ABSTRACT

Title of Dissertation: CAMERA-BASED ODOMETRY IN PROXIMITY FLIGHT UNDER ADVERSE VISUAL FACTORS

Alexander Teacu
Master of Science, 2026

Dissertation Directed by: Professor Derek Paley
Department of Aerospace Engineering

This thesis investigates the robustness of vision-based odometry to adverse visual factors in the context of proximity flight of micro air vehicles (MAVs). Vision-based odometry is widely used by MAVs in mapping applications and in GPS-denied environments due to the low cost and small form factor of the required hardware. However, vision-based odometry can fail in scenarios presenting adverse visual factors, resulting in a loss of localization, which is critical for autonomy and mapping. This thesis applies and validates a methodology for capturing data containing adverse visual factors which crucially allows for the isolation of the effect of these adverse visual factors relative to a baseline. This concept is extended to an augmentation-based setup, which leverages a simulation environment to add smoke as a visual factor on top of existing real-world data. Finally, this thesis evaluates the performance of visual-inertial odometry on the adverse visual factor data, of which low-contrast surfaces, distorted reflections, glare, and smoke were found to adversely affect odometry.

CAMERA-BASED ODOMETRY IN PROXIMITY FLIGHT UNDER ADVERSE VISUAL FACTORS

by

Alexander Teacu

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
2026

Advisory Committee:
Professor Derek Paley, Chair/Advisor
Professor Christopher Metzler
Associate Professor Joseph Conroy

Acknowledgments

I would like to thank the many people in my life who have made this thesis possible and given me an unforgettable experience as I pursued my master's.

First, I would like to thank my advisor, Dr. Derek Paley, for giving me the resources and freedom I needed to complete this work as well as for connecting me with many of the other people in this acknowledgment. I don't know where I would be if you hadn't taken me in those years ago at UROC.

I would like to thank my family for standing by me when I needed them most. Thank you mom and dad for all your support from the start. Thank you to my brother, Liviu, and to Alexandra for taking me in when I needed it.

I could not have done this without my wonderful robotics lab manager, Ivan Penskiy. You gave me a space and the resources I needed to complete this, not to mention the encouragement to follow through. Thank you for your patience and all the great times.

I would also like to thank Animesh Shastry, for providing me with an amazing, fully functional custom-built MAV and helping me to understand it, use it, and ultimately write this paper using it as a foundation. Without your prior work and support, this thesis would be unrecognizable.

My committee members, Dr. Metzler and Joe Conroy, sparked my curiosity for many topics in and related to this thesis, and the knowledge I gained from their classes have both significantly aided me in completing this work and shown me what is and could be possible. I am fortunate to have been your student.

Finally, I would like to thank all the people I'm grateful to call friends in my life who have kept me afloat, given me fulfillment outside of my studies, and helped me grow as a person throughout. You know who you are.

Table of Contents

Acknowledgements ii

Table of Contents iii

List of Tables v

List of Figures vi

List of Abbreviations x

Chapter 1: Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Relation to State of The Art . . . 4
- 1.3 Technical Approach . . . 7
- 1.4 Contributions . . . 12
- 1.5 Outline of Thesis . . . 13

Chapter 2: Background and Methodology 14
- 2.1 Visual-Inertial Odometry . . . 14
- 2.2 Experimental Data Collection Setup . . . 15
  - 2.2.1 Intrigue MAV Platform . . . 15
  - 2.2.2 UR5 Robot Arm . . . 17
  - 2.2.3 MAV-Robot Mount . . . 18
  - 2.2.4 Visual Factor Environment Setup . . . 18
  - 2.2.5 Software Integration of Experimental Setup . . . 24
  - 2.2.6 Camera Calibration . . . 26
  - 2.2.7 IMU Calibration . . . 29
  - 2.2.8 Camera-IMU Calibration . . . 33
  - 2.2.9 Sinusoidal Trajectory Parameterization Methodology . . . 33
- 2.3 Data Augmentation Setup . . . 36
  - 2.3.1 EuRoC MAV Dataset . . . 36
  - 2.3.2 Computational Simulation Methods with Blender . . . 38
  - 2.3.3 Blender Simulation Environment . . . 39
  - 2.3.4 Data Augmentation Methodology . . . 40
  - 2.3.5 Augmentation Software Infrastructure . . . 41
- 2.4 Performance Metrics . . . 41

Chapter 3: Experimental Data Containing Adverse Visual Factors 45
- 3.1 Generation of MAV Trajectory . . . 45
  - 3.1.1 Definition of Reference Frames . . . 45
  - 3.1.2 Sinusoidal Trajectory Parameters . . . 47
  - 3.1.3 UR5 Joint Poses Generation and Validation . . . 47

3.2 Experimental Data Collection . . . 53
3.2.1 Preparation of Experimental Setup . . . 54
3.2.2 Recording of Experimental Data . . . 55
3.2.3 Data Post-Processing . . . 55
3.2.4 Resultant Ground-truth Trajectory . . . 56
3.2.5 Resultant Experimental MAV Data . . . 57
3.3 Evaluation and Discussion of VIO Performance on Experimental Data . . . 61
3.3.1 Overall VIO Performance under Adverse Visual Factors in Experimental Data . 65
3.3.2 Feature-rich (baseline) Sequence Analysis . . . 79
3.3.3 Feature-sparse Sequence Analysis . . . 80
3.3.4 Repeating Features Sequence Analysis . . . 81
3.3.5 Mirror Sequence Analysis . . . 81
3.3.6 Direct Glare Sequence Analysis . . . 82
3.3.7 Lens Dirt Sequence Analysis . . . 83

Chapter 4: Augmented Data Containing Adverse Visual Factors 84
4.1 Smoke Simulation and Rendering Parameters . . . 84
4.2 Augmentation of Smoke to the EuRoC Dataset . . . 85
4.3 Evaluation and Discussion of VIO Performance on Smoke-Augmented Data . . . 87

Chapter 5: Conclusion 96
5.1 Summary of Contributions . . . 96
5.2 Summary of Findings . . . 96
5.3 Ongoing and Future Work . . . 97

Appendix A: Other Evaluation Data 99
A.1 Supplemental Plots for OpenVINS Analysis on Experimental Data . . . 99
A.2 VINS-Mono Results on Experimental Data . . . 99
A.2.1 Feature-rich Sequence Analysis . . . 99
A.2.2 Feature-sparse Sequence Analysis . . . 114
A.2.3 Repeating Features Sequence Analysis . . . 114
A.2.4 Mirrored Features Sequence Analysis . . . 115
A.2.5 Direct Glare Sequence Analysis . . . 116
A.2.6 Dirty Lens Sequence Analysis . . . 118
A.2.7 Overall VIO Performance under Adverse Visual Factors in Experimental Data . 119

Bibliography 120

List of Tables

2.1 Intrigue Camera Characteristics 16
2.2 Camera-IMU Calibration Residuals 33

3.1 OpenVINS performance metrics on the experimental data for each visual factor sequence across all sub-sequences 66
3.2 Velocity RMSE before (t_0–t_1), during (t_1–t_2) and after (t_2–t_f) presence of interchangeable visual factor panel, highlighted in red in Figure 1.1 70
3.3 Angular velocity RMSE before (t_0–t_1), during (t_1–t_2) and after (t_2–t_f) presence of interchangeable visual factor panel, highlighted in red in Figure 1.1 70

4.1 Images of smoke-augmented EuRoC V1_01_easy dataset, at select times [1] 88
4.2 Performance metrics for smoke-augmented data 90

A.1 VINS-Mono performance metrics on the experimental data for each visual factor sequence across all sub-sequences 99
A.2 VINS-Mono Velocity RMSE before (t_0–t_1), during (t_1–t_2) and after (t_2–t_f) presence of interchangeable visual factor panel, highlighted in red in Figure 1.1 104
A.3 VINS-Mono angular velocity RMSE before (t_0–t_1), during (t_1–t_2) and after (t_2–t_f) presence of interchangeable visual factor panel, highlighted in red in Figure 1.1 104

List of Figures

1.1 Experimental data collection setup [1] . . . 9
1.2 Concept of dataset augmentation using Blender [1] . . . 11

2.1 The early version of the Intrigue MAV used to collect experimental data . . . 16
2.2 UR5 robot arm mounted to a movable table . . . 17
2.3 The mount which attaches the Intrigue MAV to the UR5 robot arm . . . 18
2.4 Steps of mounting Intrigue MAV to UR5 end effector . . . 19
2.5 Blank panel without attached visual factor surface, mounted on a stand . . . 20
2.6 Generated feature-rich pattern with grid overlaid . . . 22
2.7 Panels used in real-world dataset setup . . . 23
2.8 Visual factor setups that did not require a panel . . . 25
2.9 Real-world data collection hardware setup, with components highlighted . . . 26
2.10 Devices used in real-world dataset collection . . . 27
2.11 Sample images from camera calibration dataset . . . 28
2.12 Kalibr reprojection error results for Kannala-Brandt pinhole camera model with radtan distortion on IMX412 hires camera . . . 30
2.13 Kalibr reprojection error results for Mei omnidirectional camera model with radtan distortion on IMX412 hires camera . . . 30
2.14 Kalibr reprojection error results for Kannala-Brandt pinhole camera with equi distortion on IMX412 hires camera . . . 31
2.15 Kalibr reprojection error results for Mei omnidirectional camera model with radtan distortion on M0014 tracking camera . . . 31
2.16 24-hour log of ICM2098 IMU accelerometer calibration data with parameter t lines . . . 32
2.17 24-hour log of ICM2098 IMU gyroscope calibration data with parameter t lines . . . 32
2.18 Kalibr Camera-IMU reprojection errors . . . 34
2.19 Estimated poses from Camera-IMU calibration . . . 35
2.20 Stages of parameterizing trajectory in lissajous-like fashion . . . 37
2.21 EuRoC Vicon room 1 3D scanned point cloud, visualized in MeshLab . . . 38
2.22 Sample of smoke simulated using SPH in Blender . . . 39
2.23 Blender simulation environment of EuRoC Vicon room 1 . . . 40

3.1 All reference frames used in experimental dataset collection . . . 46
3.2 Lissajous-like parameterization of the four state variables of ten sequences in the local MAV frame . . . 48
3.3 Trajectory created by lissajous-like parameterization with ten sequences, with darker points corresponding to earlier times, and arrows indicating the forward vector of the camera . . . 49
3.4 Solved UR5 joint angles, with limits shown by dashed line . . . 51
3.5 Solved UR5 joint velocities, with limits shown by dashed line . . . 52
3.6 Trajectory before and after inverse kinematics with near-goal perturbation done on eight poses . . . 53

3.7 UR5 Gazebo simulation environment during trajectory execution with MAV attached to end effector . . . 54
3.8 Ground-truth trajectory in 3D, position-only multi-view . . . 57
3.9 Ground-truth trajectory sub-sequences in 3D, with orientation shown by drawn reference frame vectors . . . 58
3.10 Ground truth acceleration magnitude along all ten sub-sequences, with pauses marked . 59
3.11 Ground-truth inertial position . . . 59
3.12 Ground-truth inertial quaternion rotation . . . 60
3.13 Ground-truth MAV-local velocity . . . 60
3.14 Ground-truth MAV-local acceleration . . . 61
3.15 Samples from each visual factor sequence with camera facing the highlighted red panel in Figure 1.1 [1] . . . 62
3.16 Samples at t = 0 from each visual factor sequence, for sub-sequence 2 . . . 63
3.17 OpenVINS root mean squared error (RMSE) of performance metrics across all sub-sequences of experimental data . . . 67
3.18 Velocity RMSE before (t_0–t_1), during (t_1–t_2) and after (t_2–t_f) presence of interchangeable visual factor panel, highlighted in red in Figure 1.1 . . . 68
3.19 Angular velocity RMSE before (t_0–t_1), during (t_1–t_2) and after (t_2–t_f) presence of interchangeable visual factor panel, highlighted in red in Figure 1.1 . . . 68
3.20 OpenVINS local linear and angular velocity power spectral density across all sub-sequences, cropped to 5 Hz, with the signal frequency marked by a black vertical line . 69
3.21 OpenVINS trajectory sub-sequences in 3D, with orientation shown by drawn reference frame vectors . . . 71
3.22 OpenVINS position errors for each sub-sequence, with sub-sequence 2 at top and sub-sequence 8 at bottom . . . 72
3.23 OpenVINS rotational errors for each sub-sequence, with sub-sequence 2 at top and sub-sequence 8 at bottom . . . 73
3.24 OpenVINS velocity errors for each sub-sequence, with sub-sequence 2 at top and sub-sequence 8 at bottom . . . 74
3.25 OpenVINS angular velocity errors for each sub-sequence, with sub-sequence 2 at top and sub-sequence 8 at bottom . . . 75
3.26 OpenVINS local linear and angular velocity (left) and power spectral density (right) for sub-sequence 2, with interchangeable visual factor panel presence highlighted in grey 76
3.27 OpenVINS local linear and angular velocity (left) and power spectral density (right) for sub-sequence 3, with interchangeable visual factor panel presence highlighted in grey 76
3.28 OpenVINS local linear and angular velocity (left) and power spectral density (right) for sub-sequence 4, with interchangeable visual factor panel presence highlighted in grey 77
3.29 OpenVINS local linear and angular velocity (left) and power spectral density (right) for sub-sequence 5, with interchangeable visual factor panel presence highlighted in grey 77
3.30 OpenVINS local linear and angular velocity (left) and power spectral density (right) for sub-sequence 6, with interchangeable visual factor panel presence highlighted in grey 78
3.31 OpenVINS local linear and angular velocity (left) and power spectral density (right) for sub-sequence 7, with interchangeable visual factor panel presence highlighted in grey 78
3.32 OpenVINS local linear and angular velocity (left) and power spectral density (right) for sub-sequence 8, with interchangeable visual factor panel presence highlighted in grey 79

4.1 Variation of smoke simulation shader parameters 86
4.2 Variation of smoke simulation quality parameters, including dataset generation time . . 86
4.3 Sample augmented images, before and after composition of the smoke-only image frame onto the original EuRoC image 87
4.4 OpenVINS trajectory on augmented data, including orientation expressed by reference frames 89
4.5 OpenVINS position errors on augmented data 90
4.6 OpenVINS rotation errors on augmented data 91
4.7 OpenVINS velocity errors on augmented data 91
4.8 OpenVINS angular velocity errors on augmented data, with logarithmic y-axis scale . . 92
4.9 OpenVINS local velocity and power spectral density on augmented data 92
4.10 OpenVINS root mean squared error (RMSE) of performance metrics 93

A.1 Samples at t = 0 in the feature-rich baseline sequence, for all evaluated sub-sequences 100
A.2 VINS-Mono root mean squared error (RMSE) of performance metrics across all sub-sequences of experimental data 101
A.3 VINS-Mono velocity RMSE before (t_0–t_1), during (t_1–t_2) and after (t_2–t_f) presence of interchangeable visual factor panel, highlighted in red in Figure 1.1 102
A.4 VINS-Mono angular velocity RMSE before (t_0–t_1), during (t_1–t_2) and after (t_2–t_f) presence of interchangeable visual factor panel, highlighted in red in Figure 1.1 102
A.5 VINS-Mono local linear and angular velocity power spectral density across all sub-sequences, cropped to 5 Hz, with the signal frequency marked by a black vertical line . 103
A.6 VINS-Mono trajectory sub-sequences in 3D, with orientation shown by drawn reference frame vectors 105
A.7 VINS-Mono position errors for each sub-sequence, with sub-sequence 2 at top and sub-sequence 8 at bottom 106
A.8 VINS-Mono rotational errors for each sub-sequence, with sub-sequence 2 at top and sub-sequence 8 at bottom 107
A.9 VINS-Mono velocity errors for each sub-sequence, with sub-sequence 2 at top and sub-sequence 8 at bottom 108
A.10 VINS-Mono angular velocity errors for each sub-sequence, with sub-sequence 2 at top and sub-sequence 8 at bottom 109
A.11 VINS-Mono local linear and angular velocity (left) and power spectral density (right) for sub-sequence 2, with interchangeable visual factor panel presence highlighted in grey 110
A.12 VINS-Mono local linear and angular velocity (left) and power spectral density (right) for sub-sequence 3, with interchangeable visual factor panel presence highlighted in grey 110
A.13 VINS-Mono local linear and angular velocity (left) and power spectral density (right) for sub-sequence 4, with interchangeable visual factor panel presence highlighted in grey 111
A.14 VINS-Mono local linear and angular velocity (left) and power spectral density (right) for sub-sequence 5, with interchangeable visual factor panel presence highlighted in grey 111
A.15 VINS-Mono local linear and angular velocity (left) and power spectral density (right) for sub-sequence 6, with interchangeable visual factor panel presence highlighted in grey 112
A.16 VINS-Mono local linear and angular velocity (left) and power spectral density (right) for sub-sequence 7, with interchangeable visual factor panel presence highlighted in grey 112

A.17 VINS-Mono local linear and angular velocity (left) and power spectral density (right) for sub-sequence 8, with interchangeable visual factor panel presence highlighted in grey 113

List of Abbreviations

AVF	Adverse visual factor
DNN	Deep Neural Network
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
MAV	Micro air vehicle
SAR	Search and Rescue
SLAM	Simultaneous localization and mapping
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
VO	Visual odometry
VIO	Visual-inertial odometry

Chapter 1: Introduction

1.1 Motivation

Unmanned Aerial Systems (UAS) are uniquely useful in hazardous scenarios, such as disaster and emergency response, where they have gained recent attention [2] [3] for their capability to enter these hazardous sites, off-loading responders from the risk and communicating critical information to them. These scenarios often require close approach to hazards and obstacles to collect adequate information, often indoors, where maneuverability of the UAV is critical. In these tight spaces, a miniaturized UAV, also known as a Micro Air Vehicle (MAV), offers the maneuverability and dimensions necessary. Generally, a pilot is required for the MAV in such tight spaces, necessitating a line of sight to the vehicle or a live video feed which is not always feasible and runs the risk of human error. By increasing UAV autonomy under these hazardous conditions, the pilot's performance can be improved by collision avoidance or even minimized altogether by of oading more piloting tasks to the UAV. By of oading a majority of the piloting tasks to the UAV, a lower communications bandwidth is required and non-line-of-sight operations may be enabled, reducing the burden on human operators while simultaneously increasing capabilities.

Challenges to UAV autonomy in these environments have gathered much attention recently, for example, indoors [4], collapsed buildings [5], res [6], caves [7], and even other planets [8]. One major challenge for autonomy in these scenarios is a loss of position estimation, which is critical for any level of autonomous operations, and many other operations, such as mapping. UAVs commonly use a Global Navigation Satellite System (GNSS) such as the Global Positioning System (GPS) to estimate their global position. In these indoor environments, or in the case of jamming or use on other planets, GNSS localization can be denied to a UAS, leaving it without a global position estimate, and

as such, GPS-denied navigation has become a subject of interest in the eld [9]. Without a global position estimate, the UAS must fall back on using its estimated change in position, also known as odometry.

Most GPS-denied localization methods involve one or multiple onboard sensors that are used to estimate difference in position as the vehicle navigates the environment. A wide range of sensors have been applied for GPS-denied localization, for example, LiDaR, IMU, sonar, pressure sensors, time of ight cameras, and monocular or stereo cameras in multiple spectra (e.g. RGB, thermal) [10]. These sensors are typically combined with one another in the process of sensor fusion [11] in order to create a more accurate nal estimation. Additionally, even with a GPS signal, non-GPS localization methods can be combined with GPS to create a more accurate estimate of the UAV location [11].

There are a wide range of solutions to localization without GPS [9] [12]. However, many of them are impractical for the applications discussed in this paper. For example: LiDaR requires a heavy, power-consuming, and costly sensor which is impractical for a MAV; inertial navigation systems (INS) quickly accumulate high amounts of drift [13], becoming highly inaccurate; and beacon-based navigation requires placement of beacons beforehand, which is impossible in exploration scenarios. As such, many MAVs in these scenarios choose to use a camera as the primary sensor for odometry. However, using only a camera leads to long-term drift, and as such, an IMU is often included in the state estimation in order to correct for this drift. Such a robust system combining a camera and IMU is known as visual-inertial odometry (VIO).

Because VIO requires visual information to function, certain adverse visual factors can severely impact its accuracy. Adverse visual factors such as smoke, fog, glare, or a dirty lens impact visibility, but VIO can lose accuracy even in perfect visibility: a featureless image, an image containing identical features, or a dynamic environment with moving features are all possible failure modes of VIO.

Additionally, VIO is not the only UAS process that suffers from visual failure modes: any visual or visual-inertial system may exhibit many of the same failure modes, such as simultaneous localization and mapping (SLAM) [14].

Research into visual-inertial failure modes has gathered attention in recent years as MAVs have become increasingly commonplace, along with research into their applications in hazardous scenarios such as disaster response. As discussed by [1], a VIO algorithm to improve robustness to variable lighting conditions was presented in [15]; in [16], lens contaminants were computationally removed, whereas [17] tested VIO algorithms on both lighting conditions and lens contamination; [18] dealt with foggy weather and similarly [19] addressed smoke; and [20] addressed dynamic features caused by motion in the environment; [21] tested visual odometry performance in rain. In recent years, neural networks have been increasingly applied to VIO; [22] and [23] have speci cally studied applications of neural networks in generally challenging environments for VIO, while [24] investigated application of DNNs speci cally to address dynamic features and loop-closure challenges. Other studies looked at different camera con gurations; [25], [26], and [27] combined multiple cameras to provide more information to the VIO algorithm, while [28] took this concept all the way with a 360-degree camera.

Despite the current research into VIO failure modes, there seems to be an overall lack of public visual-inertial ight datasets containing both adverse visual factors and a corresponding non-adverse baseline, which would be a dataset comprised of the same images and the same ground truth trajectory, only without the adverse visual factor present. Existing public visual datasets containing adverse visual factors either do not have a baseline [29] or are challenging to reproduce [30]. Additionally, ight datasets can be prohibitively dif cult to reproduce due to the uncertain nature of ight and visual factors such as smoke and dust. Such a visual-inertial adverse visual factor dataset that is public, contains a baseline, and addresses the reproducibility problem could enable the evaluation of

the robustness of many different VIO approaches to adverse visual factors, which may drive innovation in VIO algorithms and related fields, such as autonomy countermeasures and mapping.

One possible solution to the dataset reproducibility and baseline problem is the usage of a simulation-based synthetic dataset; the nature of a simulation allows for the exact replication of conditions between collection of data sequences. However, a simulation-based approach cannot perfectly mirror reality, and computational requirements for a simulation become increasingly logistically challenging the closer it must mirror reality. In contrast, an augmentation-based approach combines the realism of a real-world, non-synthetic dataset with simulated adverse visual factors.

This thesis aims to address both the reproducibility and baseline problems in current visual-inertial adverse visual factor datasets by applying a reproducible methodology to create such datasets and evaluating the performance of VIO on them. Two approaches are used: first, a real-world experimental dataset is collected containing non-synthetic adverse visual factors in a constrained laboratory setup utilizing a robot arm setup that has high reproducibility and ground truth accuracy. Second, an existing real-world visual-inertial dataset is augmented with simulated adverse visual factors which would be problematic to replicate in a real-world setup. By undertaking the creation and evaluation of these real and augmented adverse visual factor datasets, this thesis offers insights to how visual-inertial odometry may be affected in these hazardous environments along with a pipeline and methodology for further testing in such conditions.

1.2 Relation to State of The Art

Investigations into the robustness of visual-inertial algorithms to various environmental conditions have been undertaken before. However, there seems to be a significant lack in reproducible

public datasets that also contain a valid baseline which would allow for isolation of the adverse visual factor. This thesis intends to contribute to this growing body of work by applying a methodology that provides a baseline alongside adverse visual factors using a reproducible foundation for future investigations into visual-inertial robustness to build on top of.

Visual localization may be the most common solution to localization in GPS-denied environments, and is implementable on a MAV due to the small form factor of a camera. Visual localization can be relative or absolute. Relative visual localization, which measures position relative to the starting point, is often achieved using either visual odometry (VO), visual-inertial odometry (VIO) or visual SLAM [14]. Some methods exist to localize these relative methods on a global map using satellite imagery and pre-existing maps, giving vision-based methods the capability to offer the absolute positioning that GPS would otherwise provide [13]. However, this thesis focuses only on the relative visual localization problem, as it appears to be more of a bottleneck to increased autonomy in these scenarios.

The SubT-MRS dataset [29] is perhaps the most comprehensive visual-inertial dataset currently available that focuses on adverse visual factors. While it includes a wide breadth of data, each sequence presents a different trajectory, with no baseline non-adverse sequence provided that would allow for the isolation of the adverse visual factor's effect on performance. This no-baseline quality seems to be shared by nearly all public real-world adverse-visual-factor datasets [31] [32] [33], which may be due to the dif culty in repeating the exact same trajectory during the chaotic movement of ight. In contrast, this thesis demonstrates the use of a highly repeatable and accurate robot arm to provide a baseline sequence with no adverse visual factor which shares the same trajectory as the sequence with adverse visual factor, enabling the isolation of the adverse visual factor's effect.

Published datasets are far from the only evaluations of adverse visual factors on visual-inertial

odometry. Many papers follow the paradigm of proposing a new VIO algorithm to address a speci c adverse visual factor, which necessitates creation or usage of such a dataset to evaluate the new algorithm. This paper instead does not aim to propose any new VIO algorithms, instead focusing on isolating the effects of adverse visual factors on existing VIO algorithms. The effect of adverse visual factors has also been investigated in vision-based detection of UAVs [34]. However, this thesis focuses on odometry from the perspective of the UAV, not detection of UAVs within an image.

The synthetic generation of a dataset via simulation can solve both the baseline and reproducibility problems due to the completely reproducible nature of a simulation. Synthetic data has gained much attention in recent years, for example being applied to vision-based UAV navigation with the TartanAir dataset [35] and used to simulate smoke-heavy environments in UAV ight [6]. However, while simulation environments grow increasingly realistic, they are still not a complete substitute for the real world, where VIO algorithms will be applied. By including real-world data in the dataset, VIO algorithms tested on it are brought one step closer to real-world application. As such, this thesis uses real-world visual-inertial data in all sequences, and uses a hybrid augmentation-based approach to simulate some of the adverse visual factors on top of real-world data.

Augmentation of adverse visual factors to existing real-world data has been investigated in order to improve robustness in machine-learning models [36]. However, a variety of different approaches for the augmentation have been attempted. Rain, fog, and snow were augmented onto existing datasets in both [37] and [38]; however, the former used diffusion and generative adversarial networks (GAN), while the latter used a simulation environment in the Unity game engine. Several lens and lighting effects were augmented onto real images in [39], which used a custom parameter-based image processing algorithm. This paper chooses a method similar to the one presented in [38], but instead of using the Unity game engine, the 3D graphics application called Blender is used. In contrast to Unity,

Blender, as a 3D graphics program, is built with photorealism, physically-based rendering, fluid simulations, ray-tracing, and compositing integrated and out-of-the-box [40]. Meanwhile, other approaches such as diffusion or generative AI struggle with consistency and physically-accurate simulation. For these reasons, this thesis uses Blender to simulate select adverse visual factors for augmentation on top of real-world imagery.

Many VIO performance evaluations to date have used the absolute trajectory error (ATE), which represents the average difference in position or rotation between the estimated trajectory and ground truth [41]. Evaluating VIO only based on ATE presents serious drawbacks for autonomy applications, since the stability of control loops and autonomy processes will heavily depend on relative pose differences, i.e. velocity. The ATE has been supplemented before with relative pose error (RPE) [42], which evaluates the short-term local consistencies of the estimates trajectory relative to ground truth, providing a more drift-robust and relevant performance metric for autonomous applications. However, short spikes in velocity can still crash the vehicle when VIO is running within the control loop, which [29] addresses through the development of a robustness metric based on velocity. A handful of studies [43] have analyzed the frequency domain of trajectories, but such analyses remain rare. This thesis evaluates VIO by using both ATE and linear/angular velocity errors, applying RMSE to both to penalize outliers, and additionally analyzes the frequency domain to characterize noise in velocity estimation.

1.3 Technical Approach

This thesis applies a methodology to test and evaluate visual-inertial odometry performance under adverse visual factors. In contrast to previous work, this methodology includes a highly repro-

ducible setup with sub-mm accuracy ground truth, which crucially allows for recording baseline data which is used to isolate the effect of the adverse visual factor.

In order to create such a precise setup, a UR5 robot arm is used to manipulate the camera-IMU system, selected due to its sub-mm accuracy and precision [44] in executing a trajectory and recording it, providing both a reproducible trajectory and accurate ground-truth trajectory data. A MAV containing cameras and an IMU is then mounted on the end effector of the arm, providing the camera-IMU sensor package used to record the visual-inertial dataset. The robot-MAV system is then surrounded by a circle of foam panels which have a baseline high-feature pattern on their surface. The MAV is then connected to a ground station, which gives the command to begin recording visual-inertial data. The robot end effector then executes a pre-planned trajectory, moving the MAV around the environment in a ight-analogous pattern. This same trajectory is then repeated with a different visual factor present, for example, by changing one of the panels out. Together, this creates one dataset with multiple sequences containing adverse visual factors as well as a baseline, which all share the same ground truth trajectory, allowing for direct comparison between the VIO performance on the baseline sequence versus the adverse-visual-factor sequence. This setup is shown in Figure 1.1, with highlighted MAV (blue) ; UR5 robot arm (orange) ; camera ground truth trajectory (green) ; and foam panel which is replaced between runs (red).

Augmentation of adverse visual factors on top of a real-world dataset has several advantages over the precise real-world setup shown in Figure 1.1. Most notably, the baseline in any augmentation methodology is simply the non-augmented data for any visual-inertial dataset. This means that the non-augmented baseline in an augmented methodology can be guaranteed to be exactly the same as the data with augmented visual factors, minus the visual factor, which cannot be guaranteed in the experimental setup. Additionally, visual factors that are logistically challenging and highly random,

Figure 1.1: Experimental data collection setup [1]

like smoke, are possible to replicate exactly and with minimal resources in an augmentation-based setup. As such, as a proof-of-concept, this paper chooses to add the adverse visual factor of smoke to the real-world EuRoC dataset [45], which is widely used in the literature as a standard UAV ight dataset for testing VIO.

The augmentation of smoke to the EuRoC dataset is achieved through the use of the 3D computer graphics software Blender. First, a 3D scan of the environment provided by the EuRoC dataset is loaded into Blender. This environment does not need to be completely accurate, and is removed later; as such, a scan is not required, and running SLAM on the source dataset could provide such a 3D environment if desired. The smoke is then simulated in this 3D environment using Blender's smoothed particle hydrodynamics (SPH) system, which is capable of creating photorealistic uid simulations [46]. Then, each image in the dataset is stepped through, with the simulation moving the camera to the exact position where the image was captured using the provided ground truth data. The simulation is then also stepped forward to the timestamp of the source image. Finally, the simulation image is rendered, the 3D environment subtracted, and the result re-combined with the original non-adverse image to create an augmented image. This process is shown in Figure 1.2.

VIO is run of ine on the datasets for convenience, using a VIO algorithm which is capable of real-time performance, in this case, VINS-Mono and OpenVINS. Evaluation of the adverse visual factors effects on VIO performance is achieved through a combination of practical performance metrics. Estimated pose errors versus ground truth are broadly divided into four components: translational and rotational, and their derivatives, velocity and angular velocity. Each of these errors are compared to the baseline data error in combination with the known time of visual factor presence in order to isolate the effect of the adverse visual factor with respect to time. The errors over time are then each combined into an RMSE in order to obtain one performance metric of each component for each visual factor.

Figure 1.2: Concept of dataset augmentation using Blender [1]

Additionally, a frequency-domain analysis is performed on the experimental data in order to characterize the short-term noise added by the adverse visual factor, which is enabled by the use of sinusoidal parameterization of the trajectory, i.e. a signal frequency exists in the experimental data which can be used to compute signal-to-noise ratio (SNR). These performance metrics are qualitatively weighted in an analysis of the impact of each visual factor on the VIO performance.

1.4 Contributions

The three major contributions of this thesis are:

1. Design and implementation of an experimental methodology to isolate the effect of adverse visual factors on VIO: An experimental laboratory setup to record real-world visual-inertial data containing adverse visual factors is presented and used to test VIO performance. The setup notably has the repeatability necessary to produce multiple sequences that contain the same ground truth trajectory, enabling collection of a baseline sequence of images with no adverse visual factor. The baseline sequence may be compared to sequences containing adverse visual factors, allowing for isolation of the effect of the adverse visual factor.

2. Application of an augmentation-based approach to isolate the effect of synthetic adverse visual factors on VIO: An augmentation-based setup is presented and applied which can add photorealistic simulated adverse visual factors to existing real-world visual-inertial datasets. The use of augmentation allows for both reproducibility of logistically-challenging adverse visual factors and for the creation of a corresponding baseline sequence on any existing visual-inertial dataset, allowing for the isolation of the effect of the simulated adverse visual factors in a less-constrained setup.

3. A quantitative evaluation of the effect of selected adverse visual factors on VIO: An investigation into the effects of the adverse visual factors from the real-world and augmented datasets on VIO performance is presented. This evaluation represents a contribution to a wider effort to understand the effects of adverse visual factors on VIO, serving as a proof-of-concept for the ability of the experimental and augmentation-based methodologies to isolate the effect of adverse visual factors.

This thesis builds upon previous research by the author published in [1]. In [1], the same methodology to collect and create data containing adverse visual factors is presented, and an analysis of VIO on the data is conducted, using VINS-Mono on the experimental data and OpenVINS on the augmented data. However, in contrast to [1], this thesis uses a larger set of experimental data, presents a more detailed methodology, and applies new evaluation and analysis techniques to arrive at novel findings.

1.5 Outline of Thesis

Chapter 2 provides a detailed background on visual-inertial odometry, information necessary to understand the process of creating the real-world and augmented adverse visual factor datasets, and methods used to analyze VIO performance. Chapter 3 then details the collection of the lab-based experimental adverse visual factor data and an evaluation of VIO performance on it. Similarly, Chapter 4 details the augmentation of the existing EuRoC visual-inertial dataset with the adverse visual factor of smoke, and proceeds to evaluate VIO performance in an attempt to isolate the effect of the smoke. Finally, Chapter 5 concludes this thesis with a summary of research contributions along with suggestions for future work to further the field.

Chapter 2: Background and Methodology

This chapter provides a detailed background on the concepts and methodology necessary to understand this thesis.

2.1 Visual-Inertial Odometry

Visual-inertial odometry (VIO), which fuses information from a camera and IMU for pose estimation, may be the most common GPS-denied localization methods [10], is easily implemented on a MAV, can deal with some loss of vision, and requires relatively lower cost and mass than alternatives such as Lidar. As such, this paper primarily focuses on VIO for evaluation, but the data and methodology can be extended to any camera-based navigation system.

Fusing the visual information from a camera with inertial information from an IMU (i.e. visual-inertial odometry, or VIO) results in several major bene ts for visual localization [14]. First, vision-only pose estimates accumulate long-term drift as errors between images stack up, while IMU-only pose estimates accumulate large, random spikes of drift in the short term. Therefore, combining the IMU and camera data results in a solution which is both stable in the short term and long term. Additionally, monocular (single-camera) visual odometry has ambiguous scale, and so by adding inertial data, the trajectory scale can be accurately measured without any priors. This combination makes visual-inertial odometry (VIO) signi cantly more robust and applicable than vision-only odometry, and as such, this paper focuses on the use of VIO for localization.

VIO necessitates tracking of motion between frames, which can be done with either direct or

indirect methods. Direct methods work by directly comparing all pixel intensities between two frames and minimizing a photometric error between them [47]. However, direct methods require comparison of all pixels, resulting in a large computational overhead, and direct methods struggle with illumination changes due to operating based off of pixel intensity. In comparison, indirect methods which operate by tracking speci c points, also known as features, require relatively fewer resources and provide greater robustness to illumination variation since the features are de ned by their relationship to the surroundings. As such, this paper focuses on feature-based VIO methods due to their robustness and capability to run in real-time on a MAV.

2.2 Experimental Data Collection Setup

2.2.1 Intrigue MAV Platform

The MAV platform used in the experimental lab setup is an early version of the custom-built MAV named Intrigue, detailed in [5]. Intrigue was purpose-built to y in tight, GPS-denied indoors spaces under hazardous conditions in missions such as search-and-rescue. As such, Intrigue serves as a representative MAV for the scenarios that motivate this paper. A detailed view of the early version of Intrigue is shown in Figure 2.1.

The Intrigue MAV was out tted with an IMU and cameras for GPS-denied navigation using visual-inertial odometry. Onboard its ModalAI VOXL 2 computer, it ran visual-inertial odometry using its proprietary QVIO service. Intrigue was intended for SLAM applications, and achieved detailed, real-time RGB mapping and localization in the 2023 NIST UAS 3D Mapping Challenge 5.0 [5].

Two cameras on Intrigue are used for VIO and SLAM: the small M0014 greyscale tracking camera, and the high-resolution IMX412 RGB camera. In [5] the M0014 camera was used for tracking