



UNIVERSITEIT•STELLENBOSCH•UNIVERSITY
jou kennisvennoot • your knowledge partner

A Critical Analysis of Design Flaws in the Death Star

Luke Skywalker
99652154

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 O. W. Kenobi

October 2099



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

English

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.

Afrikaans

Onbemande Lugvaartuie (UAV's) is afhanklik van die Globale Positieseringsstelsel (GPS) vir presiese navigasie, maar kwesbaarhede soos storings en spogging kan missie-sukses in gevaar stel. Hierdie projek het 'n beeld-gebaseerde GPS-lokaliseringsstelsel ontwikkel om betroubare UAV-navigasie in GPS-ontkende omgewings moontlik te maak. Die stelsel optimaliseer die lokalisasie-pyplyn deur gevorderde kenmerkuittrekking, ooreenstemming en planêre transformasie tegnieke te integreer, en bereik radiale lokalisasie-foute onder 10% van die UAV se verskuiwing vanaf 'n verwysingsbeeld. Ontwerp vir regstreekse operasie, reageer die stelsel binne twee sekondes na GPS-siggewigsverlies, wat vlieërs in staat stel om effektiewe beheer te behou. Omvattende toetsing oor diverse omgewings het die stelsel se hoë akkuraatheid, robuustheid en generaliseerbaarheid getoon sonder om omgewings-spesifieke afstelling nodig te hê. Die stelsel het uitnemend gepresteer in lae-lig en lae-ooreenkoms toestande, wat betroubare navigasiedata lewer in verskeie scenario's. Hierdie beeld-gebaseerde lokalisasie bied 'n praktiese en betroubare alternatief vir GPS, wat UAV-operasionele veiligheid en doeltreffendheid in beide militêre en siviele toepassings

verbeter. Met verdere verbeterings is die stelsel gereed vir werklike implementering, wat veiliger en meer betroubare UAV-missies belowe.

Contents

Declaration	i
Abstract	ii
List of Figures	vii
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. Summary of Work	3
1.5. Scope	3
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	8
2.2.4. LIDAR and Radar for Ground Mapping	8
2.2.5. Image-Based Homography Navigation	8
2.3. Background	10
2.3.1. Planar Transforms	10
2.3.2. Feature Detectors	11
2.3.3. Feature Matching	13
2.3.4. Image Similarity Computation	14
2.3.5. Planar Transformation Estimators	16
2.3.6. Optimization Techniques	17

3. System Design	20
3.1. System Pipeline	20
3.1.1. Pipeline Stages	20
3.2. Dynamic Methods and Techniques	23
3.3. Rotational Normalization Strategies	24
3.3.1. Empirical Validation	25
3.3.2. Method Conclusion	25
3.4. Testing Shortlist	26
3.5. Conclusion	26
4. Comparison of Methods	27
4.1. Testing Setup	27
4.1.1. Datasets	27
4.1.2. Testing Structure	29
4.1.3. Parameters	30
4.2. Feature Detectors	30
4.2.1. Accuracy	31
4.2.2. Final Selection of Feature Detectors	32
4.3. Local Feature Matchers	32
4.3.1. Testing Overview	32
4.3.2. Accuracy Evaluation	32
4.3.3. Runtime Evaluation	33
4.3.4. Robustness Under Varying Detector Thresholds	33
4.3.5. Final Selection of Local Feature Matcher	34
4.4. Rotational Estimators	35
4.4.1. Accuracy Evaluation	35
4.4.2. Runtime Evaluation	36
4.4.3. Robustness Testing	36
4.4.4. Final Selection of Rotational Estimator	36
4.5. Image Similarity Estimators	37
4.5.1. Accuracy Evaluation	37
4.5.2. Runtime Evaluation	38
4.5.3. Robustness to Rotational Offsets	38
4.5.4. Improvement Technique	39
4.5.5. Final Selection of Global Matching Technique	39
4.6. Translational Estimators	39
4.6.1. Accuracy Evaluation	40
4.6.2. Runtime Evaluation	40
4.6.3. Robustness Testing	41

4.6.4. Final Choice of Translational Estimator	42
4.7. Optimization Techniques	42
4.7.1. Planar Transform Outlier Rejection Method	42
4.7.2. Lowe's Ratio Test	43
4.7.3. Parallel Flow Filtering	44
4.7.4. Empirical Testing Only	44
5. Results	46
5.1. Detailed Performance Analysis	46
5.1.1. Key Performance Metrics	46
5.1.2. Detailed Performance Metrics	47
5.1.3. Dataset Performance Analysis	48
5.1.4. Error Distribution Map	49
5.1.5. An Account of Variance Between and Error in Datasets	49
5.2. Robustness Results	51
5.2.1. Mutual Information Results	51
5.2.2. Low-Light Testing	53
5.3. Conclusion	56
5.4. Engineering Summary	56
5.5. Future Work	58
5.5.1. Multi-Image Weighted Inference System	58
5.5.2. Non-Planar Stereo Matching	58
5.5.3. Integration with Mapping and Reference Images	58
5.5.4. Distortion Corrections	58
5.5.5. Implementation on a Single Board Computer (SBC)	59
5.5.6. Higher Resolution Imaging	59
A. Project Planning Schedule	60
B. Outcomes Compliance	61

List of Figures

3.1. High-Level Flow of the System	20
3.2. Loss comparison between pre-normalization and post-normalization methods.	25
4.1. Examples of the CITY1 and CITY2 Datasets	29
4.2. Example of the ROCKY Dataset	29
4.3. Example of the DESERT Dataset	29
4.4. Example of the AMAZON Dataset	29
4.5. Overview of Datasets	29
4.6. Divergence in RMSE GPS Error Between FLANN and BFMatcher Across Keypoint Targets	34
5.1. Violin Plot of Pixel Deviations in X and Y Directions	46
5.2. Accuracy Metrics for Each Dataset	47
5.3. Runtime Metrics for Each Stage of the Pipeline	48
5.4. Plot of distribution of radial errors in each dataset	48
5.5. Heatmap of Pixel Deviations in X and Y Directions	49
5.6. Accuracy vs Mutual Information	52
5.7. Daytime Image of Cape Town	54
5.8. Evening Image of Cape Town	54
5.9. Night Image of Cape Town	55

List of Tables

4.1.	RMSE for Various Local Detectors (in meters)	31
4.2.	Runtime for Various Local Detectors (in seconds)	31
4.3.	RMSE GPS Accuracy for BFMatcher and FLANN (in meters)	33
4.4.	Runtime Comparison for BFMatcher and FLANN (in seconds)	33
4.5.	RMSE Comparison Across Datasets for Different Rotational Estimators . . .	35
4.6.	Runtime Comparison Across Datasets for Different Rotational Estimators . .	36
4.7.	RMSE Comparison Across Datasets for Various Global Matching Techniques (in meters)	37
4.8.	Runtime Comparison Across Global Matching Techniques (in seconds)	38
4.9.	RMSE (GPS Error) and Percentage Change with 10-degree Rotational Offset	38
4.10.	RMSE GPS Error Comparison Across Datasets for Different Translation Methods	40
4.11.	Runtime Comparison Across Datasets for Different Translation Methods (in seconds)	41
4.12.	RMSE Variance Across Datasets for Different Translation Methods (Ro- bustness Test)	41
4.13.	RMSE Comparison Across Datasets for LMEDS and RANSAC	42
4.14.	Runtime Comparison Across Datasets for LMEDS and RANSAC	43
5.1.	Nighttime and Evening Performance RMSE GPS for Various Datasets . . .	55

Nomenclature

Variables and Functions

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.
$RMSE$	Root Mean Square Error, representing the average magnitude of localization (GPS) errors.
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

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.
Features	Unique keypoints in an image along with their descriptors, used for identifying and matching points across different images for localization.
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 images.

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 [?]. 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.

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. 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.

1.5. 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:

- **Camera Orientation:** The UAV's downward-facing camera is assumed to remain perfectly aligned downward throughout all operations.
- **Image Quality:** It is assumed that the captured images are of sufficient resolution and free from significant distortions such as warping, blurring, or obstruction by clouds.
- **Operational Environment:** The UAV is expected to operate over predominantly static land surfaces with minimal dynamic objects, ensuring a stable environment for image-based localization.
- **Altitude Consistency:** The UAV maintains a constant altitude, negating the need for scale estimation in localization calculations.
- **Path Loss:** The UAV uses complementary, odometry-based systems to find its path back in the event of GPS signal loss, ensuring continuous operation.

- **Image Overlap:** Images used for localization have an overlap exceeding 60%, ensuring sufficient reference information.
- **Method Choice:** The selected methods are based on comprehensive research and empirical testing, with the understanding that not all potential methods could be included within the scope of this study.
- **Ground Planarity:** The terrain is assumed to be significantly planar, minimizing perspective distortions that could affect localization accuracy.
- **UAV Stability:** The UAV and its camera system remain level with the ground, eliminating pitch and roll variations.
- **Training Requirements:** The localization methods employed do not require pre-training or live training, ensuring operational simplicity and efficiency.
- **Data Provisioning:** The project used data from Google Earth due to delays in receiving the agreed-upon prerequisite data, with a minor impact on accuracy acknowledged due to the use of estimated headings.
- **System Output:** The system provides an estimated GPS location and heading, intended for manual UAV control by pilots rather than automated control systems.
- **Optimization:** The system is optimized for real-time performance and generalizability. However, the focus is on testing the viability of the system rather than achieving the absolute best performance.

While these scope limitations streamline the project, they do not preclude the potential for future enhancements, such as incorporating scale estimation techniques or expanding the system's robustness to dynamic environmental conditions.

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, evaluates the viability of the developed system, and discusses its potential impact on UAV navigation.

- **Chapter 7: Future Work**

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

References

1. Sathyamoorthy, D., et al. (2020). Evaluation of the Vulnerabilities of Unmanned Aerial Vehicles (UAVs) to Global Positioning System (GPS) Jamming and Spoofing. *Defense Science & Technology Journal*, 33, 334-343.
2. Rusnák, M. & Vásárhelyi, J. (2023). A review of using visual odometry methods in autonomous UAV navigation in GPS-denied environment. *Acta Universitatis Sapientiae Electrical and Mechanical Engineering*, 15, 14-32. <https://doi.org/10.2478/auseme-2023-0002>
3. Karab, E., Prasad, S., & Shahato, M. (2017). Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images. *Acta Polytechnica Hungarica*, 14(6), 183-206.
4. Luglio-Miguel, F., Iores, G., & Salazar, S. & Lozanc, R. (2014). Dubins path generation for a fixed-wing UAV. *Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS)*, 339-345.

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 [?].

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 [?].

Such interference has direct implications for national security, as border surveillance and reconnaissance missions are heavily dependent on accurate GPS data [?]. 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. This method offers exceptionally accurate acceleration and rotation data, theoretically minimizing navigation drift [?].

Despite its high precision, quantum sensing navigation presents significant challenges. The primary drawbacks include:

- **Cost and Size:** Cold atom sensors are expensive and bulky, making them impractical for most UAV applications where cost-efficiency and compactness are crucial.
- **Power Consumption:** These sensors require substantial power, limiting their feasibility for battery-operated UAVs.
- **Operational Limitations:** Although aviation-grade systems incorporating cold atom sensors can reduce drift by approximately half, they still suffer from significant drift over extended periods. Additionally, sensor misalignment and noise, especially when gyroscope noise surpasses thresholds like 0.05, 0.10, or 0.20 rad/s at higher frequencies, can exacerbate errors [?].

These factors collectively restrict the adoption of quantum sensors in UAV navigation systems, particularly those that are cost-sensitive and size-constrained.

2.2.2. Odometry-Based Solutions

Odometry-based navigation estimates UAV displacement by tracking wheel rotation or utilizing accelerometer readings. This method provides a direct measurement of movement, allowing for real-time tracking of the UAV's position.

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 [?]. Specifically:

- **Error Accumulation:** Small discrepancies in sensor measurements can accumulate, resulting in substantial positional errors over extended flights.
- **Environmental Factors:** Variations in terrain, wheel slippage, or unexpected accelerations can further degrade odometry accuracy.

For UAVs, 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 infrastructure limits its use in remote areas, and the Earth's curvature further restricts line-of-sight over long distances or at lower altitudes, reducing signal strength and accuracy [?].

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 are advantageous for precise altitude measurements and mapping, allowing for accurate location inference. However, LIDAR and radar systems emit detectable signals, which may be undesirable for applications requiring minimal detectability. Additionally, these systems can be resource-intensive and exhibit limitations in dense environments, where interference and signal scattering may impair accuracy and operational range [?].

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. In this application, 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:

- **Cost and Size:** Unlike quantum sensing systems, image-based navigation relies on standard imaging hardware, which is typically more cost-effective and compact, making it suitable for UAV applications where size and budget constraints are critical.
- **Stealth:** Unlike LIDAR and radar, which emit detectable signals, image-based navigation does not "announce" the UAV's presence, allowing for lower detectability in sensitive or security-focused applications.
- **Power Consumption:** Image processing can be optimized for low power consumption compared to the high energy demands of cold atom sensors, thereby enhancing the feasibility for battery-operated UAVs.
- **Cumulative Drift and Error Accumulation:** While odometry-based systems suffer from cumulative drift over time, the usage of this device specifically to navigate back to base ensures constant correction and is thereby not affected by drift.
- **Environmental Adaptability:** The proposed system is designed to handle environmental variability by incorporating robust feature detection and noise filtering, ensuring reliable performance across diverse operational contexts.
- **Real-Time Processing:** The implementation focuses on real-time image processing and transformation estimation, enabling immediate adjustments to the UAV's navigation path and enhancing overall responsiveness.

Much research exists in the field of homography, including its application to aerial imagery, as demonstrated in [?], 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 to 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, being robust to distortions, generalizing across various environments, and attaining 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. Planar Transforms

Planar transformations are mathematical operations that map points from one plane to another, accounting for changes in position, orientation, and perspective. In the context of UAV navigation, the output of these transformations are directly used to infer the UAV's location and heading based on prior reference images and telemetry data.

Two primary approaches exist for estimating planar transforms: direct methods and feature-based methods, each with distinct strengths and limitations [?].

Direct Methods

Direct, or global, methods utilize the entire image to estimate planar transforms by comparing pixel intensities and minimizing the differences through optimization techniques such as gradient descent [?]. Key characteristics include:

Direct Methods

Direct methods, also known as global methods, utilize the entire image to estimate planar transformations by comparing pixel intensities and minimizing differences through optimization techniques such as gradient descent [?]. Key characteristics of direct methods include:

- **Robustness for Small Viewpoint Changes:** These methods perform well when dealing with minor adjustments in viewpoint, maintaining accuracy during incremental movements.

- **Sensitivity to Noise:** Direct methods are highly susceptible to noise from defective pixels or environmental factors, which can degrade transformation accuracy.
- **Assumption of Smooth Movements:** They rely on the premise of smooth, incremental UAV movements. Large changes in rotation, perspective, or translation, such as those encountered when a UAV turns back to base, render direct methods unreliable [?].
- **Similarity Comparison:** Direct methods evaluate image similarity based on the global context, which is beneficial for understanding overall similarity between images.
- **Computational Stability:** Although direct methods involve pixel-wise comparisons that may be seem computationally expensive, the relatively simple and single layer computations allow for efficient and stable computations across different operational conditions [?]. In contrast, local methods require multiple stages of processing and become significantly slower if the number of features becomes too large.

Feature-Based Methods

In contrast, feature-based methods extract and match keypoints from both reference and real-time images, often employing multiple stages of outlier removal [?]. Key advantages of feature-based methods include:

- **Suitability for Large Transformations:** Feature-based methods excel in scenarios involving significant viewpoint changes and rotations, as they focus on distinctive features rather than pixel-wise comparisons.
- **Computational Efficiency:** By targeting salient features, these methods can perform multiple stages of processing without having to analyze every pixel, thereby enhancing computational efficiency. However, care must be taken to ensure that the number of features does not become excessive, as this can lead to performance degradation.
- **Robustness to Distortions:** Feature-based methods are resilient to various forms of distortion and environmental variability, ensuring reliable performance across different operational conditions.

2.3.2. 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.

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 [?].

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 [?].

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 [?]. 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 [?]. Despite their high accuracy, SuperPoint and LightGlue are computationally intensive, necessitating GPU acceleration for real-time applications.

2.3.3. 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.

Types of Feature Matchers

Two primary feature matchers are employed in this system: the **Brute-Force Matcher (BFMatcher)** and the **Fast Library for Approximate Nearest Neighbours (FLANN)**. Additionally, **LightGlue**, a machine-learning-based matcher, is utilized for its enhanced accuracy despite its higher computational demands.

- **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 [?].
- **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 [?]

- **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 [?]. The enhanced accuracy comes at the cost of increased computational demands, requiring GPU acceleration for optimal performance.

Search Techniques

The search technique determines which potential matches are retained for further processing. Various methods, including **Radius Search**, **Vanilla Matching**, and **K-Nearest Neighbours (KNN) Matching**, are explored to control the number and quality of matches.

- **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 [?].
- **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 [?].
- **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 [?].

2.3.4. 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

Global matching techniques evaluate image similarity based on the overall context of the images. Although these methods are less precise for pose inference compared to local feature matchers, their ability to capture the holistic visual information makes them effective for similarity comparisons. Additionally, their computational efficiency is advantageous for processing large image datasets. These methods necessitate rotational alignment of images to eliminate bias introduced by orientation discrepancies [?].

Several global matching techniques are employed to score image similarity:

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 [?].

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 [?].

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 [?].

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.

2.3.5. Planar Transformation Estimators

Planar transformation estimators are pivotal for calculating the UAV's changes in position and orientation between consecutive images. 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 [?]. 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:

1. Computing the weighted centroids of both point sets.
2. Centering the points by subtracting their respective centroids.
3. Calculating the covariance matrix of the centered points.
4. Performing SVD on the covariance matrix to derive the rotation matrix.
5. 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 [?].

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 [?].

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 [?]. 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.

2.3.6. Optimization Techniques

Optimizing parameter sets is crucial 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 [?]. The process involves:

1. Randomly selecting a minimal subset of point pairs.
2. Estimating the transformation model (e.g., affine or homography) based on the selected subset.
3. Determining the number of inliers that fit the estimated model within a predefined threshold.
4. 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, as it focuses 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 [?].

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 [?]. This technique involves:

1. Analyzing the image to identify regions with high feature density.
2. Selecting keypoints located at local maxima within these dense regions.
3. 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. This method is effective in outlier rejection, leading to more reliable transformation estimations.

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 [?]. The procedure involves:

1. For each keypoint match, calculating the ratio of the distance of the best match to that of the second-best match.
2. Retaining the match if this ratio is below a predefined threshold (commonly 0.75).

A lower ratio indicates that the best match is significantly better than the alternatives, thereby increasing the likelihood of the match being correct. This test is essential for ensuring that only the most reliable correspondences are used in subsequent processing stages, thereby enhancing the overall accuracy of the transformation estimation.

Parallel Flow Filtering

Parallel Flow Filtering ensures consistency in feature matches by aligning them with the average motion flow resulting from the UAV's rotation and translation. In typical UAV movements, feature flow is linear and parallel to the direction of motion, assuming no scaling as per the system's scope. This technique involves:

1. Calculating an average motion vector by combining mean and median motion vectors derived from the keypoint matches.
2. Filtering out matches that deviate beyond a specified angular threshold from the average motion vector.

By removing matches that do not conform to the expected motion pattern, Parallel Flow Filtering ensures that only consistent and reliable matches contribute to the transformation estimation. This significantly improves the overall accuracy and reliability of the navigation system by maintaining alignment with the UAV's actual motion dynamics.

n-Match Thresholding

This method involves setting a threshold that allows only a specific number of matches with the smallest descriptor distances to be retained. This approach ensures that only the most confident matches are considered, reducing the impact of weaker or less reliable matches.

Absolute Thresholding of Matches

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.

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.

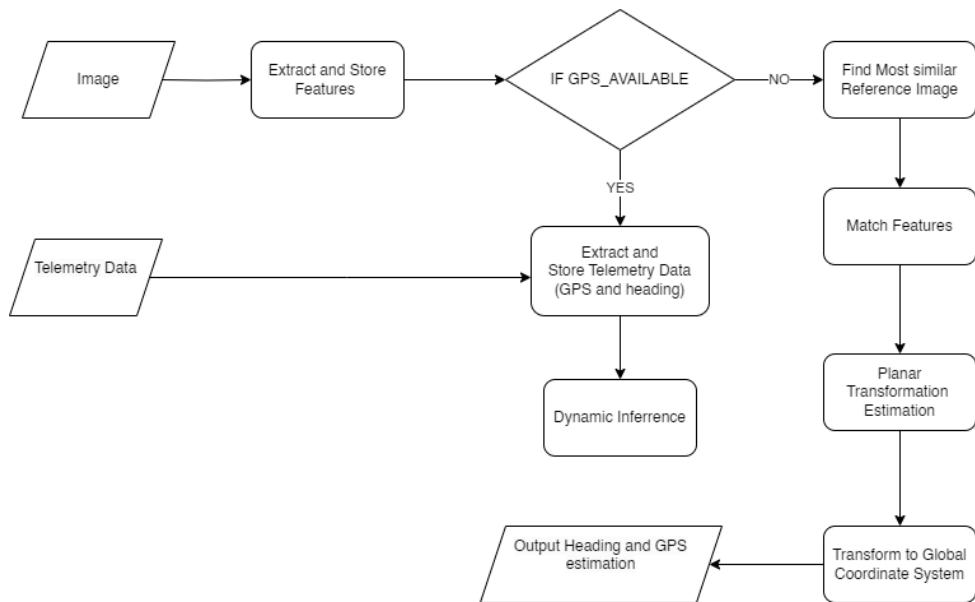


Figure 3.1: High-Level Flow of the System

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, n-Match thresholding, and absolute thresholding 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.
 - (d) **Metres to Global Coordinate System (Relative):** Translates the metre-based changes to relative changes in longitude and latitude using the following

equations:

$$\Delta\text{Longitude} = \frac{\Delta\text{Metres}}{111320 \times \cos(\text{Latitude})} \quad (3.1)$$

$$\Delta\text{Latitude} = \frac{\Delta\text{Metres}}{111320} \quad (3.2)$$

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.

- (e) **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 essential for aligning images and translating estimations accurately in image-based UAV navigation systems. Two primary strategies were considered for normalizing rotations during the flight back to the base. The first involves applying rotational normalization during image capture, which aligns both images to a global North-East (NE) reference space before keypoint detection. The second method avoids normalization at the point of capture, aligning the images relative to one another and applying normalization only after translation estimation.

The key difference between these two methods lies in their respective losses caused by image rotation, due to the fixed canvas size during normalization. When an image is rotated, parts of the image are lost or moved off-canvas. The loss function for the first approach, where images are normalized independently, is proportional to:

$$\text{Loss}_{\text{pre}} \propto |\sin(|\Delta_1|) + \sin(|\Delta_2|)| \quad (3.3)$$

In the second approach, where images are aligned relative to each other before normalization is applied, the loss function is:

$$\text{Loss}_{\text{post}} \propto |\sin(|\Delta_2 - \Delta_1|)| \quad (3.4)$$

Here, the loss depends on the relative difference in rotation between the two images, reducing the potential for compounded information loss. In the given application, during

the return flight to base, the UAV will undergo an initial 180-degree rotation relative to the outbound path with an expected maximum tolerance of ± 20 degrees as the UAV attempts to as accurately as possible follow the path back to base. Given this constraint, the loss functions of both approaches are proportional to the following diagram:

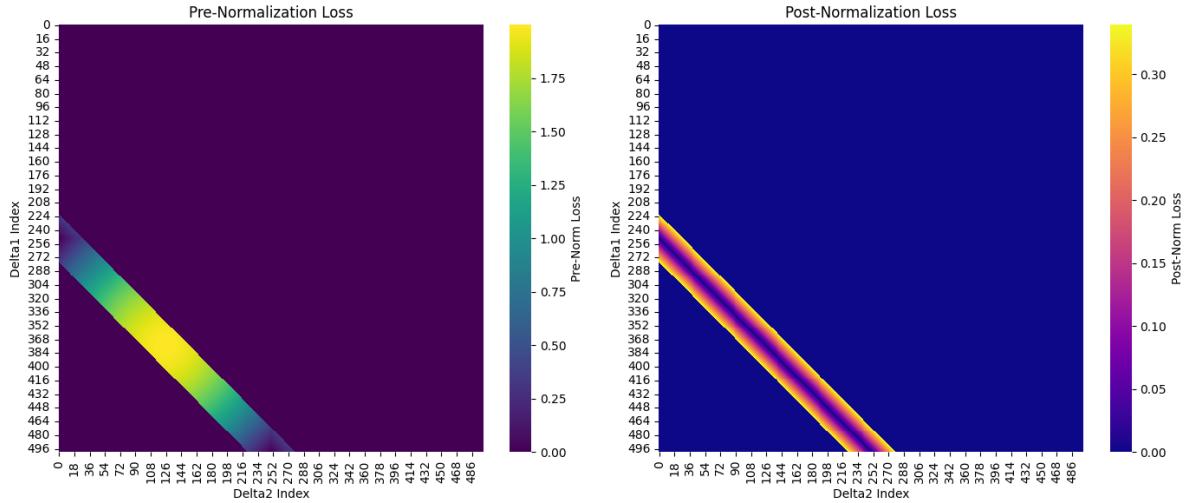


Figure 3.2: Loss comparison between pre-normalization and post-normalization methods.

As seen in Figure 3.2, the pre-normalization method results in losses exceeding 2.0 due to the compounding effect of treating both images independently. In contrast, the post-normalization method, which works with relative rotation, consistently yields lower losses, with a maximum loss of approximately 0.35 under these typical UAV conditions. Further, the losses in the post-normalization method consists of almost all non-mutual information. Therefore, post-normalization is expected to provide more accurate translation estimations due to better retention of keypoint information.

3.3.1. Empirical Validation

In empirical tests, these results were confirmed, with post-normalization consistently providing higher accuracy due to better retention of keypoint information. The effects were most significant in the DESERT dataset, where the sparsity of keypoints made the dataset particularly sensitive to information loss during rotation.

3.3.2. Method Conclusion

Given the UAV's typical flight pattern and the need to minimize information loss, post-normalization is applied. In this approach, images are first aligned, followed by translation estimation. Finally, the translation vector is normalized to the global NE coordinate space for consistent navigation.

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.
- **Transformation Estimation:** Homography, Affine, Partial Affine (Rigid) transformations using OpenCV, and SVD-based rigid transformations for rotation and translation estimations. Rotational and Translational Estimators are tested separately as optimal detectors and parameters may vary for each.
- **Global Image Similarity Measures:** SSIM, histogram matching, local retrofit, and cross-correlation.
- **Optimization Techniques:** Parallel Flow 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.

xxx - SIFT could not generalize

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

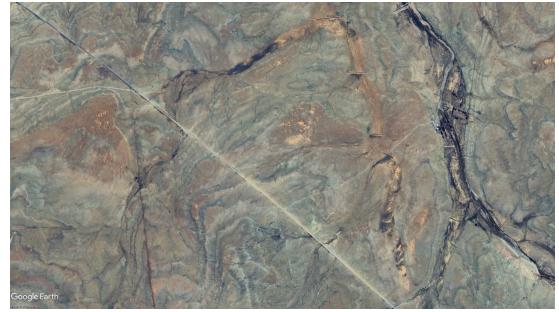


Figure 4.2: Example of the ROCKY Dataset



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, including feature extraction, local feature matching, rotational estimation, image similarity computation, translational estimation, and optimization techniques, 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.
- **Accuracy:** Evaluated using the Root Mean Square Error (RMSE) of GPS estimations, accuracy assessments focus on the end error. This approach is preferable because intermediate outputs do not always provide clear indicators of quality. For instance, the number of keypoints can be misleading; without context, it's difficult to determine if they represent good features or excessive noise. Outputs propagate through the pipeline, meaning any errors or poor choices are ultimately reflected in the final GPS error, making it more effective to measure accuracy at the end.
- **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 primary objective of this evaluation is to identify the most suitable detector for a UAV navigation system tasked with estimating rotational and translational transformations. Key performance metrics, including RMSE (Root Mean Square Error), runtime, and robustness across multiple datasets, were considered to assess each detector’s effectiveness and suitability.

The evaluation focused on three main criteria:

- **Accuracy (RMSE in GPS):** Measures the precision of transformation estimates.
- **Runtime:** Assesses the computational efficiency of each detector.
- **Robustness:** Evaluates consistency and reliability across different datasets.

The results from these tests informed the selection of the most appropriate detector and threshold parameters for the crude and dense layer of keypoints, as well as whether the optimal detector for rotational estimation is the same as for translational estimation.

4.2.1. Accuracy

Table 4.1 presents the RMSE values in meters for each feature detector applied to the precise translation estimation stage. AKAZE, utilizing dynamic keypoint targeting, demonstrated the highest accuracy across all datasets while maintaining reasonable runtime. ORB exhibited respectable accuracy but was consistently outperformed by AKAZE. SuperPoint recorded the highest RMSE values, particularly in challenging datasets such as ROCKY and AMAZON, indicating its limited generalizability across diverse environments.

Table 4.1: RMSE for Various Local Detectors (in meters)

Method	CITY1	CITY2	ROCKY	DESERT	AMAZON
ORB	70.89	10.56	43.32	80.66	46.39
Dynamic AKAZE (3000 keypoints)	66.80	6.91	22.48	33.07	39.27
SuperPoint	72.66	15.80	114.93	31.60	329.19

Runtime and Efficiency

Table 4.2 illustrates the runtime (in seconds) for each feature detector across different datasets. ORB proved to be the most efficient detector, making it suitable for applications requiring fast processing. Dynamic AKAZE, while exhibiting longer runtimes, achieved a balance between efficiency and accuracy. SuperPoint demonstrated the longest runtimes across all datasets, highlighting its limited applicability for time-sensitive applications unless optimized with GPU acceleration.

Table 4.2: Runtime for Various Local Detectors (in seconds)

Method	CITY1	CITY2	ROCKY	DESERT	AMAZON
ORB	75.25	70.77	111.64	58.29	75.34
Dynamic AKAZE (3000 keypoints)	112.18	104.16	127.86	108.86	119.76
SuperPoint	338.21	292.11	307.93	277.73	291.01

Robustness

In terms of robustness, AKAZE exhibited the highest consistency in accuracy across different datasets. ORB also performed well but showed greater variability in accuracy across datasets compared to AKAZE. SuperPoint, however, demonstrated significant performance variability, indicating lower robustness across diverse environments.

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. ORB's robustness and consistent runtime performance across datasets make it an ideal choice for processing multiple images.
- **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.

These selections ensure that each stage of the UAV navigation system leverages the most appropriate feature detector, optimizing both accuracy and computational performance.

4.3. Local Feature Matchers

This section evaluates two prominent local matchers, BFMatcher and FLANN, within the context of a UAV navigation system. The primary objective of this evaluation is to determine the most efficient and robust matcher in terms of accuracy, runtime, and consistency across diverse datasets. These matchers play a crucial role in the system's ability to accurately estimate rotational and translational transformations by effectively pairing detected feature points.

4.3.1. Testing Overview

The local matchers were assessed under various conditions to evaluate their performance concerning accuracy (RMSE in GPS), runtime, and robustness. The evaluation was meticulously designed to understand how each matcher handles noisy keypoints, limited post-match filtering, and varying detector thresholds. The goal was to identify which matcher offers the most suitable balance between match quality and computational efficiency, thereby ensuring real-time applicability in UAV navigation.

4.3.2. Accuracy Evaluation

Table 4.3 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.

Table 4.3: RMSE GPS Accuracy for BFMatcher and FLANN (in meters)

Matcher	CITY1	CITY2	ROCKY	DESERT	AMAZON
FLANN	56.59	4.70	14.63	71.20	32.35
BFMatcher	53.89	3.79	16.36	68.64	33.24

Observations:

- **BFMatcher Accuracy:** BFMatcher demonstrated slightly superior accuracy in CITY1 and CITY2 compared to FLANN. However, the improvement was not substantial enough to outweigh its increased computational demands.
- **FLANN Accuracy:** FLANN maintained competitive accuracy levels, trailing BFMatcher only marginally in most datasets. Notably, FLANN exhibited strong performance in the ROCKY dataset, underscoring its reliability in challenging environments.

4.3.3. Runtime Evaluation

Table 4.4 illustrates the runtime (in seconds) for BFMatcher and FLANN across different datasets. The results clearly show that FLANN consistently outperforms BFMatcher in terms of speed, with significantly lower execution times across all datasets.

Table 4.4: Runtime Comparison for BFMatcher and FLANN (in seconds)

Matcher	CITY1	CITY2	ROCKY	DESERT	AMAZON
FLANN	42.72	41.61	41.96	43.42	53.62
BFMatcher	203.49	228.59	46.33	52.59	88.95

Observations:

- **FLANN Efficiency:** FLANN exhibited significantly faster runtimes across all datasets, particularly excelling in CITY1 and CITY2 where it completed tasks in less than a quarter of the time required by BFMatcher.
- **BFMatcher Runtime:** The exhaustive matching approach of BFMatcher resulted in considerably higher runtimes, rendering it less suitable for real-time applications where speed is critical.

4.3.4. Robustness Under Varying Detector Thresholds

Figure 4.6 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.

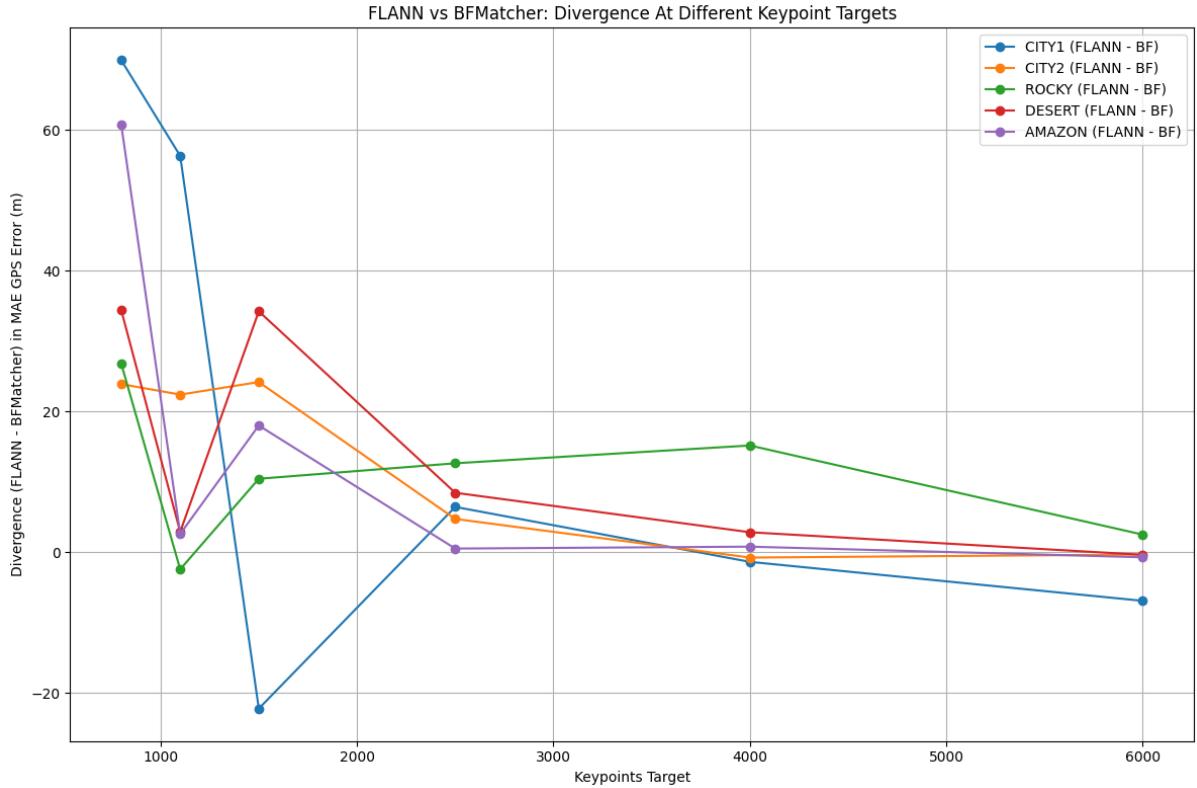


Figure 4.6: Divergence in RMSE GPS Error Between FLANN and BFMatcher Across Keypoint Targets

Observations:

- **Convergence:** Both matchers demonstrated similar accuracy levels as the number of keypoints increased, with performance becoming nearly identical beyond 2500 keypoints. Given that feature detectors are typically configured with a keypoint target of at least 3000, both matchers are expected to perform comparably in standard operational conditions.
- **Outliers:** Although BFMatcher generally outperforms FLANN, there are instances where FLANN achieves better accuracy. This can occur due to FLANN's approximate matching, which may preserve valid matches that BFMatcher might discard using Lowe's ratio thresholding.
- **Scalability:** FLANN's runtime scalability with increasing keypoint counts was superior to that of BFMatcher, making FLANN a more scalable solution for larger datasets.

4.3.5. 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.

- **Overall Choice:** FLANN is selected as the primary local feature matcher for the UAV navigation system, balancing high performance and efficiency across diverse operational conditions.

These selections ensure that the UAV navigation system leverages the most appropriate local matcher, optimizing both accuracy and computational performance to achieve reliable and efficient navigation.

4.4. Rotational Estimators

xxx fix This section evaluates the performance of four rotational estimation methods: Partial Affine 2D, Affine 2D, Homography, and Rigid Via SVD. The objective is to identify the most suitable method for UAV navigation by comparing accuracy, runtime, and robustness across various datasets. These estimators are critical for accurately determining the rotational transformations required for precise navigation and alignment of UAV imagery.

4.4.1. Accuracy Evaluation

The Rigid transforms, SVD and Partial Affine 2D provided the most accurate and consistent results across datasets. They were subsequently tested in conjunction to see if their benefits could be added. However, the SVD only rigid transform performed marginally better across the datasets. Ultimately, the OpenCV methods using outlier rejection sometimes helped improve the estimate, but their use of static thresholds and increased degrees of freedom rendered them less able to maintain this accuracy across datasets. Table 4.5 summarizes the RMSE values (in meters) for the given methods.

xxx - just use rigid its too slow otherwise

Table 4.5: RMSE Comparison Across Datasets for Different Rotational Estimators

Method	CITY1	CITY2	ROCKY	DESERT	AMAZON
Partial Affine 2D	70.18	7.54	27.98	44.94	46.98
Affine 2D	81.22	7.80	22.10	43.11	52.56
Homography	76.25	7.16	24.40	48.56	48.04
Rigid Affine + SVD	51.72	5.98	21.18	36.27	30.84
Rigid SVD	51.74	6.16	19.03	36.48	31.57

4.4.2. Runtime Evaluation

The runtime results below relate mostly to the degrees of freedom in the method, with the simplest rigid transforms performing the best. The SVD only based method performs the fastest due to its lack of iterative optimization and singular method. Table 4.6 presents the runtime (in seconds) for each rotational estimator across the five datasets.

Table 4.6: Runtime Comparison Across Datasets for Different Rotational Estimators

Method	CITY1	CITY2	ROCKY	DESERT	AMAZON
Partial Affine 2D	56.76	52.20	69.44	56.40	50.34
Affine 2D	65.27	54.55	76.16	71.52	58.16
Homography	68.79	65.11	78.92	69.97	58.27
Rigid Affine + SVD	90.74	77.45	81.19	90.74	94.25
Rigid SVD	83.33	61.47	77.94	71.27	74.77

4.4.3. Robustness Testing

Robustness testing assessed each method's sensitivity to parameter variations, including Lowe's ratio, RANSAC thresholds, and keypoint quantity. The generalization to datasets was spoken to in accuracy testing. 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 excellent at being robust against parameter variations, maintaining consistent performance across different settings.
- **Best Robustness:** The Rigid via SVD only transform exhibited the highest robustness across all parameters, especially at low keypoint counts, maintaining consistent performance under varying conditions.

4.4.4. Final Selection of Rotational Estimator

Based on the comprehensive evaluation of accuracy, runtime, and robustness, the rigid transform by SVD only emerged as the most suitable rotational estimator for the UAV navigation system. It demonstrated the lowest combined normalized RMSE across all datasets, the fastest runtime, and high robustness.

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. This section evaluates different global matching techniques across multiple datasets to identify the most suitable method for UAV navigation based on accuracy, runtime, and robustness.

4.5.1. Accuracy 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. Root Mean Square Error (RMSE) was utilized to assess the GPS estimation accuracy of each global matching technique across the five datasets. Table 4.7 summarizes the RMSE values (in meters) for Local Retrofit, Cross Correlation, Histogram, and SSIM.

Table 4.7: RMSE Comparison Across Datasets for Various Global Matching Techniques (in meters)

Global Matching Technique	CITY1	CITY2	ROCKY	DESERT	AMAZON
Local Retrofit	46.56	7.37	17.32	128.43	34.54
Cross Correlation	49.88	5.12	15.77	26.84	31.50
Histogram	46.35	5.12	15.77	27.28	31.50
SSIM	50.03	6.16	15.77	27.28	31.44

Observations:

- **Histogram Accuracy:** The Histogram technique consistently achieved the lowest RMSE across most datasets, particularly excelling in CITY1, CITY2, and ROCKY.
- **Cross Correlation Accuracy:** Cross Correlation closely followed Histogram in accuracy, demonstrating strong performance in CITY2 and DESERT.
- **SSIM Accuracy:** SSIM maintained comparable accuracy to Histogram and Cross Correlation but exhibited slightly higher errors in CITY1 and DESERT.
- **Local Retrofit Accuracy:** The Local Retrofit method recorded the highest RMSE values, especially in the DESERT dataset, indicating poor generalizability and higher complexity. Consequently, it was excluded from further analysis.

4.5.2. Runtime Evaluation

Table 4.8 compares the computational efficiency of each global matching technique across the five datasets.

Table 4.8: Runtime Comparison Across Global Matching Techniques (in seconds)

Global Matching Technique	CITY1	CITY2	ROCKY	DESERT	AMAZON
Local Retrofit	84.56	76.34	70.22	63.47	90.69
Cross Correlation	59.62	56.67	54.14	59.90	65.92
Histogram	57.06	56.33	56.68	51.98	59.09
SSIM	91.60	77.99	83.31	83.08	114.07

Observations:

- Histogram Efficiency:** The Histogram technique was the most efficient in terms of runtime, outperforming all other methods consistently across all datasets.
- Cross Correlation Efficiency:** Cross Correlation followed closely behind Histogram, offering slightly higher runtimes but still maintaining high computational efficiency.
- SSIM Efficiency:** SSIM exhibited the longest runtimes, particularly in the AMAZON and DESERT datasets, rendering it less suitable for real-time applications.
- Local Retrofit Efficiency:** The Local Retrofit method had the highest runtimes and was deemed non-viable due to its excessive computational cost and poor accuracy.

4.5.3. Robustness to Rotational Offsets

Robustness testing evaluated each global matching technique's sensitivity to a 10-degree rotational offset. All matchers that made it this far saw no change in choice combination below a 5-degree offset. The impact on RMSE GPS error and the percentage change in error were recorded to assess each method's stability under rotational misalignments.

Table 4.9: RMSE (GPS Error) and Percentage Change with 10-degree Rotational Offset

Method	Metric	CITY1	CITY2	ROCKY	DESERT	AMAZON
Cross Correlation	RMSE	54.81	5.34	17.55	32.51	31.52
	% Change	5.89%	4.29%	11.54%	21.92%	0.06%
Histogram	RMSE	56.17	4.45	16.83	37.20	36.35
	% Change	20.93%	-13.09%	6.73%	36.73%	15.38%
SSIM	RMSE	55.93	5.53	16.25	41.01	31.67
	% Change	11.79%	-10.23%	3.04%	50.43%	0.73%

Observations:

- **Cross Correlation Robustness:** Cross Correlation exhibited the lowest percentage change in GPS error under a 10-degree rotational offset, indicating strong robustness.
- **Histogram Robustness:** While Histogram demonstrated competitive RMSE values, it showed significant deviations in the DESERT and AMAZON datasets when subjected to rotational offsets.
- **SSIM Robustness:** SSIM displayed higher error rates and greater variability, particularly in the DESERT dataset, making it less robust to significant rotational misalignments.

4.5.4. Improvement Technique

An additional method, the rigid transform using Singular Value Decomposition (SVD), was initially evaluated as a primary global matching technique. However, it performed significantly worse than the other methods and was subsequently excluded from standalone consideration. However, when the rigid transform was applied after other estimation methods, it consistently enhanced overall accuracy. This improvement is likely due to the rigid transform's ability to make minor, precise corrections once the point cloud is largely aligned by other methods, despite its initial inefficiency in handling large misalignments.

4.5.5. 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, particularly under large rotational offsets. Although Cross Correlation also demonstrated strong performance, Histogram's marginally better accuracy and runtime efficiency make it the preferred choice.

4.6. Translational Estimators

This section evaluates the performance of various translational estimation methods for UAV navigation. The methods assessed include Phase Correlation, Rigid Transform, Affine Transform with RANSAC, Homography Transform, and Partial Affine 2D. Each method is evaluated based on accuracy, runtime, and robustness across multiple datasets to identify the most effective option for UAV navigation.

4.6.1. Accuracy Evaluation

Root Mean Square Error (RMSE) was utilized to assess the GPS estimation accuracy of each translational estimation method across five datasets. Table 4.10 summarizes the RMSE values (in meters) for each method.

Table 4.10: RMSE GPS Error Comparison Across Datasets for Different Translation Methods

Method	CITY1	CITY2	ROCKY	DESERT	AMAZON
Phase Corr	1437.85	1349.16	629.82	1263.15	1121.81
Rigid Transform	64.31	7.42	21.82	29.01	38.52
Affine Transform	127.77	88.41	57.01	70.96	118.77
Homography	168.35	92.54	58.69	120.15	242.97
Partial Affine 2D	64.11	6.34	19.07	29.94	40.08

Observations:

- **Partial Affine 2D** achieved the lowest RMSE, establishing it as the most accurate estimator across all datasets. Its superior performance compared to the direct algebraic **Rigid Transform** is attributed to the utilization of RANSAC for effective outlier rejection.
- Both **Homography** and **Affine Transform** methods exhibited significantly higher RMSE values than Partial Affine 2D and Rigid Transform. This increase is primarily due to their greater degrees of freedom and inherent complexity.
- **Phase Correlation** recorded the highest RMSE values, indicating lower accuracy relative to the other methods. This reduced performance is likely due to its heightened sensitivity to noise, as it processes the entire image context.

4.6.2. Runtime Evaluation

Table 4.11 presents the runtime (in seconds) for each translational estimation method across the five datasets.

Table 4.11: Runtime Comparison Across Datasets for Different Translation Methods (in seconds)

Method	CITY1	CITY2	ROCKY	DESERT	AMAZON
Phase Corr	1437.85	1349.16	629.82	1263.15	1121.81
Rigid Transform	89.05	99.74	103.46	104.84	87.43
Affine Transform	127.77	88.41	57.01	70.96	118.77
Homography	168.35	92.54	58.69	120.15	242.97
Partial Affine 2D	94.62	82.77	85.94	72.43	78.33

Observations:

- **Partial Affine 2D** demonstrated the fastest runtime among the translational methods, making it the most computationally efficient. It was followed by the **Rigid Transform**.
- **Phase Correlation** exhibited significantly longer runtimes, making it less viable for real-time UAV applications.

4.6.3. Robustness Testing

Robustness was assessed by evaluating each method's sensitivity to parameter variations, such as keypoint quantity and threshold changes. Table 4.12 highlights the standard deviation of RMSE across datasets to indicate consistency.

Table 4.12: RMSE Variance Across Datasets for Different Translation Methods (Robustness Test)

Method	CITY1	CITY2	ROCKY	DESERT	AMAZON
Phase Corr	111.35	76.54	55.69	110.15	101.97
Rigid Transform	64.61	7.53	21.14	31.31	37.96
Affine Transform	127.77	88.41	57.01	70.96	118.77
Homography	168.35	92.54	58.69	120.15	242.97
Partial Affine 2D	64.11	6.34	19.07	29.94	40.08

Observations:

- **Partial Affine 2D** and **Rigid Transform** exhibited the lowest variance, indicating strong robustness to parameter changes.
- **Homography** showed the most variability and performed inconsistently under different conditions.

4.6.4. Final Choice of Translational Estimator

Based on the evaluation of accuracy, runtime, and robustness, **Partial Affine 2D** was selected as the most suitable translational estimator for the UAV navigation system. It demonstrated the highest accuracy, fastest runtime, and strong robustness across all datasets.

- **Primary Translational Estimator:** **Partial Affine 2D** is chosen for its optimal performance in minimizing GPS error, computational efficiency, and consistent robustness across diverse operational conditions.

4.7. Optimization Techniques

This section outlines the selected optimization methods employed to improve the performance of the UAV navigation system. The optimization techniques focus primarily on filtering matches between images. Since the accuracy of transformation techniques is largely governed by the quantity of matches that adhere to the correct model, optimizing these matches is essential to the overall performance of the system.

4.7.1. Planar Transform Outlier Rejection Method

Two outlier rejection methods, LMEDS (Least Median of Squares) and RANSAC (Random Sample Consensus), were evaluated for planar transform estimation. These techniques are applied during transformation estimation when inferring rotation or translation using an OpenCV method. Although applied during the transformation phase, they still function to filter matches, fitting an iteratively adjusted model. The performance comparison is summarized in Tables 4.13 and 4.14.

Table 4.13: RMSE Comparison Across Datasets for LMEDS and RANSAC

Dataset	RMSE GPS Error (LMEDS)	RMSE GPS Error (RANSAC)
CITY1	51.78	59.57
CITY2	6.16	5.84
ROCKY	19.04	18.59
DESERT	36.48	35.90
AMAZON	31.57	34.48

Table 4.14: Runtime Comparison Across Datasets for LMEDS and RANSAC

Dataset	Runtime (LMEDS)	Runtime (RANSAC)
CITY1	68.72	126.14
CITY2	52.20	64.65
ROCKY	69.44	126.97
DESERT	56.40	126.97
AMAZON	50.34	64.65

LMEDS marginally outperforms **RANSAC** in terms of mean accuracy across datasets, achieving a higher mean accuracy despite RANSAC performing better in three out of five instances. Additionally, LMEDS demonstrates significantly faster runtimes across all datasets, as depicted in Table 4.14. This superior balance of accuracy and efficiency makes LMEDS the preferred method for planar transform outlier rejection. The default parameters for LMEDS were utilized, as they provided sufficiently optimal performance across the tested datasets.

4.7.2. 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. This method required careful tuning of several parameters to balance match quality and quantity:

- **Initial Lowe's Ratio:** Configured between 0.5 and 0.7 to strike a balance between generalization (ensuring a sufficient number of matches under various conditions) and match quality (preventing an overload of poor matches).
- **Step Size:** Ranged from 0.025 to 0.1 to balance the number of iterations and the quality of matches.
- **Maximum Iterations:** Limited to 10-30 iterations to prevent excessive inclusion of poor matches, especially in sparse datasets.
- **Minimum Matches:** Set between 2000 and 3000 to ensure a sufficient number of reliable matches without compromising match quality.

This dynamic thresholding strategy minimizes the impact of Lowe's ratio by adjusting the threshold only when necessary, thereby maintaining stability and ensuring a robust number of matches across varying datasets.

4.7.3. Parallel Flow Filtering

Parallel Flow Filtering enhances the consistency of feature matches by aligning them with the average motion flow resulting from the UAV's rotation and translation. The following parameters were optimized based on empirical testing:

- **Average Flow Measurement:** Utilized the median exclusively, leveraging its robustness against outliers.
- **Tolerance:** Set between 1.5 and 4 degrees to balance noise stability and the exclusion of false positives.
- **Internal Angle Tolerance Scaling:** Applied an exponential scaling based on the internal angle between image pairs using the equation:

$$T_s = T_i \cdot \left(1 - \left(\frac{\alpha}{\alpha_m}\right)\right)^2 \quad (4.1)$$

where:

- T_s is the scaled tolerance.
- T_i is the initial tolerance.
- α is the internal angle.
- α_m is the maximum possible internal angle.

This scaling ensures that the tolerance adapts appropriately to the degree of image rotation, effectively modelling the exponentially increasing likelihood of outliers in image pairs as their internal angles increase. This model works on the basis of being stricter when the likelihood of outliers is higher. This was empirically found to occur when the internal angle between images is higher.

4.7.4. Empirical Testing Only

n-Match Thresholding

Empirical tests revealed that the optimal number of matches varies significantly across datasets and to other parameters, making it unreliable. This approach risks either retaining too many false matches or discarding too many good matches, depending on the dataset characteristics. This method was not pursued due to its high sensitivity to the number and quality of keypoints.

Absolute Thresholding of Matches

Empirical tests revealed that this method was largely redundant, as Lowe's ratio test inherently accounts for descriptor distances. Additionally, absolute thresholding led to

large performance variability, either having no meaningful effect or removing too many matches. Given these limitations, absolute thresholding was deemed ineffective and was consequently excluded from the optimization techniques.

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%.

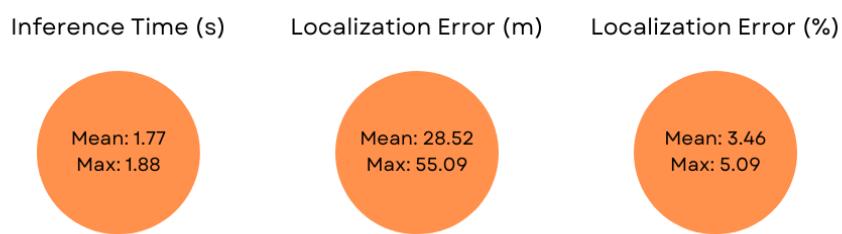


Figure 5.1: Violin Plot of Pixel Deviations in X and Y Directions

5.1.2. 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 5.2 and 5.2.

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.

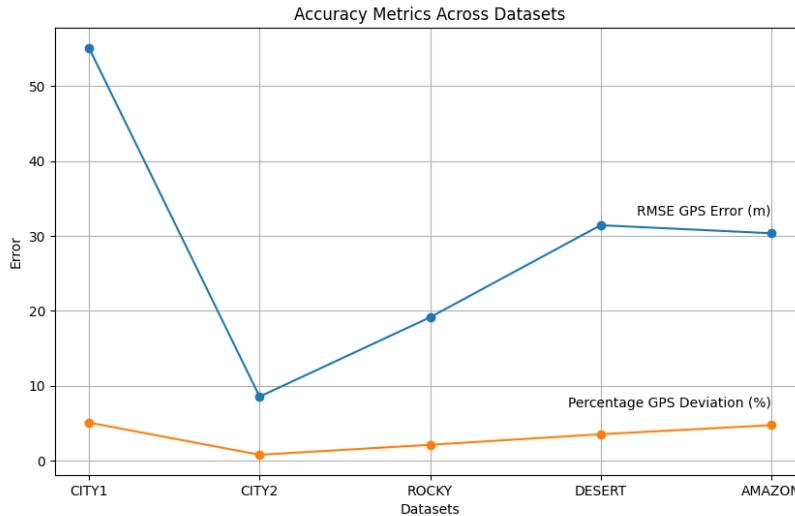


Figure 5.2: Accuracy Metrics for Each Dataset

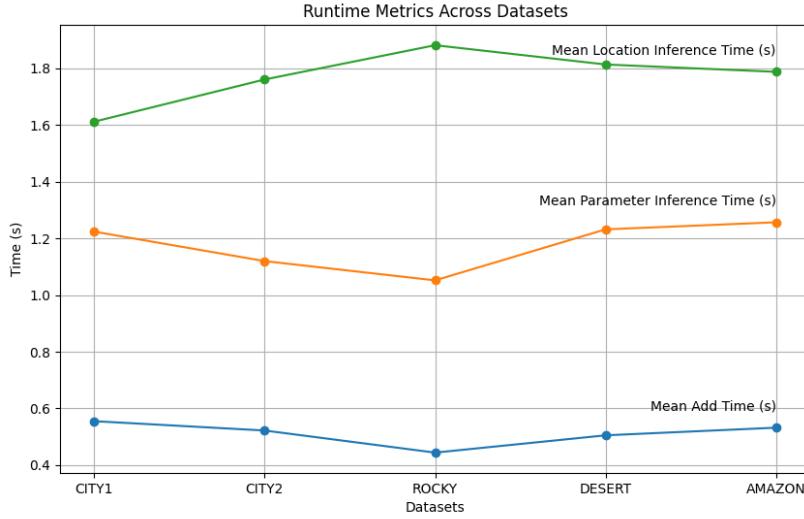


Figure 5.3: Runtime Metrics for Each Stage of the Pipeline

5.1.3. Dataset Performance Analysis

This section utilizes the violin plot, in Figure 5.4, 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.

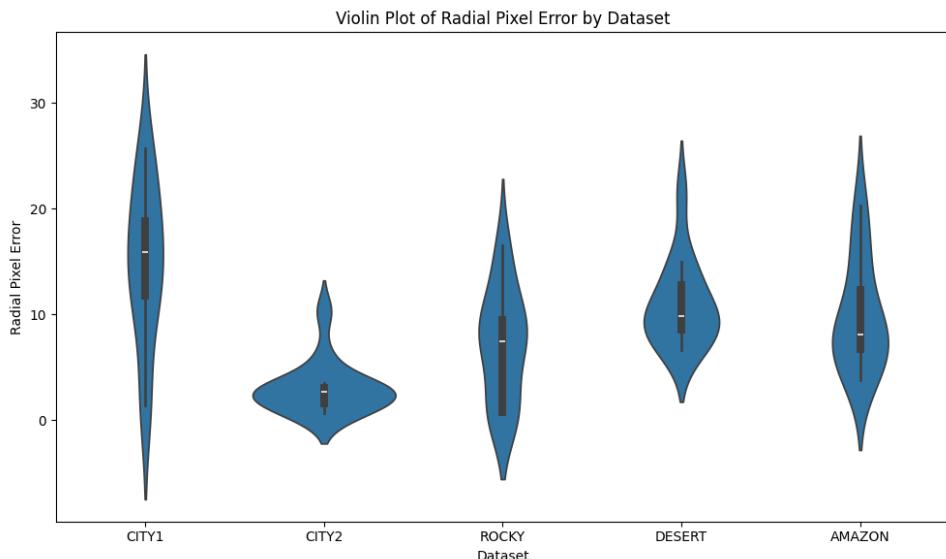


Figure 5.4: Plot of distribution of radial errors in each dataset

5.1.4. Error Distribution Map

The heatmap in Figure 5.5 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.

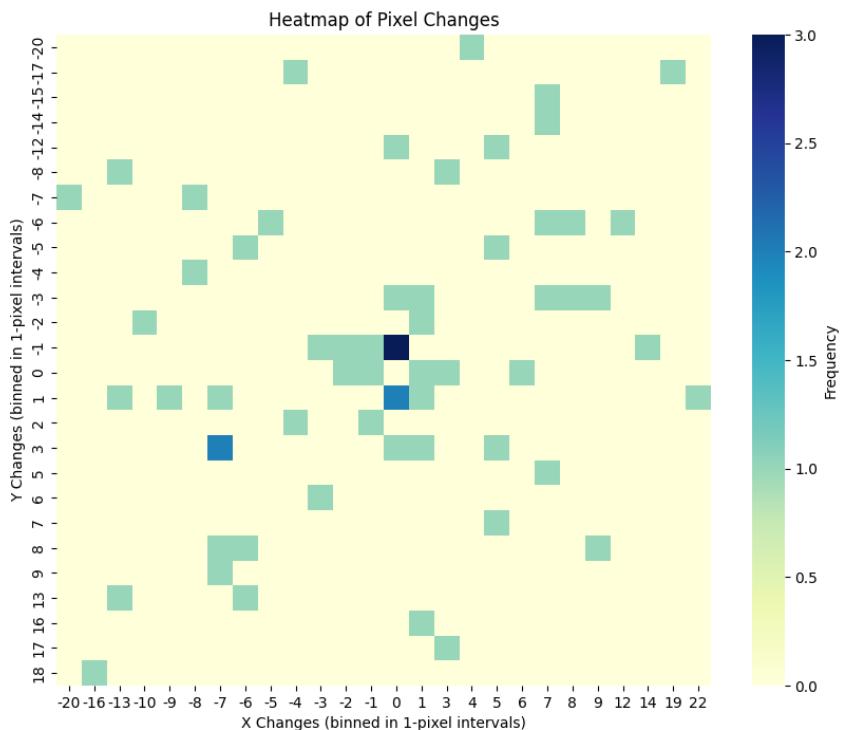


Figure 5.5: Heatmap of Pixel Deviations in X and Y Directions

5.1.5. 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 5.4 and the heatmap in figure 5.5, 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 magnitude and measures of uncertainty, specifically the standard deviation and mean-median difference in match vector magnitudes, were assessed to test whether large variability in the pipeline caused the fluctuations in error. The variability in these vectors was used as a proxy for the uncertainty of the method. This method

tested the likelihood that image-sensitive extraction and matching led to variability across datasets. The results, in terms of correlation coefficients, as well as visual inspection, showed no relationship between uncertainty in the pipeline and error, indicating that the method stability was likely not the primary reason for fluctuating errors in the system. This laid the foundation for considering external or practical reasons for the observed errors.

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 1.5, 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. Robustness Results

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.

Objectives

1. **Determine the Minimum Viable Mutual Information:** Identify the threshold of mutual information required to maintain an RMSE below 10%, ensuring

navigational accuracy without inducing additional drift.

2. **Optimize Runtime Efficiency:** 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

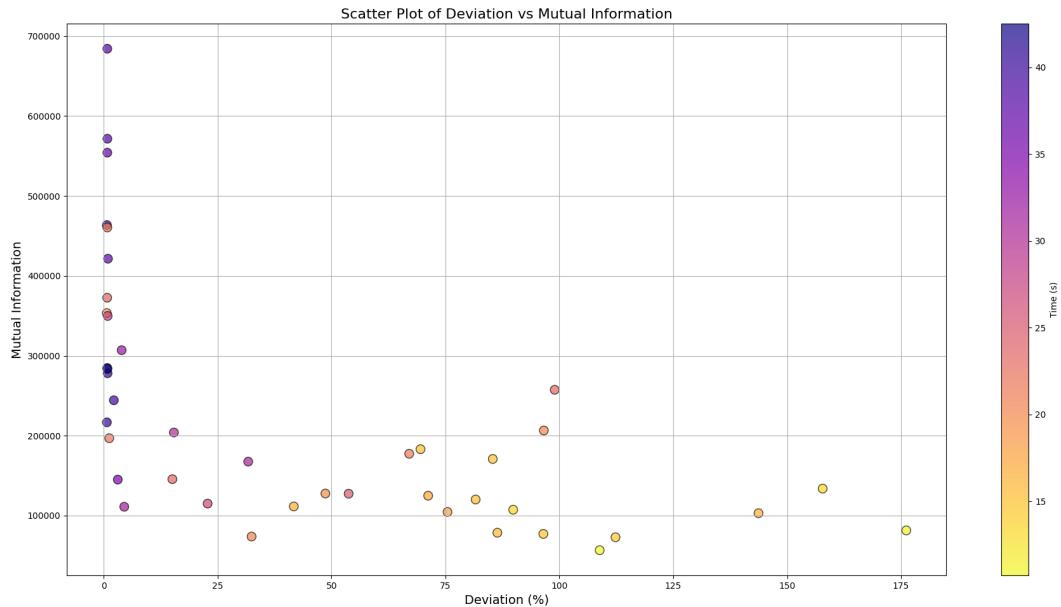


Figure 5.6: Accuracy vs Mutual Information

Observations

- **Critical Threshold Identification:** Figure 5.6 shows that navigational accuracy remains stable until mutual information decreases to approximately 250,000 pixels (15.67% of a 1920x1080 image). Below this threshold, accuracy sharply declines, indicating unreliable estimates around this range.

- **Minimum Mutual Information for Acceptable Accuracy:** To maintain an RMSE below 10%, mutual information must exceed 300,000 pixels (18.81% of a 1920x1080 image).
- **Runtime Optimization through Reduced Mutual Information:** Reducing mutual information by cropping images significantly decreases runtime.
- **Dataset-Specific Considerations:** The identified mutual information thresholds are specific to this dataset and pipeline. However, the results demonstrate the feasibility of balancing mutual information and runtime for effective navigation.

Conclusions

variance in direction. 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 1.80-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. 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.

Objectives

1. **Assess Robustness in Low-Light Conditions:** Evaluate how evening and nighttime lighting conditions affect the accuracy of UAV navigational estimates.

2. **Compare Performance Across Diverse Environments:** Determine the method's effectiveness in different datasets with varying feature densities under low-light scenarios.

Representative examples from the CITY dataset illustrate these conditions:



Figure 5.7: Daytime Image of Cape Town

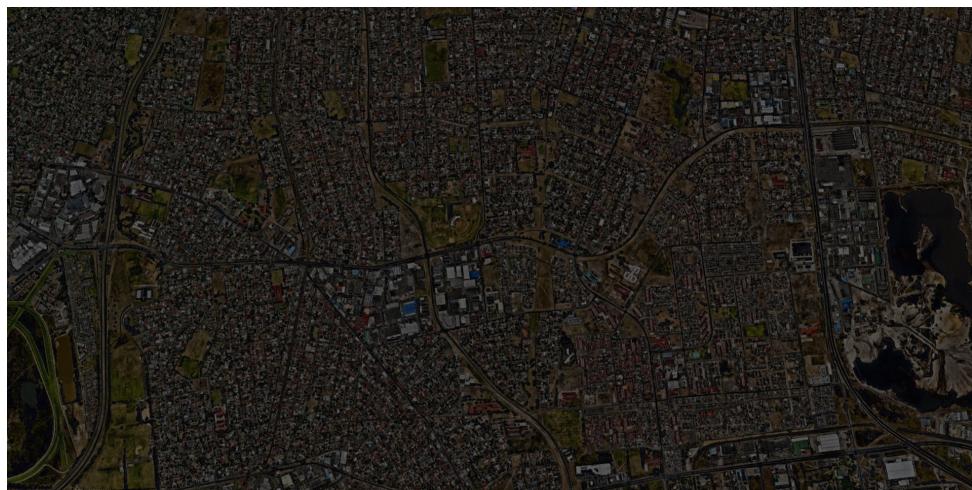


Figure 5.8: Evening Image of Cape Town

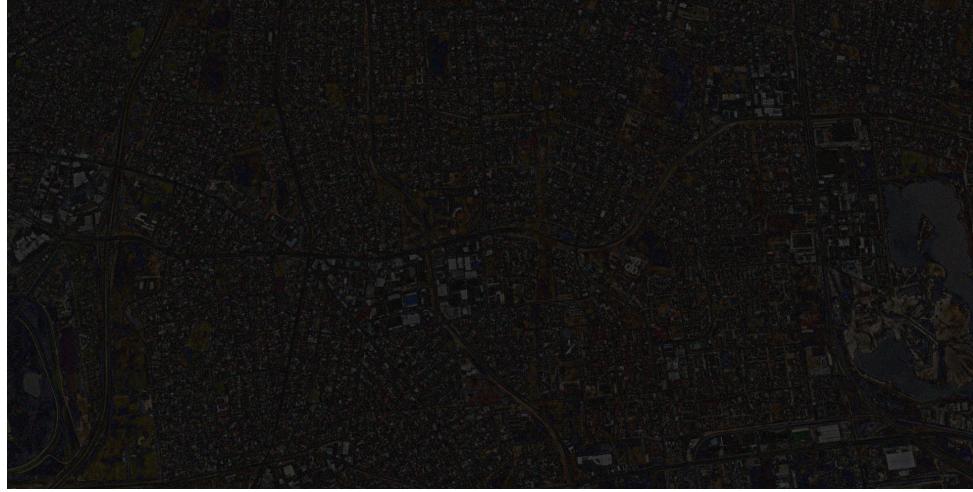


Figure 5.9: Night Image of Cape Town

Evaluation Metrics The Root Mean Square Error (RMSE) of GPS error is used to quantify navigational accuracy under different lighting conditions.

Results

The following tables summarize the accuracy of the navigation method under evening and nighttime conditions for each dataset:

Table 5.1: Nighttime and Evening Performance RMSE GPS for Various Datasets

Dataset	Night RMSE GPS (m)	Evening RMSE GPS (m)
CITY1	68.65	59.41
CITY2	57.77	6.44
ROCKY	91.42	32.14
DESERT	1588.15	156.34
AMAZON	1271.68	32.90

Observations

- Nighttime Performance:** The CITY and ROCKY datasets maintain relatively low RMSE values under nighttime conditions, indicating resilience to severe low-light environments. Conversely, the DESERT and AMAZON datasets exhibit substantial increases in RMSE, primarily due to their sparse and repetitive features, which hinder effective keypoint matching and lead to inaccurate positional estimates.
- Evening Performance:** Evening conditions result in minimal degradation of navigational accuracy for most datasets. The DESERT dataset experiences a moderate error increase of approximately from its initial RMSE of 31.44 m to 156.34 m. This corresponds to a 17.51% RMSE error, which is still

usable. This study suggests that moderate low-light conditions are manageable for environments with sufficient distinct features.

- **Feature Density Impact:** The performance disparity between datasets highlights the importance of feature density in low-light navigation. Environments with rich and distinct landmarks (e.g., CITY and ROCKY) demonstrate greater robustness, whereas feature-poor environments (e.g., DESERT and AMAZON) suffer more significant accuracy declines.

Conclusions

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.

5.3. 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.

5.4. Engineering Summary

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.

5.5. 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.

5.5.1. 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.

5.5.2. 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.

5.5.3. 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.

5.5.4. 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.

5.5.5. 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.

5.5.6. 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.

Appendix A

Project Planning Schedule

This is an appendix.

Appendix B

Outcomes Compliance

This is another appendix.