliable performance across various datasets, including sparse-feature environments, showcasing its ability to generalize effectively. Despite practical limitations such as resolution constraints, performance limitations, and fluctuating terrain feature densities, the system maintained robust accuracy.

The analysis further indicates that accuracy would be significantly improved with access to accurate ground truth headings and known camera parameters, reducing potential sources of error. Furthermore, the system showed resilience under challenging conditions, including low-light scenarios and images with reduced mutual information. Even with limited pixel overlap, the system continued to deliver reliable navigation data.

Overall, the system has proven to be a versatile and effective solution for UAV navigation, capable of sustaining performance under a wide range of operational challenges and environmental conditions.

Chapter 6

Summary and Conclusion

Global Positioning System (GPS) vulnerabilities, such as jamming and spoofing, pose significant risks to Unmanned Aerial Vehicle (UAV) navigation, potentially leading to loss of control and mission failure. Recognizing the limitations of existing alternatives—which are often prohibitively expensive or unreliable—this project addressed the critical need for a robust, GPS-independent navigation solution for UAVs operating in GPS-denied environments.

The primary objective was to develop an accurate and generalizable image-based GPS localization system. To achieve this, the project optimized the localization pipeline by selecting and integrating effective feature extraction, matching, and planar transformation estimation techniques. The system was designed to estimate GPS locations using only prior telemetry data and image features, ensuring radial errors remained below 10% of the UAV's displacement from a reference image. Additionally, the system was engineered for real-time operation, achieving response times under two seconds following GPS signal loss, thereby allowing pilots to maintain effective manual control without significant delays. Comprehensive validation across diverse environments confirmed the system's generalizability and robustness, eliminating the need for environment-specific parameter tuning.

The implementation successfully met all outlined objectives. The image-based localization system demonstrated high accuracy and real-time performance across multiple datasets, including challenging sparse-feature environments. It maintained robust accuracy despite practical limitations such as resolution constraints, varying terrain feature densities, and fluctuating environmental conditions. The system's resilience was further evidenced under low-light conditions and scenarios with limited pixel overlap, ensuring reliable navigation data delivery even in adverse situations.

The successful development and validation of this system signify a substantial advancement in UAV navigation technology. By providing a reliable alternative to GPS, the system enhances UAV operational safety and effectiveness in both military and civilian applications.

However, the project also identified several areas that warrant further research and development. The system was unable to consistently balance the quantity (stability) and quality (outliers) of feature detection across diverse feature landscapes, highlighting

the need for more sophisticated algorithms or machine-learning techniques. Additionally, sparse-feature environments in low-light conditions continue to pose significant challenges. Furthermore, this study did not account for distortions resulting from scale and perspective changes, necessitating the application and testing of homography transformations to address these distortions effectively.

In summary, this project successfully developed a versatile and effective image-based GPS localization system for UAVs, addressing the critical need for reliable navigation in GPS-denied environments. The system has proven to be practical for real-world applications, demonstrating robust performance across a wide range of operational conditions. With further enhancements to improve accuracy and overcome existing limitations, this localization system holds substantial promise for enhancing UAV navigation reliability in both military and civilian sectors, supporting a broader scope of missions with increased safety.

Chapter 7

Future Work

This study establishes a foundational framework for image-based UAV navigation. While effective, several advancements could enhance the accuracy, practicality, and efficiency of the system in future implementations.

7.0.1. Multi-Image Weighted Inference System

Future implementations could improve location accuracy by using a weighted estimation approach based on multiple reference images, rather than relying on a single image. Aggregating data from multiple images would provide a more reliable position estimate, mitigating the effects of individual image distortions or outliers. Although this method was not tested in this study due to time constraints, it has the potential to enhance the robustness of location estimation with minimal increase in computational load.

7.0.2. Non-Planar Stereo Matching

For navigating complex terrains with non-planar ground surfaces, incorporating a non-planar stereo matching approach could improve depth perception and accuracy. By accounting for variations in surface elevation, the UAV could more accurately approximate altitude, enhancing its location estimates over diverse landscapes.

7.0.3. Integration with Mapping and Reference Images

Integrating reference images with up-to-date mapping data could eliminate the need for prior image capture and a fixed return path to base. This approach would enable the UAV to navigate freely within a mapped region without accumulating positional drift. Using prior flight data or collaboratively obtained maps would allow the system to maintain accuracy over a larger operational area, providing extreme reliability in the event of GPS signal loss.

7.0.4. Distortion Corrections

Although this system did not account for the effects of roll and pitch changes or altitude variations, these distortions are common in real-world applications. Future implementations could integrate correction mechanisms for these factors to enhance accuracy. Solutions such as OpenCV's homography estimators, offer built-in methods for compensating for such distortions, allowing easy to implement, efficient real-time correction on compatible devices.

7.0.5. Implementation on a Single Board Computer (SBC)

To improve system performance and enable more complex computations, future work could explore implementing the system on a higher-performance Single Board Computer (SBC) with enhanced processing power. A more capable SBC would support the integration of multiple estimation techniques, facilitating more sophisticated data fusion to enhance overall accuracy.

7.0.6. Higher Resolution Imaging

Incorporating higher-resolution imaging in future systems could significantly improve feature extraction and matching accuracy by capturing more detailed visual information. However, as resolution increases, so do the processing and memory demands. To maintain real-time performance, careful management of computational load and memory usage will be essential.

In summary, the proposed future work addresses a range of enhancements, from improving altitude and position inference to optimizing system performance and accuracy through advanced image processing and storage management techniques. These improvements would ensure a more robust, adaptable, and scalable navigation system suitable for various UAV applications.

Bibliography

- [1] M. Weiss, “Personal communication,” RF Engineer, 2024, june 26.
- [2] Geotab Team, “What is gps?” 2024, accessed: October 17, 2024. [Online]. Available: <https://www.geotab.com/blog/what-is-gps/>
- [3] J. Khalil, “Gnss spoofing threatens airline safety, alarming pilots and aviation officials,” *GPS World*, 2024. [Online]. Available: <https://www.gpsworld.com/gnss-spoofing-threatens-airline-safety-alarming-pilots-and-aviation-officials/>
- [4] M. J. Wright, L. Anastassiou, C. Mishra, J. M. Davies, A. M. Phillips, S. Maskell, and J. F. Ralph, “Cold atom inertial sensors for navigation applications,” *Frontiers in Physics*, vol. 10, p. 994459, 2022.
- [5] Y. Zhuang, X. Sun, Y. Li, J. Huai, L. Hua, X. Yang, X. Cao, P. Zhang, Y. Cao, L. Qi, J. Yang, N. El-Bendary, N. El-Sheimy, J. Thompson, and R. Chen, “Multi-sensor integrated navigation/positioning systems using data fusion: From analytics-based to learning-based approaches,” *Information Fusion*, vol. 95, pp. 62–90, 2023.
- [6] J. Brewer, “Line of sight requirements for reliable low-power rf communication,” White Paper, ATEK Access Technologies, 2024, senior Electrical Design Engineer.
- [7] Scout Aerial Australia, “An introduction to lidar technology and its applications,” 2024, accessed: October 17, 2024. [Online]. Available: <https://www.scoutaerial.com.au/article-lidar/>
- [8] X. Zhang, F. Xiao, M. Zheng, and Z. Xie, “Uav image matching from handcrafted to deep local features,” *European Journal of Remote Sensing*, vol. 57, no. 1, 2024.
- [9] OpenCV, *ORB (Oriented FAST and Rotated BRIEF) - Feature Detection*, 2022. [Online]. Available: https://docs.opencv.org/4.x/d1/d89/tutorial_py_orb.html
- [10] ——, *cv::AKAZE Class Reference*, 2018. [Online]. Available: https://docsopencv.org/3.4/d8/d30/classcv_1_1AKAZE.html
- [11] Rpaultrat, “Efficient neural feature detector and descriptor,” GitHub, 2023, accessed: 2024-10-24. [Online]. Available: <https://github.com/rpaultrat/SuperPoint>
- [12] Cvg, “Lightglue: Local feature matching at light speed,” GitHub, 2023, accessed: 2024-10-24. [Online]. Available: <https://github.com/cvg/LightGlue>

- [13] OpenCV, *cv::BFMatcher Class Reference*, 2022. [Online]. Available: https://docs.opencv.org/4.x/d3/da1/classcv_1_1BFMatcher.html
- [14] ——, *FLANN-based Matcher - Feature Matching with FLANN*, 2018. [Online]. Available: https://docs.opencv.org/3.4/d5/d6f/tutorial_feature_flann_matcher.html
- [15] ——, *Feature Matching in OpenCV*, 2018. [Online]. Available: https://docs.opencv.org/3.4/dc/dc3/tutorial_py_matcher.html
- [16] Z. Rezvani, S. Shekarizeh, and M. Sabokrou, “Global-local processing in convolutional neural networks,” 2023. [Online]. Available: <https://doi.org/10.48550/arxiv.2306.08336>
- [17] V. Sharma, “Image-template matching using cross-correlation,” *Medium*, 2022, accessed: 2024-10-26. [Online]. Available: <https://vipin-sharma.medium.com/image-template-matching-using-cross-correlation-2f2b8e59f254>
- [18] A. Rosebrock, “How-to: 3 ways to compare histograms using opencv and python,” *PyImageSearch*, 2014, accessed: 2024-10-26. [Online]. Available: <https://pyimagesearch.com/2014/07/14/3-ways-compare-histograms-using-opencv-python/>
- [19] ——, “Image difference with opencv and python,” *PyImageSearch*, 2017, accessed: 2024-10-26. [Online]. Available: <https://pyimagesearch.com/2017/06/19/image-difference-with-opencv-and-python/>
- [20] T. Nguyen, S. W. Chen, S. S. Shivakumar, C. J. Taylor, and V. Kumar, “Unsupervised deep homography: A fast and robust homography estimation model,” *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2346–2353, 2018.
- [21] OpenCV, “Warp affine transformation,” 2024, accessed: 2024-10-24. [Online]. Available: https://docs.opencv.org/4.x/d4/d61/tutorial_warp_affine.html
- [22] O. Sorkine-Hornung and M. Rabinovich, “Least-squares rigid motion using svd,” 2017, january 16, 2017.
- [23] OpenCV, “Homography estimation,” 2024, accessed: 2024-10-24. [Online]. Available: https://docs.opencv.org/4.x/d9/dab/tutorial_homography.html
- [24] R. B. Fisher, “The ransac (random sample consensus) algorithm,” 2002, accessed: 2024-10-28. [Online]. Available: https://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/FISHER/RANSAC/
- [25] D. S. Farin, “Automatic video segmentation employing object/camera modeling,” Ph.D. dissertation, Technische Universiteit Eindhoven, 2005.

- [26] J.-W. Bian, W.-Y. Lin, Y. Liu, L. Zhang, S.-K. Yeung, M.-M. Cheng, and I. Reid, “Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence,” *International Journal of Computer Vision*, vol. 128, June 2020.

Appendix A

Project Planning Schedule

This is an appendix.

Appendix B

Outcomes Compliance

Student's Graduate Attribute (GA) Achievement Plan

GA 1: Problem Solving

The navigation system I developed effectively addresses complex engineering and software problems to achieve a robust and accurate image-based localization system. By leveraging mathematical and statistical methods, I crafted a system that accurately identifies, matches, and estimates the transformations of the UAV even in challenging GPS-denied environments. Throughout the project, I prioritized methods that could reliably extract features and estimate transformations in real time, allowing the UAV to autonomously navigate with precision. I rigorously tested, through over 10,000 lines of code, various algorithms and parameters to ensure high accuracy, outlier rejection, and robustness across different environmental conditions. Furthermore, throughout the project, I faced numerous hurdles regarding issues in the accuracy and runtime of the solution; I had to perform extensive debugging, address prior assumptions, and explore different ways to solve problems. For instance, normalizing images to the global space might seem straightforward, but after noting errors, I realized I had to find their source. Rotational loss was found to be minimized when alignment was applied between images only. Additionally, I used literature to identify innovative approaches that could be adapted to overcome specific challenges related to feature scarcity, low-light conditions, and image warping.

GA 2: Application of Scientific and Engineering Knowledge

My project extensively utilized computer vision theories, image processing techniques, and pre-trained machine learning models to develop an effective navigation system. I applied various algorithms grounded in computer vision to accurately detect and analyze landscape changes. Leveraging principles of homographic transformations, I was able to account for the UAV's motion and orientation without relying on external GPS signals. Furthermore, I incorporated statistical methods to quantify model output and to improve accuracy by systematically testing different outlier rejection methods. This application of scientific and engineering knowledge was instrumental in developing a system that could

operate reliably across multiple terrains, validating the project's approach to image-based navigation.

GA 3: Engineering Design

The engineering design process began with creating a high-level flow diagram that outlined each stage in the image processing pipeline. This design incorporated multiple resources and techniques and integrated them effectively with robust design measures to handle inaccuracies. I utilized my knowledge of engineering principles gained during the course of my studies in design skills and software development to design a modular, robust system capable of adapting to different environments. This structured approach allowed the system to maintain stability and precision, even in scenarios with sparse image features or environmental variations, ultimately fulfilling the project's goals of reliable UAV navigation.

GA 4: Investigations, Experiments, and Data Analysis

To support the development and validation of the system, I collected extensive data across various environments and under different conditions. I systematically tested the navigation system using diverse datasets and conditions, such as sparse desert landscapes and low-light repetitive terrain, to identify the suitability, generalizability, accuracy, and robustness of different techniques. Using statistical analysis, I was able to quantify the variance in results, leading to the subsequent discovery of faults in logic and techniques, as well as understanding the true performance of the system. The ability to leverage systematic investigation and experimentation, as well as my knowledge of data analytics, allowed me to implement, test, and optimize various methods with limited guidance. The project required ongoing experimentation with different feature detection and matching algorithms, including neural network-based methods, to fine-tune the system's performance. The large array of parameters needed to be tuned meant I could not simply test every possible solution; I needed to systematically investigate from principles and then experiment based on those to ensure efficiency and not forgo a better alternative.

GA 5: Engineering Methods, Skills, and Tools (including IT)

I employed a wide range of tools and engineering methods, including Python libraries such as OpenCV and advanced frameworks available on GitHub for image processing. Furthermore, this project involved rigorous research into the best and most current solutions for all stages of the pipeline. By using the latest methods, I was able to demonstrate the applicability of this system in difficult conditions, something made possible by recent advancements in computer vision techniques. This project also required rigorous adherence to software engineering practices, ensuring modularity, code optimization, and thorough

error handling. I adopted computational techniques to evaluate the system's accuracy and performance, allowing for the assessment and validation of methods in a controlled environment. By following best practices and leveraging tools to address issues in real time, I was able to identify areas for improvement and continuously optimize the system's accuracy and response.

GA 6: Professional and Technical Communication

Throughout the project, I demonstrated strong communication skills by preparing both written reports and oral presentations. These presentations communicated complex technical concepts in a clear and concise manner to stakeholders and supervisors. Additionally, I documented my methods and findings in detail, ensuring reproducibility and clarity for future work. The project's outcomes were communicated effectively in both technical and non-technical formats, highlighting my ability to translate technical progress and results into comprehensible information.

GA 8: Individual Work

As the lead on this project, I took primary responsibility for all aspects of the development and implementation process. This included problem-solving, design, and optimization, as well as the testing and analysis of results. My role required me to work independently to research, develop, and refine solutions, utilizing various resources and overcoming challenges without external assistance. This project reflects my commitment to delivering high-quality work and achieving the outlined objectives independently.

GA 9: Independent Learning Ability

This project demanded a high level of independent learning to solve complex navigation challenges in GPS-denied environments. I actively sought out research materials, consulted technical documentation, and applied knowledge from relevant scientific literature. I expanded my understanding of image processing techniques, statistical analysis, and feature extraction methods independently, equipping myself with the skills necessary to achieve project goals without requiring extensive guidance. This process underscored my ability to learn independently and apply new knowledge effectively to overcome project-specific challenges.