

Visual Localization in Challenging Environments

vorgelegt von
M. Sc.
Patrick Irmisch

von der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
-Dr.-Ing.-

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender: Prof. Dr. Marc Toussaint

Gutachter: Prof. Dr. Olaf Hellwich

Gutachter: Prof. Dr. Cyrill Stachniss

Gutachter: Dr. Anko Börner

Tag der wissenschaftlichen Aussprache: 23. Mai 2022

Berlin 2022

Abstract

Visual localization, the method of self-localization based on camera images, has established as an additional, GNSS-free technology that is investigated in increasingly real and challenging applications. Particularly demanding is the self-localization of first responders in unstructured and unknown environments, for which visual localization can substantially contribute to increase the situational awareness and safety of first responders. Challenges arise from the operation under adverse conditions on computationally restricted platforms in the presence of dynamic objects. Current solutions are quickly pushed to their limits and the development of more robust approaches is of high demand. This thesis investigates the application of visual localization in dynamic, adverse environments to identify challenges and accordingly to increase the robustness, on the example of a dedicated visual-inertial navigation system.

The methodical contributions of this work relate to the introduction of semantic understanding, improvements in error propagation and the development of a digital twin. The geometric visual odometry component is extended to a hybrid approach that includes a deep neural network for semantic segmentation to ignore distracting image areas of certain object classes. A Sensor-AI approach complements this method by directly training the network to segment image areas that are critical for the considered visual odometry system. Another improvement results from analyses and modifications of the existing error propagation in visual odometry. Furthermore, a digital twin is presented that closely replicates geometric and radiometric properties of the real sensor system in simulation in order to multiply experimental possibilities.

The experiments are based on datasets from inspections that are used to motivate three first responder scenarios, namely indoor rescue, flood disaster and wildfire. The datasets were recorded in corridor, mall, coast, river and fumarole environments and aim to analyze the influence of the dynamic elements person, water and smoke. Each investigation starts with extensive in-depth analyses in simulation based on created synthetic video clones of the respective dynamic environments. Specifically, a combined sensitivity analysis allows to jointly consider environment, system design, sensor property and calibration error parameters to account for adverse conditions. All investigations are verified with experiments based on the real system.

The results show the susceptibility of geometric approaches to dynamic objects in challenging scenarios. The introduction of the segmentation aid within the hybrid system contributes well in terms of robustness by preventing significant failures, but understandably it cannot compensate for a lack of visible static backgrounds. As a consequence, future visual localization systems require both the ability of semantic understanding and its integration into a complementary multi-sensor system.

Zusammenfassung

Die visuelle Lokalisierung, die Methode der Selbstlokalisierung anhand von Kamerabildern, hat sich als eine zusätzliche, GNSS-freie Technologie etabliert, die in immer mehr realen und anspruchsvollen Anwendungen untersucht wird. Besonders anspruchsvoll ist die Selbstlokalisierung von Ersthelfern in unstrukturierten und unbekanntem Umgebungen, bei der die visuelle Lokalisierung wesentlich dazu beitragen kann, das Situationsbewusstsein und die Sicherheit von Ersthelfern zu erhöhen. Herausforderungen ergeben sich durch den Betrieb auf rechenbeschränkten Plattformen unter widrigen Bedingungen und in Gegenwart dynamischer Objekte. Aktuelle Lösungen stoßen schnell an ihre Grenzen und die Nachfrage nach der Entwicklung von robusteren Ansätzen ist hoch. Diese Arbeit untersucht die Anwendung der visuellen Lokalisierung in widrigen, dynamischen Umgebungen, um Herausforderungen zu identifizieren und die Robustheit der Methode zu erhöhen, am Beispiel eines dedizierten visuell-inertialen Navigationssystems.

Die methodischen Beiträge dieser Arbeit beziehen sich auf die Integration des semantischen Verstehens, Verbesserungen in der Fehlerfortpflanzung und die Entwicklung eines digitalen Zwillinges. Die geometrische Methode zur visuellen Odometrie wird zu einem hybriden Ansatz weiterentwickelt, in dem markante Bildpunkte auf bestimmten Objektklassen basierend auf einem neuronalen Netz zur semantischen Segmentierung aussortiert werden. Ein entwickelter Ansatz aus dem Bereich der sensor-nahen künstlichen Intelligenz ergänzt diese Methode, indem das Netz direkt darauf trainiert wird, Bildbereiche zu erkennen, welche für die betrachtete visuelle Odometrie kritisch sind. Eine weitere Verbesserung ergibt sich aus der Analyse und der Modifikation einer bestehenden Fehlerfortpflanzung innerhalb der betrachteten visuellen Odometrie. Außerdem wird ein digitaler Zwilling vorgestellt, der die geometrischen und radiometrischen Eigenschaften des realen Sensorsystems in der Simulation nachbildet mit dem Ziel, die experimentellen Untersuchungsmöglichkeiten zu vervielfachen.

Die Experimente basieren vorrangig auf Inspektionsdatensätzen, die verwendet werden, um drei Ersthelferszenarien zu untersuchen, nämlich Rettung in Gebäudekomplexen, Flutkatastrophe, und Waldbrand. Die Datensätze wurden in Flur-, Einkaufszentrum-, Küsten-, Fluss- und Fumarolenumgebungen aufgezeichnet und werden verwendet, um den Einfluss der dynamischen Elemente Person, Wasser und Rauch zu analysieren. Jede Untersuchung beginnt mit einer ausführlichen Analyse in der Simulation auf der Grundlage von synthetischen Videoklonen der jeweiligen dynamischen Umgebungen. Insbesondere ermöglicht die kombinierte Sensitivitätsanalyse die gemeinsame Betrachtung von Umgebungs-, Systemdesign-, Sensoreigenschaften- und Kalibrierungsfehlerparametern, um widrige Bedingungen zu berücksichtigen. Alle Untersuchungen werden durch Experimente am realen System verifiziert.

Die Ergebnisse zeigen deutlich die Anfälligkeit von geometrischen Ansätzen für dynamische Objekte in anspruchsvollen Szenarien. Die Einführung des Segmentierungszusatzes innerhalb des hybriden Systems verbessert deutlich dessen Robustheit, indem erhebliche Fehler verhindert werden. Das Fehlen eines sichtbaren, statischen Hintergrunds kann es jedoch verständlicherweise nicht kompensieren. Zukünftige visuelle Lokalisierungssysteme erfordern daher sowohl die Fähigkeit zum semantischen Verständnis als auch die Integration in ein komplementäres Multisensorsystem.

Acknowledgements

I would like to take this opportunity to thank all the people who have supported me during this exiting PhD journey.

I would like to express my sincere gratitude to Prof. Dr. Olaf Hellwich for his great encouragement, confidence, patience and valuable advice throughout this study and these years. I am very thankful to him for the time he invested in the many fruitful discussions and for the opportunity to also complete this thesis in his field at the TU Berlin. I am sincerely grateful to Dr. Anko Börner for the opportunity to write this thesis in his research team at the DLR, for his help and guidance as a mentor, and for his ideas and advice for a coherent story for this work. I appreciate working in his lab with all the diverse and exiting scientific tasks within the scope of this work and beyond. My further thanks go to Prof. Dr. Cyrill Stachniss for reviewing this dissertation.

I would like to thank my colleagues from the IPS research lab for their support, motivation and a pleasant working environment. My sincere gratitude goes to Dr. Jürgen Wohlfeil for supervising this thesis, for frequent discussions and good advice, for reviewing this work, and for sharing his experience of writing a dissertation. Special thanks to Ines Ernst, who always had an open ear when I got stuck, encouraged me, and gave good advice on many aspects of this work. Thanks to my office mate Dirk Baumbach for a friendly working relationship and his unconditional support. Appreciation is due to Dr. Denis Griebach for the many discussions on uncertainties in visual odometry and advice on aesthetic structuring of the dissertation. Great thanks to Dr. Adrian Schischmanow for his detailed and constructive review. Further, I would like to thank Maik Wischow for proofreading, Dr. Hongmou Zhang for advice, and Dr. Sergey Zuev, André Choinowski, Dr. Maximilian Buder, Magdalena M. Linkiewicz, and Dennis Dahlke for their support with respect to IPS questions, 3D reconstruction, and calibration.

I would like to extend my gratitude to people outside of the IPS lab. Many thanks to Dr. Julia Gonschorek for the good advice on possible first responder scenarios. I would like to thank Prof. Dr. Vikram Unnithan and Dr. Frank Sohl for giving me the opportunity to participate at the Vulcano summer schools with IPS, and for helping me to record the volcanic datasets that finally shaped the experiments of this thesis. Further thanks go to all the students who assisted me during the measurement campaigns.

I would like to thank my family and friends for their endless trust and unconditional support. My mother, who taught me the value of carefulness. My father, who taught me the meaning of dedication. My sister and nephews, with whom I enjoyed spending so much time over the last years, gave me the strength to continue this work to completion. A sincere thanks to Robert for his diligent proofreading of this thesis.

Contents

List of Tables	ix
List of Figures	xi
Notation	xiii
Acronyms	xiv
1 Introduction	1
1.1 Motivation	2
1.2 First Responder Scenarios	3
1.3 Research Focus	5
1.4 Organization of the Thesis	6
2 Related Work	9
2.1 Self-Localization Technologies and their Operational Capability for First Responders	9
2.2 Methods for Visual(-Inertial) Localization	11
2.3 Visual Localization in Dynamic Environments	14
2.4 Simulation for Visual Localization	17
I The Geometric System	19
3 Fundamentals - Visual-Inertial Navigation with IPS	21
3.1 Sensors	22
3.1.1 Physical Sensor Composition	22
3.1.2 Camera Sensor Modeling	23
3.2 Data Processing	26
3.2.1 Feature Detection and Tracking	27
3.2.2 Ego-Motion from Visual Odometry	29
3.2.3 Inertial Navigation and Data Fusion	30
3.3 Evaluation Metrics	32
3.3.1 Trajectory Evaluation	33
3.3.2 Error Propagation and Uncertainty Evaluation	35
3.4 Summary	37

4	A Digital Twin for IPS	39
4.1	Simulation Framework	40
4.1.1	Movement Profile Generation	40
4.1.2	Rendering of Camera Images	41
4.1.3	IMU Simulation	42
4.1.4	Additional Ground Truth Data	44
4.2	Synthetic Datasets	45
4.2.1	Synthetic Corridor Dataset	45
4.2.2	Synthetic Smoke Dataset	46
4.2.3	Synthetic Water Dataset	47
4.3	Simulation Strategies	48
4.3.1	Sensitivity Analysis	48
4.3.2	Geometric Monte Carlo Simulation	49
4.3.3	Combined Sensitivity Analysis	50
4.4	Summary	51
5	Analysis and Improvement of Uncertainty Estimation in Visual Odometry	53
5.1	Geometric System Calibration	54
5.1.1	Camera Calibration for Inspection and First Responder	54
5.1.2	Derivation of Calibration Uncertainties	55
5.2	Uncertainties in Visual Odometry	57
5.2.1	Feature Matching Uncertainties	58
5.2.2	Covariances during Feature Transformation	60
5.2.3	Weighted Least-Squares	63
5.3	Experiments	65
5.3.1	Geometric Monte Carlo Simulation	66
5.3.2	Confirmation with Real World Data	68
5.4	Discussion	69
5.5	Summary	71
II	The Hybrid System	73
6	Fundamentals - Semantic Segmentation	75
6.1	Developments in Deep Learning	76
6.2	Evaluation Metrics	78
7	Analysis and Improvement of Visual-Inertial Navigation in Dynamic Indoor Environments	81
7.1	Semantic Segmentation for Feature Selection	82
7.1.1	Segmentation Aid	82
7.1.2	Technical Notes	83
7.2	Corridor - Sensitivity Analysis	85
7.2.1	Simulation and Real-World Sensitivity Analysis	85
7.2.2	Combined Sensitivity Analysis	89
7.3	Confirmation with Real World Data	92

7.4	Summary	94
8	A Sensor-AI Approach to Improve Visual Odometry in Adverse, Dynamic Environments	97
8.1	Automatic Training Data Generation	98
8.1.1	Pixel-level Labeling	98
8.1.2	Timestamp Selection	101
8.2	Training for Semantic Segmentation	102
8.2.1	Technical Notes	103
8.2.2	Datasets	103
8.2.3	Evaluation	104
8.3	Combined Sensitivity Analysis for Coast and Fumarole Environments .	106
8.3.1	Coast	107
8.3.2	Fumaroles	108
8.4	Confirmation with Real World Data	109
8.5	Summary	111
9	Discussion	113
9.1	The Influence of Dynamic Objects	113
9.2	The Deep Learning Module	115
9.3	The Digital Twin	116
9.4	IPS for Self-Localization of First Responders	117
10	Conclusion	119
10.1	Summary	119
10.2	Outlook	121
	References	i
	Technology List	xii
A	Supplementary Material	xiii
A.1	The Least-Squares Problem	xiii
A.2	Datasets	xv
A.3	Geometric Calibration Parameters	xix
B	Supplementary Experiments	xxi
B.1	Sensitivity Analysis	xxi
B.2	Feature Matching Evaluation	xxii
B.3	Error Propagation for Feature Undistortion	xxii
B.4	Monte-Carlo-Simulation for Visual Odometry	xxiii
B.5	Semantic Segmentation	xxiv

List of Tables

3.1	IPS hardware parameters	22
5.1	Definition of prior matching uncertainties	60
5.2	Results from the geometric Monte Carlo simulation	66
5.3	Real-world experiments	68
7.1	The role of internal uncertainties in dynamic environment	88
7.2	The influence of masking features on static objects	88
7.3	Parameters for the combined sensitivity analysis	89
7.4	Quantitative results of the combined sensitivity analysis	91
7.5	Quantitative results for the IPIN dataset	94
8.1	Constraints for the automatic timestamp selection method	102
8.2	Evaluation of trained DNNs for semantic segmentation	106
8.3	Parameters of the combined sensitivity analysis	106
8.4	Evaluation of IPS in dynamic real world environments	110
A.1	Synthetic datasets used for the geometric MCS	xv
A.2	Synthetic datasets from the corridor environment	xv
A.3	Real datasets without dynamic elements	xvi
A.4	Real datasets from the corridor environment	xvii
A.5	Synthetic dataset used for the combined sensitivity analysis	xvii
A.6	Real datasets from the mall environment	xviii
A.7	Real datasets from the fumaroles, coast, river environments	xviii
A.8	Geometric calibration parameters	xix
A.9	Geometric calibration uncertainties	xx
B.1	Supplementary evaluation of feature matching errors	xxii
B.2	Supplementary experiment to validate the WLS method	xxiv
B.3	Supplementary segmentation results	xxiv
B.4	Supplementary quantitative segmentation results	xxv

List of Figures

1.1	Exemplary IPS dataset	2
3.1	IPS technology demonstrators	21
3.2	IPS coordinate systems and transformations	23
3.3	Camera coordinate frames and transformations	23
3.4	Central projection and image distortion	24
3.5	Image noise model	25
3.6	Stereo camera geometry	26
3.7	IPS software processing pipeline	27
3.8	Feature matching procedure in IPS	28
3.9	3D-to-2D visual odometry	30
3.10	Strapdown mechanization	31
3.11	Error state spatial navigation filter design	32
3.12	The absolute trajectory error	33
3.13	The normalized error	36
4.1	A digital twin for IPS	39
4.2	Overview of the simulation framework	40
4.3	Processing steps of movement profile generation	40
4.4	The graphic rendering pipeline	41
4.5	The extended rendering pipeline	42
4.6	Simulation of an inertial measurement unit	43
4.7	Synthetic image-based ground truth data	44
4.8	Synthetic corridor dataset	45
4.9	Synthetic fumarole dataset	46
4.10	Synthetic coast dataset	47
4.11	Conceptual design of the strategy sensitivity analysis	48
4.12	Conceptual design of the strategy geometric Monte Carlo simulation	49
4.13	Conceptual design of the combined sensitivity analysis	50
5.1	Calibration setups for inspection and first responder applications	55
5.2	Exceptional change in camera calibration under physical stress	57
5.3	Feature matching evaluation procedure	58
5.4	Evaluation of feature matching errors	59
5.5	Propagated uncertainties during feature transformation	63
5.6	Visualization of residual covariances	65
5.7	Evaluation of propagated uncertainties	67

5.8	Distributions of single samples	67
5.9	Correlation of calibration parameters	69
6.1	Different types of semantic segmentation	75
6.2	Principle of a feedforward, fully connected neural network	76
6.3	General structure of the encoder-decoder design of Deeplabv3+	78
6.4	Mean intersection over union	79
7.1	Examples with significant influence of dynamic objects	81
7.2	Integration of a segmentation aid into IPS	83
7.3	Run-times of different IPS-configurations	84
7.4	Sensitivity analysis in a simulated and real corridor environment	86
7.5	Used features in selected corridor scenarios	87
7.6	The effect of image frequency and image resolution	87
7.7	Variations in the corridor dataset	90
7.8	Correlation analysis for the corridor dataset	90
7.9	Sensitivity analyzes for the corridor dataset	91
7.10	Evaluation of uncertainties for the corridor dataset	92
7.11	Trajectory of the mall dataset	92
7.12	Used features in selected mall scenarios	93
7.13	Analysis of the escalator scene	94
8.1	The proposed Sensor-AI approach	97
8.2	The automatic pixel-level labeling procedure	99
8.3	The automatic timestamp selection procedure	101
8.4	Visualization and statistics of training, validation and test datasets	104
8.5	Visualization of semantic segmentations from the test dataset	105
8.6	Variations in the fumaroles and coast datasets	107
8.7	Correlation analysis for the coast dataset	107
8.8	Sensitivity analysis for the coast dataset	108
8.9	Correlation analysis for the fumaroles dataset	108
8.10	Sensitivity analysis for the fumaroles dataset	109
8.11	Used features in selected fumaroles, coast and river scenarios	111
10.1	Exploration of a fumarole field with IPS	121
B.1	Supplementary sensitivity analysis	xxi
B.2	Supplementary evaluation of error propagation for feature undistortion	xxiii
B.3	Supplementary qualitative segmentation results	xxv

Notation

Scalar notation:

- s : Scalars are denoted in lower case letters.
 $\{s_i\}_{i=0}^{N-1}$: A set of N scalars.
 s_i : i -th scalar of a set of scalars.

Vector notation:

- \mathbf{v} : Vectors are denoted in lower case bold letters.
 v_j : j -th value of a vector (with n values) with $\mathbf{v} = (v_0, \dots, v_{n-1})^T$.
 $\{\mathbf{v}_i\}_{i=0}^{N-1}$: A set of N vectors.
 \mathbf{v}_i : i -th vector of a set of m vectors with $\mathbf{v}_i \in \{\mathbf{v}_0, \dots, \mathbf{v}_{m-1}\}$.
 v_{ij} : j -th value of the i -th vector of a set of vectors.
 \mathbf{v}_{-j} : j -th values of m vectors with $\mathbf{v}_{-j} = (v_{0j}, \dots, v_{m-1j})^T$.

Matrix notation:

- \mathbf{M} : Matrices are denoted in upper case bold letters.
 M_{ij} : Scalar value in matrix \mathbf{M} at i -th row and j -th column.
 $\{\mathbf{M}_i\}_{i=0}^{N-1}$: A set of N matrices.
 \mathbf{M}_i : i -th matrix of a set of matrices.
 \mathbf{J}_a^b : Jacobian matrix that describes the derivative $\frac{\delta b}{\delta a}$.
 Σ_a : Covariance matrix for vector quantity \mathbf{a} .

Operators:

- $\hat{[\cdot]}$: Apparent quantity (estimated, measured value).
 $\tilde{[\cdot]}$: Normalized entity.
 $\vec{[\cdot]}$: Homogeneous vector.
 $\check{[\cdot]}$: Experimentally measured quantity.
 $[\cdot]^T$: Matrix transposed.
 $\|\cdot\|$: Euclidean norm.
 ${}_{(k,l)}[\cdot]$: Matrix of shape (k, l) with k rows and l columns.
 $\overset{\frown}{[\cdot]}$: Part of a matrix or of a continuous trajectory.
 $\text{sd}(\mathbf{e}_{-i})$: Returns the SD of the i -th element of a set of vectors.
 $\text{param}(\mathbf{M})$: Returns a vector with translation and rotation parameters of a homogeneous transformation \mathbf{M} .
 $\text{diag}(\mathbf{v})$: Returns a diagonal matrix based on the vector values.

Frequently used symbols to denote image points:

- \mathbf{m} Point $(u, v)^T$ in image coordinates.
- $\tilde{\mathbf{m}}$ Point $(x, y)^T$ in normalized camera coordinates.
- $\hat{\mathbf{m}}$ Estimated point $(\hat{u}, \hat{v})^T$ in image coordinates (e.g., a matched feature).
- \mathbf{m}^δ Point $(u, v)^T$ in distorted image coordinates.
- \mathbf{m}^{l2} Point $(u, v)^T$ in image coordinates of the second left stereo camera image.
- $\hat{\mathbf{m}}_j^{r1\delta}$ j-th point from a set of estimated feature points in distorted normalized camera coordinates of the first right camera image.

Frequently used symbols:

- l Reft stereo camera frame.
- $l1, l_1$ First left stereo camera framemage.
- $l2, l_2$ Second left stereo camera frame.
- r Right stereo camera frame.
- $r1, r_1$ First right stereo camera frame.
- $r2, r_2$ Second right stereo camera frame.
- n Navigation frame.
- b Body frame.
- $\boldsymbol{\kappa}$ Parameters of the interior orientation with $\boldsymbol{\kappa} = (u_0, v_0, f_0, f_1)^T$
- $\boldsymbol{\delta}$ Distortion parameters with $\boldsymbol{\delta} = (k_1, k_2, k_3, p_1, p_2)^T$

Acronyms

F calibration for First responders

I calibration for Inspection

ATE Absolute Trajectory Error

BA Bundle Adjustment

BLE Bluetooth Low Energy

CNN Convolutional Neural Network

DL Deep Learning

DLR German Aerospace Center

DNN Deep Neural Network

DoF Degrees-of-Freedom

EKF Extended Kalman Filter

FPGA Field Programmable Gate Array

GCP Ground Control Points

GNSS Global Navigation Satellite System

GPU Graphic Processing Unit

GT Ground Truth

IMU Inertial Measurement Unit

IN Inertial Navigation

IoU Intersection over Union

IPS Integrated Positioning System

KLT Kanade-Lucas-Tomasi

LiDAR	Light Detection and Ranging
MAV	Micro Air Vehicle
MCS	Monte-Carlo-Simulation
MEMS	Microelectromechanical System
MSCKF	Multi-State-Constraint Kalman Filter
NCC	Normalized Cross Correlation
nCLE	Normalized Closed Loop Error
PDF	Probability Density Function
RANSAC	Random Samples Consensus
RCNN	Recurrent Convolutional Neural Network
RFID	Radio-Frequency Identification
RTE	Relative Translation Error
SAD	Sum of Absolute Differences
SD	Standard Deviation
SfM	Structure from Motion
SGM	Semi-Global-Matching
SLAM	Simultaneous Localization and Mapping
SVM	State Vector Machine
ToF	Time-of-Flight
UAV	Unmanned Aerial Vehicle
UKF	Unscented Kalman Filter
UWB	Ultra-Wide-Band
VINS	Visual-Inertial Navigation System
VIO	Visual-Inertial Odometry
VO	Visual Odometry
WLS	Weighted Least-Squares
ZUPT	Zero Velocity Updates

Chapter 1

Introduction

Visual localization describes the estimation of the 6 Degrees-of-Freedom (DoF) pose, consisting of position and orientation, of a mobile sensor-system from camera images in a confined space. A basic building block is Visual Odometry (VO) that incrementally estimates the camera pose based on a sequence of camera images. A complementary sensor to cameras is the Inertial Measurement Unit (IMU) that measures accelerations and angular rates and also can be used to estimate the trajectory. The combination of both sensors within a Visual-Inertial Navigation System (VINS) provides a more robust and accurate localization solution compared to the individual sensors. Further developments, such as their integration into a Simultaneous Localization and Mapping (SLAM) framework or their combination with a Global Navigation Satellite System (GNSS) receiver, can significantly increase the accuracy of the system. Moreover, recent advances in Deep Learning (DL) lead to the development of hybrid systems, combining model- and data-based approaches, to further improve localization capabilities.

One of such visual localization systems is the Integrated Positioning System (IPS) (Börner et al., 2017). It was developed at the German Aerospace Center (DLR) and is designated for navigation, 3D reconstruction and inspections. IPS performs dead reckoning based on Inertial Navigation (IN) that is aided by measurements from stereo-camera-based VO. By consisting of both, the sensors and the localization solution, it represents a complete sensor system. Optionally, measurements from GNSS or infrastructure-based fiducial markers can be used to enable global localization.

Visual localization is an essential tool for robust autonomous navigation in unknown environments and has become indispensable for robotic applications such as planetary exploration or vehicle navigation within environments where GNSS is not available. Visual cameras are valuable components for sensor systems, because they are based on passive measurements, do not include mechanical elements, provide a large amount of information and are light-weight and inexpensive. Furthermore, the relatively large computational requirements of related computer vision algorithms are cushioned by the rapid improvements of computing hardware in the past few decades. These advantages motivate practitioners to apply visual localization in more and more difficult environments, but in which such methods quickly reach their limits. As a consequence, researchers have recently focused on increasing robustness of visual localization systems in high dynamic environments and under adverse conditions. Coined by Cadena et al. (2016), visual localization has entered the *robust-perception age*.

1.1 Motivation

In line with the global trend, IPS is exploited to be applied in more and more challenging environments. Some of those current and future challenging applications are shown in Figure 1.1. This includes localization in the context of dynamic hand-held indoor (a) and outdoor localization (d) and vehicle navigation (g). This includes inspection in the context of geological applications (b, c, f), forestry (e), and on-orbit-servicing (h). Associated challenges include reduced calibration accuracy and sensor properties due to adverse conditions and visual distractions from dynamic environmental elements that can severely impact the localization solution. Analyzing this impact and increasing robustness against dynamic elements is a hot topic in current research.

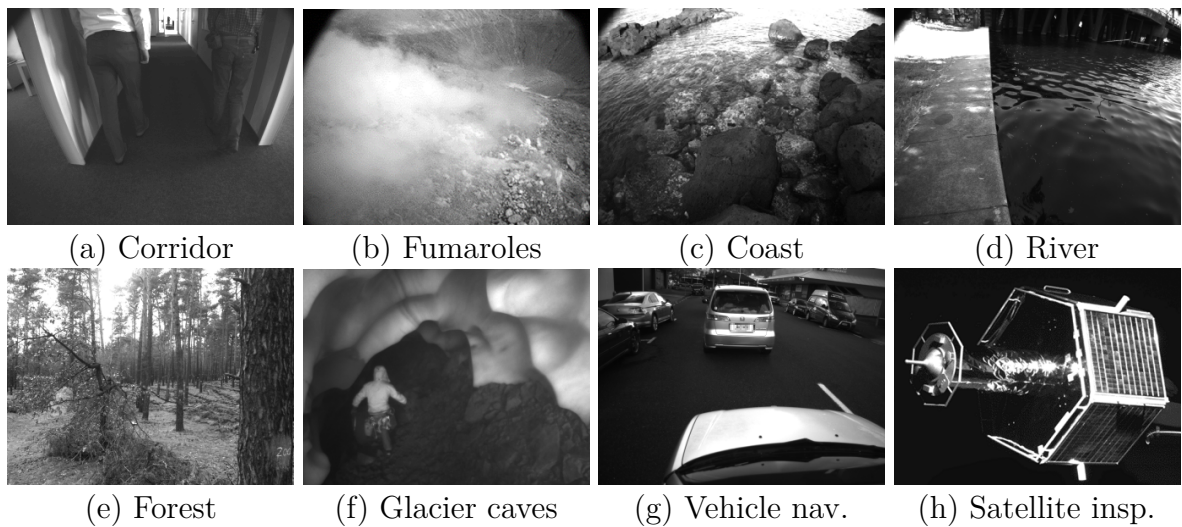


Figure 1.1: Exemplary IPS datasets recorded in challenging environments. Datasets of (a-d) are used in this thesis. References: (a,b) Irmisch et al., 2020, 2021, (e) Thiele, 2015, (f) Auf dem Kampe, 2021, (g) Ernst et al., 2018, (h) Benninghoff et al., 2018.

A particularly demanding application is the localization of first responders in unstructured or unknown environments. In emergency response, a reliable and accurate localization system with seamless indoor and outdoor navigation capability is considered as an essential tool to improve situational awareness and safety (Ferreira et al., 2017; Rantakokko et al., 2011). This real-time system must provide meter-level accuracy, specifically less than one meter horizontally and floor-level accuracy vertically, and integrity monitoring capabilities such as estimation of localization errors in form of uncertainties (Rantakokko et al., 2010). The accuracy requirement is a lot less than for instance from industry and construction with at least centimeter-level requirements (El-Sheimy and Li, 2021). Though, localization systems for first responders must be able to work reliable without using technical infrastructure and without any a priori knowledge (e.g., maps, floor plans). Furthermore, the performance of the sensor system will be severely constrained by the demands for energy-efficiency with batteries that last longer than 24 hours, inexpensiveness with a cost less than 1000 \$, handiness and light weight of less than 1 kg (Rantakokko et al., 2010). Moreover, it must operate in harsh environments and is frequently exposed to physical stress, which can reduce sensing quality and lower calibration accuracy.

Visual localization is generally considered as a promising component for sensor-fusion-based localization systems of first responders. But the mentioned restrictions (has to work under adverse conditions in harsh environments and without any self-localization infrastructure and a priori knowledge) are challenging and topic of research and development. In this context, the IPS research team of the DLR is currently engaged in the projects INGENIOUS (2019) and RESCUER (2018). For instance, an on-ground IPS is deployed in combination with airborne MACS (Hein et al., 2019) technology in INGENIOUS to develop methods for real-time localization of first responders in combined indoor and outdoor environments and rapid environmental mapping. A critical point to consider is that the localization system must not interfere with the first responder activity (Ferreira et al., 2017). This reduces the applicability of infrastructure-based or SLAM-based visual localization methods that require to look at specific way points. Therefore, IPS must constantly guarantee an accurate dead reckoning solution in all environments.

The development of solutions for real-time tracking of first responders is of high demand (IFAFRI, 2022). A robust visual localization method integrated in a multi-sensory system will substantially contribute to increase the situational awareness and safety of rescuers in future first responder scenarios. In this context, the main challenges are operation under adverse conditions on a computationally restricted platform and in the presence of a high number of dynamic objects. For example, adverse conditions can lead to reduced sensing capabilities due to poor visibility or to reduced calibration accuracy due to high physical stress on the system. Dynamic objects can create strong visual distractions that can even cause the system to fail.

Motivated by this promising use case, this work aims at investigating the application of visual localization in dynamic, adverse environments to identify challenges and accordingly to increase the robustness of visual localization, on the example of IPS.

1.2 First Responder Scenarios

First responders have to operate in adverse environments that present unique challenges and consist of diverse dynamic environmental elements, such as in Figure 1.1 (a-d). Their influence on the localization solution has to be studied in detail in order to develop robust localization systems, which requires the existence of appropriate datasets. However, obtaining such data is complicated, because, similar to the deployment of robotic platforms in disaster response (Murphy, 2021), a real first responder operation in a catastrophic scenario is not the time to collect data for research experiments or for system development. As a consequence, the required experiments must be based either on a simulated scenario in a real confined environment (as in INGENIOUS or RESCUER), abstracted from other applications or modeled with synthetic data.

In this thesis, inspection datasets from IPS are used to abstract and motivate three theoretical first responder scenarios that are characterized by unique environmental elements. The scenarios include (i) *indoor rescue*, (ii) *flood disaster* and (iii) *wildfire*.

(i) *Indoor rescue* scenarios can be versatile and can take place in complex buildings. For instance, groups of civilians might have locked themselves in different building parts during a rampage scenario and must be rescued by groups of first responders without them coming into contact with the attacker. The rescuers themselves are unaware of

the attacker’s position, which however might be known to the incident commander due to contact with other rescue teams. The commander’s task is then to navigate the rescuers safely through the building. In a complex building with no or damaged pre-installed infrastructure for localization, visual localization can be a valuable asset, but might be impaired by the frequent presence of moving persons in front of the cameras. For instance, team members might walk constantly in front of the camera as they navigate through the building. Further, groups of people to be rescued walk disoriented through the area or move together in one direction. The dynamic element *person* is considered in this thesis based on a large-scale *mall* dataset and the *corridor* dataset of Figure 1.1 (a).

(ii) The *flood disaster* in Germany 2021 resulted in considerable destruction of infrastructure and claimed over 180 lives (Kreienkamp et al., 2021). Even though such exceptional event is considered to happen only once every few hundreds years, the climate change increases the likelihood and intensity of extreme rainfall events. (Fekete and Sandholz, 2021) attribute the main problems of the 2021 flood disaster response to awareness, assessment, construction and planning gaps. They further noted that navigation through flooded areas and identification of alternative access routes has proven to be a challenge. In this context, accurate localization and navigation is essential for a systematic securing of large terrains, possibly including digital capture of secured areas and location marking of, in the worst case, corpses to be recovered. This requires an accurate self-localization, which is severely restricted by GNSS availability in urban canyons or dense forests, missing natural reference points such as bridges or buildings due to the destruction caused by the flood, and unfamiliarity of non-resident first responders with the environment. Visual localization might be a valuable asset, but might be impaired doing an ongoing flood by flowing water that appears opaque and consists of floating debris. The dynamic element *water* is considered in this thesis based on the *coast* (c) and *river* (d) datasets of Figure 1.1.

(iii) *Wildfire*. A self-localization system for fire fighters can be beneficial to quickly reach individuals that lost orientation or behave erratically during a fire fighting scenario (Rantakokko et al., 2010). Another application for self-localization of fire fighters could be the efficient localization and extinguishing of embers after a main wildfire fighting event in combination with satellites and drones. Space- and airborne technologies are the key for large-scale fire monitoring. However, they cannot observe embers through clouds, treetops or dense smoke. In this case, ground-based fire fighters could efficiently support the ember mapping process through the use of thermal camera devices and precise localization methods. In this context, visual localization can be a valuable asset, but might be impaired by the frequent presence of rising smoke and steam that partly cover and blur image areas. The dynamic element *smoke*¹ is considered in this thesis based on the *fumaroles* dataset of Figure 1.1 (b).

The datasets for these three scenarios have mostly in common that they are recorded in non-laboratory and relatively inaccessible environments. As a consequence, the amount of ground truth information is severely limited. Furthermore, the datasets are mostly restricted to one survey day with each a limited number of recordings with just a few tens of thousands of images. This limits the amount of variation of environmental

¹The term *smoke* is used in this work for better generalization in the context of first responders, but is imprecise from a geological point of view, since fumaroles emit steam and volcanic gases (vapor).

factors in the images and consequently reduces the significance of results. Therefore, each scenario is considered additionally in simulation to create highly diverse synthetic datasets using the digital twin of IPS that was developed within the scope of this thesis.

1.3 Research Focus

This section outlines the research questions, briefly summarizes the main contributions of this thesis and lists publications, in which parts of thesis have been already published.

Research Questions

Following the motivation, this work shall answer the research question:

1. What is the influence of typical dynamic objects from different challenging environments on (stereo) visual localization and how can it be assessed?

In connection with this research focus, the capability of Deep Neural Networks (DNNs) for object detection is exploited in this thesis. A key component will be the consideration of strategies from the research field Sensor-AI (Börner et al., 2020), which foresees a close interaction and combination of physical models, data-based models and classical approaches in one sensor system. Therefore, the following questions are formulated:

2. With specific focus on the dynamic elements *person*, *water* and *smoke*, can a DNN be used to identify critical image areas that are not suitable for VO?
3. Can the gained additional knowledge of this DNN be used to improve visual localization in the respective dynamic environments?

Since the main motivation is the application for first responders, which involves the topics uncertainty and sensor degradation, the following question is also considered:

4. Which error sources, either from the environment or the sensor system, have the most erroneous influence and need to be considered first in future developments?

Contributions

To answer the given research questions, new methods and tools for evaluation must be developed. The three main contributions of this thesis summarize as follows.

The first contribution is the analysis of the influence of different dynamic elements from challenging environments on visual localization and the development of methods to reduce their influence. First, the object *person* is considered in indoor environments, which is frequently studied in literature. Therefore, a basic mask approach based on a pre-trained DNN for semantic segmentation is used to ignore distracting image areas. Second, the dynamic elements *smoke* and *water* are considered, which are studied in this work based on fumarole, coastal and river datasets. Therefore, a Sensor-AI approach is proposed to improve VO-based sensor systems. Specifically, a DNN is trained to segment image areas that are critical for a specific VO system and use this knowledge to improve the same VO system.

The second contribution is the consistent simultaneous investigation in real world and in simulation. Therefore, a digital twin was developed that closely replicates the

geometry and radiometric properties of the real-world IPS in simulation. It was used to create synthetic videos clones of each considered dynamic element in the respective environments. In-depth analyzes are conducted in simulation based on three extensive simulation strategies. First, the *sensitivity analysis* allows to assess the influence of single parameters. Second, the *geometric Monte-Carlo-Simulation (MCS)* allows to assess the quality of propagated uncertainties. Third, the *combined sensitivity analysis* allows to consider multiple environment, system design, sensor property, and calibration error parameters at once and to weight their individual influence. All investigations are verified with experiments based on the real IPS.

The third contribution is the continuous consideration of degraded camera parameters caused by adverse conditions. First, degraded geometric system calibration is considered that can result, for instance, from high physical stress on the system. Therefore, realistic calibration errors in form of uncertainties are assessed and deployed in all experiments. Further, the error propagation concept in IPS is analyzed and improved, which is validated using the simulation strategy *geometric MCS*. Moreover, system uncertainties are brought into relation with the influence of dynamic objects and are also deployed to identify distracting image areas in order to generate reference data within the Sensor-AI approach. Second, degraded camera properties are considered that can result from adverse environmental conditions. Specifically, image blur and noise are considered as part of the *combined sensitivity analysis*.

Associated Publications

Parts of this thesis have been published in the following publications:

- Irmisch et al. (2019). “Simulation Framework for a Visual-Inertial Navigation System”. In: *International Conference on Image Processing (ICIP)*.
- Irmisch et al. (2020). “Robust Visual-Inertial Odometry in Dynamic Environments using Semantic Segmentation for Feature Selection”. In: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*.
- Irmisch et al. (2021). “A Hand-Held Sensor System for Exploration and Thermal Mapping of Volcanic Fumarole Fields”. In: *Geometry and Vision. Communication in Computer and Information Science (CCIS)*.

1.4 Organization of the Thesis

This thesis is mainly separated into two parts. Part I (Chapters 3-5) considers the basic geometric approach of IPS in order to prepare the investigation of the main research questions. This includes a methodical description of IPS, the development of a digital twin and an investigation of uncertainties in VO. Part II (Chapters 6-8) evolves IPS to a hybrid system by introducing a learning-based module. This includes the introduction of a segmentation aid and a related Sensor-AI approach to improve IPS in diverse dynamic environments. The results of both parts are then jointly discussed with respect to the research focus in a subsequent chapter.

Chapter 2 reviews state-of-the art methods available in the field of (visual) localization. Special focus is on: navigation in first responder applications; different methods

for visual localization, including the combination of classical approaches with DL modules; visual localization in dynamic environments; and simulation tools to analyze and improve visual localization.

Chapter 3 introduces the basic concepts of the existing IPS with focus on its sensors, physical models, localization approach and metrics to evaluate estimated trajectory and uncertainty quantities. Chapter 4 introduces the digital twin that was developed within this thesis. This includes a description of its functionality, its synthetic video clones for the environments *corridor*, *fumaroles* and *coast*, and the used simulation strategies (i) *sensitivity analysis*, (ii) *geometric MCS*, and (iii) *combined sensitivity analysis*. Chapter 5 investigates the influence of geometric calibration errors on IPS and analyzes its error propagation concept. Therefore, factors are first discussed that influence the geometric registration during application for inspection and first responders and respective calibration settings I and F are derived. Then, the VO pipeline is analyzed with respect to propagation of uncertainties and three related improvements are proposed. Finally, the improvement of the VO and navigation solution of IPS is validated in simulation with strategy (ii) and confirmed on real-world data. The improved method and the derived calibration settings are used in the following chapters.

Chapter 6 introduces the basic concept of semantic segmentation, which is integrated as a DL module in the proposed hybrid system, and briefly summarizes related developments. Chapter 7 investigates the influence of the dynamic object *person* on the localization in indoor environments by using a pre-trained DNN for semantic segmentation. Three experiments are presented. Strategy (i) is conducted simultaneously in simulation and real-world in a highly dynamic corridor environment to investigate the influence of different system parameters. Strategy (iii) is applied to weight the influence of dynamic environmental parameters to other error sources. Real-world data is used to confirm the observations based on a large-scale mall dataset. In Chapter 8, a Sensor-AI approach is proposed that learns critical image areas from VO and uses this knowledge to improve the same VO method. This method helps to extend the investigation of the influence of dynamic objects on visual localization to the objects *smoke* and *water*. The experiments are conducted in simulation with strategy (ii) and are also based on real-world data from the environments *fumaroles*, *coast* and *river*.

The results are jointly discussed and interpreted in Chapter 9 and are brought into context with existing literature. Finally, Chapter 10 summarizes gained insights and gives an outlook on future work.

Chapter 2

Related Work

This chapter provides a concise overview about related literature with focus on self-localization for first responders, different realizations of vision-based localization methods, visual localization in dynamic environments and the use of digital twins to develop related vision-based methods.

2.1 Self-Localization Technologies and their Operational Capability for First Responders

A variety of solutions for self-localization of individuals or robotic platforms exist. This section provides an outline of the main technologies from research for the localization of first responders. The reader might be referred to El-Sheimy and Li (2021) for a comprehensive review of sensors and techniques for indoor navigation and to Ferreira et al. (2017) for a detailed survey of research work for localization of first responders.

GNSS is a collective term for global-navigation systems such as the NAVSTAR or Galileo. Based on multilateration with at least 4 satellites within line of sight, a low-cost receiver can estimate its position with an accuracy of a few meters. Although this revolutionary technology has become a standard for many outdoor applications, it shows major performance loss and outages in urban canyons, tunnels, dense forests and indoor environments. As a result, GNSS transfers to a secondary sensor even in autonomous driving (El-Sheimy and Li, 2021).

Infrastructure-based methods depend on pre-installed or pre-deployed local area networks. For instance, pre-installed Wifi-networks are widely used for indoor navigation and provide accurate localization. Further, 5G technology shows great potential for localization and navigation (El-Sheimy and Li, 2021). However, the signal can be blocked by multiple floor-levels or underground, jammed or destroyed during major disaster events such as fires or floods. Pre-deployed infrastructure has to be established first on arrival of the first responders, but similarly enable reliable and accurate localization. This includes technologies such as Radio-Frequency Identification (RFID), Ultra-Wide-Band (UWB), Zigbee or Bluetooth Low Energy (BLE).

Infrastructure-less approaches allow relative positioning in a local reference frame. They can be self-contained or dependent on active or passive measurements of the environments. IN provides a self-contained localization solution, which does not require ex-

ternal references, by integrating measured acceleration and angular rate measurements in a strapdown manner (Woodman, 2007). The development of small and lightweight Microelectromechanical System (MEMS) inertial sensors allow the deployment of IMUs in various applications. However, they are prone to strong drift accumulation due to varying unknown measurement bias terms and therefore, they must be aided by Zero Velocity Updates (ZUPT) or external sensor measurements for accurate localization.

Active sensors such as Light Detection and Ranging (LiDAR) efficiently capture the 3D-structure of the environment based on the Time-of-Flight (ToF) principle and can provide accurate localization results. Though, they are usually based on mechanical laser arrays (e.g., Velodyne VLP-16) that spin at high frequency and are relatively expensive and bulky. This limits its application for small localization devices that are subjected to high physical stress. A recent development is the MEMS-based solid-state LiDAR (e.g., RealSense L515), which was already successfully applied for SLAM (Wang et al., 2021). It is inexpensive and light-weight, but comes with a smaller field of view. Related, the RGB-D sensor is frequently used for localization in indoor environments (e.g., Azure Kinect) as it provides highly accurate depth maps at close distances. However, due to its infrared-based ToF technology, it is unusable in outdoor environments with direct sunlight (Tölgyessy et al., 2021) and is therefore less suited for outdoor first responder applications. Other active sensors are based on radar and sonar technologies, that however show low measurement density or reflection issues on object surfaces.

Cameras are passive sensors and a popular choice for localization approaches. Visual localization comes at low cost, provides a large amount of information and shows accurate localization results. Though, it is sensitive to illumination, depends on the significance of environmental features and is relatively computational expensive. Besides standard visual camera, new camera sensor technology is developed that is promising for localization under adverse environmental conditions. Examples are the event camera sensor for extremely fast camera movements, the SWIR camera that is possibly well suited for navigation in dense smoke or RGB-D, thermal and light-field cameras.

A first responder localization system is a safety-critical component and cannot be based on one sensor only. Data fusion of complementary sensors must be realized. For instance, an early sensor fusion approach was described by Fischer et al. (2008), which is based on the combination of foot-mounted IMUs and ultrasound beacons. The IMU is used for IN with periodical application of ZUPTs during standstill phases. The ultrasound beacons were simulated, but were thought to be deployed by the first responders, leaving a “breadcrumb” trail. The authors demonstrated that the inherent drift of the IN solution could be compensated by the measurements from the beacons. Rantakokko et al. (2011) deployed two foot-mounted IMUs for two units that were additionally equipped with UWB ranging devices. They demonstrated a cooperative approach, where the shared position and distance information between both units improved localization results over basic ZUPT-based IN. Kachurka et al. (2021) presented a real-time cooperative SLAM system for indoor localization of first responders, fusing GNSS with two complementary SLAM approaches, namely LiDAR-Inertial-SLAM and Visual-Inertial-SLAM. By testing their approach in a challenging indoor scenario, they showed that although the individual SLAM methods fail in different scenes, the cooperative approach offers a consistent and accurate trajectory.

2.2 Methods for Visual(-Inertial) Localization

Visual localization is an extensive field of research that is targeted with most different approaches. This section provides an outline of different categories with selected examples of the main directions for vision-based localization with focus on VO, VINS and SLAM approaches. For a complete and comprehensive overview, the reader might be referred to the classics (VO: Scaramuzza and Fraundorfer, 2011, Aqel et al., 2016; VINS: Huang, 2019; SLAM: Durrant-Whyte and Bailey, 2006, Cadena et al., 2016) or to Alkendi et al. (2021) for a recent review about state-of-the-art approaches.

Visual Odometry

VO is a map-less approach to estimate the ego-motion based on images sequences and is most often applied in a monocular or stereo-camera setup (Scaramuzza and Fraundorfer, 2011). Substantially, the methods are divided into the two broad categories *geometric (model-based)* and *non-geometric (learning-based)* approaches (Chen et al., 2020; Poddar et al., 2018). Depending on the amount of image information used, it is classified as *sparse* or *dense* (Engel et al., 2016). The type of used image information, either based on salient features or pixel intensities, classifies the approach as *feature-* or *appearance-based* (Scaramuzza and Fraundorfer, 2011). Closely related, the optimization objective determines whether it is an *indirect* or *direct* approach by respectively formulating a geometric or a photometric error (Engel et al., 2016). The exact classification is often vague and many hybrid methods exist.

A geometric, sparse, indirect VO approach was presented by Nister et al. (2004), who estimated the ego-motion based on tracked features either for a monocular or stereo-camera setup. They detected salient feature points in every image, used image patches as descriptors and applied feature matching with normalized correlation as similarity metric. The camera motion estimation was combined with an outlier detection approach to improve robustness and iterative refinement that optimized the geometric reprojection error. This work has coined the term visual odometry.

A geometric, sparse, direct, monocular VO approach was presented by Engel et al. (2016), who estimated the ego-motion directly based on measured pixel intensities. They optimized the photometric error over multiple recent keyframes to jointly account for intrinsic and extrinsic camera parameters and inverse depth values. This approach inherits the advantages of direct methods, i.e., the ability to use all image points, and of sparse methods, i.e., efficiency. Though, they further showed that direct methods are more prone to geometric noise such as calibration errors than indirect methods. Therefore, indirect methods might be better suited for first responder applications where high physical stress and the use of low-cost cameras are to be expected.

Learning-based methods recently gained enormous attention due to the immense progress in the DL domain. Chen et al. (2020) distinguished between hybrid and end-to-end approaches and further divided them into supervised and unsupervised approaches. The stated main advantages over model-based approaches are their capability to automatically discover relevant and resilient task-specific features, their ability to learn from past experience and their capability to account for the increasing amount of sensor data. Though, disadvantages arise from its dependence on giant datasets, lack of interpretability and required amount of computational power.

For instance, Zhan et al. (2020) proposed a hybrid approach that aids geometric monocular pose estimation with two DNNs. They are deployed to find reliable 2D-2D point correspondences based on optical flow and to recover the metric scale based on depth prediction. In such way, hybrid approaches of model- and learning-based approaches usually outperform pure geometric approaches, since the DNNs can provide additional information that can easily be integrated into the model-based method.

Wang et al. (2017) proposed DeepVO, a supervised, end-to-end, monocular approach that is based on a Recurrent Convolutional Neural Network (RCNN). It is able to compute the metric scale without prior information during operation and showed superior performance over a monocular geometric approach of LIBVISO2 (Geiger et al., 2011).

End-to-end approaches are promising, but are still outperformed by model-based (or hybrid) approaches in the VINS domain (Chen et al., 2020). Though, they show better resistance to sensor degradation such as calibration errors (Clark et al., 2017).

Visual-Inertial Odometry

VINS utilize high-informative but low frequency cameras and self-contained but drift-prone IMU in a complementary sensor setup. Substantially, VINS methods are categorized based on the processing stage used for data fusion, i.e., *loosely-* or *tightly-coupled*, and the type of fusion method, i.e., *filter-* or *optimization-based* (Huang, 2019). VINS approaches that essentially perform dead reckoning are referred to as Visual-Inertial Odometry (VIO).

Optimization-based approaches solve a nonlinear least-squares problem over a set of measurements in a Bundle Adjustment (BA) manner, which is constrained to a few keyframes due to high computational costs. E.g., Stumberg et al. (2018) proposed an optimization-based, tightly-coupled, direct, sparse, monocular VIO method that minimizes photometric errors and IMU measurements simultaneously to jointly estimate camera poses and sparse scene geometry. By integrating the scale and gravity direction into the state vector, they eliminated the need for a tedious state initialization phase.

Filter-based methods perform data fusion in a filtering manner, for instance based on the Extended Kalman Filter (EKF) or the Unscented Kalman Filter (UKF). Loosely-coupled, filter-based VINS estimate the VO with a covariance independently and then fuse the estimated transformation with IMU measurements in the filter (e.g., IPS, Griebbach et al., 2014). This is computationally highly efficient, but geometric information are lost. A tightly-coupled, filter-based VINS uses information of 3D points to optimize the IMU states, which leads to improved accuracy, but comes at higher computational costs. Geometric constraints can be induced by adding tracked 3D points to the state vector, such as described by Huang (2019). Though, this approach is computational expensive when there are many features involved. To reduce computational costs, for instance Mourikis and Roumeliotis (2007) proposed the EKF-based Multi-State-Constraint Kalman Filter (MSCKF) in the context of monocular, tightly-coupled, feature-based VIO. The derived measurement model includes camera poses of recent keyframes in the state vector instead of 3D points and uses multi-view-tracked feature points during the measurement update to induce geometric constraints.

Anderson et al. (2019) investigated the covariance estimation for a geometric, feature-based, indirect RGBD-VO approach, using a Monte-Carlo method for error

propagation. The estimated relative transformation was used in a simplified EKF of a loosely-coupled, filter-based VIO approach for an Unmanned Aerial Vehicle (UAV). They showed that dynamically propagated covariances for VO lead to more accurate filter estimates than fixed values. They formulated a *normalized error* that allows a gross evaluation of propagated uncertainties, see Section 3.3.2.

Simultaneous Localization and Mapping

SLAM describes the simultaneous state estimation of a sensor system and map generation of the environment based on the perceived sensor data in real-time (Cadena et al., 2016), starting in an unknown location and unknown environment (Durrant-Whyte and Bailey, 2006). SLAM is a superset of VO since the trajectory estimation of the sensor system is an essential part (Poddar et al., 2018). Further, VINS are considered as an instance of SLAM, while it is specifically addressed as Visual-Inertial (VI)-SLAM, if the 3D feature points are contained in the state vector and jointly optimized with the camera/IMU pose (Huang, 2019). The mentioned taxonomy for VO and VIO is directly applicable for SLAM approaches. The reader might be referred to Campos et al. (2021) for a concise overview of the most representative approaches.

The main advantage of SLAM is its potential to eliminate the drift, which basically is the Achilles heel of all dead reckoning systems, in already mapped areas based on keyframe- and loop closure techniques. Though, the disadvantage is the enormous required computational costs to contain and update the map and the corresponding system poses. Therewith, additional challenges arise in terms of robustness, scalability and efficient map representation, see Cadena et al. (2016).

Most recently, the open-source library ORBSLAM3 was published by Campos et al. (2021), which contains state-of-the-art methods for geometric, sparse, indirect V-SLAM and optimization-based, tightly-coupled VI-SLAM for mono-, stereo- and other cameras types. It builds up on ORBSLAM2 (Mur-Artal and Tardós, 2016) that combines tracking and mapping of 3D points based on descriptor-based ORB features, keyframes, loop closure and relocalization based on bag-of-words (Galvez-López and Tardos, 2012). The backbone of ORBSLAM3 is a sophisticated map-handling approach based on Atlas (Elvira et al., 2019) that allows multi-map handling, smooth loop-closure and relocalization. Thereby, they strongly contributed to the open problem of “how to store the map during long-term operation” (Cadena et al., 2016). ORBSLAM3 includes a VINS-option based on Mur-Artal and Tardos (2017) and a favorable short initialization phase of less than 15s based on Campos et al. (2020). Interestingly, they pointed out that image segmentation could be used to discard features on the sky that can corrupt the system due to small motions, indicating that a geometric- and learning-based hybrid approach still could improve the system in dynamic environments.

2.3 Visual Localization in Dynamic Environments

In the last decade, visual localization research has significantly shifted to the improvement of robustness in challenging environments. This section provides an outline of selected strategies and examples with focus on visual, feature-based approaches. The reader might be referred to Saputra et al. (2018) for a comprehensive overview for V-SLAM and Structure from Motion (SfM) for localization in dynamic environments.

Geometric Approaches

Feature-based VO and SLAM approaches typically rely on the same principles for robust localization. Geometric models in form of fundamental matrices or homographies are estimated based on tracked features, while Random Samples Consensus (RANSAC) (Fischler and Bolles, 1981) is used to statistically exclude outliers and non-static feature points. Thereby, different geometric constraints are applied, such as from epipolar geometry (Hartley and Zisserman, 2003) or motion constraints. For instance, MonoSLAM (Davison et al., 2007) predicts the camera motion with corresponding uncertainties to restrict the search space during feature tracking. The space is defined by the position and propagated uncertainty of the projected map point in the new camera frame.

Sensor fusion such as with an IMU can significantly increase the robustness of the localization approach. Similar to motion constraints, Hwangbo et al. (2011) used gyroscope measurements from an IMU to predict feature positions in sequential images under strong camera motion and used them as initialization for the Kanade-Lucas-Tomasi (KLT) tracker. In terms of localization robustness, Zhang et al. (2018) developed a tightly-coupled, multi-keyframe VINS and could show a superior performance of VIO over pure visual methods in a dynamic office environment by experimenting on datasets consisting of pedestrians and strong camera motion.

Temporal information from feature tracking or optical flow ease feature classification. Extending MonoSLAM, Migliore et al. (2009) implemented tracking and classification of dynamic features in a parallel module. An intersection test of three projected viewing rays from the same feature in different views is used for classification. It formulates a chi-squared test based on corresponding propagated uncertainties to account for uncertain measurements. Zou and Tan (2013) proposed a collaborative SLAM for localization of multiple monocular cameras tracking static and dynamic points. In scenarios with high content of moving objects, a camera view might be obscured by a moving object. In such a scenario, they preserve robustness by estimating camera poses of multiple cameras jointly together with the positions of dynamic 3D points. They maintain uncertainties of map points, which allows to use its reprojection error for classification based on a chi-squared test. Alcantarilla et al. (2012) identified features on moving objects based on dense scene flow from two subsequent stereo frames within V-SLAM. First, dense scene flow is computed based on depths maps, optical flow and an initial VO estimation. Measurement uncertainties are propagated to derive corresponding motion uncertainties. Second, non-static motions are identified with a chi-squared test and corresponding features are discarded within a second VO estimation.

Geometric constraints play an essential role in IPS and are used in this thesis in combination with a chi-squared test to identify moving features. Though, the proposed approach of Chapter 8 is deployed offline to train a DNN based on past experiences.

Semantic Approaches

The integration of a DNN for semantic segmentation into VO or SLAM is an efficient way to introduce prior knowledge about potentially moving objects. Kaneko et al. (2018) generated a mask for the object classes *car* and *sky* and used it to prevent the feature detection within monocular SLAM in corresponding image areas. They found mostly superior performance over the base method by experimenting on synthetic datasets with diverse weather conditions. Bescos et al. (2018) could show similarly good results of the basic mask approach over the base method in dynamic environments with respect to monocular and stereo SLAM. Though, they reported slight deterioration in the presence of parking cars, due to hard rejection of features on cars, which generally provide good and reliable static features for the base method. Schorghuber et al. (2019) relaxed those semantic constraints by introducing a confidence factor that continuously evaluates features based on the number of observations and semantic classifications to certain classes. Features on parking cars, if observed to be static multiple times, are therefore used for localization. Contrary to discarding dynamic features, motion information of rigid bodies can provide additional information during the optimization process. Bescos et al. (2021) tightly coupled the SLAM objectives and multi-object tracking in dynamic environments for stereo or RGBD systems, i.e., they optimize the camera trajectory, object trajectory and static and dynamic objects points jointly.

Segmentation of only objects that actually move is a hot topic in research. For instance, Siam et al. (2018) proposed a two-stream architecture for joint object detection and motion segmentation based on appearance and motion clues. They could show that this multi-task approach outperforms independently-trained networks. Apart from vehicle navigation, Dave et al. (2019) proposed a class-agnostic approach for instance segmentation of arbitrary moving objects. Their architecture similarly fuses motion cues from optical flow and appearance cues from a segmentation model backbone.

Segmentation of arbitrary moving objects can significantly improve visual localization. For instance, Barnes et al. (2018) trained a DNN to simultaneously predict a depth map and an ephemerality mask to improve either a direct or an indirect monocular VO approach in a vehicle navigation context. Reference ephemerality image data was generated offline based on additional sensor data from a stereo camera and LiDAR and a generated 3D model of the static background. During application in indirect VO, the ephemerality mask is used to strictly classify each feature as static or dynamic. Recently, Bojko et al. (2021) generated class-agnostic masks of moving objects by only using monocular camera images in connection with the targeted SLAM approach. In training runs, they deployed SLAM outliers to derive binary masks for the whole image sequence, which are used to train the DNN. In their experiments, they focused on *consensus inversion*, which describes situations where more feature on dynamic than on static objects are present, and could show significant improvements.

In this thesis, the basic mask approach is used to assess the influence of different dynamic objects. In Chapter 8, geometric constraints are deployed to generate training data offline for unknown object types, which shows parallels to Barnes et al. (2018) and Bojko et al. (2021). Contrary to Barnes et al. (2018), the proposed method does not require an additional sensor setup. Contrary to Bojko et al. (2021), the proposed method is not restricted to dynamic objects and identifies method-specific distractions in general, such as homogeneous surfaces or unusable fine-structured vegetation.

Application in Challenging Environments

Visual localization and semantic segmentation are applied to increasingly challenging environments, which can contain significant amounts of water or smoke.

In terms of water detection, for instance, Lopez-Fuentes et al. (2017) trained DNNs for semantic segmentation in the context of flood monitoring based on a proposed river water segmentation dataset and demonstrated the potential of DL for water detection. Though, water appearance comes with high variability due to constant motion, reflection of surrounding vegetation and structures and changes of weather conditions. Scherer et al. (2012) proposed an automatic river mapping system based on a low-flying Micro Air Vehicle (MAV) that contains an image-based, self-supervised river segmentation module. This approach uses a State Vector Machine (SVM) to classify patch- and descriptor-based feature vectors. The classifier is continuously updated with generated training samples based on prior scene knowledge, e.g., a predicted horizon to partition the image or detection of reflected features that directly identify river patches, and therefore can cope with the high variety of river appearances. In a similar riverine mapping context, Yang et al. (2017) proposed to include the reflection of feature points into the localization procedure of Unmanned Aerial Vehicle (UAV)-based VINS. They proved the benefit of the additional geometric information from reflections in terms of observability analysis, numerical simulation and real-world experiments. Besides drone applications, visual localization is further considered for surface vessel applications. E.g., Kriechbaumer et al. (2015) evaluated two stereo-VO methods and found better performance of an indirect- over a direct approach. They stated that they found no evidence for static scene violations in their experiments after RANSAC filtering.

Visual localization seems to be relatively robust against distractions from the water. Reflection has even shown to be a key component for efficient river segmentation and localization. Though, the considered scenario in this thesis differs in the way that no prior assumptions about water can be made, water can obscure very large parts of the image and flotsam might introduce challenges besides ripples or reflections.

In terms of smoke detection, for instance, Yuan et al. (2019) proposed a DNN for smoke segmentation that was trained on synthetically generated data. Their results show relatively accurate segmentation results and they highlighted difficulties arising from high variation in smoke appearances, blurry boundaries and possible confusion with similar appearing objects. Furthermore, the visibility of smoke depends on the observed wavelength range of the cameras. Starr and Lattimer (2014) conducted extensive experiments to evaluate the performance of different navigation sensors under low visibility due to smoke. Visual cameras have shown to be unaffected in light smoke conditions and to be strongly affected in dense smoke conditions. Brunner et al. (2013) combined visual and thermal cameras in V-SLAM, which improved localization over single domain approaches in night, smoke and fire conditions. A proposed pre-selection of local image areas for feature matching based on an entropy-based evaluation of structural information further improved their localization results by masking distractions from low-light or smoke conditions.

The observability of smoke at different wavelength indicates that the training of a DNN for smoke segmentation should be directly linked to the used cameras. This is targeted in this thesis with a Sensor-AI approach and class-agnostic image segmentation in Chapter 8.

2.4 Simulation for Visual Localization

Simulation is essential for the efficient development and safe testing of visual localization approaches in hazardous or difficult to access environments. This chapter presents an outline of existing simulation tools and datasets. The reader might be referred to Liu et al. (2021) for a comprehensive survey about related real and synthetic datasets.

Modular simulation tools are essential to create versatile synthetic datasets. A popular tool from the robotics community is Gazebo (Koenig and Howard, 2004) that allows to use different sensor models and physics engines in order to model different complex robot platforms. OpenSceneGraph [OSG] is an OpenGL-based graphics toolkit which was frequently used for visual simulation or scientific visualization, but is currently succeeded by VulcanSceneGraph. Blender is a popular 3D modeling and animation tool that allows realistic rendering based on computational expensive raytracing. It is frequently used in scientific research, such in the context of 3D reconstruction (SyB3R, Ley et al., 2016) or training of DNNs (BlenderProc, Denninger et al., 2019). Different real-time game engines with realistic large-scale environments are exploited, such as Unreal Engine or Unity. For instance, AirSim (Shah et al., 2017) presents a multi-sensor-platform that generates synthetic camera-images and IMU data in real-time primarily for UAV applications. CARLA (Dosovitskiy et al., 2017) presents a vehicle platform simulator that focuses on urban environments and simulates dynamic objects with high diversity in their appearance and behavior.

Many challenging simulated datasets were proposed recently. For instance, Wang et al. (2020) proposed the challenging TartanAir dataset that consists of several sequences in diverse environments. Trajectories were generated automatically by using sampling-based planning techniques and occupancy grid maps. Jeon et al. (2019) proposed the disaster scenario dataset DISC that contains environmental challenges such as fire, smoke and collapse. They further demonstrated the benefit of using disaster scenarios during the training of DNNs in different tasks, such as semantic segmentation or VO.

Realism comes not only from high-quality synthetic sensor data, but also from a close connection to real-world scenarios and motions. Pham and Suh (2018) simulated sensor data for a foot-mounted IMU based on measured real-world trajectory and a smoothing algorithm to introduce corrections based on externally measured way points. A similar approach is developed in this thesis for the motion profile transfer. Gaidon et al. (2016) cloned real-world sequences in a synthetic environment and introduced the term *synthetic video clone*. They explicitly considered the transferability of experiment observations across real and virtual worlds, exemplary on the task of complementing training sets for multi-object trackers based on DL methods. However, even though the realism of simulations improves continuously, testing should not be based on synthetic data only, due to the gap between synthetic and real-world data (Vaudrey et al., 2008).

The term *digital twin* is used in this thesis to describe the synthetic projection of a real sensor system. However, it can also be used to describe the synthetic projection of a real environment. In this context, Smit et al. (2021) demonstrated and thoroughly discussed the combined use of real-time, vision-based 3D mapping and 3D data visualization to increase situational awareness of first responders and their commanders.

This thesis focuses on the use of synthetic video clones from mid-scale, real environments and realistic trajectories in combination with extensive statistical evaluation.

Part I
The Geometric System

Chapter 3

Fundamentals - Visual-Inertial Navigation with IPS

In short, IPS (Börner et al., 2017) is a stereo-vision-aided inertial navigation system that loosely-couples indirect, sparse VO and inertial navigation based on a Kalman filter. It describes a sensor system that consists of the sensor components and a localization solution. The function demonstrator used in this work is shown in Figure 3.1 (left). The localization functionality was designed and implemented by the authors of Griebach et al. (2014) [OSLib]. This includes the implementation of a multi-threading processing framework, mathematical camera models, feature detection and matching, VO, strapdown and data fusion, and analytical error propagation (Griebach, 2015). Zhang (2018) extended this work by improving the AGAST feature detector, implementing an error propagation for the matching procedure, and investigating a first keyframe-based approach in IPS. Outside of the focus of this thesis, it further consists of a 3D reconstruction module based on the stereo camera (e.g., used in Ernst et al., 2018) and can be fused with GPS to allow global localization (Baumbach et al., 2018).

This chapter describes the basic functionality of the VINS implementation in IPS, focusing on the VO component. The chapter begins with an introduction of the sensors of the used prototype and mathematical models used to model the camera system. The second section describes the localization functionality. This includes a detailed description of the image processing pipeline, which is the focus for improvements in

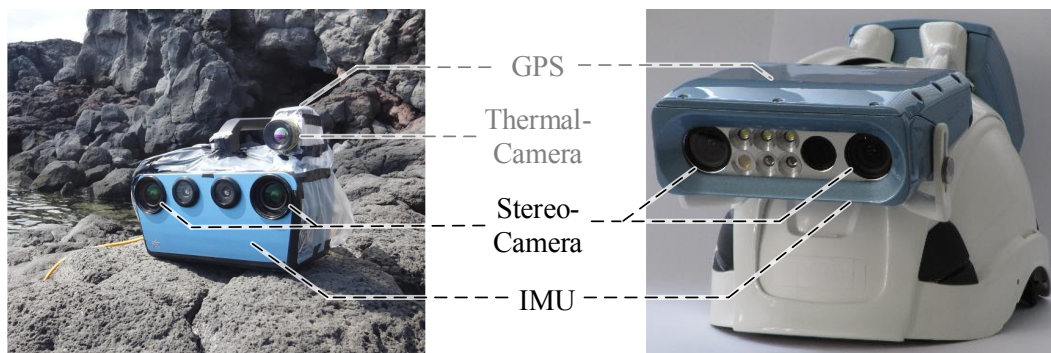


Figure 3.1: IPS hardware demonstrator (left, image from Irmisch et al., 2021) that is used in this work and a IPS helmet (right) that might be suitable for first responders.

Chapters 5, 7 and 8, and a brief summary of the IN and data fusion component. The third section describes evaluation metrics that are used in later experiments. This concerns trajectory evaluation and the evaluation of propagated uncertainties. Finally, a short summary follows, which highlights geometric properties that should make IPS less vulnerable to dynamic objects.

3.1 Sensors

This section introduces the sensors of IPS and the used mathematical camera models.

3.1.1 Physical Sensor Composition

Different prototypes exist for IPS and consist at least of a stereo camera and IMU. The hand-held IPS technology demonstrator of Figure 3.1 (left) is used in this thesis. The image shows the setup with an additional GNSS and a thermal camera in a volcanic coast environment. Figure 3.1 (right) shows a prototypical IPS helmet system that was for instance used during inspection of maritime hull structures (Wilken et al., 2015). This system can be used hand-free, which makes it suitable for use by first responders in principle. Further systems exist, such as used for vehicle navigation in combination with GNSS (Baumbach et al., 2018; Zuev et al., 2019), which could be suitable for global navigation of emergency vehicles in partly GNSS-denied environments.

Table 3.1: Hardware parameters for the camera (left) and IMU (right).

AVT Prosilica GC1380H		ADIS-16488	Gyroscope	Accelerometer
Resolution	1360×1024 px	Bandwidth	330 Hz	330 Hz
Pixel size	6.25 μm	Scale-factor stab. (\mathbf{s})	10 000 ppm	5000 ppm
Focal length	4.8 mm	Random walk	0.3°/h	0.029 m/s/√h
Sensor type	CCD-panchrom.	Bias repeatability (\mathbf{b}_c)	±0.2 °/s	±16 mg
Field of view	98°	output noise (\mathbf{n})	0.16 °/s	1.5 mg

The main hardware components of IPS are an industrial-grade, MEMS-based IMU and a stereo camera setup based on industrial-grade, panchromatic cameras, see Table 3.1. The chosen camera parameters in IPS can be seen as a trade-off between inspection with accurate 3D reconstruction and robust localization in real-time. On the one hand, inspection based on stereo cameras favors long focal length cameras and high resolution. On the other hand, localization favors a large field of view to ensure a large coverage area between sequential camera images during motion and to reduce the image area of individual moving objects in the camera view. For real-time localization, the images are usually downscaled to half resolution (640×512 px) and captured at 10 Hz. A Field Programmable Gate Array (FPGA) synchronously triggers both visual cameras and handles the accurate timestamp assignment for all sensor data. The data is recorded and processed on an external laptop.

The main camera coordinates and transformation are shown in Figure 3.2. The Kalman filter estimates the state of the system for timestamp t . The state includes the

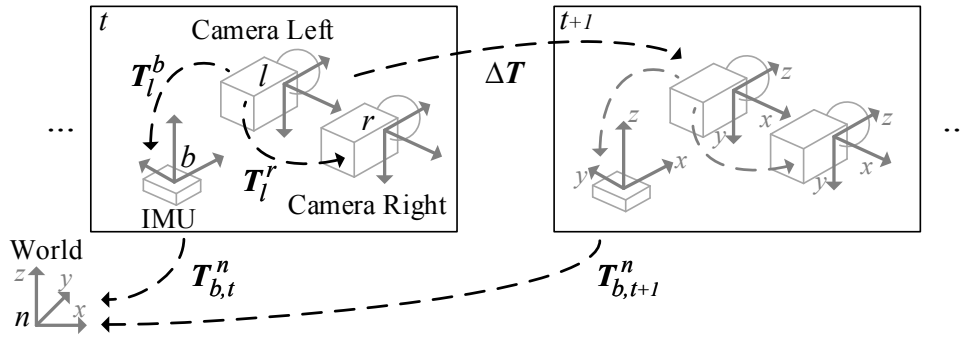


Figure 3.2: Illustration of the main coordinate systems and transformations in IPS.

transformation $\mathbf{T}_{b,t}^n$ that describes the orientation and position of IMU body frame b in world navigation frame n . The coordinate system of n is defined by the position and yaw-angle of the first recorded system state of each recording (run), while the z-axis is aligned to the direction of gravity. The stereo camera consists of the left camera frame l and the right camera frame r . The VO estimates the relative transformation $\Delta\mathbf{T}$ between consecutive timestamps for l . Frames b, l, r of the sensor system are spatially referenced by the calibrated static transformations \mathbf{T}_l^b and \mathbf{T}_l^r .

3.1.2 Camera Sensor Modeling

In this section, mathematical models are introduced that are applied in IPS to model a camera system, including geometric and radiometric components. Figure 3.3 illustrates the relation of different camera coordinates that are used in the following.

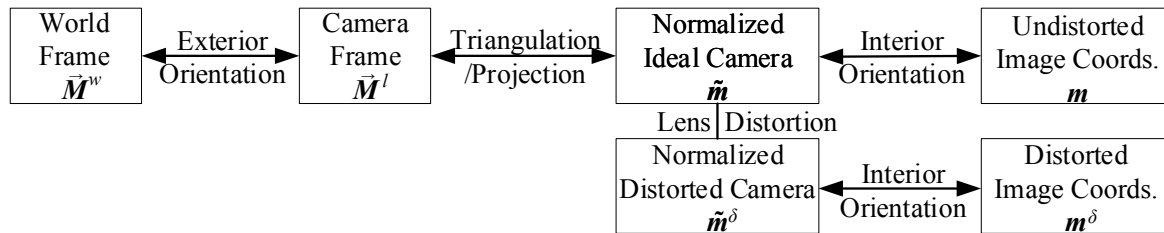


Figure 3.3: Camera coordinate frames and transformations, based on Griebach, 2015.

Camera Model

The pinhole camera model (Schreer, 2005, p.40) is used to model the projection of a camera and is illustrated in Figure 3.4 (a). It describes the central projection of an object point \vec{M}^l onto the image plane Π , which results in the projected image point $\mathbf{m} = (u, v)^T$. Π is defined in parallel to the xy-plane of the camera frame l and is placed in front of the camera in the mathematical model. Related, the principal point \mathbf{c} describes the point on the image plane that is the intersection of the principal axis z and Π . This projection is realized by the projection matrix \mathbf{P} in Equation 3.1. The camera matrix \mathbf{K} is composed of the principal point $\mathbf{c} = (u_0, v_0)^T$ and the principal

distance in pixel units α , based on the focal length f and pixel size d . s is a scale factor for the transformation into the two-dimensional Euclidean space.

$$s\mathbf{m}^i = \mathbf{K}\tilde{\mathbf{m}} = \mathbf{K}\cdot\tilde{\mathbf{P}}\cdot\tilde{\mathbf{M}}^l = {}_{(3,3)}\mathbf{K}\cdot{}_{(3,4)}\tilde{\mathbf{P}}\cdot{}_{(4,4)}\mathbf{T}_w^l\cdot{}_{(4,1)}\tilde{\mathbf{M}}^w = {}_{(3,4)}\mathbf{P}\cdot\tilde{\mathbf{M}}^w \quad (3.1)$$

$$\text{with } \mathbf{K} = \begin{bmatrix} \alpha & 0 & u_0 \\ 0 & \alpha & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{and } \alpha = f/d \quad (3.2)$$

\mathbf{P} composes first transformation \mathbf{T}_w^l that transforms object point $\tilde{\mathbf{M}}^w$ from an arbitrary world frame w into camera frame l . Second, it consists of the normalized projection matrix $\tilde{\mathbf{P}} = {}_{(3,4)}\mathbf{I}$, transforming $\tilde{\mathbf{M}}^l$ into normalized camera image coordinates $\tilde{\mathbf{m}}$. And third, it consists of the camera matrix \mathbf{K} , transforming $\tilde{\mathbf{m}}$ into image coordinates to \mathbf{m} . The coordinate transformation steps are noted in Figure 3.3.

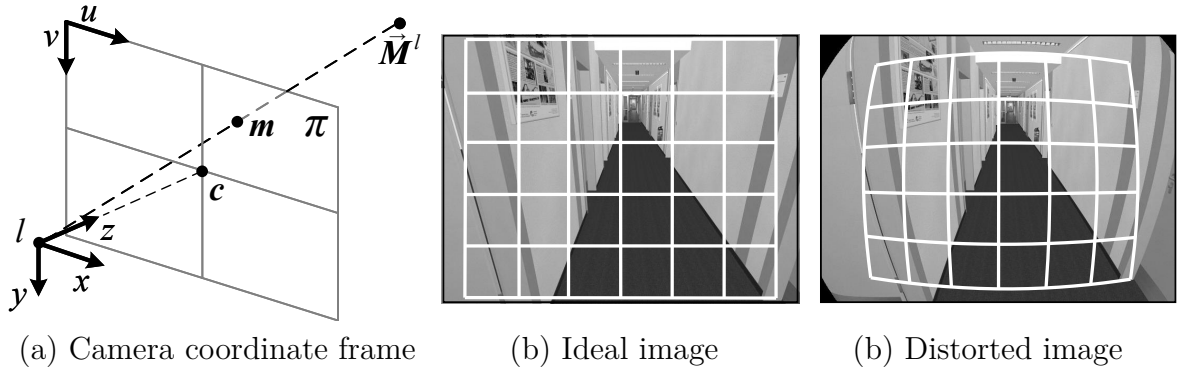


Figure 3.4: Illustration of central camera projection (a) and of the observed distortion effect for the used system in simulation, showing ideal (b) and distorted images (c).

Distortion model

Real lenses are usually not sufficiently represented by the ideal pinhole model, due to aberrations such as image distortion, defocus, spherical and chromatic aberration, coma, and similar. Geometrically most significant is image distortion (Hartley and Zisserman, 2003) and needs to be considered using a fitting distortion model. A common approach for standard field-of-view cameras, and fitting for the used IPS demonstrator, is the Brown distortion model (Brown, 1971). It models radial symmetric distortion δ_r and decenteric distortion δ_t in Equation 3.3 to transform the normalized camera point $\tilde{\mathbf{m}} = (x, y)^T$ to the distorted point $\tilde{\mathbf{m}}^\delta$.

$$\tilde{\mathbf{m}}^\delta = \tilde{\mathbf{m}} + \delta_r(\tilde{\mathbf{m}}, \mathbf{k}) + \delta_t(\tilde{\mathbf{m}}, \mathbf{p}) \quad (3.3)$$

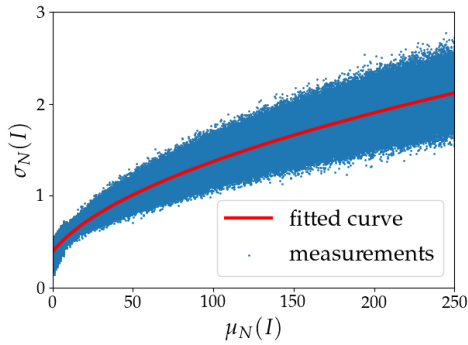
$$\delta_r(\tilde{\mathbf{m}}, \mathbf{k}) = \tilde{\mathbf{m}} \cdot \sum_{n=1}^N (k_n r^{2n}) \quad \text{with } r^2 = x^2 + y^2 \quad (3.4)$$

$$\delta_t(\tilde{\mathbf{m}}, \mathbf{p}) = \begin{pmatrix} p_1(3x^2 + y^2) + 2p_2xy \\ p_2(x^2 + 3y^2) + 2p_1xy \end{pmatrix} \quad (3.5)$$

Radial distortion is usually the most important deviation (Hartley and Zisserman, 2003, p.189) and appears in pincushion or barrel distortion. The latter is the case for the used IPS demonstrator, visualized in Figure 3.4 (right). Radial distortion is described as a polynomial with coefficients $\mathbf{k} = (k_1, k_2, \dots, k_N)$, while N is usually set to 3. In the case of cheap camera lenses, the deviation might also include tangential distortion and decentric distortion should additionally be considered. It is described in Equation 3.5 with $\mathbf{p} = (p_1, p_2)$. For the used IPS demonstrator, \mathbf{p} is omitted.

Noise model

In addition to degradation caused by the lens, the conversion of the captured light into a digital signal adds noise to the image. Image noise can degrade the feature matching process and needs to be considered (Zhang, 2018). It can roughly be separated into two main sources, which are fixed pattern noise and dynamic noise. Fixed pattern noise is usually automatically corrected by the camera itself. In contrast, dynamic noise varies between each captured frame due to read-out noise and photon noise. The noise model of Zhang (2018) is used for error propagation from noise during feature matching in IPS [OSLib] and for image degradation in the developed simulation tool of this thesis (Chapter 4). The model is formulated in Equation 3.7 with the resulting pixel value μ_N and resulting noise σ_N . It depends on electronic noise N_E of the camera and shot noise, represented by $\mu_N(I)/G$ with pixel intensity I and a gain parameter G . For instance, the parameters $N_E = 0.32$ and $G = 58.12$ are used in thesis for the considered camera system and were provided by (Zhang, 2018). Figure 3.5 shows the characteristic square root shape of the mean Standard Deviation (SD) for each pixel intensity during the calibration procedure based on 100 simulated images, degraded using given radiometric parameters.



$$\mu_N(I) = I \quad (3.6)$$

$$\sigma_N(I) = \sqrt{N_E^2 + \frac{\mu_N(I)}{G}} \quad (3.7)$$

Figure 3.5: Illustration and image noise model formulation based on (Zhang, 2018).

In the course of this thesis, camera binning and camera capture gain is applied in simulation. Both affect image noise and therefore requires an adaption of N_E and G . An implementation was provided in [OSLib]. Binning for IPS cameras refers to a summation of intensity values of pixels $\mathbf{I} = \{I_i\}_{i=0}^{b-1}$ in the binning area $b = b_{width} * b_{height}$, formulated as

$$\mu_B(\mathbf{I}) = \sum_{i=0}^{b-1} I_i, \quad (3.8)$$

$$\sigma_B(\mathbf{I}) = \sqrt{\sum_{i=0}^{b-1} \sigma_N(I_i)^2} = \sqrt{N_{E,B}^2 + \frac{\mu_B(\mathbf{I})}{G_B}}, \quad (3.9)$$

and with $N_{E,B} = \sqrt{b} * N_E$ and $G_B = G$. The analogue amplification using camera capture gain C results in scaling based on the decibel equation and is formulated as

$$\mu_D(I) = 10^{\frac{C}{20}} * I, \quad (3.10)$$

$$\sigma_D(I) = 10^{\frac{C}{20}} * \sigma_N(I) = \sqrt{N_{E,D}^2 + \frac{\mu_D(I)}{G_D}} \quad (3.11)$$

with $N_{E,D} = 10^{\frac{C}{20}} * N_E$ and $G_D = 10^{-\frac{C}{20}} * G$.

Stereo Camera

Figure 3.6 shows a stereo setup of two pinhole cameras, displaced by the fixed relative transformation T_l^r . In this setup, the projection of \vec{M} onto the image plane Π^r to m^r is constraint by the epipolar geometry. The baseline B describes the connection of the origins of the camera frames l and r . Its intersections with the image planes Π and Π^r define the epipols e^l and e^r . Related, the object point \vec{M} and the origins of l and r define the epipolar plane, while m^l , m^r , e^l , e^r lie on this plane. Its intersections with the image planes denote the epipolar lines ι^l and ι^r . The epipolar geometry states that the related image point of m^r in image plane Π^r lies on the epipolar line ι^r .

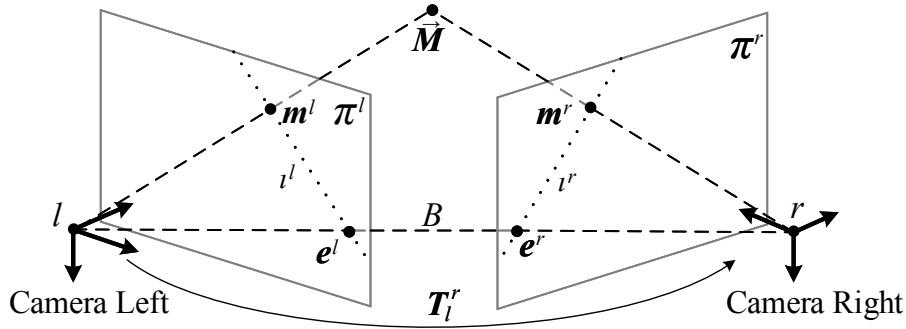


Figure 3.6: Stereo camera setup with epipolar geometry, based on Schreer (2005, p.69).

3.2 Data Processing

The main components of the IPS processing pipeline are summarized in Figure 3.7 and will be explained in detail in the following sections. In summary, the sensor data consists of consecutive camera images and IMU acceleration \hat{a} and angular rate $\hat{\omega}$ measurements. For VO, features are detected in the left camera image and tracked within the right image and one consecutive stereo pair. Successfully tracked features are used to compute the ego-motion between both stereo pairs. An additional ego-motion is computed based on the IMU data using the common strapdown mechanism. A Kalman filter fuses both information and computes the system pose. Optionally, the stereo images can be used to generate depth images based on a Semi-Global-Matching (SGM) GPU-implementation (Ernst and Hirschmüller, 2008). Using the estimated poses, the depth images can be accumulated to generate a 3D point cloud (e.g., used in Irmisch et al., 2021) in the context of fumarole mapping.

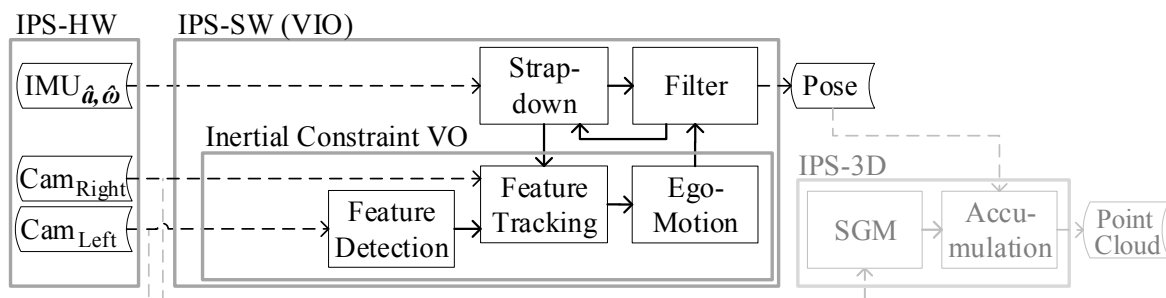


Figure 3.7: Software processing pipeline and its main components for IPS localization (black) and optional standard IPS 3D reconstruction (grey, not used in this thesis).

3.2.1 Feature Detection and Tracking

Feature-based VO relies on point-based image correspondences between images. Features or keypoints are distinctive image points such as corners or blobs, where (Harris and Stephens, 1988; Lowe, 2004) are commonly used representatives. Image correspondences are defined as “two-dimensional features that are the reprojection of the same 3D feature across different frames” (Scaramuzza and Fraundorfer, 2011). There are two major techniques in literature to find image correspondences (Fraundorfer and Scaramuzza, 2012). First, features can be detected in one image and tracked in other images using local search techniques, such as correlation. This approach is relatively fast, but assumes that the viewpoint change between cameras is small. Second, features can be detected in each image separately and matched using feature descriptors. This procedure is more computational expensive, but allows larger view point changes and is the current absolute choice for feature-based state-of-the-art SLAM approaches, such as in (Campos et al., 2020).

The first technique is applied in IPS, because detected features are only needed to be tracked in one consecutive stereo image. Feature tracking over more images is currently avoided, because it might lead to correlated VO estimations. All image operations for detection and tracking are applied in the original distorted image (superscript δ) to keep computational costs low and to not disturb information on image noise by interpolating pixel values. Visualized in Figure 3.8, features are detected in the first left camera frame only (e.g., $\mathbf{m}^{l\delta}$) and are tracked in the corresponding right camera image (intra-matching) as $\hat{\mathbf{m}}^{r1\delta}$ and in the subsequent stereo frame (inter-matching) as $\hat{\mathbf{m}}^{l2\delta}$ and $\hat{\mathbf{m}}^{r2\delta}$. $\{t; t+1\}$ is used interchangeably with $\{1; 2\}$ for subsequent timestamps.

An extended feature detector of AGAST (Mair et al., 2010) is used, that was proposed and implemented for IPS by (Zhang, 2018). The basic principle is the comparison of pixels in a circular mask to the center value and the derivation of corner candidates based on the number of similar intensity pixels (Smith and Brady, 1997). To improve efficiency, the Features from Accelerated Segment Test (FAST) is applied that considers only pixels on the outer circle with a Bresenham circle of 3, reducing the number of considered pixels to 16 (Rosten and Drummond, 2005). A feature is found if at least $S = 9$ contiguous pixels have brighter or darker pixel values, delimited by a threshold Θ_f . Further improvements are made in the Adaptive and Generic Accelerated Segment Test (AGAST, Mair et al., 2010), in which comparisons are formulated in a dynamic decision tree that automatically switches between two trained trees, optimized for ei-

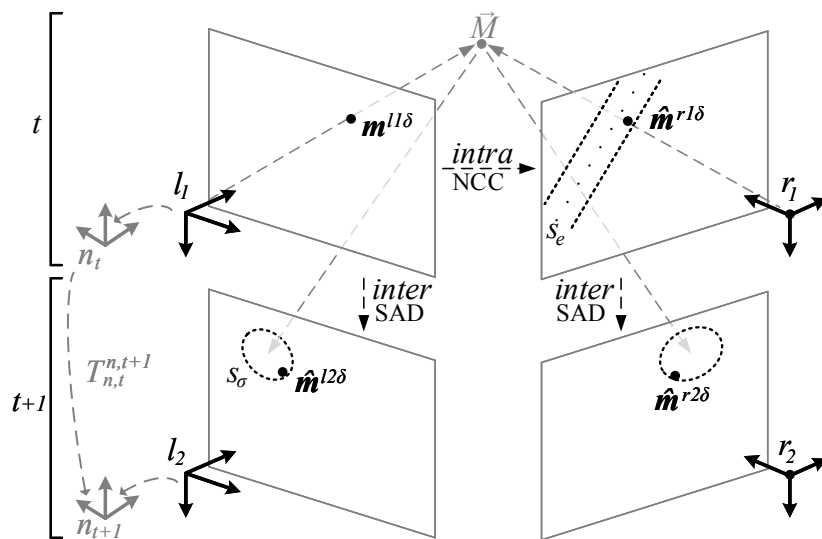


Figure 3.8: Feature matching process in IPS. The explanation is provided in the text.

ther homogeneous and heterogeneous areas. (Zhang, 2018) ensures a stable number of intra-matched feature a well spatial distribution of detected features. The stable number of feature matches is achieved by first using a relatively low intensity threshold Θ_I during detection (e.g., 8 stated by Zhang, 2018). Then, a number of Θ_D features with the highest score are selected based on a circular non-max suppression approach, before intra-matching is applied. Θ_D is set dynamically and orientes on the number of successfully matched features from intra-matching in comparison to a user-defined desired number of feature matches Θ_M for intra-matching (e.g., $\Theta_M = 100$). The well spatial distribution of detected feature is achieved by dividing the image into $c \times c$ grid cells and detecting Θ_D/c^2 features in each cell (e.g., $c = 2$ used in this thesis).

Feature tracking is based on patch-based template matching in IPS. The template is defined based on the feature of $m^{l\delta}$ in the first image and is matched with a number of candidates in the second image within a local search area (s_e, s_σ). A relatively small template size (e.g., 5×5 px) is used in IPS, which is possible due to tracking of keypoints over only two stereo frames and strict reduction of the search space.

For intra-matching, Normalized Cross Correlation (NCC) is used as matching metric with

$$\gamma_{NCC}(u, v) = \frac{1}{n} \sum_{i,j} \frac{(\mathbf{S}(u+i, v+j) - \mu_{\mathbf{S}})(\mathbf{T}(i, j) - \mu_{\mathbf{T}})}{\sigma_{\mathbf{S}}\sigma_{\mathbf{T}}}, \quad (3.12)$$

with search image patch \mathbf{S} and template patch \mathbf{T} of size $n = i \cdot j$ pixels. $\{\mu_{\mathbf{T}}, \mu_{\mathbf{S}}\}$ and $\{\sigma_{\mathbf{T}}, \sigma_{\mathbf{S}}\}$ describe the mean and SD of the patch in both images, respectively. Its advantage is an invariance to pixel intensity differences, e.g., caused by different exposure times of both cameras. Its disadvantage is a currently lacking subpixel matching approach. During intra-matching, the search space is effectively reduced using epipolar constraints, visualized in Figure 3.8 with dashed lines in the right camera image in r_1 . The object point \tilde{M} is triangulated in the stereo setup based on $m^{l1\delta}$ and $\hat{m}^{r1\delta}$, whereby the Sampson-approximation (Hartley and Zisserman, 2003) is used for correction of first order measurement errors in feature positions (Grießbach, 2015, p.33).

Sum of Absolute Differences (SAD) is applied in IPS during inter-matching, under the assumption that corresponding pixels of consecutive images keep the same intensity.

$$\gamma_{SAD}(u, v) = 1.0 - \frac{1}{n \cdot 255} \sum_{i,j} |\mathbf{S}(u+i, v+j) - \mathbf{T}(i, j)| \quad (3.13)$$

Its advantage is fast processing and the existence of a subpixel matching approach with analytical error propagation from image noise, proposed by Zhang (2018). The disadvantage is its susceptibility to changing pixel intensities, which can limit its usability in outdoor environments. The search space is reduced by inertial constraints, visualized with a dashed ellipses in Figure 3.8. Its center is the projection of $\vec{\mathbf{M}}$ into the stereo frame at $t+1$, using the IN strapdown solution $\mathbf{T}_{n,t}^{n,t+1}$ to predict the movement. The elliptical search space s_σ itself is defined by the elliptical confidence region, resulting from the propagated covariance from the uncertain object point $\vec{\mathbf{M}}$, uncertain calibration parameters and uncertain navigation solution $\mathbf{T}_{n,t}^{n,t+1}$.

3.2.2 Ego-Motion from Visual Odometry

“Visual odometry (VO) is the process of estimating the egomotion of an agent (e.g., vehicle, human and robot) using only the input of a single or multiple cameras attached to it” (Scaramuzza and Fraundorfer, 2011). In general, a cost function is defined and minimized using a least-square variant to estimate the optimal 6D transformation parameters based on measured 3D object point or 2D image point correspondences. Scaramuzza and Fraundorfer (2011) categorized this optimization problem in three different approaches, which are 2D-to-2D, 3D-to-2D, 3D-to-3D, while the former two are found to be generally more accurate. Most suited for a stereo setup is 3D-to-2D, since the set of 3D points directly inherent the correct metric scale (Nister et al., 2004).

Accordingly, the stereo-based visual odometry approach in IPS implements the 3D-to-2D approach. The non-linear least-squares problem is formulated by (Grießbach, 2015, p.35) with

$$\min_{\Delta \mathbf{T}} \|\hat{\mathbf{m}}^{l2} - \mathbf{m}^{l2}\|^2, \quad (3.14)$$

exemplary for the left camera. The parameters for the relative transformation $\Delta \mathbf{T}$ are optimized by minimizing the error between the matched feature $\hat{\mathbf{m}}^{l2}$ and the projected point \mathbf{m}^{l2} of the object point $\vec{\mathbf{M}}$, as visualized in Figure 3.9. In the sense of least-squares (Appendix A.1), $\hat{\mathbf{m}}^{l2}$ is the observation and $\vec{\mathbf{M}}$ is the condition. The parameters of $\Delta \mathbf{T}$ with $(t_x, t_y, t_z, \alpha_x, \alpha_y, \alpha_z)^T$ define the model parameters. The model function is defined by the transformation of $\vec{\mathbf{M}}$ by $\Delta \mathbf{T}$ and its projection into l_2 .

The least-squares problem is solved using the Gauss-Newton algorithm (Appendix A.1), for which required Jacobiens with partial derivatives were formulated and implemented by Grießbach (2015). Assuming that only small movements are possible between consecutive camera frames, the identity matrix is used as initial guess and a Levenberg-Marquardt approach is not required. Grießbach (2015) further implemented an analytical error propagation to estimate model parameter uncertainties, propagating feature matching uncertainties and camera model uncertainties through the least-squares solution. The error propagation is only done once during the last

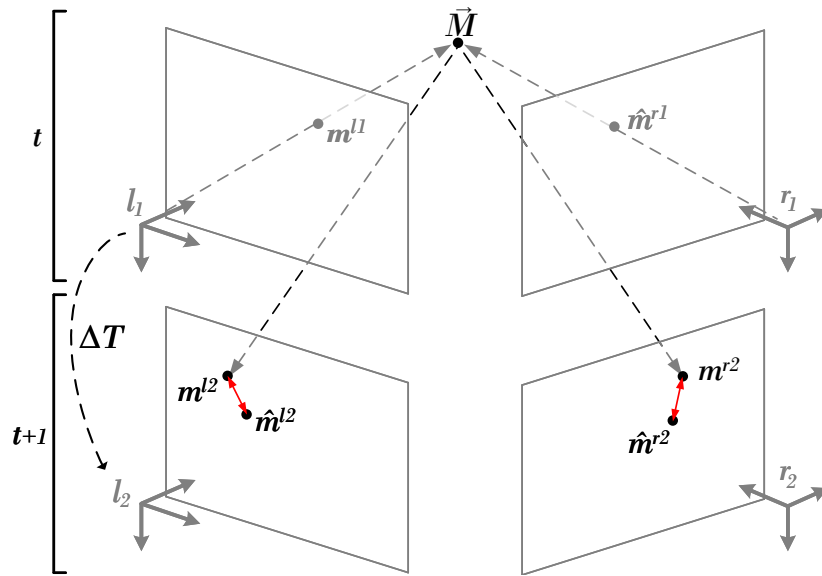


Figure 3.9: The 3D-to-2D VO problem in IPS. ΔT is optimized to minimize the distance (red) between the projection m^{l2} of \hat{M} and the tracked feature m^l2 .

optimization step, where Δa is close to zero and eases the computation, see Griebach (2015, p.86).

Further, single outliers in Equation 3.14 can introduce significant errors or even cause the minimization to fail. A common approach to achieve robustness against outliers is RANSAC (Fischler and Bolles, 1981). It estimates a solution for model parameters based on a randomly sampled minimal set of observations, which are three feature points for the given optimization problem. This solution is assessed by counting inliers from the full feature set to this solution. A feature is counted as an inlier, if the squared distance from their projection to the corresponding matched feature point is less than a user defined threshold. This procedure is done iteratively multiple times, while the best solution is further optimized based on its inlier set to compute the final model parameters.

3.2.3 Inertial Navigation and Data Fusion

“Inertial navigation is a self-contained navigation technique in which measurements provided by accelerometers and gyroscopes are used to track the position and orientation of an object relative to a known starting point, orientation and velocity” (Woodman, 2007). IPS deploys an IMU that consists of each three perpendicular accelerometers and gyroscopes. It is based on MEMS technology, which allows rigid construction and efficient design in terms of small size and weight, power consumption, cost, maintenance and allows operation in hostile environments (Titterton and Weston, 2004; Woodman, 2007), such as experienced with IPS. In this context, this section provides a brief introduction to the IN concept in IPS. However, since the focus of this thesis is on visual localization, the filter is mostly considered as black box in the further course of this thesis and not considered for improvements.

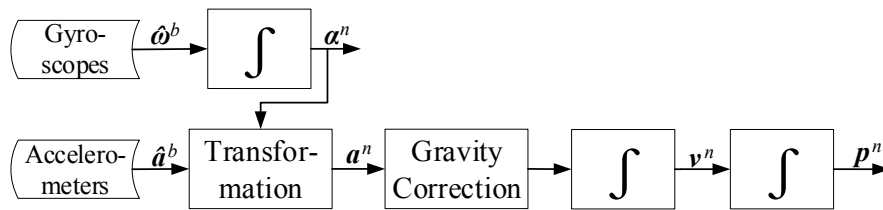


Figure 3.10: Strapdown mechanization, based on Woodman, 2007.

The strapdown mechanism is used to estimate the current 6D pose $\{\alpha^n, \mathbf{p}^n\}$ of the system in the navigation frame based on the measured accelerations $\hat{\mathbf{a}}^b$ and angular velocities $\hat{\boldsymbol{\omega}}^b$ in body frame. Formulas for the IPS specific implementation are detailed in Griebach et al. (2014). Figure 3.10 shows the overall principle of the strapdown mechanism, that consists of five steps. First, the orientation or attitude $\hat{\boldsymbol{\alpha}}^n$ of the system is estimated through integration of $\hat{\boldsymbol{\omega}}$. The estimated orientation is secondly used to project $\hat{\mathbf{a}}^b$ into the navigation frame, where the gravitation is on the z-axis and can be subtracted in the third step. With the acceleration being in the navigation frame, it can be integrated twice, resulting fourth in the velocity and fifth in the 3D position \mathbf{p}^n of the system. Several assumptions are made that require a relatively high data rate of the IMU, especially for the first step. For instance, the small angle assumption is made to simplify the orientation vector differential equation (Wendel, 2011, p. 46). Also, earth rotation is neglected in IPS. Therefore, a sampling period of 2.44 ms was selected in IPS (Griebach et al., 2014), which corresponds to a clock rate of 410 Hz.

Measurements of gyroscopes and accelerations are subjected to perturbations that can lead to drifts when propagated through the strapdown mechanism (Woodman, 2007). Most significantly, the measurements are subject to white noise and a bias that can show a significant constant and instabilities. A calibration is usually provided by the manufacturer. Further error sources are for instance temperature effects and calibration errors, such as errors in scale factors, alignments and output nonlinearities, but are not explicitly considered in the current development stage of IPS navigation.

To additionally estimate the bias terms, a Kalman filter is implemented in IPS that is aided by relative transformation measurements from the visual odometry component. In general, a Kalman filter is a recursive probabilistic state estimation technique to estimate the state of a discrete-data (or continuous in other settings) linear dynamic system based on noisy measurements, which are assumed to be from Gaussian nature. For nonlinear filter problems, the EKF is a popular choice that linearizes about the current mean and covariance using first-order Taylor expansion. However, EKF are considered as ad hoc state estimator (Welch and Bishop, 1995) or suboptimal approximation method (Wendel, 2011), if non-linearities are significant, which would be the case when filtering total-space states in IN. Therefore, an error state space EKF variant is applied in IPS that severely reduces nonlinearities by filtering the error state

$$\Delta \mathbf{x} = (\Delta \boldsymbol{\alpha}_b^n, \Delta \mathbf{p}_b^n, \Delta \mathbf{v}^n, \Delta \mathbf{b}_\omega^b, \Delta \mathbf{b}_a^b)^T, \quad (3.15)$$

which consists of error estimates of the attitude $\boldsymbol{\alpha}_b^n$, position \mathbf{p}_b^n , velocity \mathbf{v}^n and bias terms. The principle is visualized in Figure 3.11, while the mathematical formulation is derived and detailed in Griebach et al. (2014). In the time update step, a priori

state \mathbf{x}^- is predicted based on the strapdown mechanism, where IMU measurements are treated as known input variables. During the measurement update, the error measurement $\Delta\mathbf{y}$ is defined based on the difference of \mathbf{x}^- to the reference measure \mathbf{y} from VO, which is used to filter the error state and update a priori \mathbf{x}^- , resulting in a posteriori \mathbf{x}^+ . The method of stochastic cloning (Roumeliotis and Burdick, 2002) is applied to correctly take correlations between the current and the previous frame into account.

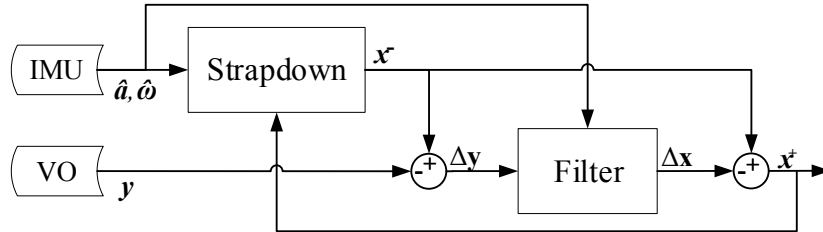


Figure 3.11: Error state spatial navigation filter design (Grießbach et al., 2014).

The backbone of a Kalman filter are propagated uncertainties for measurements and internal states, which are required to correctly weight the influence of the reference measurement. Uncertainties need to be consistent, which means that estimated errors match to their theoretical statistical characteristics with zero mean and correctly scaled covariance (Grießbach et al., 2014). The consistency of uncertainties will be exemplary addressed in Chapter 5 for the VO results based on statistical analysis with simulated data. To reach covariance consistency of the filter solution, the Kalman filter was tuned once in 2014 based on a grid search approach on a set of real world datasets.

False measurements with inconsistent uncertainties from VO can corrupt the filter states and lead to a false navigation solution. To identify significant deviations, a chi-squared test is applied on the innovation $\Delta\mathbf{y}$ to verify that the VO measurement is consistent to the filter state \mathbf{x}^- . The chi-squared test compares the error $\Delta\mathbf{y}$ with the propagated uncertainties based on the Mahalanobis distance. The measurement is ignored if $\chi^2 > c$, while c is set to 22.46. This value results from a chi-squared distribution with 6 degrees of freedom and a significance number of 0.1 %.

IPS requires an initial two-step procedure to initialize the strapdown and filter states, which is essential for an accurate navigation solution. First, the system is set at rest for a short time to average angular velocity measurements in order to estimate their bias terms. Though, the acceleration bias can not directly be estimated, due to the superimposed acceleration from gravity. Therefore, small movements of the system are required with a reference measurement from VO in a second phase. Ideally, each body frame axis is individually aligned to the gravitation direction. This procedure takes around 45s for an experienced user with feedback from the filter.

3.3 Evaluation Metrics

Different evaluation metrics are required to assess the localization performance of IPS. This section briefly introduces required metrics for trajectory and transformation evaluation and the evaluation of propagated uncertainties.

3.3.1 Trajectory Evaluation

The goal of trajectory evaluation is an assessment of the consistency and accuracy of the temporally- and spatially depended consecutively estimated 6D poses. Generally, an end-to-end performance evaluation based on suitable Ground Truth (GT) is preferred over intrinsic measures, such as the reprojection error for SLAM systems (Sturm et al., 2012). However, it is subjected to difficulties, such as the high dimensionality of the data and the definition of GT and estimation in different reference frames (Zhang and Scaramuzza, 2018). Further, the choice of the metric to evaluate an estimated trajectory depends on the available GT data. In this thesis, a complete GT is provided for synthetic datasets, whereas only a few Ground Control Points (GCPs) or even just a few closed loops are available for real-world datasets.

Absolute Trajectory Error (ATE)

In simulation, the ATE (Sturm et al., 2012) is used as primary metric in this thesis to evaluate the final localization output. A visualization is given in Figure 3.12. Its computation composes of two steps. First, trajectory alignment is applied to estimate transformation \mathbf{S} matrix that optimally transforms the estimated trajectory $\{\hat{\mathbf{P}}_i\}_{i=0}^{N-1}$ with N poses to the reference frame of the GT trajectory $\{\mathbf{P}_i\}_{i=0}^{N-1}$. Specifically, the distances between positions of estimated and GT poses are minimized in a least-squares manner with the method of (Umeyama, 1991) using the implementation of (Zhang and Scaramuzza, 2018). Second, distances between corresponding poses are estimated with

$$\text{mATE} := \frac{1}{N} \sum_{i=0}^{N-1} \text{ATE}_i = \frac{1}{N} \sum_{i=0}^{N-1} \|\mathbf{P}_i^{-1} \mathbf{S} \hat{\mathbf{P}}_i\|_T, \quad (3.16)$$

and stated by their mean in this thesis. $\|\cdot\|_T$ describes the norm of the translational components. The advantage of the ATE is that it captures translational and rotational errors simultaneously, since the latter manifest themselves in wrong translations.

In real world, the GT is limited to GCPs $\{\mathbf{p}\}_{i=0}^{M-1}$ with $\mathbf{p} \in \mathbb{R}^3$, which can be used for evaluation using the ATE. GCPs are 3D positions where the system was placed or hold for a short time. Due to the possibility of only very few GCPs being available, two special cases needs to be considered and are visualized in Figure 3.12 (b) and (c). If exactly two GCPs are available, which were reached exactly once, the GT can be simply defined by a distance between both GCPs. The measured distance error is

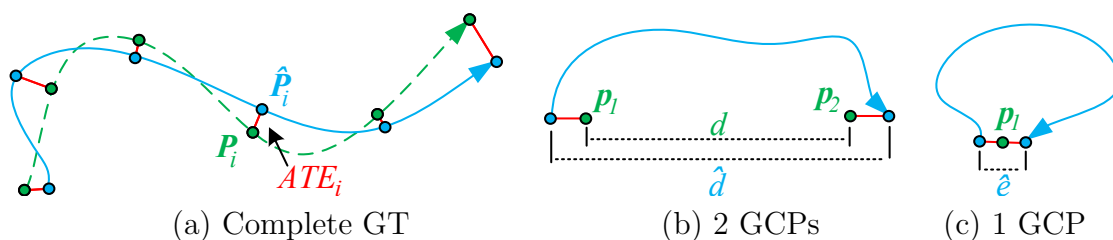


Figure 3.12: Computation of the ATE (red) for the estimated trajectory (blue) that is aligned to the ground truth (green). Three cases are visualized with different number of ground truth data available.

twice the mean ATE. If only one GCP is available which was reached exactly twice, the problem reduces to a closed loop and no GCP parameters need to be defined. The common Closed Loop Error (CLE) is twice the mean ATE.

Normalized Closed Loop Error (nCLE)

Many IPS datasets only consist of a number of closed loops. The comparison of results between many different closed loop trajectories is difficult due to different trajectory lengths. The drift of VO and VIO will lead to a higher CLE, if the trajectory is longer. A common solution is to normalize the CLE by the traveled distance, given with

$$\text{nCLE} = \frac{\|\hat{\mathbf{p}}_{N-1} - \hat{\mathbf{p}}_0\|}{\sum_{i=1}^{N-1} \|\hat{\mathbf{p}}_i - \hat{\mathbf{p}}_{i-1}\|}. \quad (3.17)$$

Relative Translation Error (RTE)

In simulation, the estimated relative transformation from VO can also be evaluated using GT. Error matrix \mathbf{E}_i describes the VO error with

$$\mathbf{E}_i := (\mathbf{P}_{i+1}^{-1} \mathbf{P}_i)^{-1} \hat{\mathbf{T}}_{l,i}^{l,i+1}. \quad (3.18)$$

Following, the computation of the mean RTE is straight forward with

$$\text{mRTE} = \frac{1}{N} \sum_{i=0}^{N-1} \text{RTE}_i = \frac{1}{N} \sum_{i=0}^{N-1} \|\mathbf{E}_i\|_T. \quad (3.19)$$

The rotational error can be computed in a similar way, under the assumption that the small angle approximation holds. Considering the applied 10 Hz image frame rate, this rotational metric is less meaningful, if fast rotations are observable in the dataset.

It is obvious that an investigation based on datasets with such limited amount of GCP is doomed to be of less meaning. Due to this, researchers often expensively generate GT data using motion capture systems or use synthetic data with complete and exact GT. I chose the latter and developed a digital twin.

In literature, there are several alternatives described for trajectory alignment, error formulation and application of statistical metrics. (Zhang and Scaramuzza, 2018) describe additional problem-specific variations of the trajectory alignment. For instance, the scale of \mathbf{S} can be estimated additionally to tackle scale ambiguity of monocular localization methods. Or, since IMU estimates a roll and pitch based on the IMU, a yaw-only optimization for the rotational part can be used. Other error formulations are, for instance, the Relative Pose Error (RPE) that considers the relative error between pairs of poses or the Relative Error (RE) that applies the ATE procedure to sub parts of the trajectory (Zhang and Scaramuzza, 2018). In this thesis, however, the basic ATE is used for most investigations, which is most common and different evaluation metrics are usually highly correlated (Sturm et al., 2012). Common alternatives for the mean are the Root Mean Squared Error (RMSE) or the median, which weight outliers stronger or weaker, respectively.

3.3.2 Error Propagation and Uncertainty Evaluation

A consistent estimated uncertainty for VO is a hard requirement for a loosely-coupled, filter-based VINS. Griebach (2015) implemented analytical error propagation from feature matching and camera model uncertainties throughout the presented VO pipeline. The individual steps with error propagation will be revisited in Chapter 5 in detail. In the following, the main idea of analytical error propagation and two evaluation metrics to assess the quality of the estimated uncertainties are introduced.

Analytical Error Propagation

In error propagation, the goal is to find the distribution of a function of random variables. Specifically, the distribution of an output \mathbf{y} is desired that results from a function f with random variables \mathbf{x} , composed in $\mathbf{y} = f(\mathbf{x})$. Formulated by the law of error propagation with

$$\Sigma_{\mathbf{y}} = \mathbf{F}_{\mathbf{x}} \Sigma_{\mathbf{x}} \mathbf{F}_{\mathbf{x}}^T, \quad (3.20)$$

the input uncertainty $\Sigma_{\mathbf{x}}$ is propagated through the function f with jacobian $\mathbf{F}_{\mathbf{x}}$ and approximately mapped to the output $\Sigma_{\mathbf{y}}$ (Arras, 1998). A linear equation system $\mathbf{y} = \mathbf{F}\mathbf{x}$ can be solved directly.

In the nonlinear case, the function f can be approximated by Taylor series. Based on Arras (1998), the first order approximation results in a mean \mathbf{y}_0 and covariance $\Sigma_{\mathbf{y}}$ formulated as

$$\mathbf{y}_0 = f(\mathbf{x}_0), \quad (3.21)$$

$$\Sigma_{\mathbf{y}} = \mathbf{J}_{\mathbf{x}}^{\mathbf{y}} \Sigma_{\mathbf{x}} \mathbf{J}_{\mathbf{x}}^{\mathbf{y}T}, \quad (3.22)$$

where the Jacobien matrix $\mathbf{J}_{\mathbf{x}}^{\mathbf{y}}$ contains the partial derivatives $J_{ij} = \delta f_i / \delta x_j$.

Analytical error propagation is applicable, if the following simplified conditions are met: *(i)* the distribution of \mathbf{x} can be described by a mean \mathbf{x}_0 and covariance matrix $\Sigma_{\mathbf{x}}$; *(ii)* the SD is relatively small with respect to the range of \mathbf{x} ; *(iii)* f is continuously differentiable for elements \mathbf{x}_i of \mathbf{x} in the neighborhood of \mathbf{x}_0 ; *(iv)* higher order terms of the Taylor series approximation are negligible.

Monte Carlo Simulation

MCS is commonly used as reference to verify functionality of developed methods for uncertainty propagation. It describes a statistical approach for error propagation, that can be applied on general multivariate distributions (JCGM 102, 2011), if a number of conditions are met (JCGM 101, 2008, p.14).

Only multi-variate Gaussian distributions are considered in this thesis, for which the procedure can be summarized in three steps. First, M samples $\{\mathbf{x}_i\}_{i=0}^{M-1}$ are drawn from the joint Probability Density Function (PDF) of \mathbf{x} , defined by \mathbf{x}_0 and $\Sigma_{\mathbf{x}}$. Then, each sample is applied in f to retrieve M output quantities $\{\mathbf{y}_i\}_{i=0}^{M-1}$. Finally, mean $\check{\mathbf{y}}_0$ and covariance $\check{\Sigma}_{\mathbf{y}}$ are estimated using the set of generated quantities $\{\mathbf{y}_i\}_{i=0}^{M-1}$.

This method is applicable, if the following simplified conditions are met: *(i)* the distribution of \mathbf{x} can be described by a mean \mathbf{x}_0 and covariance matrix $\Sigma_{\mathbf{x}}$; *(ii)* f is continuous for elements \mathbf{x}_i of \mathbf{x} in the neighborhood of \mathbf{x}_0 ; *(iii)* the distribution of

\mathbf{y} can be described by a mean $\check{\mathbf{y}}_0$ and covariance matrix $\check{\Sigma}_{\mathbf{y}}$; (iv) a sufficiently large number of samples M are drawn.

The analytically estimated uncertainty can be validated based on a relative test with

$$|\hat{\sigma}_i - \check{\sigma}_i| < \epsilon \cdot \check{\sigma}_i, \quad (3.23)$$

by comparing the analytically propagated SD $\hat{\sigma}_i$ to the statistically propagated SD $\check{\sigma}_i$, with $\hat{\sigma}_i^2 = \Sigma_{\mathbf{y},ii}$ and $\check{\sigma}_i^2 = \check{\Sigma}_{\mathbf{y},ii}$ and a tolerance value ϵ .

Normalized Error

The normalized error (Anderson et al., 2019) can be used to judge the estimated uncertainty, if a large number of estimated quantities are present, which are independent and consist each of an individual (differently scaled) covariance. An accurate GT is required. The error e_i is estimated based on the estimate \hat{x}_i and GT x_i , and then normalized by the estimated SD $\hat{\sigma}_i$ to get the normalized error \tilde{e}_i :

$$e_i = \hat{x}_i - x_i, \quad (3.24)$$

$$\tilde{e}_i = e_i / \hat{\sigma}_i, \quad (3.25)$$

$$\check{\sigma}_i = \text{sd}(\tilde{e}_i). \quad (3.26)$$

Note that e_i is notated in this thesis as the i -th element of an error \mathbf{e} of a measured vector quantity $\hat{\mathbf{x}}$. Further, gross outliers are rejected during this procedure based on the condition¹ $|e_i| < 6\hat{\sigma}_i$, as suggested by Anderson et al., 2019.

Ideally, the resulting distribution of a set of normalized errors \tilde{e}_i should follow a SD with $\check{\sigma}_i \approx 1$, due to normalization. If $\check{\sigma}_i > 1$, the estimated uncertainty $\hat{\sigma}_i$ is optimistic or overconfident, i.e., the error is underestimated. If $\check{\sigma}_i < 1$, the estimated uncertainty $\hat{\sigma}_i$ is conservative or underconfident, i.e., the error is overestimated. The different manifestations of the normalized error are exemplified in Figure 3.13.

This error metric only allows a gross evaluation of the uncertainty, but might offer useful insights, if MCS cannot be applied directly.

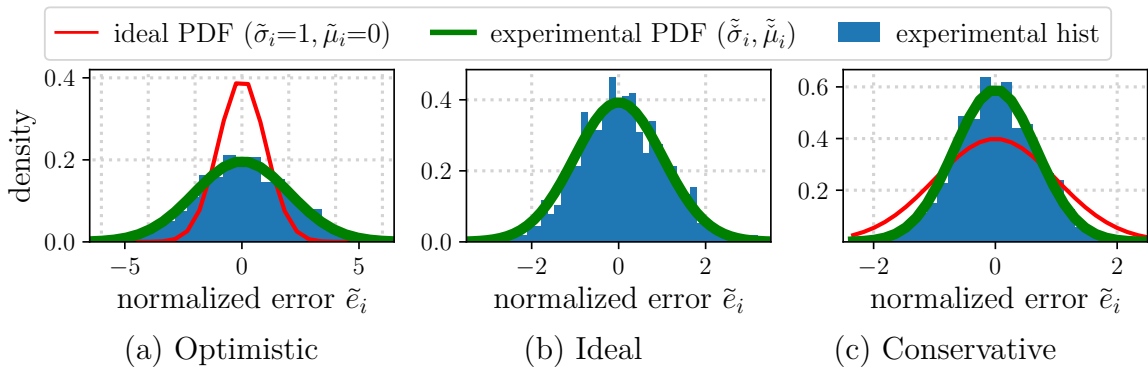


Figure 3.13: Exemplary manifestations of the normalized error.

¹This condition is omitted in Section 5.2.1 due to partly highly optimistic predictions.

3.4 Summary

In this chapter, various outlier rejection methods were introduced that help to make IPS generally robust against outliers and also moving objects. In the following, four aspects are recapitulated that will be revisited in the following chapters.

First, the VO is constrained by inertial measurements. Specifically, the strapdown-based ego-motion prediction is used to predict the feature positions in the second stereo frame images and to restrict the search area based on estimated uncertainties. Consequentially, features on moving objects are rejected beforehand, if they show a larger displacement in image coordinates than the magnitude of the estimated uncertainty.

Second, the VO of IPS uses generally tight bounds for several geometric thresholds. Concerning outliers in general, epipolar constraints are applied during feature tracking with a tight threshold (e.g., 0.8 px) This requires accurate camera calibration.

Third, RANSAC effectively helps to find the subset of data (feature matches) that can be described jointly by a relative transformation. In combination with the extended AGAST detector, which guaranties a well distribution of detected features in the image, the object must be of significant relative size to disturb the system.

Fourth, the estimated VO can be rejected by the Kalman filter based on the chi-square test, if the new measurement does not fit to the internal state with the given propagated uncertainties.

Chapter 4

A Digital Twin for IPS

The ability to test methods based on simulated data is important for the development of localization systems that have to operate reliably in hazardous environments. This includes the replication of realistic environments and sensors on the one hand. On the other hand, an implementation of a correct movement profile of the mobile platform is required. This is complicated by the large variety of possible platform variants.

In (Irmisch et al., 2019), we presented a method to transfer movement profiles that have been recorded in real world into a simulation environment. The associated modular simulation framework was designed especially, but not limiting, for the development of VIO of the handheld localization system IPS. The digital twin is used to generate *synthetic video clones* (Gaidon et al., 2016), exemplified in Figure 4.1.

This chapter extends this work by two aspects. First, the simulation tool is used to replicate three dynamic real-world scenarios, whose procedure and dynamic elements are explained. Second, different strategies to evaluate and analyze methods for visual localization with statistical meaning are presented, which are used in later chapters.

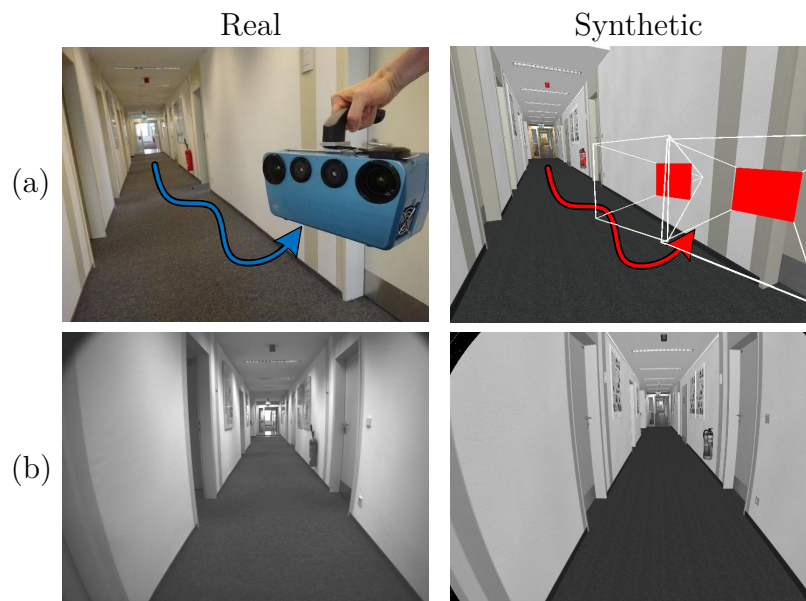


Figure 4.1: (a) Illustration of using the estimated trajectory from the real IPS (blue) for a digital twin (red). (b) Exemplary real and synthetic camera images.

4.1 Simulation Framework

This section describes the three processing steps of the digital twin, shown in Figure 4.2, and names additional generated data that can be used for evaluation purposes.

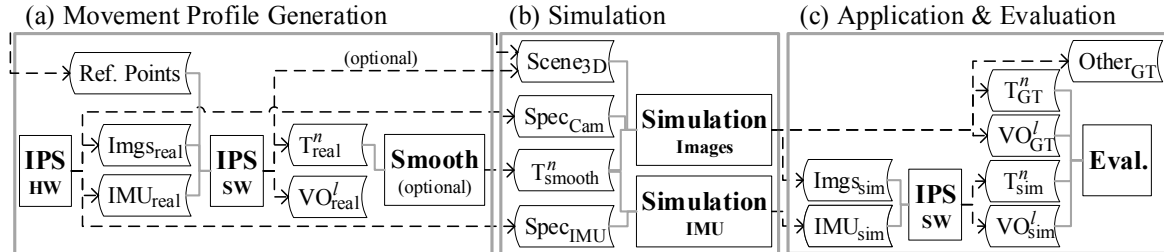


Figure 4.2: Proposed pipeline of the digital twin. An estimated trajectory is transferred from real world into simulation (a). Synthetic images and IMU data are generated (b). IPS is applied and evaluated based on synthetic data and complete ground truth (c).

4.1.1 Movement Profile Generation

The objective is to create a trajectory with a realistic motion profile, such as motion from walking with a hand-held system. The required steps are noted in Figure 4.2 (a).

First, a handheld device (IPS HW) is used to record stereo images and IMU data. Second, the stereo-vision aided inertial navigation engine (IPS SW) estimates a trajectory, that consists of time-aligned six degrees-of-freedom poses and related covariance matrices with respect to a local reference frame. Due to the relative nature of VO, the positional uncertainty in the fusion process with IMU data continues to increase over time (Figure 4.3 (a)). An unbounded increase and a possible deviation in the position estimation can lead to incorrectly simulated camera poses that intersect with walls or the ground. This is prevented by inserting absolute measurements, for instance at the end of the run (Figure 4.3 (b) at 72s). Third, the smoothed trajectory T_{Smooth}^n can be determined by the Rauch-Tung-Striebel algorithm (Rauch et al., 1965). It uses the stored a priori and a posteriori state estimates including their covariances from the navigation engine forward pass to calculate optimal estimates and covariances in a second backward pass. T_{Smooth}^n serves as motion profile in simulation (Figure 4.2 (b)).

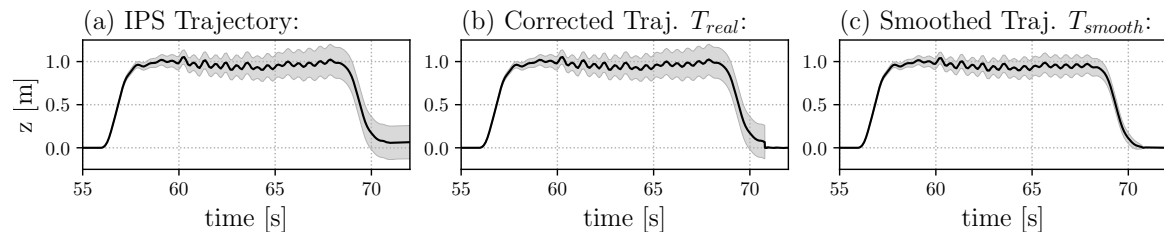


Figure 4.3: Processing steps of movement profile generation: basic IPS trajectory in z direction (a), aiding with reference points (b) and subsequent smoothing (c). The trajectory (black line) shows the pickup of the IPS, walking straight for 12s and putting it down. The associated standard deviation (grey area) was scaled for visualization.

The second and third step can be replaced by a bundle adjustment approach that simultaneously estimates a 3D model and a fitting trajectory, see Section 4.2.

4.1.2 Rendering of Camera Images

The objective is to simulate stereo camera images based on real world geometric and radiometric calibrations, including image distortion and noise. Image simulation is based on the standard rendering pipeline and was extended with subsequent shaders to simulate sensor effects. [OSG] was chosen as the rendering platform. It provides free and open source accessibility, high integrability in C++ frameworks and extensive expansion options due to its scene graph technology. The implemented virtual stereo camera originates from (Lehmann, 2015, 2016) and was improved in (Irmisch, 2017) with respect to image quality. In this thesis, the virtual camera was validated and integrated into the developed simulation tool with several extensions.

The standard rendering pipeline is used “to generate, or render, a two-dimensional image, given a virtual camera, three-dimensional objects, light sources, shading equations, textures and more” (Akenine-Möller et al., 2008). This pipeline consists of three conceptual steps, illustrated in Figure 4.4. First, the geometry of the scene is set in the application step for a defined timestamp, including objects, positions and movements, described as $Scene_{3D}$ in Figure 4.2. The timestamps to render are determined based on the frequency and exposure time of the target camera $Spec_{Cam}$. The poses are extracted from a cubic spline of T_{smooth}^n , detailed in Section 4.1.3, and taking into account the rigid relative transformation T_l^n . Mentioned objects consist of points (vertices), lines and faces, where the latter are each defined by three vertices and represent the surface of the object. Second, all positions of the objects to be rendered are projected into image coordinates during the geometry stage based on a normalized camera model, while vertices that are not bordered by the image are clipped. Last, the rasterizer stage uses the projected vertices to compute the nearest face and set colors for each pixel defined by the face. The result is an ideal image that coincides with the pinhole model, as shown in Figure 4.4 (iii) or Figure 4.5 (ii).

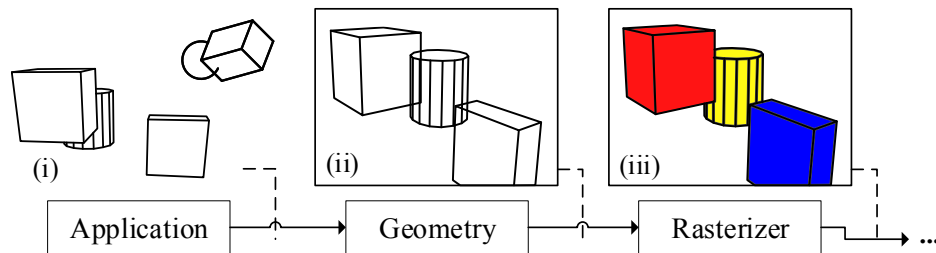


Figure 4.4: The graphic rendering pipeline (illustration from Irmisch, 2017, p.18).

Sufficient image quality is required for a meaningful investigation of computer vision algorithms. It can be ensured by texture filtering (mip-mapping), anti-aliasing (multi-sampling) and rendering in high resolution (super-sampling), see (Irmisch, 2017, p.19). Further, horizontal and vertical dimensions of the viewing frustum are set to cover the corresponding border pixels resulting from the camera distortion, to facilitate the largest possible image coverage after distortion.

The extended rendering pipeline introduces three additional steps, which are realized by subsequent fragment shaders and visualized in Figure 4.5. First, the *lens-shader* simultaneously distorts the ideal camera image based on the Brown distortion model and down samples the high-resolution image with extended image borders (w', h') to the original image size (w, h). From an implementational perspective, each pixel of (iii) needs to be undistorted to sample at the correct position in image (ii). For efficiency, this step is based on a pre-computed lookup table, since undistortion requires an iterative non-linear solving algorithm, such as Gauss-Newton implemented in [OSLib]. Second, an *accumulation buffer* (Navarro et al., 2011) is implemented to integrate a defined odd number ($2n + 1$) of distorted images that are rendered within a given exposure time to simulate motion blur. The timestamps for the integrated images are equally placed in the exposure interval and the pose of image ($n + 1$) is stored as GT. Motion blur requires high computing effort and therefore, it is used sparsely in this thesis. Third, the *sensor-shader* is used to model various image degradation effects. In this work, it includes blurring with a Gaussian kernel, greyscaling and image noise based on (Zhang, 2018), described in Section 3.1.2.

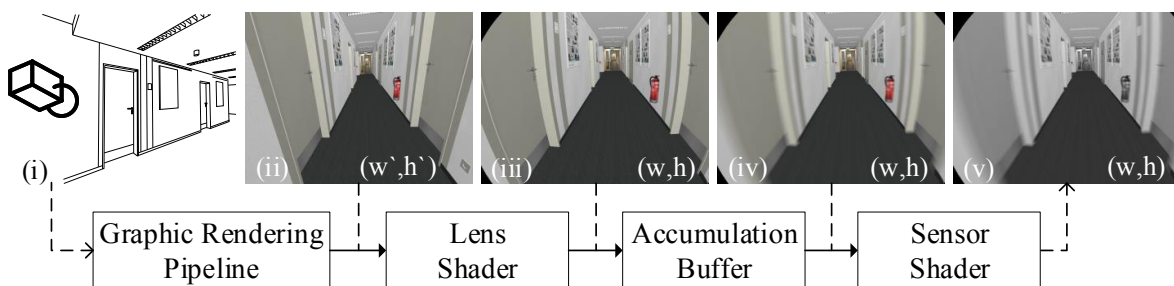


Figure 4.5: The extended rendering pipeline (extending figure of Irmisch, 2017, p.19).

The modular implementation allows the simulation of any number of cameras with individual exposure times, frequencies, time-offsets, grey scaling, interior and exterior geometric camera parameters and the radiometric property noise. The camera can be attached to any movement profile and rendered in any user-defined synthetic 3D-world.

4.1.3 IMU Simulation

The objective is to create degraded synthetic acceleration \mathbf{a}^b and angular rate $\boldsymbol{\omega}^b$ measurements in body frame b based on a given discrete trajectory, realized in three steps. This component was implemented by co-authors of (Irmisch et al., 2019) [OSLib].

First, the objective requires a representation of a discrete trajectory as a continuous function with C^2 continuity, i.e., continuous derivatives up to second order. The considered discrete trajectory T_{Smooth}^n consists of sampled poses for camera rendering based on T_{Smooth}^n and equals T_{GT}^n . C^2 continuity for continuous speed and acceleration are fulfilled by cubic spline interpolation for positions [SciPy]. Similar, the C^2 continuity for continuous angular velocity and acceleration are met by cubic spline interpolation for rotation vectors [pyins]. While Euler vector representation is used, the difference to generated trajectories based on other rotation matrix representations is negligible for the used multiple-point interpolation with points spaced close together (Kang and

Park, 1999). Nevertheless, future IMU data simulation could profit from a trajectory with improved properties from higher order rigid body motion interpolation methods (Haarbach et al., 2018).

Second, based on the cubic spline functions, new discrete poses and their derivatives can be extracted for any intermediate points in time and at arbitrary frequencies. The desired angular velocity $\boldsymbol{\omega}^b$ is sampled directly. The acceleration \boldsymbol{a}^n is sampled in the navigation frame n , superimposed by gravity \boldsymbol{g} and rotated into the body frame b based on the corresponding sampled Euler angles, resulting in \boldsymbol{a}^b .

Third, perfect IMU measurements need to be degraded, since real gyroscope and acceleration measurements are subject to deterministic and stochastic errors such as from constant bias, white noise, temperature effects, calibration or bias instabilities (Woodman, 2007). The applied error model (Wendel, 2011, p. 68) is formulated as

$$\hat{\boldsymbol{x}}^b = \boldsymbol{M}\boldsymbol{x}^b + \boldsymbol{b}_x + \boldsymbol{n}_x, \quad \text{with } \boldsymbol{M} = \begin{pmatrix} s_x & \delta_{z_x} & -\delta_{y_x} \\ -\delta_{z_y} & s_y & \delta_{x_y} \\ \delta_{y_z} & -\delta_{x_z} & s_z \end{pmatrix}. \quad (4.1)$$

Vectors $\hat{\boldsymbol{x}}$ and \boldsymbol{x} denote the measured and true quantities of either the accelerations or angular rates. \boldsymbol{M} denotes misalignments of sensor axes for off-diagonal elements and scale error \boldsymbol{s} on the main diagonal. The bias \boldsymbol{b}_x consists additively of a constant \boldsymbol{b}_c and an instability component \boldsymbol{b}_i . \boldsymbol{n}_x denotes zero-mean Gaussian noise. We consider \boldsymbol{s} , \boldsymbol{b}_c and \boldsymbol{n}_x in the current implementation based on Table 3.1 (p.22).

Figure 4.6 shows the effect of sensor degradation on the IN solution based on basic strapdown. The result worsens with increasing number of added error sources. Finally, it shows the application of the EKF approach with aid from VO that successfully corrects the pure IN.

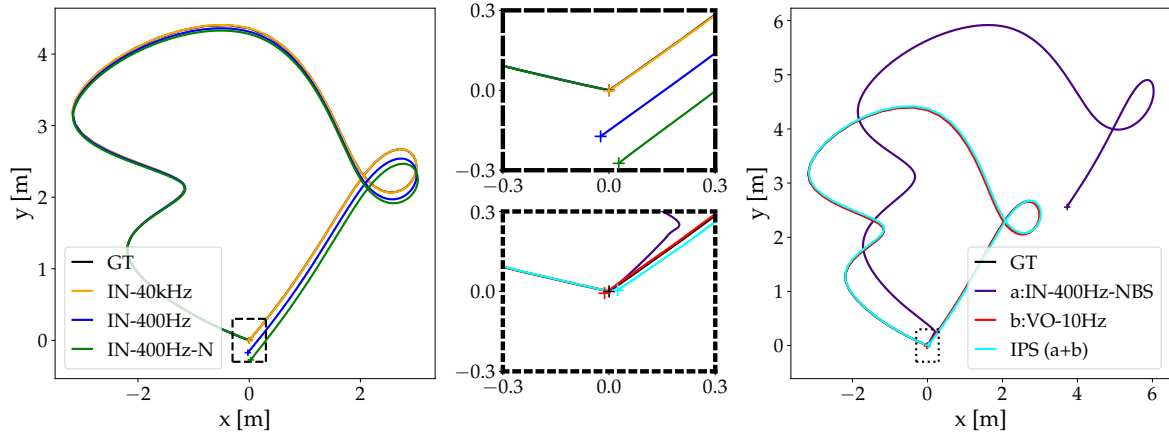


Figure 4.6: Inertial navigation (IN) based on strapdown with degrading synthetic IMU data. Left: IN at 40 kHz fits the ground truth (GT) well (IN-40kHz). Using 400 Hz for IN degrades the results (IN-400Hz). The error increases when adding noise (IN-400Hz-N). Right: An additional constant bias and scale error results in a large drift (IN-400Hz-NBS). Meanwhile, the integrated estimated visual odometry (VO) results in an accurate trajectory (VO-10Hz). Using an EKF with VO as aid corrects the IN (IPS).

4.1.4 Additional Ground Truth Data

The objective is to extract versatile information that can be used to generate substantial Ground Truth (GT) information. This includes absolute and relative transformations, semantic segmentation, depth maps, optical flow and point clouds, see Figure 4.7.

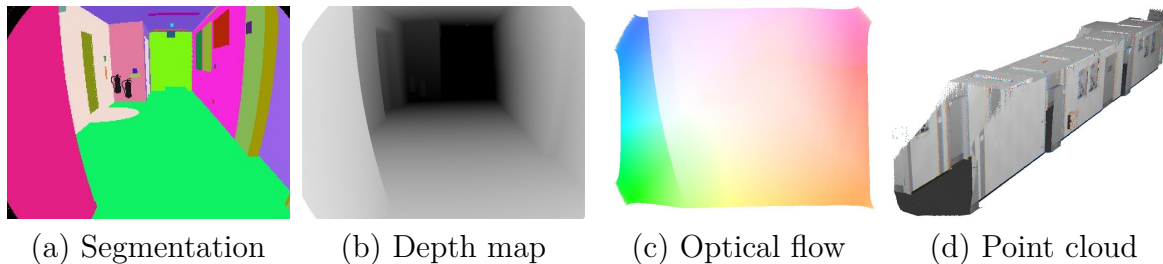


Figure 4.7: Synthetic image-based GT data from the presented simulation tool.

First, *GT trajectory* T_{GT}^n and *GT visual odometry transformations* VO_{GT}^l are generated based on a generated logfile. It holds intermediate representations of the scene graph at all rendered times by saving the structure and all matrix transformations

GT semantic segmentation can be generated. Similar to (Gaidon et al., 2016), the scene is rendered a second time to generate time-consistent per-pixel category- and instance-level semantic GT. During the second rendering, all lighting, shading and material effects are disabled and a unique label is assigned to each object, decoded in a RGB color value and set as ambient material property. Though, this procedure is not applicable for partially transparent objects, such as smoke (Section 4.2.2) or fire, as observed by Jeon et al., 2019. If transparent foreground particles do not change the received color from the background object, due to a very low alpha value of the particle texture, the label of the background object should be used. Therefore, during second rendering, smoke is simulated identically to the first rendering. All pixels with RGB values that do not belong to known background objects get the label smoke.

GT depth maps can be extracted from the z-buffer, which is used in the rasterizer step to identify the nearest object. Its value represents the depth between the frustum near z_{near} and far z_{far} plane, while higher precision is preserved for near values due to perspective division (OpenGL, 2012). z_{near} is set to 5 cm within this work, since ground objects appear close, if the hand-held system is put on the ground. This might lead to an ambiguous order of objects in far distance (z-fighting, Akenine-Möller et al., 2008) and needs to be checked during the design of experiments. The extracted depth map is distorted to match the RGB image. Nearest-neighbor interpolation is used to avoid averaging depth values from fore- and background objects at object edges. A high super-sampling factor of 5 is used to reduce resulting inaccuracies from interpolation.

GT optical flow can be generated that describes the displacement of a projected object point between two images. Similarly to (Wang et al., 2020), the optical flow is computed using logged camera poses and depth maps. A related procedure will be presented in Section 5.2.1. Additionally, moving objects are identified using GT semantic segmentation to compensate for their relative movements between considered timestamps. Though, the implemented GT optical flow is limited to static objects and moving rigid objects, which is not the case for person and particles. Finally, a *GT point cloud* can be generated using logged camera poses, calibration and depth maps.

4.2 Synthetic Datasets

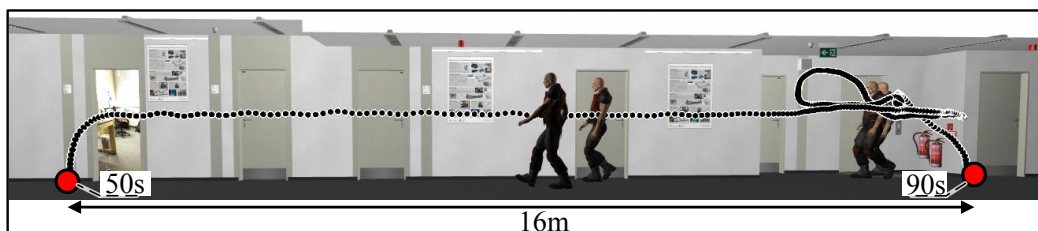
The synthetic video clone closely resembles real-world data (Gaidon et al., 2016). In this thesis, three synthetic clones were generated based on real-world datasets that can be used for detailed analysis by changing different properties. This chapter describes the construction process of each clone. While the corridor datasets was constructed by hand, the other two were automatically generated using a photogrammetry tool. Dynamic elements were added manually, including persons, smoke and water. All datasets start with an initialization phase, which lasts up to one minute to facilitate a good initialization without examination by an expert.

4.2.1 Synthetic Corridor Dataset

The 3D model for the corridor dataset was created manually and recreates an office corridor of the DLR department in Berlin Adlershof in detail. The position and structure of real objects were measured using measuring tapes and laser range finders and used to reconstruct the objects in [Blender]. Textures and materials were chosen and adapted to visually match recorded real imagery data. The corridor has a length of 21.5 m, height of 2.5 m and width of 1.7 m in the beginning.

Trajectories were estimated and recorded following the procedure described in Section 4.1.1. Two GCPs were introduced with a distance of around 16m. A rigid metal construction was installed on the ground to ensure reproducibility. The initialization phase is an inherent part of each recorded trajectory.

The dynamic element person was introduced manually. An animated human model [XNA] was used in the experiments, which is limited to walking straight with adjustable speed. Two adjustable dynamic scene properties were defined: P_s describes walking speed of the person; P_h describes height of the person (e.g., 1.95 m).



(a) Synthetic corridor scene



(b) Real

(c) Synthetic at t_1

(d) Synthetic at t_2

Figure 4.8: Visualization of the synthetic corridor environment with optional persons. In (c,d) two persons are observed slowly starting moving, where $t_2 - t_1 = 4s$.

4.2.2 Synthetic Smoke Dataset

The 3D model and trajectory were generated simultaneously with the photogrammetry tool [Pix4D] and recreate a part of the active volcanic fumarole fields of Vulcano, Sicily (Irmisch et al., 2021). Fumaroles are openings (vents) in the surface that emit steam and corrosive volcanic gases due to underground thermal volcanic activity. The 3D reconstruction is based on the recorded images of IPS in full resolution and uses the estimated local trajectory and calibration parameters of IPS as a priori. The advantage is a fast, automated, high quality and consistent 3D modeling that intuitively only reconstructs surfaces in detail that are observed based on the used trajectory and images. A disadvantage is that it can only be used for this specific trajectory. The initialization sequence is added manually in the beginning of the trajectory.

The dynamic element smoke was introduced manually. Therefore, the particle editor of McDowell et al. (2006) was used to design smoke that visually matches the appearance of smoke (correct: vapor) in recorded real imagery data of fumaroles. The texture of each particle is a simple grey pattern with a two-dimensional Gaussian-like distributed alpha, which is high in the middle and zero at texture borders. More complex or varying particle patterns result in flickering of the smoke in subsequent images, which might be due to z-fighting of close particles. Due to this, the synthetic smoke only shows soft edges and primarily softens the edges from background objects. While the synthetic smoke looks relatively realistic in Figure 4.9, this possible gap to real data needs to be kept in mind when evaluating vision algorithms. Two adjustable dynamic scene properties were defined: S_{PS} describes the particle speed that proportional adapts the number of particles to keep a constant smoke density; S_N describes a factor for the alpha value of the particles texture and is used to adjust the smoke density.

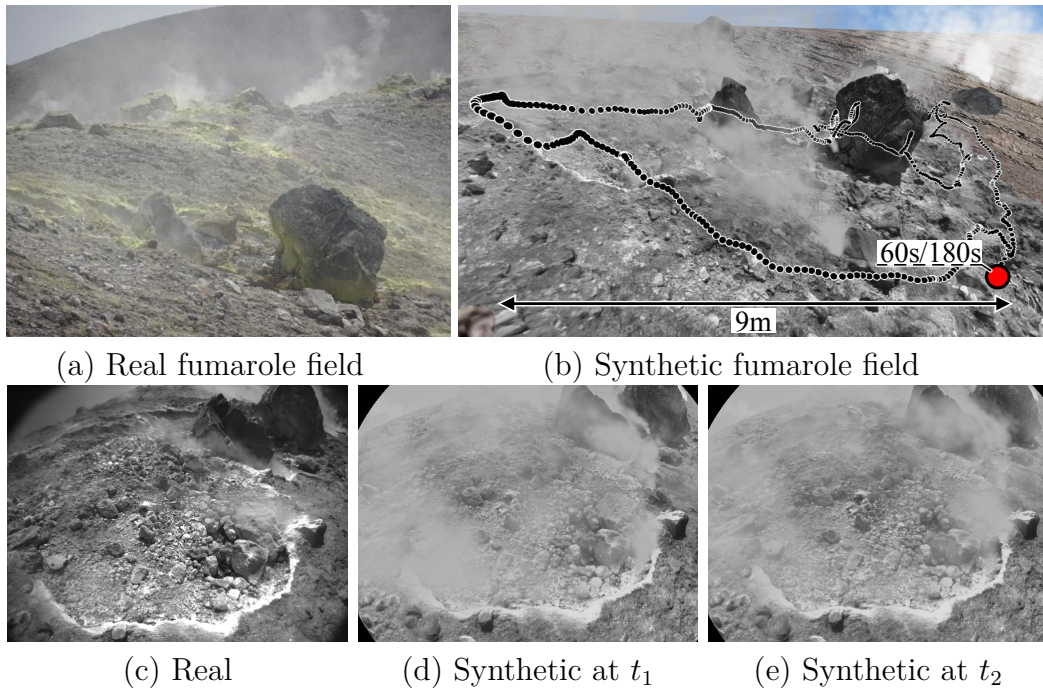


Figure 4.9: Visualization of the synthetic fumarole environment with optional smoke. In (d,e) the development of smoke is exemplified, where $t_2 - t_1 = 1s$.

4.2.3 Synthetic Water Dataset

The 3D model and trajectory were generated similarly to the fumarole dataset and recreates a part of the volcanic coast of Vulcano, Sicily. The coast formation was created by a slowly cooling laval flow, where the typical polygonal joint pattern of basalt columns can be observed at the steep wall in Figure 4.10 (a). The real dataset targets to inspect the coast line, while the presence of water was kept to a minimum during recording to avoid degrading effects by water dynamics. To increase the severity of the water element in simulation, the trajectory was moved 1m closer to the water.

The dynamic element water was introduced manually. The basic water implementation of Wang and Qian (2012, p.262) is used that simulates water based on texture operations on a 2D plane. It first generates a reflection map for the water surface that shows the rendered scene mirrored at the water plane. A normal map and its derivative (du/dv map) with wave structures is then applied to distort the reflection in order to create water effects. To sample from both maps, flow and ripple coordinates are estimated based on current texture coordinate \mathbf{c}_{curr} and current time t with $\mathbf{t} = (t, t)^T$, as exemplified for the flow component in Equation 4.2. W_{flow} is introduced to control water flow speed, where the water flows slow or fast, if W_{flow} is low or high, respectively.

$$\mathbf{c}_{flow} = 5.0 \cdot \mathbf{c}_{curr} + 0.02 \cdot W_{flow} \cdot \mathbf{t} \quad (4.2)$$

In the same way, W_{ripple} is introduced to vary the speed component of the ripple effect, which basically adds a repetitive back and forth motion. Further relations and equations can be looked up in the open source package of Wang and Qian (2012). Finally, W_{scale} is introduced to control the size of waves by scaling the water plane.

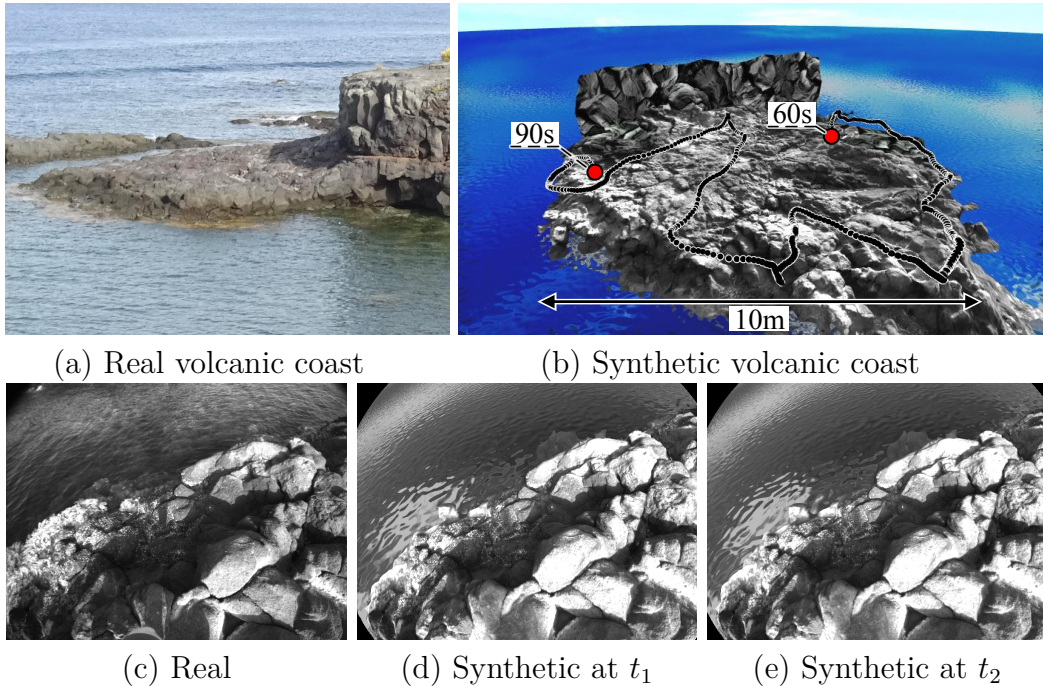


Figure 4.10: Visualization of the synthetic coast environment with optional water. In (d,e) the development of water is exemplified, where $t_2 - t_1 = 1s$.

4.3 Simulation Strategies

The application of a computer vision algorithm on synthetic data can provide substantial insights with the right choice of simulation and evaluation strategy. It can provide clues of how this method depend on specific parameters and how the robustness can be increased. This can be attributed to the substantial GT information and absolute control over system and environment parameters. The following parameter types are considered in this thesis:

- P_E *Environment parameters*: either control the appearance of the environment (static) or the proportion of dynamic elements contributing to the scene (dynamic). Applied during simulation.
- P_D *System design parameters*: control design choices of the constructed virtual system, such as the dimension of the stereo baseline B . Applied during simulation.
- P_P *Sensor property parameters*: control the degradation of sensor data, such as camera capture gain, which affects image noise. Applied during simulation.
- P_C *Calibration error parameters*: control the severity of calibration errors by falsifying true simulated values. Applied during application.

Three different strategies are presented in this section and are used with different goals in this thesis. This includes the basic *sensitivity analysis*, a *geometric MCS*, and a proposed Monte-Carlo-based *combined sensitivity analysis*. They are based on the same overall structure, but differ in the used type of parameters, the number of parameters and the evaluation method. Hereinafter, each strategy is introduced with an explanation of its goal, its simulation procedure and its specific evaluation method.

4.3.1 Sensitivity Analysis

The main objective is a quantitative study of the impact of single factors on the considered computer vision method. A side objective is the analysis of the method under difficult conditions that are rarely observable in real world. This approach is common and the objectives are also addressed as “ceteris paribus analysis” (main objective) or “what-if analysis” (side objective) by (Gaidon et al., 2016).

Figure 4.11 (left) shows the conceptual design. One parameter is selected and set to m distinctive values. The value is added to modify the corresponding configuration

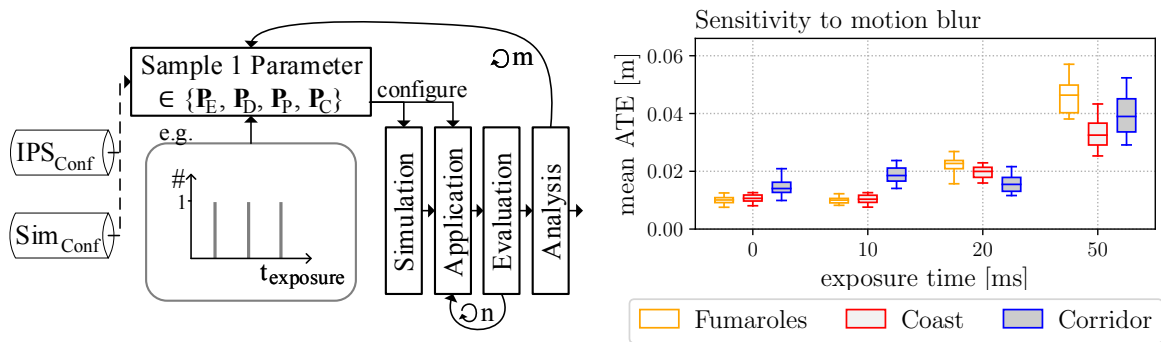


Figure 4.11: Conceptual design (left) of the strategy *sensitivity analysis* with an exemplary statistical analysis (right). The explanation is provided in the text.

files Sim_{Conf} or IPS_{Conf} for subsequent data simulation and application of IPS. Each estimated trajectory is evaluated based on GT from the synthetic dataset. Application and Evaluation is repeated n times to account for the non-deterministic nature of RANSAC in the VO component. The n results are then used for analysis based on some statistical metric, which is done for each of the m parameter settings. This simulation strategy is used in Section 7.2.

The statistical data can for instance be visualized in a box plot with the varied parameter on the x-axis and resulting error on the y-axis. One example is shown in Figure 4.11 (right) for the environment parameter *motion blur* that was introduced based on four different exposure times. One IPS setting is evaluated based on the mean ATE in the different environments *fumaroles*, *coast*, *corridor* and with $m = 50$ repetitions. Similar sensitivity experiments are provided and explained in Appendix B.1.

4.3.2 Geometric Monte Carlo Simulation

One characteristic of the strategy *sensitivity analysis* is that it does not consider calibration errors, which, however, can have a strong influence on the navigation result. This severely increases the gap between real world and simulation, especially if the evaluated algorithm makes assumptions about the existence of errors. Therefore, the strategy *geometric MCS* is proposed to evaluate the considered navigation method based on simulated datasets in combination with a MCS to model errors from extrinsic and intrinsic calibration parameters. The overall objective is to introduce calibration errors into the analysis process to provide a meaningful comparison of methods and to enable an evaluation of propagated uncertainties.

The procedure consists of three steps, which are illustrated in Figure 4.12. First, sensor data for one complete run is simulated based on perfect calibration parameters. Second, IPS is applied and evaluated on this run n -times. For each application, a calibration parameter set is sampled once from their defined error distributions, i.e., altogether n sets are sampled. Third, the estimated values can be analyzed based on their accuracy and uncertainty.

The accuracy analysis is based on common metrics for trajectory evaluation (Section 3.3.1). The uncertainty of VO estimations is evaluated based on the normalized error (Section 3.3.2), which provides a rough impression about the uncertainty qual-

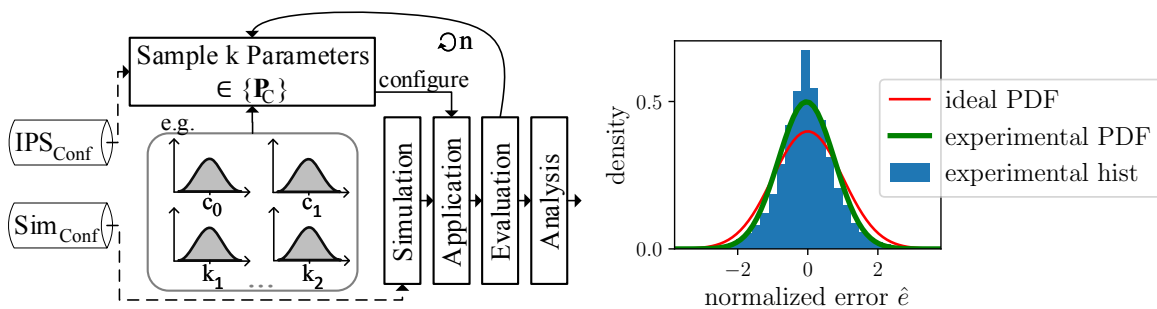


Figure 4.12: Conceptual design (left) of the *geometric Monte Carlo Simulation* with an exemplary statistical analysis (right). The explanation is provided in the text.

ity. The information about the considered calibration error distributions are provided to the specific localization method in order to provide a comprehensive evaluation of propagated uncertainties. Further, only measurements during motion of the system are considered for uncertainty evaluation, because a system at stand still can result in VO measurements that show a systematic error and would distort the statistic with a bias. Figure 4.12 (right) shows an exemplary analysis of the normalized error \hat{e} , which experimental distribution ideally should match the estimated distribution with $\tilde{\sigma} \approx 1$ (Section 3.3.2). This approach will be used in Section 5.3.1.

The difference to a basic MCS is that detected features, feature matching errors and system movements are not based on a theoretical or random generation, but based on a realistic setup and can be considered in an end-to-end setup. The approach is applied in Section 5.3.1 to evaluate uncertainty assumptions and error propagation in the VO component of IPS.

4.3.3 Combined Sensitivity Analysis

A characteristic of the previous two strategies is that they only observe one state of the highly complex system that the real world represents. Even small deviation in the observed scene, such as the size of a moving object, might completely change the navigation results. This severely limits the meaningfulness of the two previous presented strategies. Therefore, the strategy *combined sensitivity analysis* is proposed that jointly considers environment, system design, sensor property and calibration parameters for simulation and application in a Monte Carlo manner. This approach allows to cover a large set of possible scenarios and enables a statistical evaluation of the considered localization methods.

The procedure is illustrated in Figure 4.13 (left). First, n parameter sets are sampled for simulation and application based on defined distributions for $\mathbf{P}_{E,D,P,C}$. In the current implementation, they are mostly defined as Gaussian distributions and within a reasonable range for each individual parameter. Second, the datasets are simulated and the localization method is applied for each sampled parameter set and the ATE is estimated. Third, a correlation analysis between the varied parameters and the mean ATE is conducted to quickly find the potentially most influencing parameters, such as exemplified in Figure 4.13 (right). Fourth, a sensitivity analysis is conducted based on

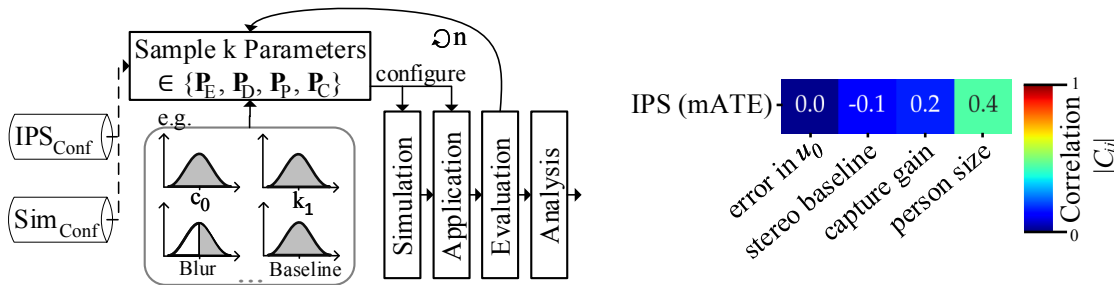


Figure 4.13: Conceptual design (left) of the strategy *combined sensitivity analysis* with an exemplary correlation analysis (right). The explanation is provided in the text.

scatter plots for the most influential parameters to validate the observations from the correlation analysis. This strategy is used in Section 7.2.2 and Section 8.3.

The correlation analysis is based on the Pearson correlation coefficient that describes the degree of linear relationship between two variables. The two fundamental assumptions are normality and linearity of the data. To meet normality, the parameters are preferably sampled from normal distributions. Meeting linearity is more difficult, since calibration errors lead to higher mean ATE either with a positive or negative error value. Therefore, only the absolute value of calibration error parameters are used during correlation analysis. All in all, the translation from sampled input parameters to the final mean ATE is highly nonlinear, which severely reduces meaningfulness of the correlation coefficient. Nevertheless, it shows to provide important insights in later experiments, which can be verified by observing the scatter plot.

4.4 Summary

In this section, the developed digital twin of IPS was presented that is used to generate synthetic video clones of scenarios that were recorded with the real-world system. Three synthetic environments were presented, which will be used in the following chapters for in-depth analysis. Furthermore, three simulation strategies are presented that target different objectives and will support the investigation in the following chapters. The potential gap between the real-world and synthetic datasets and experiments will be discussed in Section 9.3.

Chapter 5

Analysis and Improvement of Uncertainty Estimation in Visual Odometry

Measurement results require a reliable estimate of uncertainty to be used in subsequent decision making. This applies equally to localization for inspection and by first responders. For instance, the visualization of uncertainty regions for position monitoring of first responders in the command and control center can increase efficiency and helps to build trust in the system (Rantakokko et al., 2011).

In IPS, uncertainty estimation based on analytical error propagation is an essential component. It is required for the Kalman filter that predicts the current pose with assigned uncertainties. The filter can only generate reliable results, if all input measurements provide consistent uncertainties that match their true error distribution. Particularly, the quality of a VO estimate is highly variable and depends on the number and distribution of features in the image, feature matching quality and camera calibration accuracy. Therefore, the VO implementation includes an analytical error propagation to propagate uncertainties from feature matching and camera calibration through the least-squares solution. This method was developed by (Grießbach, 2015) and expanded by (Zhang, 2018) for error propagation from image noise through the feature matching process. Each individual step was verified by MCS and the overall IPS has proven itself by successful use in many applications over years. Though, the quality and consistency of the VO measurements itself, including all VO steps, assumptions and approximations, has not been verified.

This chapter has two aims, which form the basis for investigations in the following chapters. First, a way is needed in the evaluation process to account for degradation in geometric calibration, which, for example, is evoked by limitations in the calibration process or by high physical stress on the system in a first responder context. This is important to assess the influence of calibration errors for a specific application and to guide the focus of future developments to increase the overall robustness of the system. Second, the VO pipeline is revisited with focus on error propagation to assess the quality of propagated uncertainties. This is important, because system uncertainties are brought into relation with the influence of moving objects in Section 7.2.1 and are exploited to identify distracting image regions in Section 8.1.1.

This chapter consists of four sections. Section 5.1 defines and reasons for two geometric calibration settings that differ in the severity of calibration errors. Specifically, a calibration for Inspection (I) is defined that assumes small errors and a calibration for First responders (F) is defined that assumes large errors. Section 5.2 revisits the VO pipeline with focus on uncertainty propagation and includes three related and proposed minor improvements, which will be used during application of IPS in the following chapters. In Section 5.3, the *geometric MCS* is used to analyze the influence of calibration errors on IPS and to assess the propagated uncertainties. Section 5.4 discusses the results and the currently used concept in IPS for error propagation with focus on propagating calibration uncertainties. Finally, a short resume summarizes the main insights and consequences for the following chapters.

5.1 Geometric System Calibration

Geometric system calibration is a mandatory requirement in photogrammetry and is essential for accurate visual localization. In IPS, this concerns the spatial transformations \mathbf{T}_l^b and \mathbf{T}_l^r (Section 3.1.1), the camera model parameters $\boldsymbol{\kappa}=(u_0, v_0, f_u, f_v)^T$ and the camera distortion parameters $\boldsymbol{\delta}=(k_1, k_2, k_3, p_1, p_2)^T$ (Section 3.1.2). The calibration itself is performed beforehand and is usually based on a least-squares optimization. The quality of the calibration depends strongly on the used setup, camera properties, and number of images and number of corners of the used chessboard calibration pattern (other options do exist, see Wohlfeil et al., 2019). The outcome are calibration parameters and a covariance matrix.

In this section, two representative calibration settings (I , F) are derived for the application of IPS in inspection and by first responders. The first section describes the selected calibration setups based on laboratory calibration for I and based on in-situ calibration for F . The second section discusses challenges that can degrade the geometric calibration during operation and representative error distributions are derived, which are used in following investigations.

5.1.1 Camera Calibration for Inspection and First Responder

The objective of this section is to choose calibration settings that relate to the applications inspection and first responder. For inspection, I assume that the system can be regularly and thoroughly calibrated in a laboratory and is treated with care during operation. Thus, the calibration should be trustworthy and show only small errors. For first responder, I assume that the system cannot be calibrated regularly, often needs to be done in-situ and is exposed to physical stress during operation. Thus, the calibration can only be rough and might show larger errors.

The used settings for camera calibration are shown in Figure 5.1. The multi-camera calibration is based on a chessboard pattern of known size, a sophisticated corner assignment based on additional AprilTags to enable full image coverage, and a nonlinear optimization problem to estimate intrinsic and extrinsic camera parameters. Analytical error propagation is implemented to propagate uncertainties from feature and object points to the resulting camera parameters. For inspection, the laboratory calibration uses a high number of 72 stereo pairs, recorded with a camera tripod. The

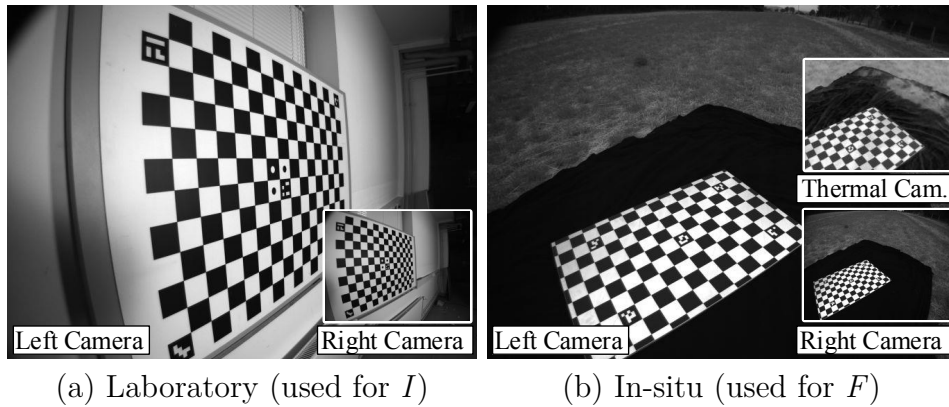


Figure 5.1: Considered calibration setups for inspection and first responder applications. Datasets and calibrations were provided by the authors of Choinowski et al. (2019) and Wohlfeil et al. (2019).

result is an accurate calibration with small uncertainties, e.g., with a SD of 0.073 px for u_0 . For first responder, the in-situ calibration is based on 40 image triplets, which were recorded without a camera tripod, and simultaneously calibrates the stereo camera and a thermal camera. This results in a more uncertain calibration, e.g., with a SD of 0.538 px for u_0 . The complete sets of calibration parameters and uncertainties can be found in Appendix A.3.

5.1.2 Derivation of Calibration Uncertainties

The objective of this section is to critically consider the trustworthiness of the geometric calibration during operation with IPS and consequentially to propose a more realistic error assessment in form of scaled calibration parameter uncertainties. A realistic assessment of possible calibration errors is required to correctly weight the calibration errors, for instance, within the combined sensitivity analyses (Section 7.2.2, 8.3) and to verify that errors of these considered dimensions can be correctly propagated in the VO component of IPS. Therefore, this section starts by discussing different significant factors that might degrade the geometric calibration. This discussion is used as basis to define multiplication factors to scale the propagated uncertainties from calibration.

The multiplication of derived uncertainties with a defined factor is common for particular applications, but the factor must be stated (JCGM 101, 2008, p. ix). However, the resulting differently scaled uncertainties of this section are primarily based on expert judgment, as they can currently not be estimated based on statistical methods. I.e., they fall into the category of uncertainties that are “evaluated by other means” (category B evaluation of uncertainty, JCGM 101, 2008, p. ix). Another point to consider is that IPS currently only supports the definition of a SD for each calibration parameter and does not allow to use the full covariance matrix from calibration, due to its modular design. Therefore, only the SDs for each parameter are considered in this thesis for all experiments, which is necessary for a meaningful evaluation of propagated uncertainties of the VO component. The use of marginal distributions for error propagation and also the systematic error influence of the calibration parameters during application will be discussed in Section 5.4.

Four effects are considered that can introduce additional errors into the calibration:

- **The quality of propagated uncertainty** itself needs to be considered. For instance, a determining factor for the resulting calibration uncertainty is the assumed uniform uncertainty of corner positions (Wohlfeil et al., 2019). The true uncertainty of detected corners will increase, if the corner is observed from poses with close distance and flat angles to the chessboard due to projective distortion, or if they are projected into regions with strong camera distortion. Other properties that have similar effects are camera noise and blur from depth of field. Consequently, the estimated uncertainty from calibration might be optimistic.
- **Physical stress** can deteriorate the calibration quality during operation. Acceleration from movement might slightly change the geometric sensor composition temporarily, due to possible elasticity of materials or (despite all efforts) loosely-mounted sensor components. Long-term vibrations, material fatigue and strong impacts on the system can lead to permanent changes. One example is provided in Figure 5.2, where the geometric sensor composition was observed to have changed during operation in an extreme environment.
- **Environmental influences** can change the sensor composition, such as air pressure, magnetic fields or temperature. I exemplarily discuss temperature that is known to cause material expansion, approximated with Equation 5.1 (Becker et al., 2003).

$$\Delta L \approx \alpha \cdot L_0 \cdot \Delta T \quad (5.1)$$

The stereo camera of the hand-held IPS consist of a calibrated baseline of $L_0=B=0.20\text{m}$ at room temperature, which is the largest calibrated dimension in this setup. The stereo camera is mounted on an aluminum rig with a length expansion coefficient $\alpha_0 = 2.4 \cdot 10^{-5} K^{-1}$. In related indoor datasets, I observed internal IMU temperatures between 30°C and 50°C . In the extreme fumarole datasets, I observed IMU temperatures of up to 70° , due to high environmental temperatures and isolation of IPS with protective gear, see Figure 3.1 (left, p.21). This suggests that differences to room temperature such as $\Delta T = 30\text{K}$ are entirely possible. This results in a material expansion of $\Delta L \approx 0.14\text{mm}$, which is far higher than the assumed SD from in-situ calibration for $B \approx x_l^r$ with 0.0083 mm .

- **Incomplete calibration** of the system lead to errors by default. Specifically, the calibration of IPS consists of two steps: calibration of the stereo camera and registration of the IMU to the left camera \mathbf{T}_l^b . For rotation estimation (Grießbach, 2015, p. 48) in the second step, the three axes of the body frame are ideally aligned with z^n axis of the navigation frame (direction of gravitation), while observing a fixed vertical calibration pattern. Parameters \mathbf{T}_b^n and the pose of the chessboard are estimated in a nonlinear optimization procedure with uncertainty estimation. The displacement is difficult to determine with the given low cost IMUs (Grießbach, 2015, p. 47). It is measured manually with a ruler and an uncertainty of 1mm is defined based on expert judgment. This second step is generally laborious and is often discarded. However, this can be crucial, because it depends on the calibrated left camera, which is slightly different for each camera calibration.

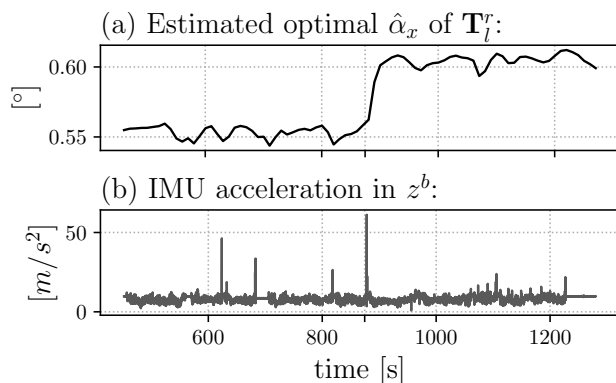


Figure 5.2: Exceptional change in camera calibration under physical stress in the extreme fumarole dataset: (a) Result from online rotation estimation for stereo transformation \mathbf{T}_l^r , provided by co-authors of (Irmisch et al., 2021). A drastical change¹ of 0.05° can be observed for $\hat{\alpha}_x$ at time 874s. (b) The change might be attributed to strong acceleration at time 874s, due to a hard put down of the system onto the ground.

To account for mentioned additional error sources, I defined multiplying factors to inflate given uncertainties and noted them in Table A.9 in Appendix A.3. Uncertainties of intrinsic camera parameters are scaled by a factor of 4 for calibration I , which results in a range of uncertainties that are usually used in IPS (*prior*, see Table A.9). Parameter uncertainties for the stereo transformation \mathbf{T}_l^r are scaled by the higher factor 5, because I assume that they are more influenced by physical stress. As a special case, the SD for baseline $B \approx x_l^r$ is multiplied by 20 to account for errors from temperature peaks. The uncertainty of calibration F is scaled with similar reasoning, but the factor is chosen generally smaller, because the main error is assumed to be already present in the uncertain in-situ calibration. Calibrated parameters for \mathbf{T}_l^b , the transformation between left camera l and body frame b of the IMU, are based on the same calibration for both applications. I chose a higher scale factor for the calibration F of \mathbf{T}_l^b based on the assumption that the transformation is more difficult to derive in a first responder sensor system, which might be more densely built and of lower material and sensor quality.

To summarize, two calibration parameter sets (I : inspection, F : first responder) were derived that mainly differ in their level of uncertainties to account for different geometric properties and calibration errors of both theoretical sensor setups.

5.2 Uncertainties in Visual Odometry

After defining reasonable calibration uncertainties as basis, the next step is to investigate the handling of uncertainties within the VO module. This section considers three aspects and proposes corresponding modifications to improve the internal handling of uncertainties. The aspects concern made assumptions during feature matching, tracking of covariances between entities and camera model parameters during feature undistortion, and Weighted Least-Squares (WLS) for ego-motion estimation.

¹The change manifested in the optimizable parameter $\hat{\alpha}_x$ of \mathbf{T}_l^r , but the real change of system might actually be in another not-optimized parameter, such as the location of the principal point.

5.2.1 Feature Matching Uncertainties

The first step with uncertainty estimation in VO is feature matching. The objective of this section is to question made assumptions about uncertainties during template feature matching (introduced in Section 3.2.1) and to suggest suitable values for IPS.

Uncertainties for template feature matching are defined in IPS as follows. The feature $\mathbf{m}^{c1\delta}$ corresponds to the considered template in the first image c_1 and is assigned an uncertainty of $\Sigma_{\mathbf{m}^{c1\delta}} = \mathbf{0}$. To define matching uncertainties, IPS differs between pixel-wise and subpixel matching, based on the introduced matching metrics NCC and SAD.

For pixel-wise matching with NCC, the uncertainty of the matched feature $\hat{\mathbf{m}}^{c2\delta}$ in the second image c_2 is

$$\Sigma_{\hat{\mathbf{m}}^{c2\delta}, \text{pixel-wise}} = \Sigma_{\text{quant.}} + \Sigma_{\text{prior, pixel-wise}}. \quad (5.2)$$

It first composes of $\Sigma_{\text{quant.}} = \text{diag}(1/12, 1/12)$ that describes uncertainty from quantization. It composes of $\Sigma_{\text{prior}} = \text{diag}(0.1^2, 0.1^2)$ to account for unknown nonlinearities that are currently not modeled in IPS, e.g., matching errors resulting from view point changes.

For subpixel matching with SAD, the uncertainty of the matched feature $\hat{\mathbf{m}}^{c2\delta}$ in the second image c_2 is

$$\Sigma_{\hat{\mathbf{m}}^{c2\delta}, \text{subpixel}} = \Sigma_{\text{noise}} + \Sigma_{\text{prior, subpixel}}. \quad (5.3)$$

It includes the covariance matrix Σ_{noise} that describes uncertainty propagated from image noise through the matching process (Zhang, 2018), which is only applicable for subpixel matching. Currently, Σ_{noise} is assumed to sufficiently cover feature matching uncertainties and Σ_{prior} is omitted. In the following, I will show that this assumption is not applicable and will derive suitable values for $\Sigma_{\text{prior, pixel-wise}}$ and also $\Sigma_{\text{prior, subpixel}}$ to account for remaining unknown nonlinearities.

Evaluation Procedure

Template feature matching can be evaluated based on the developed simulation tool, as visualized in Figure 5.3. The assumption is that $\mathbf{m}^{c1\delta}$ in image c_1 corresponds to a static object point, which is to be found in the second image c_2 using the matching procedure. (i) Then, feature $\mathbf{m}^{c1\delta}$ is detected in the first image. (ii) The corresponding GT object point $\vec{\mathbf{M}}$ is generated based on the depth value at $\mathbf{m}^{c1\delta}$ in the GT depth map

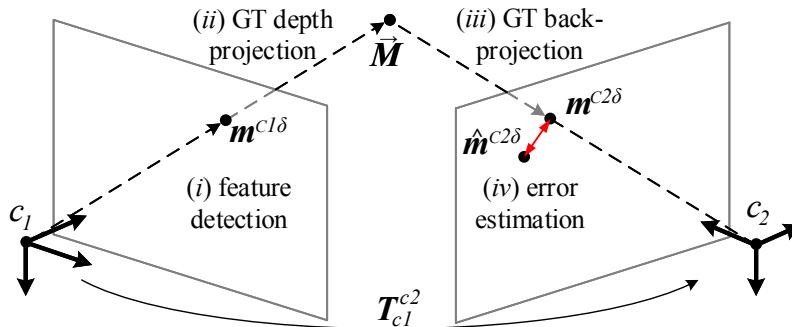


Figure 5.3: Feature matching evaluation procedure based on synthetic data.

and using GT calibration parameters and the GT camera pose. (iii) \vec{M} is projected into the second camera using GT calibration parameters and the known GT camera pose, which results in the GT image point $\mathbf{m}^{c2\delta}$. (iv) Finally, $\mathbf{m}^{c2\delta}$ can be compared to the matched feature $\hat{\mathbf{m}}^{c2\delta}$. This procedure is applied to all feature matches that were used for the final VO estimation after RANSAC filtering.

The resulting matching error is computed on each dimension i of $\mathbf{m}^{c2\delta}$ as

$$e_i = \hat{m}_i^{c2\delta} - m_i^{c2\delta} \quad (5.4)$$

and is used to compute the experimentally derived SD $\check{\sigma}_i$ of a set of feature matching errors \mathbf{e}_{-i} with

$$\check{\sigma}_i = sd(\mathbf{e}_{-i}), \quad (5.5)$$

which is computed based on the 99% interval to account for outliers.

Additionally, the estimated SD $\hat{\sigma}_i$ of the propagated feature matching uncertainty $\Sigma_{\hat{\mathbf{m}}^{c2\delta}}$ can be roughly evaluated based on the SD $\check{\sigma}_i$ of the normalized error \tilde{e}_i , as explained in Section 3.3.2.

Experiments

Figure 5.4 shows resulting experimental error distributions ($\check{\sigma}_i, \check{\mu}_i$) for one simulated experiment. This experiment evaluates intra ($\hat{\mathbf{m}}^{r1\delta}$) and inter ($\hat{\mathbf{m}}^{l2\delta}, \hat{\mathbf{m}}^{r2\delta}$) matching (superscripts were introduced in Section 3.2.1). The normalized errors $\check{\sigma}_{old}$ indicate that the estimated uncertainties $\hat{\sigma}_i$ are significantly optimistic. This clearly shows that the propagated uncertainties Σ_{noise} from noise do not sufficiently account for all unknown nonlinearities. The effect is negligible for intra matching in vertical direction with e_v^{r1} ,

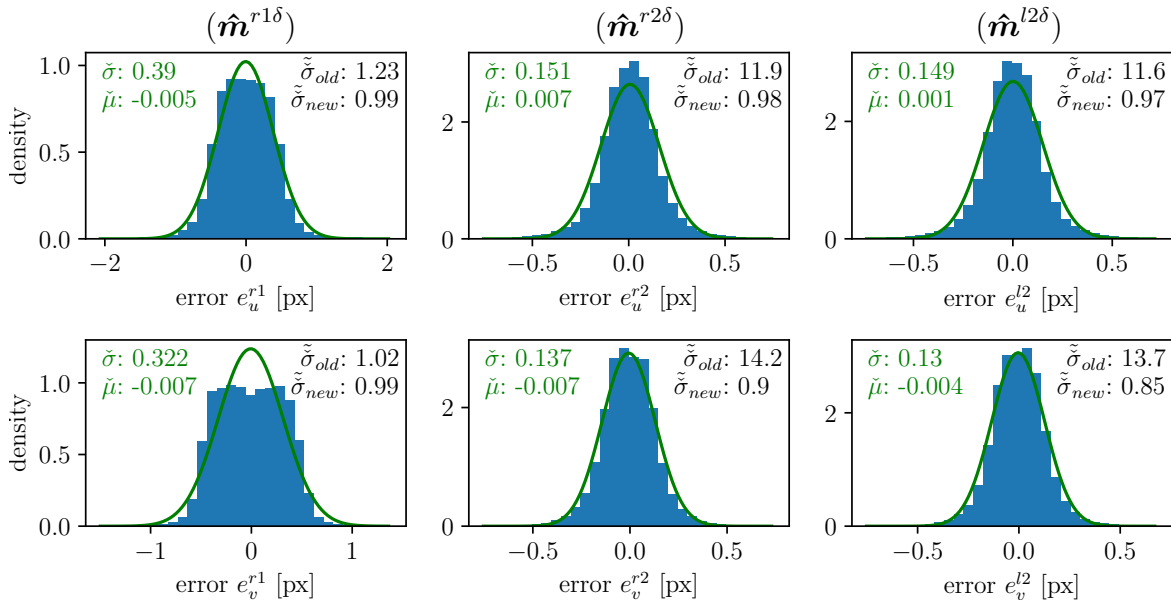


Figure 5.4: Feature matching evaluation based on the static fumarole dataset and altogether 102433 point correspondences between all four images. Errors are shown in blue and their experimentally derived distribution ($\check{\sigma}, \check{\mu}$) is plotted in green. The SD of the normalized error is written for the original ($\check{\sigma}_{old}$) and the proposed ($\check{\sigma}_{new}$) settings.

but is still significant in horizontal direction with e_u^{r1} . The errors in u -direction for intra matching is likely caused by the large view-point change introduced by the baseline in x -direction of \mathbf{T}_l^r . Similar, inter matching shows errors in x -direction, which might be due to predominant view point changes on the horizontal plane than in vertical direction.

Further experiments and evaluations are provided in Appendix B.2. They show that the feature matching error is influenced by many factors, such as the considered environment, the magnitude of view point changes and the severity of motion blur.

Definition of Prior Uncertainties

To improve the uncertainty estimation during feature matching, I defined values of Σ_{prior} for intra and inter matching, so that Σ_{match} roughly matches the measured distributions $\hat{\sigma}$ in the experiment of Figure 5.4. The chosen values are noted in Table 5.1. The uncertainties in inter matching were defined identically for the different image dimensions, because the predominant movement on the horizontal plane might be dataset specific. Based on the adjusted prior uncertainties Σ_{prior} , the normalized error $\tilde{\sigma}_{new}$ generally tends to the ideal value of 1.0, noted in Figure 5.4.

Considering an application in real world, the conditions are most likely to be more difficult than in simulation and the related true feature uncertainties will increase. Therefore, I roughly define a scaling factor of 1.2 based on expert judgment to inflate the feature matching uncertainties for application in real world, see Table 5.1.

Table 5.1: Definition of unified SD in [px] for $\Sigma_{prior} = \text{diag}(\sigma_u^2, \sigma_v^2)$ to account for unknown nonlinearities during intra ($\Sigma_{prior,r1}$) and inter matching ($\Sigma_{prior,r2}, \Sigma_{prior,l2}$).

	σ_u^{r1}	σ_v^{r1}	σ_u^{r2}	σ_v^{r2}	σ_u^{l2}	σ_v^{l2}
simulation	0.27	0.15	0.15	0.15	0.15	0.15
<i>scale factor</i>	<i>1.2</i>	<i>1.2</i>	<i>1.2</i>	<i>1.2</i>	<i>1.2</i>	<i>1.2</i>
real world	0.324	0.18	0.18	0.18	0.18	0.18

Summary

This investigation has shown that the magnitude of feature matching errors depends on many factors. It has been proven experimentally that the currently assumed feature matching uncertainties in IPS do not represent the true feature matching error well. The feature matching error is subjected to unknown nonlinearities that are not modeled in IPS, such as introduced by view point changes. To account for such nonlinearities, an additional uniform uncertainty was defined based on experiments in simulation.

5.2.2 Covariances during Feature Transformation

The second step with uncertainty propagation in VO is undistortion of detected and matched feature points. The objective of this section is to investigate the importance of tracking covariances between transformed quantities and model parameters of this

transformation. Currently, such covariances are often dropped in the modular implementation of IPS. In this section, I discuss one example from VO, where the tracking of covariances between quantities and model parameters significantly improves propagated uncertainties.

I consider the transformation of feature points (see Figure 3.3, p.23). Specifically, I consider to use 11×11 covariance matrices (*Cov11x11*), instead of 2×2 covariance matrices (*Cov2x2*), as intermediate representation of uncertainties between each of the three steps of the undistortion method, which are formulated as

$$\mathbf{m}^\delta \rightarrow \tilde{\mathbf{m}}^\delta \rightarrow \tilde{\mathbf{m}} \rightarrow \hat{\mathbf{m}}. \quad (5.6)$$

The distorted image point \mathbf{m}^δ is transformed into normal camera coordinates $\tilde{\mathbf{m}}^\delta$, undistorted to $\tilde{\mathbf{m}}$ and transformed back into image coordinates $\hat{\mathbf{m}}$.

For simplification, I concentrate in the following argumentation on a forth and back transformation of a feature point between image and normalized camera coordinates, formulated as

$$\mathbf{m} \rightarrow \tilde{\mathbf{m}} \rightarrow \hat{\mathbf{m}}, \quad (5.7)$$

based on 6×6 covariance matrices (*Cov6x6*) instead of 2×2 covariance matrices. This reasoning applies equally to undistortion, which is sketched out in Appendix B.3.

Using 2×2 Covariance Matrices

The existing implementation (*Cov2x2*) [OSLib] uses 2x2 covariance matrices to model the uncertainty of each feature point. Note that this implementation with all partial derivations was formulated and implemented by Grißbach et al. (2014, p.25).

The first transformation $\mathbf{m} \rightarrow \tilde{\mathbf{m}}$ is formulated with corresponding error propagation as

$$\tilde{m}_i = \frac{m_i - c_i}{f_i}, \quad (5.8)$$

$${}_{(2,2)}\Sigma_{\tilde{\mathbf{m}}} = {}_{(2,2)}\mathbf{J}_{\tilde{\mathbf{m}}} \cdot {}_{(2,2)}\Sigma_{\mathbf{m}} \cdot {}_{(2,2)}\mathbf{J}_{\tilde{\mathbf{m}}}^T + {}_{(2,4)}\mathbf{J}_{\boldsymbol{\kappa}} \cdot {}_{(4,4)}\Sigma_{\boldsymbol{\kappa}} \cdot {}_{(2,4)}\mathbf{J}_{\boldsymbol{\kappa}}^T. \quad (5.9)$$

It consists of the feature point uncertainty ${}_{(2,2)}\Sigma_{\mathbf{m}}$ of $\mathbf{m}=(u, v)^T$ and the model uncertainty ${}_{(4,4)}\Sigma_{\boldsymbol{\kappa}}$ of the interior orientation with $\boldsymbol{\kappa}=(u_0, v_0, f_u, f_v)^T$, including the principal point $\mathbf{c}=(u_0, v_0)^T$. Jacobian matrices ${}_{(2,2)}\mathbf{J}_{\tilde{\mathbf{m}}}$ and ${}_{(2,4)}\mathbf{J}_{\boldsymbol{\kappa}}$ describe partial derivatives of image coordinates and interior orientation w.r.t. normalized camera coordinates:

$${}_{(2,2)}\mathbf{J}_{\tilde{\mathbf{m}}} = \frac{\delta \tilde{\mathbf{m}}}{\delta \mathbf{m}} = \begin{bmatrix} f_u^{-1} & 0 \\ 0 & f_v^{-1} \end{bmatrix} \quad (5.10)$$

and

$${}_{(2,4)}\mathbf{J}_{\boldsymbol{\kappa}} = \frac{\delta \tilde{\mathbf{m}}}{\delta \boldsymbol{\kappa}} = \begin{bmatrix} -f_u^{-1} & 0 & -f_u^{-2}(u - u_0) & 0 \\ 0 & -f_v^{-1} & 0 & -f_v^{-2}(v - v_0) \end{bmatrix}. \quad (5.11)$$

The second transformation $\tilde{\mathbf{m}} \rightarrow \mathbf{m}$ with $\tilde{\mathbf{m}}=(x, y)^T$ is formulated with error propagation as:

$$\hat{m}_i = \tilde{m}_i \cdot f_i + c_i, \quad (5.12)$$

$${}_{(2,2)}\hat{\Sigma}_{\mathbf{m}} = {}_{(2,2)}\mathbf{J}_{\tilde{\mathbf{m}}}^{\mathbf{m}} \cdot {}_{(2,2)}\Sigma_{\tilde{\mathbf{m}}} \cdot {}_{(2,2)}\mathbf{J}_{\tilde{\mathbf{m}}}^{\mathbf{m}T} + {}_{(2,4)}\mathbf{J}_{\kappa}^{\mathbf{m}} \cdot {}_{(4,4)}\Sigma_{\kappa} \cdot {}_{(2,4)}\mathbf{J}_{\kappa}^{\mathbf{m}T} \quad (5.13)$$

with Jacobian matrices and partial derivatives:

$${}_{(2,2)}\mathbf{J}_{\tilde{\mathbf{m}}}^{\mathbf{m}} = \frac{\delta \mathbf{m}}{\delta \tilde{\mathbf{m}}} = \begin{bmatrix} f_u & 0 \\ 0 & f_v \end{bmatrix}, \quad (5.14)$$

and

$${}_{(2,4)}\mathbf{J}_{\kappa}^{\mathbf{m}} = \frac{\delta \mathbf{m}}{\delta \kappa} = \begin{bmatrix} 1 & 0 & x & 0 \\ 0 & 1 & 0 & y \end{bmatrix}. \quad (5.15)$$

Using 6×6 Covariance Matrices

The considered implementation *Cov6x6* tracks covariances between features and model uncertainties for the intermediate representation of $\tilde{\mathbf{m}}$ with ${}_{(6,6)}\Sigma_{\tilde{\mathbf{m}},\kappa}$. The error propagation for the first step ($\mathbf{m} \rightarrow \tilde{\mathbf{m}}$) is formulated as:

$${}_{(6,6)}\Sigma_{\tilde{\mathbf{m}},\kappa} = {}_{(6,6)}\mathbf{J}_{\tilde{\mathbf{m}},\kappa}^{\mathbf{m},\kappa} \cdot {}_{(6,6)}\Sigma_{\mathbf{m},\kappa} \cdot {}_{(6,6)}\mathbf{J}_{\tilde{\mathbf{m}},\kappa}^{\mathbf{m},\kappa T} \quad (5.16)$$

$$\begin{bmatrix} {}_{(2,2)}\Sigma_{\tilde{\mathbf{m}}} & {}_{(2,4)}\check{\Sigma}_{\kappa,\tilde{\mathbf{m}}} \\ {}_{(2,4)}\check{\Sigma}_{\kappa,\tilde{\mathbf{m}}}^T & {}_{(4,4)}\Sigma_{\kappa} \end{bmatrix} \quad \begin{bmatrix} {}_{(2,2)}\Sigma_{\mathbf{m}} & {}_{(2,4)}\mathbf{0} \\ {}_{(4,2)}\mathbf{0} & {}_{(4,4)}\Sigma_{\kappa} \end{bmatrix} \quad \begin{bmatrix} {}_{(2,2)}\mathbf{J}_{\tilde{\mathbf{m}}}^{\mathbf{m}} & {}_{(2,4)}\mathbf{J}_{\kappa}^{\mathbf{m}} \\ {}_{(4,2)}\mathbf{0} & {}_{(4,4)}\mathbf{I} \end{bmatrix}.$$

The Jacobian matrices and covariance matrices assemble of basic matrices from the implementation *Cov6x6*. ${}_{(2,2)}\Sigma_{\tilde{\mathbf{m}}}$ consists of the familiar covariance matrices ${}_{(2,2)}\Sigma_{\mathbf{m}}$ and ${}_{(4,4)}\Sigma_{\kappa}$ and the additional partial covariance matrix ${}_{(2,4)}\check{\Sigma}_{\tilde{\mathbf{m}},\kappa}$, which holds covariances between feature point $\tilde{\mathbf{m}}$ and interior orientation κ .

The error propagation for the second step ($\tilde{\mathbf{m}} \rightarrow \hat{\mathbf{m}}$) is formulated accordingly:

$${}_{(6,6)}\hat{\Sigma}_{\mathbf{m},\kappa} = {}_{(6,6)}\mathbf{J}_{\hat{\mathbf{m}},\kappa}^{\tilde{\mathbf{m}},\kappa} \cdot {}_{(6,6)}\Sigma_{\tilde{\mathbf{m}},\kappa} \cdot {}_{(6,6)}\mathbf{J}_{\hat{\mathbf{m}},\kappa}^{\tilde{\mathbf{m}},\kappa T} \quad (5.17)$$

with

$${}_{(6,6)}\hat{\Sigma}_{\mathbf{m},\kappa} = \begin{bmatrix} {}_{(2,2)}\hat{\Sigma}'_{\mathbf{m}} & {}_{(2,4)}\check{\Sigma}_{\mathbf{m},\kappa} \\ {}_{(2,4)}\check{\Sigma}_{\mathbf{m},\kappa}^T & {}_{(4,4)}\Sigma_{\kappa} \end{bmatrix}. \quad (5.18)$$

Covariance matrix ${}_{(2,2)}\hat{\Sigma}'_{\mathbf{m}}$ is the final result for this two-step procedure. Partial covariance matrix ${}_{(2,4)}\check{\Sigma}_{\mathbf{m},\kappa}$ appears to be equal ${}_{(2,4)}\mathbf{0}$ after this back and forth transformation.

Experiments

A MCS (Section 3.3.2) is conducted to evaluate the methods *Cov2x2* and *Cov6x6* for an exemplary feature point. For this experiment, ${}_{(2,2)}\Sigma_{\mathbf{m}}$ is set to $\text{diag}(1/12, 1/12)$ and the derived calibration parameters with uncertainties of calibration setting F (Section 5.1) are used. The results are visualized in Figure 5.5. After transforming image point \mathbf{m} into normalized camera coordinates $\tilde{\mathbf{m}}$, the propagated uncertainty of both analytical methods equal the MCS result. However, after transforming $\tilde{\mathbf{m}}$ back into image coordinates $\hat{\mathbf{m}}$, only *Cov6x6* with $\sigma_{6x6} = (0.29\text{px}, 0.29\text{px})^T$ equals the MCS result

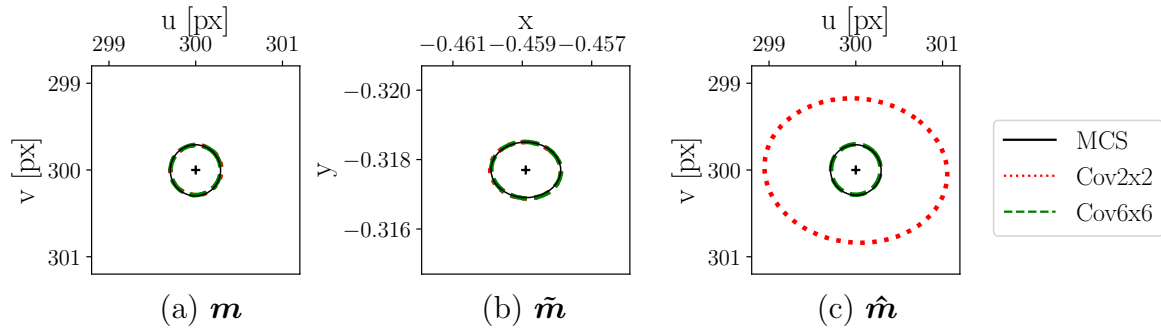


Figure 5.5: Covariance confidence ellipses from propagated uncertainties during back and forth feature transformation. Considered are Monte-Carlo Simulation (MCS) based on 100k iterations and analytical error with different covariance matrix dimensions.

with $\sigma_{MCS}=(0.29\text{px}, 0.29\text{px})^T$, while the result of $Cov2x2$ with $\sigma_{2 \times 2}=(1.06\text{px}, 0.83\text{px})^T$ is heavily conservative.

This experiment shows that the covariances in ${}_{(2,4)}\check{\Sigma}_{\mathbf{m},\kappa}$ provide the necessary information that the camera model in the second step is the same as in the first step and thus, that the introduced error in the first step will be reversed in the second step.

Summary

In summary, traversing a quantity through multiple transformations that depend on the same model parameters requires the tracking of covariances between quantity and model parameters. This insight is equally applicable for the undistortion method, consisting of three internal transformations (Equation 5.6). The equivalent covariance matrix used for tracking is ${}_{(11,11)}\Sigma_{\tilde{\mathbf{m}},\kappa,\delta}$ and additionally considers the distortion parameters $\delta = (k_1, k_2, k_3, p_1, p_2)^T$ (see Appendix B.3). In the current implementation, however, only ${}_{(2,2)}\hat{\Sigma}'_{\mathbf{m}}$ will be passed to the next step of VO in the following section, which is the ego-motion estimation.

5.2.3 Weighted Least-Squares

The third step with uncertainty propagation in VO is ego-motion estimation based on least-squares. The objective of this section is to prototypical exploit the knowledge about propagated feature uncertainties in VO. Therefore, the least-squares problem of VO (Grießbach, 2015) is extended to use feature uncertainties in a WLS manner.

Weighted Least-Squares Formulation

The formulated least-squares problem of Section 3.2.2 seeks to minimize the Euclidean distance between projected object points $\vec{\mathbf{M}}$ in the image (features \mathbf{m}^{l2}) and matched feature points $\hat{\mathbf{m}}^{l2}$ in left camera image coordinates of the second stereo frame $l2$ with

$$\min_{\Delta\mathbf{T}} \|\hat{\mathbf{m}}^{l2} - \mathbf{m}^{l2}\|^2, \quad (5.19)$$

in order to optimize model parameters that describe the relative transformation $\Delta\mathbf{T}$.

In this least-squares approach, however, the data is assumed to be isotropically Gaussian and propagated uncertainties are not considered. An alternative formulation can be formulated based on the Mahalanobis distance with

$$\min_{\Delta\mathbf{T}} \left[\sqrt{(\hat{\mathbf{m}}^{l2} - \mathbf{m}^{l2})^T \hat{\Sigma}_e^{-1} (\hat{\mathbf{m}}^{l2} - \mathbf{m}^{l2})} \right]^2. \quad (5.20)$$

Covariance matrix $\hat{\Sigma}_e$ reflects the error of observation $\hat{\mathbf{m}}^{l2}$ to the projected point \mathbf{m}^{l2} with $\mathbf{e} = \hat{\mathbf{m}}^{l2} - \mathbf{m}^{l2}$, which defines the residual of this least-squares problem. In this work, $\hat{\Sigma}_e$ is approximated as

$$\hat{\Sigma}_e \approx \Sigma_{\hat{\mathbf{m}}^{l2}} + \Sigma_{\mathbf{m}^{l2}}. \quad (5.21)$$

Covariance matrix $\Sigma_{\hat{\mathbf{m}}^{l2}}$ describes the uncertainty of the observation $\hat{\mathbf{m}}^{l2}$, that consists of uncertainties from feature matching (Section 5.2.1) and undistortion (Section 5.2.2). Covariance matrix $\Sigma_{\mathbf{m}^{l2}}$ describes the uncertainty of the projected point \mathbf{m}^{l2} and results from propagating object point uncertainties and camera model uncertainties through the projection of $\vec{\mathbf{M}}$ to \mathbf{m}^{l2} (using methods of Grißbach, 2015). In this step, the relative transformation $\Delta\mathbf{T}$ is still unknown and approximated by \mathbf{I} . This approximation is applicable since only relatively small movements in $\Delta\mathbf{T}$ are to be expected. Further inaccuracies in this approximation result from not tracking covariances between entities and camera model parameters and not modeling dependencies between observations and conditions.

Next, off-diagonal covariances are neglected to simplify the problem. Only variances are used in this approach with $\hat{\Sigma}_e = \text{diag}(\hat{\sigma}_u^2, \hat{\sigma}_v^2)$ and $\{\mathbf{m}^{l2}, \hat{\mathbf{m}}^{l2}\} = \{(u, v)^T, (\hat{u}, \hat{v})^T\}$. This restriction allows to derive

$$\min_{\Delta\mathbf{T}} \left[\frac{1}{\hat{\sigma}_u^2} (\hat{u} - u)^2 + \frac{1}{\hat{\sigma}_v^2} (\hat{v} - v)^2 \right], \quad (5.22)$$

which can be written in the original least-squares formulation of Equation A.4 (p.xiv):

$$\chi^2(\Delta\mathbf{T}) = \sum_{j=0}^N \left[\hat{w}_{ju} (\hat{u}_j - \pi_u(\vec{\mathbf{M}}_j | \Delta\mathbf{T}))^2 + \hat{w}_{jv} (\hat{v}_j - \pi_v(\vec{\mathbf{M}}_j | \Delta\mathbf{T}))^2 \right] \rightarrow \text{Min}, \quad (5.23)$$

with weights $\hat{w}_{ju} = 1/\hat{\sigma}_{ju}^2$ and $\hat{w}_{jv} = 1/\hat{\sigma}_{jv}^2$. It consists of the model function π that projects $\vec{\mathbf{M}}$ onto the image plane based on the model parameters $\text{param}(\Delta\mathbf{T}) = (t_x, t_y, t_z, \alpha_x, \alpha_y, \alpha_z)^T$ that describe the relative transformation. Additionally, π is based on $\boldsymbol{\kappa}$ and $\boldsymbol{\delta}$, which are omitted in Equation 5.23 for simplified notation. This WLS problem can be solved with the Gauss-Newton algorithm² (Appendix A.1).

Experiments

Figure 5.6 shows examples from a real-world experiment and visualizes the resulting residual uncertainties Σ_e for the feature set after RANSAC filtering. SDs are scaled for better visualization. It shows that the main difference in residual uncertainties comes

²The implementation was provided in [OSLib] with an option to define a weight matrix.

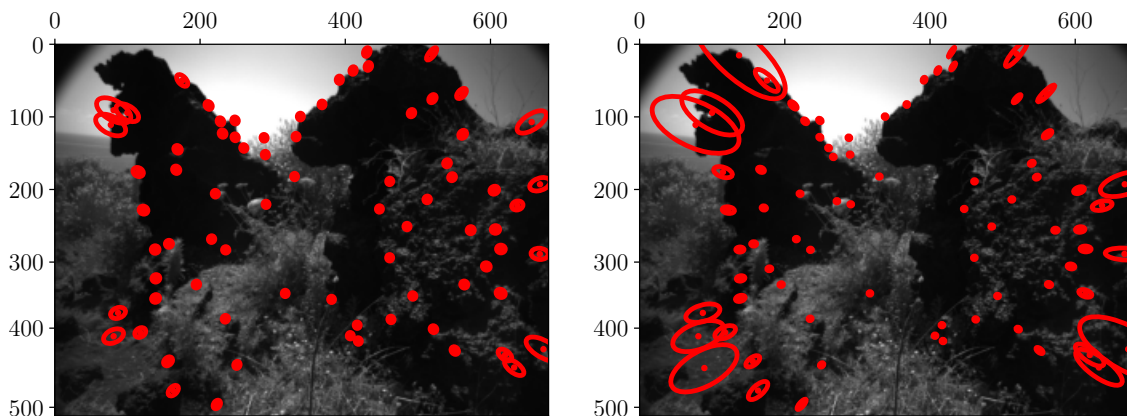
(a) Calibration I with $\Sigma_e \cdot 15^2$ (b) Calibration F with $\Sigma_e \cdot 5^2$

Figure 5.6: 68% confidence intervals of residual covariances Σ_e , which are scaled for better visualization, for one exemplary dataset, recorded at Valle dei Mostri, Sicily.

from distortion uncertainties in the considered approach. The effect is significant at image borders in (b), when using the calibration settings F .

This WLS approach is validated by a basic MCS in Appendix B.4. This experiment shows that the presented WLS method outperforms the basic least-squares approach in the presence of noisy feature points. The MCS simulation could further verify a correct analytical error propagation (developed by Grießbach, 2015) for single VO estimates, considering uncertainties from feature matching and camera model parameters κ . Though, this experiment did not consider distortion parameters δ .

Summary

In summary, VO can be improved by considering feature covariances in a WLS approach in the presence of noisy feature points. Features with high uncertainties are to be expected in IPS, primarily due to uncertainty in distortion, as exemplified in Figure 5.6. The fact that geometric distortion parameters rather constitute a systematic error than a statistical error will be discussed in Section 5.4.

5.3 Experiments

In the previous section, three modifications were proposed and validated based on small individual experiments. In this section, they are applied jointly as $WLS+$, which is evaluated on the basis of synthetic and real-world data in comparison to the original method, abbreviated as LS . First, the strategy *geometric MCS* of Section 4.3.2 is applied to introduce errors from geometric calibration into the evaluation process and to evaluate the quality of propagated uncertainties of the VO component. Second, the observations are then validated based on multiple small-scale real-world datasets of various static environments. Each experiment considers two system calibration settings (I and F , see Section 5.1) with different levels of uncertainty. (Images are processed with a resolution of 680×512 px. More technical notes are provided in Section 7.1.2)

5.3.1 Geometric Monte Carlo Simulation

The simulation strategy *geometric MCS* (Section 4.3.2) is used and applied for each of the synthetic static datasets *fumaroles*, *coast* and *corridor* (Section 4.2). Each environment is simulated once without dynamic elements. IPS is then applied 500 times for each configuration (*WLS+*, *LS*) on each dataset. For each application of IPS, geometric calibration parameters $\mathbf{P}_C := \{\boldsymbol{\kappa}^l, \boldsymbol{\kappa}^r, \boldsymbol{\delta}^l, \boldsymbol{\delta}^r, \mathbf{T}_l^b, \mathbf{T}_l^r\}$ are sampled once based on their respective error distributions. This is done equally for both calibration settings (*I*, *F*). The evaluation is based on three metrics. First, the accuracy of the VO component is evaluated based on the mean RTE (Section 3.3.1). Second, the propagated uncertainties of the translation parameters from the estimated relative VO transformation are evaluated based on the normalized error (Section 3.3.2). Third, the mean ATE (Section 3.3.1) is used to evaluate the estimated final IPS trajectory.

To recapitulate, the modifications in *WLS+* over *LS* are: (*i*) improved feature matching uncertainties; (*ii*) tracking of covariances between quantities and camera model parameters during undistortion; and (*iii*) a WLS approach.

The results of this experiments are noted in Table 5.2. They show a clear improvement of *WLS+* over *LS*. First, the SDs of the normalized errors show that the estimated uncertainties of *WLS+* better represent the true error distribution than of *LS* (closer to 1.0), which can be attributed to modifications (*i*) and (*ii*). The uncertainty estimations of *LS* show to be generally conservative (e.g., $\tilde{\sigma}_x \ll 1.0$). Second, the mean RTE implies that modification (*iii*) improves the VO estimations in *WLS+*. Further, the higher calibration errors in *F* lead to a visible increase of the mean RTE for both methods, but its relative change is less strong for *WLS+*. Third, the mean ATE shows that the improved VO of *WLS+* consequently leads to a more accurate final trajectory.

Table 5.2: Geometric MCS for *LS* and *WLS+* based on different synthetic datasets.

		calib.uncertainties: method:		<i>I</i> (small)		<i>F</i> (large)	
		LS	WLS+	LS	WLS+	LS	WLS+
Fumaroles	mATE [m]	0.0196	0.0159	0.0585	0.0371		
	mRTE [mm] (VO)	0.44	0.40	0.94	0.72		
	$\tilde{\sigma}_x$	0.6	0.94	0.43	0.85		
	$\tilde{\sigma}_y$	0.5	0.79	0.38	0.78		
	$\tilde{\sigma}_z$	0.5	0.75	0.41	0.7		
Coast	mATE [m]	0.0143	0.0127	0.0374	0.0272		
	mRTE [mm] (VO)	0.52	0.46	1.08	0.84		
	$\tilde{\sigma}_x$	0.56	0.9	0.47	0.91		
	$\tilde{\sigma}_y$	0.48	0.76	0.38	0.78		
	$\tilde{\sigma}_z$	0.54	0.85	0.43	0.77		
Corridor	mATE [m]	0.0294	0.0273	0.0785	0.0634		
	mRTE [mm] (VO)	1.31	1.28	2.56	2.25		
	$\tilde{\sigma}_x$	0.62	0.99	0.36	0.79		
	$\tilde{\sigma}_y$	0.63	0.92	0.42	0.85		
	$\tilde{\sigma}_z$	0.67	0.95	0.45	0.83		

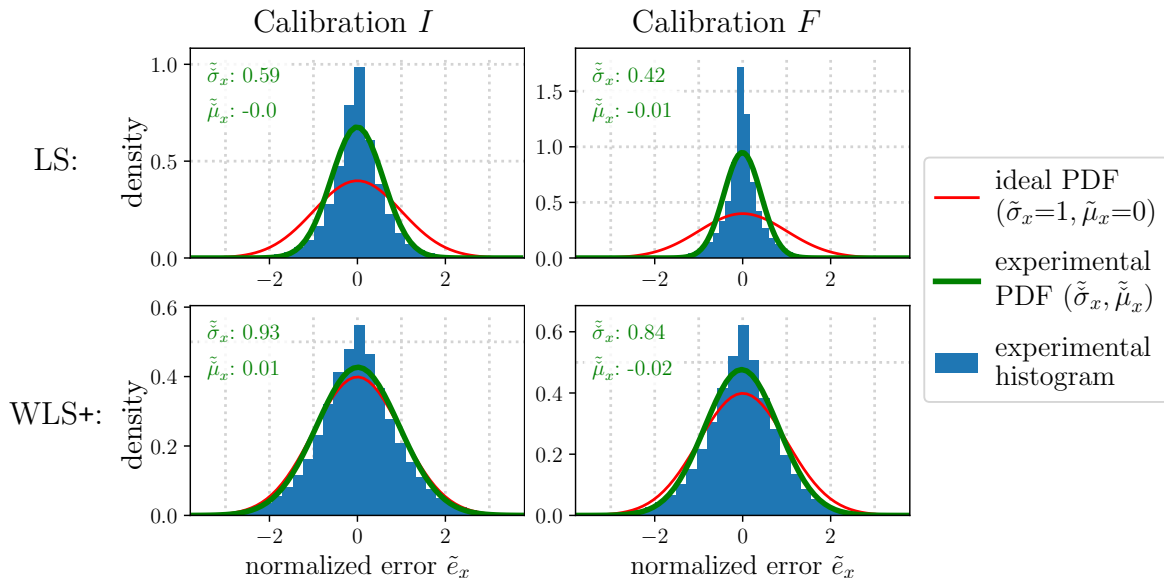


Figure 5.7: Distributions for *LS* and *WLS+*, consisting each of 606444 VO estimates from 500 samples (i.e., applications of IPS), exemplary for the fumarole dataset.

The VO estimations are further investigated in Figure 5.7. It considers the normalized error \hat{e}_x of the relative translation in x -direction of the estimated relative transformation $\Delta\mathbf{T}$. Either for *LS* and *WLS+*, the estimated uncertainties get more conservative for larger calibration uncertainties. This may indicate that still not all relevant covariances are tracked in *WLS+* between intermediate entities and calibration parameters. Specifically, the undistortion step (Section 5.2.2) and ego-motion estimation step (Section 5.2.3) are still applied individually, which should result in more conservative estimations, as similarly observed for the feature transformation steps in Section 5.2.2. Though, *WLS+* is visibly less affected by this effect than the *LS* solution.

Figure 5.8 shows the experimental distributions of the normalized error \hat{e}_x for all 500 samples (left) and single samples (center, right) for *WLS+* with calibration *F*. Single samples describe exactly one application of IPS with one set of sampled calibration parameters. It can be argued that one sample corresponds to an application in real world, where the calibration values are fixed and represent one instance of the given calibration parameter distribution. The plot implies that the estimated uncertainties

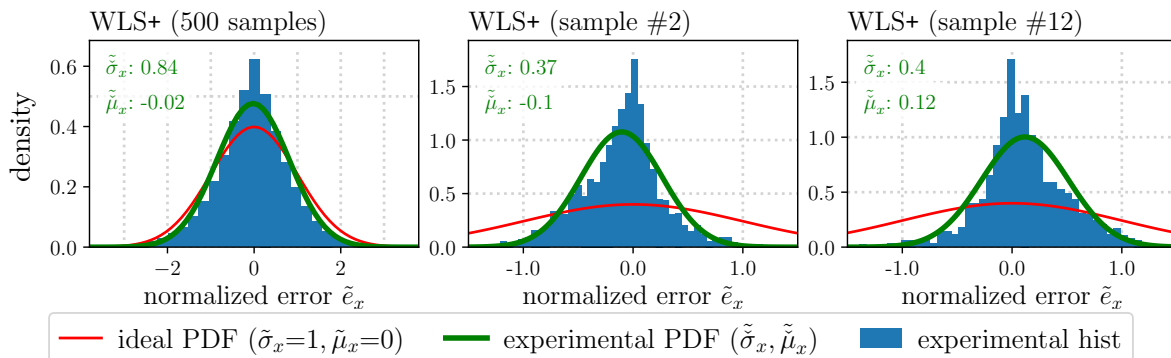


Figure 5.8: Distributions of single samples (center, right), each based on 1219 estimates.

$\hat{\sigma}_x$ of an individual sample are relatively conservative and that they are generally are not well modeled by a Gaussian distribution. Also, the mean of the experimental distribution deviates from zero, which might indicate a bias in VO. Further, it is significant that in both samples the mean for \tilde{e}_x deviates in another direction. This experiment might imply that the implemented error propagation from calibration uncertainty in IPS does not accurately model the introduced error from geometric calibration, which is rather a systematic error than a statistical error. However, since the actual calibration parameters are unknown in real world, the current solution in IPS is to model the overall uncertainty (Figure 5.8, left) that covers all possible cases as best as possible. This topic will be picked up in the discussion of Section 5.4.

5.3.2 Confirmation with Real World Data

The methods *LS* and *WLS+* are evaluated in this section based on multiple calibration settings in different static real-world environments. The used calibration parameters and uncertainty settings were introduced in Section 5.1. The *LS* approach is additionally applied with the *prior* uncertainty setting (Appendix A.3), which is frequently used in IPS as default, if reliable uncertainty values cannot be determined. A particularity is that *prior* does not define uncertainties for distortion parameters (set to 0), which is why it is considered in the following experiments. The considered datasets of Table 5.3 differ in length d and availability of GT information. The mean ATE (Section 3.3.1) is used for evaluation based on the available GCPs.

In comparison of *WLS+* with *LS*, the results imply that *WLS+* tends to show more accurate results than the basic *LS* approach. If applied based on the laboratory (lab.)

Table 5.3: Real-world experiments based on 14 different datasets (Table A.3). Each number shows the mean ATE [cm] over 20 repetitions. Bold numbers mark the best results of each dataset for each set of calibration parameters (laboratory, in-situ).

calib. param.: calib. unc.: method:	laboratory			in-situ			d[m] GCPs properties dataset-	
	prior LS	I LS	I WLS+	prior LS	F LS	F WLS+		
corridor-1	0.06	0.06	0.06	0.07	0.07	0.04	33	1
corridor-2	0.09	0.10	0.09	0.06	0.08	0.05	37	2
corridor-3	0.08	0.08	0.07	0.05	0.04	0.04	37	2
basement-1	0.43	0.43	0.45	0.43	0.45	0.40	215	6
park-area-1	0.66	0.66	0.58	1.01	0.84	0.80	364	15
park-area-2	0.08	0.08	0.07	0.11	0.10	0.08	30	1
coast-1	0.05	0.07	0.05	0.05	0.17	0.07	50	1
crater-rim-1	0.08	0.08	0.08	0.07	0.11	0.08	68	1
crater-rim-2	0.05	0.06	0.05	0.05	0.09	0.05	42	1
mars-1	0.05	0.08	0.04	0.03	0.06	0.02	68	1
hotel-1	0.09	0.08	0.10	0.11	0.27	0.08	44	1
mine-1	0.29	0.29	0.29	0.25	0.25	0.24	122	3
mine-2	0.35	0.35	0.35	0.28	0.28	0.27	134	3
park-stairs-1	0.04	0.04	0.03	0.07	0.09	0.05	36	1
Bold numbers:	7	6	11	3	1	12		

calibration, $WLS+$ achieves the best results in 11 out of 14 datasets. If applied based on the in-situ calibration, $WLS+$ achieves the best results in 12 out of 14 datasets.

In general, $WLS+$ shows to be more accurate when using uncertainty settings of calibration settings F than using I . This might imply that the derived uncertainty of F represents the true error distribution of the in-situ calibration better than the derived uncertainty of I represents the true error distribution of the laboratory calibration, at least for the presented datasets.

Despite relatively good results, their significance is impaired. The difference between results of individual methods is extremely small, on the order of cm. Furthermore, only one methodical base configuration of IPS is considered in all experiments. Due to this, the improvement can only be considered as a trend towards $WLS+$. A deeper analysis of such small differences is not suitable based on real-world dataset with such limited GT.

5.4 Discussion

The simulation-based experiments showed that increasing calibration errors (I to F) leads to a stable decrease in accuracy for the considered uncertainty ranges. They further showed the capabilities of IPS to propagate uncertainties from different error sources, which was improved based on three modifications ($WLS+$) in this chapter. Though, limitations of the current concept were observed, which are discussed in the following with respect to (i) the definition of calibration uncertainties, (ii) the systematic error in geometric calibration, (iii) feature matching uncertainties and (iv) other limitations.

(i) The first point addresses the definition of calibration uncertainties, which is a mandatory input to IPS. An assumption in IPS is that these uncertainties can be described using marginal distributions only. This restriction is required for the modular design of IPS and an efficient computation with a reduced set of covariances during each step. However, the information about the correlation of parameters is lost. For instance, Figure 5.9 shows a strong correlation between focal length f^l and distortion parameter k_1^l or between the principal point parameters u_0^l and u_0^r . Therefore, the use of only marginal distributions might lead to a conservative representation of the

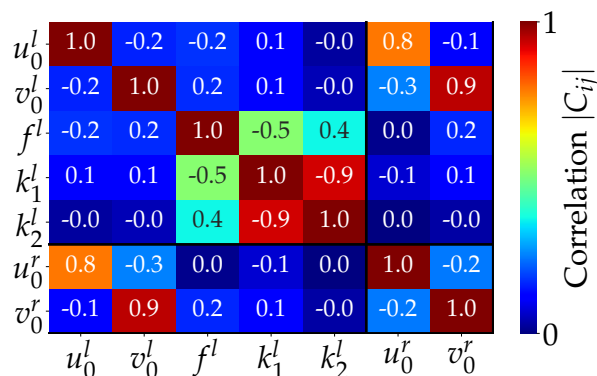


Figure 5.9: Correlation from covariances for selected calibrated camera parameters from calibration setting I . It shows high correlation between several parameters.

calibration errors. The assumption is similarly applied in all MCS experiments in this thesis, which is necessary to be able to validate propagated errors in IPS, but might exaggerate, for example, the distortion parameter errors. In turn, this assumption limits the reach of gained insights of the simulation-based experiments. Future developments could consider the correlation of calibration parameters and could start with an investigation of their influence on IPS based on the *geometric MCS*.

Further, the discussion in Section 5.1.2 found that the uncertainties from the calibration procedure do not adequately represent errors that can occur during operation. Therefore, more realistic distributions (I, F) were defined to model calibration errors in the MCS-based experiments and for propagation in IPS. However, these relatively large uncertainties are most likely conservative for the application of IPS in standard scenarios. In future, it would be desirable to observe the consistency of the system calibration during operation, or to implement an online-calibration method to continuously avoid large calibration errors and associated large uncertainties.

(ii) Calibration errors are more of a systematic nature and are most likely not well represented by Gaussian distributions during operation. In Section 5.2.2, it was shown that traversing a quantity through multiple transformations that depend on the same model parameters (if those are modeled with associated uncertainties) requires the tracking of covariances between quantity and model parameters. This was exemplified based on feature transformation, which leads to highly conservative estimates if the covariances are not tracked. As pointed out in Section 5.2.2, not all covariances between entities and camera model parameters are currently tracked through the complete VO pipeline in IPS (e.g., distortion), which might explain the conservative VO estimates of Section 5.3.1. The complete propagation of all uncertainties through the VO pipeline could be investigated in future developments.

Furthermore, different VO estimates depend on the same calibration parameters, which is currently not modeled and will most likely result in a bias drift of VO. The existence of bias and drift in VO is known and can be compensated. For instance, Dubbelman et al. (2012) proposed a projective model to compensate the bias in VO and improved their stereo-based VO method up to 50% in the closed loop error. Jiang et al. (2010) modeled the drift as a combination of wide-band noise and a first-order Gauss-Markov process and concluded that quantifying drift by related model parameters is more suited than using the closed loop error. Bias and bias instability of VO is currently not modeled in IPS. The introduction of model uncertainties that lead to more conservative VO estimates is currently the method of choice in IPS to account for degraded geometric calibration.

(iii) The estimation of feature matching uncertainties is currently not well modeled, despite the operational error propagation from image noise. High nonlinearities originate, for instance, from view point changes and are currently attempted to be compensated by an additional prior uniform uncertainty (Section 5.2.1). Though, the actual error will strongly depend on the environment and conditions during the operation, which can be highly adverse in a first responder context. Therefore, the reach of using additional uniform uncertainties is limited. Interestingly, the current error propagation approach of IPS led to a higher estimated statistical error for features at image borders (Section 5.2.3). This error propagation approach might be incomplete since calibration errors are rather systematic errors, but it might compensate for degraded

feature matching near the image border, which might arise due to strong image distortion. In future, the actual distribution of feature matching errors could be analyzed in more detail and an improved uncertainty estimation, for example, based on a DNN might be promising to improve IPS.

(*iv*) There are several more limitations that restrict the reach of the experiments and this discussion. The derived weights for the considered WLS approach (Section 5.2.3) resulted from propagated uncertainties, but are still only an approximation, as several correlations have been omitted for simplification. Also, correlations between feature points after feature transformation (e.g., undistortion) are not considered in IPS. Moreover, the investigations and improvements focused exclusively on VO and filter properties of IPS were not considered. Furthermore, the observations based the normalized error only considered all VO estimates jointly. More insights could be gained by dividing the measurements into bins, e.g., separated by the magnitude of estimated uncertainties (Anderson et al., 2019). Finally, all experiments were only based on one methodical base configuration of IPS.

5.5 Summary

This chapter created the basis for considering calibration errors in the experiments of the next chapters (7, 8). Therefore, two calibration settings were derived, motivated by the use of IPS for the applications inspection and first responder, respectively. First, calibration parameters were used from a laboratory and an in-situ camera calibration. Second, uncertainties were derived mainly based on expert judgment on the basis of a discussion of the degrading conditions during operation of IPS.

Furthermore, this chapter analyzed the error propagation concept for the VO component of IPS and proposed three modifications (*WLS+*). The improvement of IPS by the modifications was demonstrated based on a *geometric MCS* in simulation and was confirmed on real-world data based on a diverse collection of datasets. The capabilities of the current concept and its existing limitations were thoroughly discussed in the previous chapter. Based on this discussion, I conclude that uncertainties and their propagation in IPS are suitable and necessary to be involved in the following investigations and discussions.

In the experiments of the following chapters, the modified approach *WLS+* of this section will be used and abbreviated by the basic term *IPS*. Furthermore, the upcoming experiments will make use of the derived calibration settings (I, F).

Part II

The Hybrid System

Chapter 6

Fundamentals - Semantic Segmentation

Semantic understanding can provide environmental awareness capabilities for localization systems that go far beyond the possibilities of pure geometric approaches. In Part II of this work, a semantic aid is integrated into IPS in order to analyze the influence of dynamic objects and to improve the robustness of the localization system accordingly. This chapter introduces the method of semantic segmentation with primary focus on DL-based techniques and the used evaluation metrics.

“Segmentation partitions an image into its constituent parts or objects” (Gonzalez and Woods, 2018). It can be formulated as “the problem of classifying pixels with semantic labels (semantic segmentation), or partitioning of individual objects (instance segmentation), or both (panoptic segmentation)” (Minaee et al., 2021). The different types are demonstrated in Figure 6.1. Semantic segmentation performs pixel-level classification and labeling for all pixels of the image based on a set of defined categories. Pixels that do not correspond to a specific class are assigned the default background class. In contrast to instance and panoptic segmentation, it does not differentiate between individual instances of one object class, such as different humans.

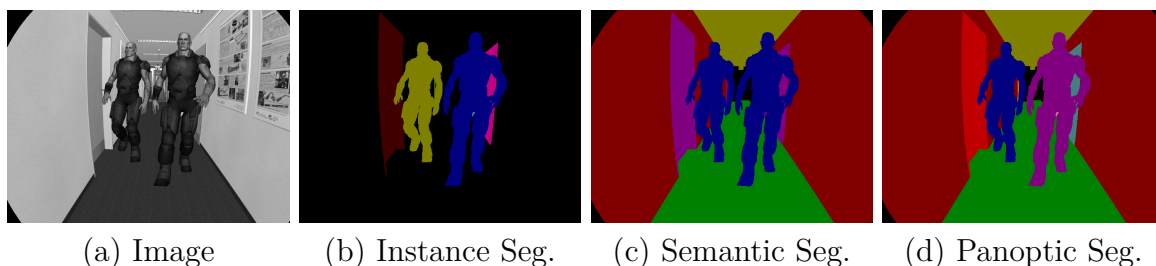


Figure 6.1: Different forms of segmentation (b-d) based on a visual image (a).

Autonomous image segmentation is a fundamental and difficult task in image processing and various problem-specific approaches exist (Gonzalez and Woods, 2018). An early classical approach is image thresholding that is a common choice if the object shows distinctive pixel intensities. It can be applied on a global level with multiple thresholds using Otsu’s method or on a local level based on moving averages. Another approach is edge based methods to detect and link object boundaries. They

are applicable if regions of object boundaries are sufficiently different from each other. Or, region-based approaches are favored for complicated scenes when object textures show patterns with notable intensity differences. Representatives are region-growing or region-splitting and -merging, k-means clustering, or watersheds. A relatively new technique that has emerged over the last decade is image segmentation based on DL. It currently represents the absolute choice that outperforms all previously mentioned classical approaches for most complicated image segmentation tasks.

6.1 Developments in Deep Learning

“Deep learning is a particular kind of machine learning that achieves great power and flexibility by representing the world as a nested hierarchy of concepts, with each concept defined in relation to simpler concepts” (Goodfellow et al., 2016, p. 8). In other words, a vector of input quantities are interpreted hierarchically in multiple steps by hidden layers, each consisting of a number of neurons. The output of each layer is a feature vector with increasing level of representation. With a significant number of layers, a high-level representation of input quantities is reached. This can finally be used to compute the output quantity, which might be a classification into user defined object categories. The mentioned flexibility refers to a possible easy adjustment of the number of layers and neurons as well as modular substitution with other mathematical problem-specific formulations, such as convolutional layers for image processing tasks. The power might further refer to the possible representation in matrix notation, which allows easy upscaling and application of hardware that is optimized for matrix processing.

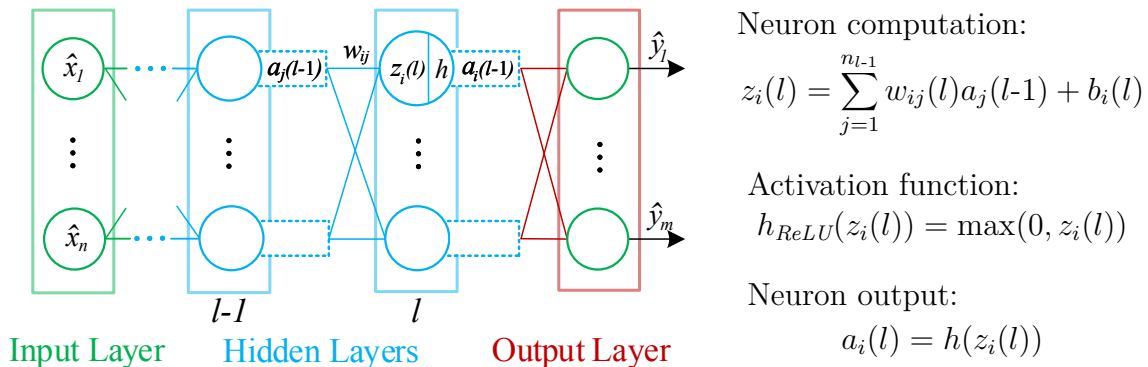


Figure 6.2: General model of a feedforward, fully connected neural network, illustration based on Gonzalez and Woods (2018, p. 946).

Figure 6.2 shows a general example of a feedforward neural network based on a fully connected neural network. The input quantity vector $\hat{\mathbf{x}}$ is processed through a number of hidden layers l and classified by an output layer, such as softmax for classification, to generate the output quantity $\hat{\mathbf{y}}$. Each layer consists of a number of neurons n_l to interpret the previous quantity vector $\mathbf{a}(l-1)$. The number of neurons of hidden layers can differ between individual layers. Each neuron consists of a mathematical formulation \mathbf{z} with trainable parameters (weights \mathbf{w} , bias \mathbf{b}) and a subsequently applied activation function h , whereby the Rectified Linear Unit (ReLU) is most common. During training, the network parameters are optimized in an iterative manner based on

backpropagation and gradient descent, where α represents the learning rate, formulated as

$$w_{ij}(l) = w_{ij}(l) - \alpha \frac{\delta L}{\delta w_{ij}(l)} \quad \text{and} \quad b_i(l) = b_i(l) - \alpha \frac{\delta L}{\delta b_i(l)}. \quad (6.1)$$

Given a dataset of corresponding input and output quantities, the current feed forward result $\hat{\mathbf{y}}$ is compared to a GT \mathbf{y} based on a loss function $L(\hat{\mathbf{y}}, \mathbf{y})$. The error is then propagated back through the network structure based on the chain rule to update network parameters. Neural networks with at least two hidden layers are called DNNs.

While the development of neural networks dates back to the 1940s (Goodfellow et al., 2016), their development and application has experienced a drastic boom in the last decade. The received attention can be subjected to four main factors. First, major methodical developments of the last decades allowed the application and training of DNNs for various tasks. For instance, Convolutional Neural Networks (CNNs) introduced spatial relationships between pixels and heavily reduced the amount of required model parameters. They have become the approach of choice for complex image recognition tasks, exemplified with the superior performance of a CNN at the ImageNet challenge (Krizhevsky et al., 2012). Second, the size of training datasets is drastically increasing, which reduces the degree to which statistical generalization is a challenge for DNNs (Goodfellow et al., 2016). Third, more powerful computing units allowed the training of larger networks and application of larger datasets. Computing based on Graphic Processing Units (GPUs) has become standard in DL. For instance, NVIDIA specifically geared toward DL. They introduced Tensor cores that are specialized on efficient matrix computation and utilize efficient training on work stations (e.g., RTX Quadro), and further developed mobile computing units for inference during operation (e.g., NVIDIA Jetson). Fourth, the software infrastructure significantly improved, due to the development of open source and user friendly software libraries for DL. They allow a flexible, modular design and efficient training, inference, and evaluation of DNNs. For instance, a popular DL library is Tensorflow (Abadi et al., 2016).

The success of DL in image segmentation is based on several developments, for which (Minaee et al., 2021) provides a comprehensive overview. A fundamental step was the application of fully CNNs (Long et al., 2015), which allow to output a prediction of the same size as arbitrarily-sized input images. Other main developments are based on encoder-decoder networks or based on spatial pyramid pooling modules.

The first approach allows to first encode a dense and high-level representation of the image and subsequently decode a high-resolution segmentation map (Noh et al., 2015). The resulting segmentation map, however, can show a loss of fine grained image information. This is addressed by U-Net (Ronneberger et al., 2015), for example, by using skip connection between a symmetrical encoder and decoder structure.

The second approach refers to networks from the DeepLab family (Chen et al., 2018a). A key component of their work is the application of atrous convolution that dilates a kernel by a specific dilation rate and thus can expand the receptive field of the kernel without increasing the number of model parameters. They further proposed Atrous Spatial Pyramid Pooling (ASPP) that applies multiple kernels with different dilation rates in one layer to handle object segmentation at different scales. Already proposed in (Chen et al., 2014), they combined their network with an iterative refine-

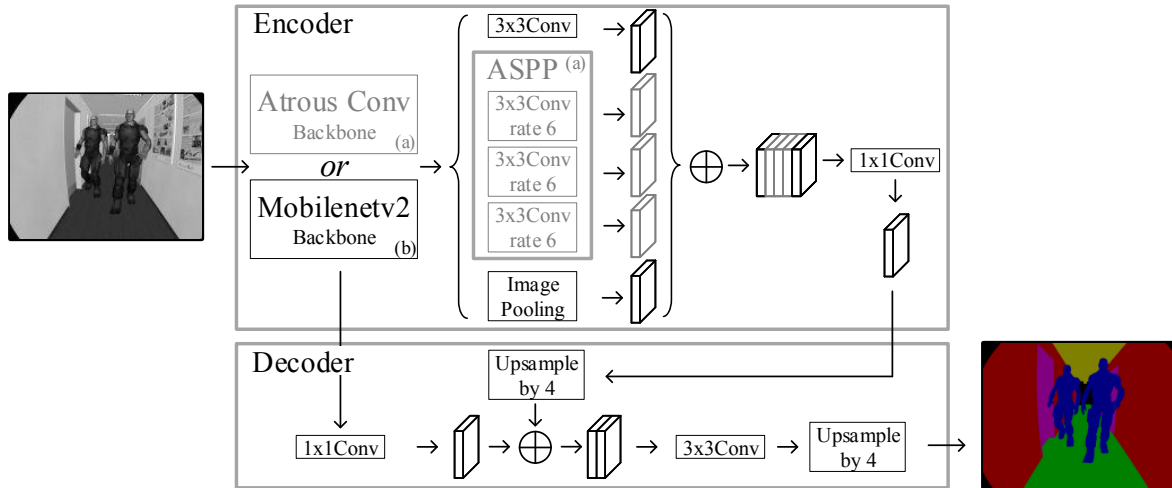


Figure 6.3: General structure of the encoder-decoder design of (a) Deeplabv3+ (Image based on Chen et al., 2018b) or (b) of Deeplabv3+ Mobilnetv2 (Sandler et al., 2018).

ment procedure based on Conditional Random Field (CRF) to improve localization of object boundaries.

A network structure that combines both approaches is Deeplabv3+ (Chen et al., 2018b) that uses an encoder-decoder architecture and includes dilated separable convolutions, sketched out in Figure 6.3 (a). The encoder consists of cascaded atrous separable convolutions, followed by an ASPP module. The decoder combines the results of both modules and refines the segmentation results. They published their Tensorflow implementation and provide pre-trained weights based on different backbones and trained on different public datasets [DeepLab]. (Sandler et al., 2018) specifically addressed the high computational demands and proposed the Mobilenetv2 architecture. This network is based on inverted residuals between bottleneck layers that comprise of depthwise separable convolutions. Separable convolution consists of spatial convolution for each channel (depthwise) and a following 1×1 convolution (point wise), and reduces the computational complexity in comparison to the standard convolution layer. It is optionally integrated into the Deeplabv3+ structure, see Figure 6.3 (b), in the way that the deep CNN structure is replaced by a Mobilenetv2 backbone and that the dilated convolution kernels of the ASPP layer are removed.

6.2 Evaluation Metrics

The evaluation of semantic segmentation usually refers to quantifying model accuracy (Minaee et al., 2021). The most commonly used metrics are the mean Pixel Accuracy (PA) and mean Intersection over Union (IoU). Further alternative metrics are the classification metrics Precision, Recall and F1-score, which however are not further considered in this work.

The PA is defined as the ratio of correctly classified pixels to the total number of pixels in the image. It is stated by its mean for $K+1$ classes with K foreground classes and one background class, and is formulated as

$$\text{mPA} = \frac{1}{K+1} \sum_{i=0}^K \text{PA} = \frac{1}{K+1} \sum_{i=0}^K \frac{p_{ii}}{\sum_{j=0}^K p_{ij}}. \quad (6.2)$$

The IoU is defined by the area of intersection between the predicted and GT segmentation map, divided by area of union. This metric formulation is visualized in Figure 6.4. The mean IoU also describes the mean over all classes and is the most popular metric for semantic segmentation. It is formulated as

$$\text{mIoU} = \frac{1}{K+1} \sum_{i=0}^K \text{IoU} = \frac{1}{K+1} \sum_{i=0}^K \frac{|A \cap B|}{|A \cup B|}. \quad (6.3)$$

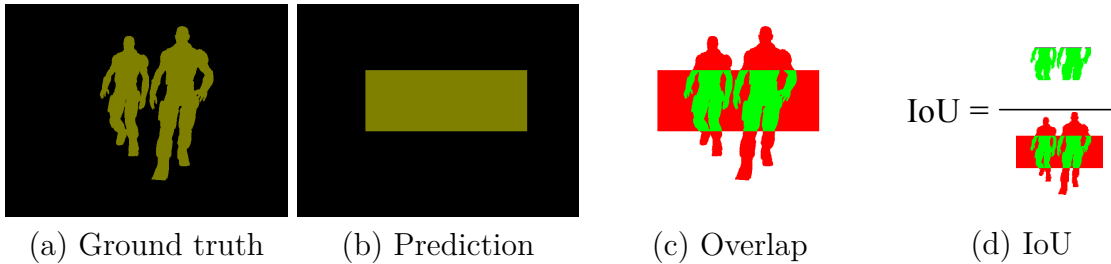


Figure 6.4: Visualization of the mean IoU estimation for semantic segmentation.

Chapter 7

Analysis and Improvement of Visual-Inertial Navigation in Dynamic Indoor Environments

Self-localization systems for first responders must be able to work reliably in highly dynamic environments, which may be characterized, for example, by a high number of moving people and vehicles. Camera-based localization in such environments is challenging. Feature-based localization methods need to reliably reject keypoints that do not belong to the static background. Here, traditional statistical methods for outlier rejection quickly reach their limits. A common approach is the combination with an

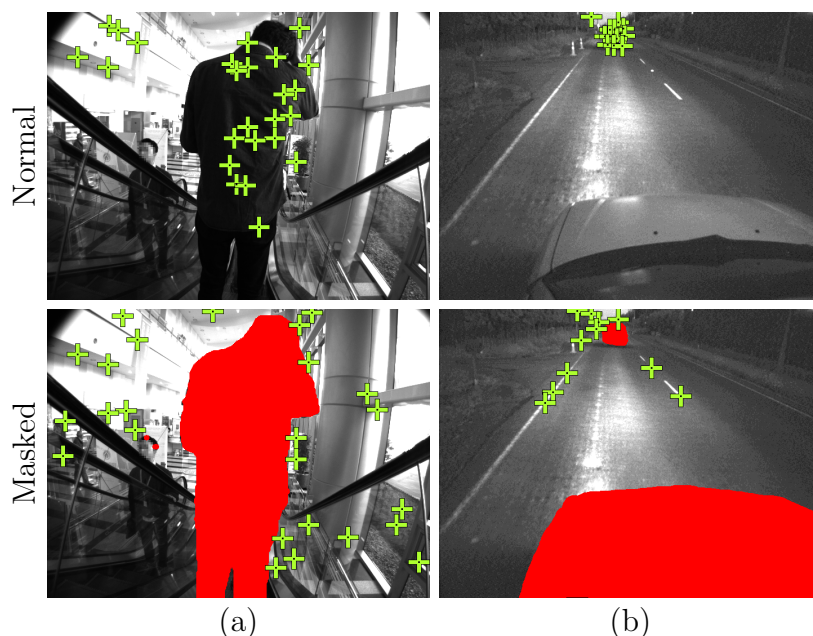


Figure 7.1: Illustration of Irmisch et al. (2020) that shows exceptional cases from hand-held localization on an escalator (a) and vehicle localization at night (b, with increased brightness for better visualization). It shows selected features (green crosses) used for pose estimation in the VO component of IPS without (Normal) and with (Masked) the segmentation aid (red area). The hand-held system is reconsidered in this chapter.

IMU for VIO, such as in IPS. Also, semantic segmentation based on DNNs was recently successfully applied in visual localization to identify features on certain object classes. In (Irmisch et al., 2020), we studied the application of mask-based feature selection based on semantic segmentation in IPS for robust localization in high dynamic environments. A pre-trained DNN was used to segment persons and cars in the image. The method was evaluated based on several different datasets from different IPS prototypes, with Figure 7.1 exemplifying two challenging scenarios. In addition to IPS, we also considered ORBSLAM2-Stereo (Mur-Artal and Tardós, 2016) to generalize our observations. In this thesis I concentrate only on the hand-held sensor system IPS.

In this chapter, the experiments with the hand-held system in indoor environments are reconsidered. The challenge is to navigate in an environment with various homogeneous surfaces on static objects and a dense presence of moving people, which in combination introduce a critical ratio of features on static and dynamic objects. This scenario is especially interesting in the context of first responders, for example, during the rescue of victims in a rampage scenario (*indoor rescue*, Section 1.2).

The contribution presented in this chapter is a detailed analysis of the influence of dynamic objects on the VIO result on the example of moving persons in indoor environments and the identification of counter measures to reduce this influence. A specific focus of this section is the application of semantic segmentation, a detailed analysis in simulation, and a confirmation in real world.

The chapter is organized as follows. At the beginning, the integration of semantic information into IPS is explained and technical aspects are noted about the configuration settings. Then, three experiments are conducted that allow in-depth analyses. First, a sensitivity analysis is conducted simultaneously in simulation and real world in a highly dynamic corridor environment (Section 7.2.1). It is used to analyze the influence of image frame rate, image resolution with stricter feature matching, level of uncertainty, and the application of semantic segmentation. Second, a *combined sensitivity analysis* follows that is used to weight the influence of dynamic environment parameters to other error sources, such as calibration inaccuracies or camera noise, and to verify propagated uncertainties from VO (Section 7.2.2). A large-scale real-world mall dataset is investigated to confirm the possible benefit of the segmentation aid in a possible first responder scenario (Section 7.3). Finally, a short summary of the results ends this chapter.

7.1 Semantic Segmentation for Feature Selection

This section shortly describes the introduction of semantic segmentation into IPS, which is processed as a mask and used for feature selection. Furthermore, technical notes are provided that comment on the used configurations of IPS and their run-times.

7.1.1 Segmentation Aid

Semantic segmentation is used to support the rejection of detected features on moving objects in the VO component. As summarized in Figure 7.2, a mask is generated based on pixel-wise classification of defined object classes and is used to reject point feature candidates during feature detection. In this chapter, I consider the class *person*, which

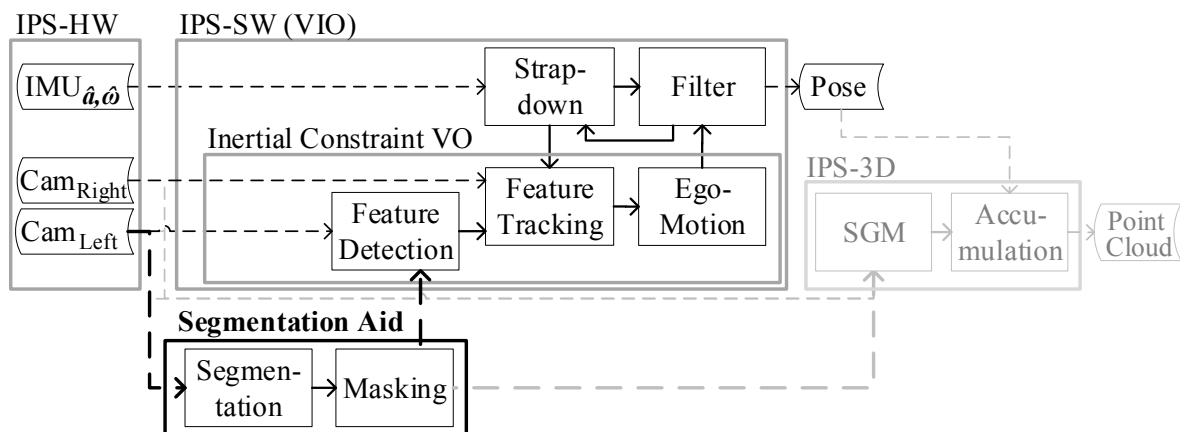


Figure 7.2: Extension of VIO (Section 3.2) with a segmentation aid and optional (not-used) extension of the 3D reconstruction component (grey, used in Irmisch et al., 2021).

I assume is constantly moving to exclude all small movements, for experiments with the hand-held system. For pixel-wise classification, Deeplabv3+ (Chen et al., 2018b) with a Mobilenetv2 (Sandler et al., 2018) backbone (Section 6.1) is used with pre-trained weights, downloaded from [Deeplab]. The network was pre-trained by the authors on the augmented training dataset of COCO (Lin et al., 2014) for semantic segmentation and is able to segment 20 different object classes and the background. Based on the segmentation of the target object class *person*, a mask is generated that defines the belonging to forbidden object classes. To compensate inaccurate segmentation borders and difficult object-assignable or object-close image features, the mask is dilated by 3 px, oriented on the feature radius of AGAST (Section 3.2.1). The application of the mask is implemented in the feature detection phase. After a point candidate is proposed by the corner detector, the image position is verified with the mask and accepted or rejected accordingly. Further selections of the features to use, e.g., with non-max suppression, follow and remain unchanged.

7.1.2 Technical Notes

In this section, I shortly comment about the implementation of IPS and the segmentation aid extension as well as its configuration and run-times to provide a rough impression about required computational resources. All experiments are conducted on a powerful work station [DellPrecision] with an Intel Xeon W-2145 processor (maximum boost frequency up to 4.5 GHz) and a Quadro RTX 6000 graphics card.

The implementation of IPS currently consists of three parallel main computational threads for (i) feature detection and intra matching, (ii) inter matching and ego-motion estimation, and (iii) the Kalman filter. The segmentation aid adds two more threads: (iv) semantic segmentation and (v) mask generation.

The segmentation aid is implemented within a development and research framework. While IPS is implemented as a multi-threading framework in a C++ environment for real-time purposes, most open source deep learning networks are realized and published in Python for fast development and deployment purposes. A replication in C++

is a laborious task and often not provided. Therefore, I used [Pybind11] to implement an interface from C++ to the Python-based Tensorflow-implementation of [Deeplab]. This approach allows to easily apply any Tensorflow- and Python-based DNN implementation in the C++ framework of IPS. Though, it complicates the installation process of IPS with the segmentation aid on new platforms.

Two different IPS configuration settings were chosen in consultation with experienced IPS users, while the base method is *WLS+*, introduced in Chapter 5. First, *IPS-fast* provides real-time localization on computational restricted platforms by processing the images in half resolution (680×512 px) at 10 Hz, while the feature matching properties are optimized for maximum speed. The semantic segmentation module for *IPS-fast-masked* processes each image at (512×384 px) resolution, which requires around 15 ms on the GPU. Second, *IPS-accurate* runs in near real-time by processing the images in full resolution (1360×1024 px) at 10Hz and uses slightly larger image patches, i.e., 7×7 instead of 5×5 , which entails a stricter selection during feature matching. The semantic segmentation module for *IPS-accurate-masked* processes each image in full resolution, which requires around 85 ms on the GPU.

Figure 7.3 summarizes the run-times. For *IPS-fast*, the IPS pipeline is able to compute a system pose for stereo images around every 50ms in average, if sensor data latency is disabled. Therefore, it is easily applicable on computational restricted hardware without losing real-time capability. The additional segmentation aid in *IPS-fast* does not decrease the throughput. For *IPS-accurate*, the observed mean throughput is around 140ms, and can therefore only be considered as near real-time capable. Interestingly, the average throughput of this configuration is around 95ms for the corridor dataset. The bottleneck appears to be the feature detection thread (*i*), which has to handle less feature candidates in the homogenous corridor environment. The average throughput is increased in *IPS-accurate-masked* by the segmentation aid, which adds additional queries for each detected feature in thread (*i*).

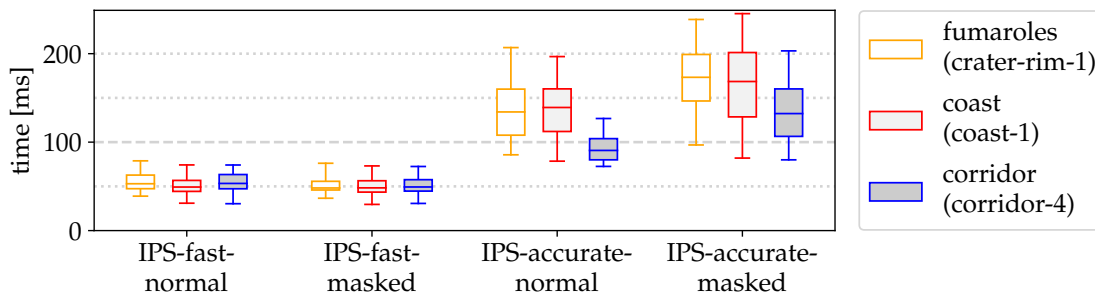


Figure 7.3: Throughput of system pose estimations for stereo images, if sensor data latency is disabled. A throughput below 100ms implies real-time capability at 10Hz.

The configuration of *IPS-fast* could be easily adapted to a specific computing platform and application to increase the localization performance without losing real-time performance. Though, this is not the focus of this thesis and only one configuration of *IPS-fast* is used for all experiments in this thesis. Furthermore, the overall implementation, specifically pointing to the developments of this work, is intended for research purposes and can be optimized in the future.

7.2 Corridor - Sensitivity Analysis

Two experiments are presented in this section to analyze the influence of dynamic objects on VIO in depth. First, a sensitivity analysis is conducted in a highly dynamic corridor environment, simultaneously in simulation and real world. It is used to analyze the influences of image frame rate, image resolution, level of uncertainty, and application of semantic segmentation. Second, a combined sensitivity analysis follows for one selected scene. It is used to weight the influence of dynamic environment parameters to other error sources and to evaluate the propagated uncertainties from VO.

7.2.1 Simulation and Real-World Sensitivity Analysis

The first experiment is designed to determine the limits of VIO in dynamic indoor environments and to exploit the potential of the segmentation aid on the example of IPS. IPS is evaluated in a similar dynamic corridor environment in real world and in simulation, using the hand-held IPS (Chapter 3) and its digital twin (Chapter 4). The experiment setup corresponds to a sensitivity analysis (Section 4.3.1). In the following, the overall setup of the experiment is explained and then the results are analyzed in detail by showing additional quantitative and qualitative results.

Experiment Setup

The experiment is based on the corridor dataset that consists of 7 real and 6 synthetic recordings in a similar corridor environment. The datasets contain humans that are mostly walking or standing with small movements. The trajectories of both sources consist each of walking a short distance, illustrated in Figure 4.8 (a, p.45), but the individual recordings differ in the level of dynamic and the presence of humans. For instance, the dataset *sim-corr-d2* consists of two humans walking consistently in front of the camera. Or, the dataset *sim-corr-d4* consists of two humans walking toward the camera, while another two are observed starting to walk slowly. Tables A.2 and A.4 (Appendix A.2) provide additional information about camera dynamics and path lengths. The real camera images were recorded with a frame rate of 30 Hz and sorted out for 10 Hz and 5 Hz. The simulation provides complete GT, while for the real-world dataset only two GCPs with a distance of 16m at the beginning and the end of each session are used as reference. To ensure the image quality of synthetic datasets, a super-sampling factor of 5 is used and 21 images are accumulated to simulate motion blur with an exposure time of 5ms.

Three different IPS configurations are examined, each applied without (*normal*) and with (*masked*) the segmentation aid. *IPS-fast* is applied with both calibration settings *I* and *F* (Section 5.1) to investigate the influence of the level of system uncertainties, with small uncertainties in *I* and large uncertainties in *F*. Additionally, *IPS-accurate* is applied to investigate the influence of image resolution with stricter feature matching, using calibration setting *F*. Each method is applied both in simulation and real world, and at different image frequencies (5 Hz, 10 Hz, 30 Hz).

For the application of IPS on synthetic datasets, the GT semantic segmentation from the simulator is used as segmentation aid in IPS. This is necessary to exclude all misleading effects from evaluation that arise from incomplete segmentation of hu-

mans. For instance, Figure 7.5 (b) shows a scene where the right person was not fully segmented by the DNN, which in turn lead to a false ego-motion estimation.

Results

The results are visualized using the Cumulative Distribution Function (CDF) in Figure 7.4, distinguished between simulation and real world and different camera frame rates. Each line shows all ATEs of one method as a whole for 6 runs in simulation and 7 in real world, and for each 20 repetitions.

The segmentation aid (dotted lines) generally increases localization results over IPS in standard configuration (solid lines). This boost is distinctive at 30Hz and negligible at 5Hz. Figure 7.5 (c) shows one example where *IPS-normal* fails to correctly distinguish between static and moving objects, which could be resolved using the segmentation aid in *IPS-masked*.

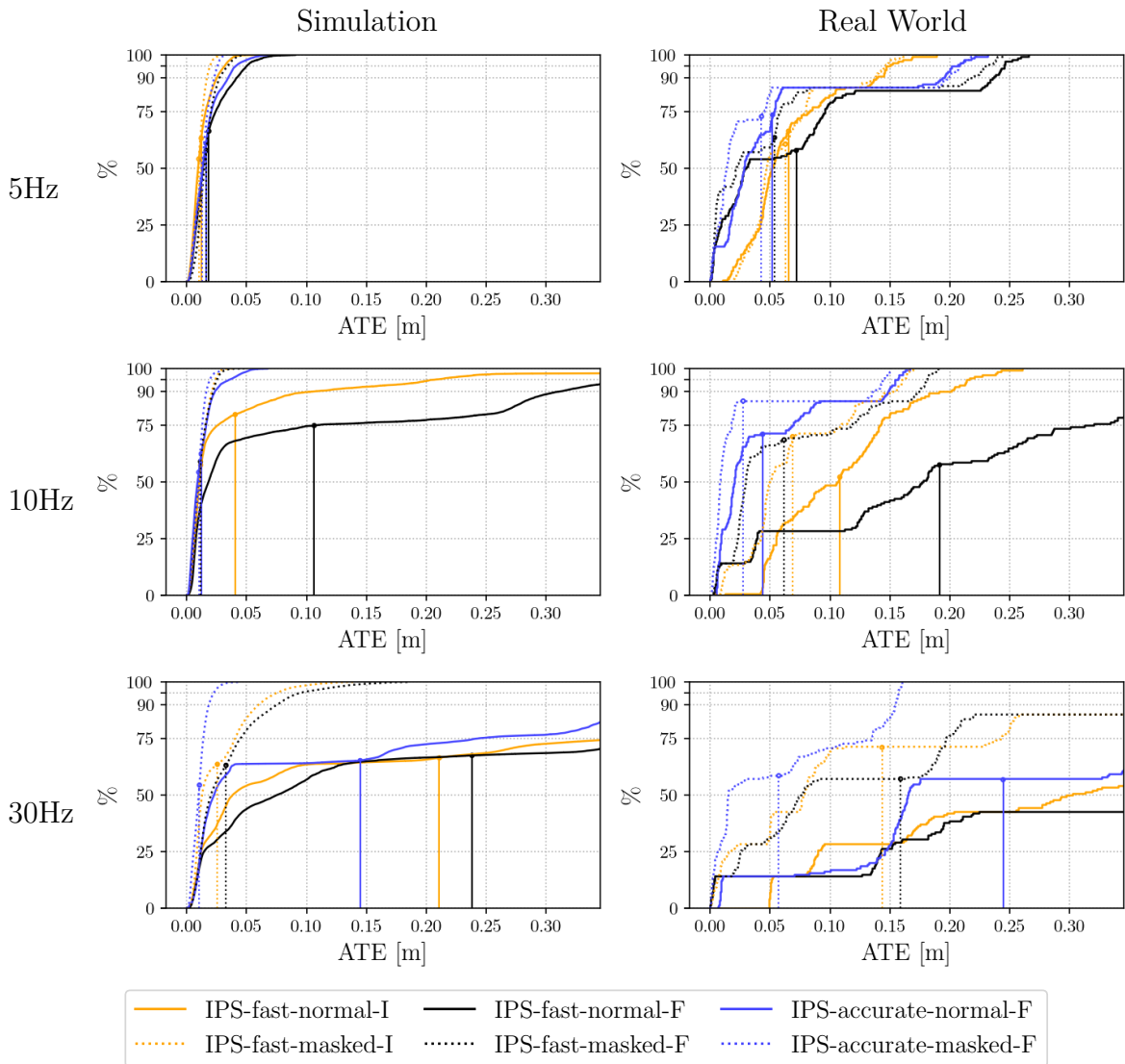


Figure 7.4: Results of the sensitivity analysis for the corridor datasets in CDF-representation. The vertical lines mark the mean ATE for each CDF.

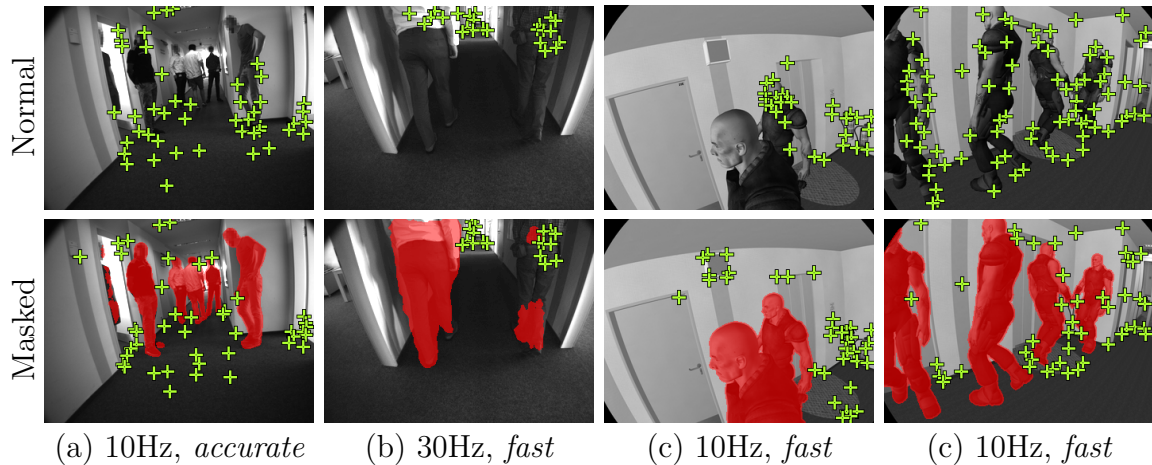


Figure 7.5: Illustration of selected features (green crosses) used for ego-motion estimation after RANSAC filtering with optional application of the segmentation aid (red area). Brightness is increased for better visualization (a,b).

A lower image frame rate increases localization results over localization with high image frequency for all *IPS-normal* settings. Due to higher frame rates, object motions are less pronounced in the image and features on slowly moving objects are more likely to be used in the VO ego-motion estimation. Illustrated in Figure 7.6 (a,b,c), more features are used on slowly moving humans with higher image frequency.

A higher image resolution with stricter feature matching (*IPS-accurate*) similarly shows to improve localization in dynamic environments in comparison to localization with lower resolution (*IPS-fast*). This might also be attributed to more pronounced object movements, such as indicated by Figure 7.6 (d) in comparison to (b).

A more certain IPS system shows to be less influenced by dynamic objects. Figure 7.4 clearly shows a better performance of *IPS-fast-normal-I* over *IPS-fast-normal-F*, which only differ in the applied calibration settings with small (*I*) and large (*F*) uncertainties. If the segmentation aid is applied, *IPS-fast-masked-I* and *IPS-fast-masked-F* do not significantly differ. This indicates a connection between the internal uncertainty of IPS and the influence of moving objects, which is analyzed further in Table 7.1. It shows the results for the simulated dataset *sim-corr-d04*, where slowly walking humans appear to have a strong influence and false VO measurements are frequently computed, such as visualized in Figure 7.5 (c). The internal uncertainty of IPS is exemplified by

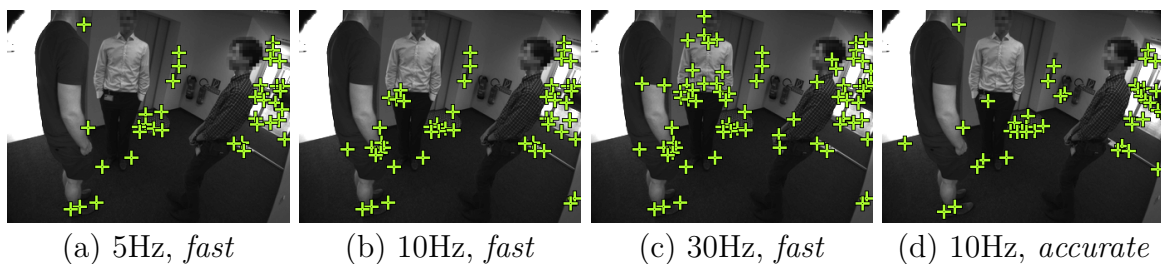


Figure 7.6: The effect of image frame rate (a,b,c) and resolution (d) on the set of selected features (green crosses) after RANSAC filtering. Brightness is increased.

Table 7.1: Analysis of the relation between internal system uncertainties and the influence of moving objects based on *IPS-fast-normal* and simulated dataset *sim-corr-d4*.

χ^2 -test:	activated		disabled	
Calibration:	(I)	(F)	(I)	(F)
mATE [m]	0.27	0.37	0.34	0.41
$\hat{\sigma}_x$ of $\Delta\mathbf{T}$ [mm]	0.64	1.24	0.63	1.23
$\Delta\mathbf{T}$ rejected	12	1	0	0

the propagated uncertainty $\hat{\sigma}_x$ for the translation in x -direction of the estimated relative transformation $\Delta\mathbf{T}$. The results show that a smaller internal uncertainty (I) enables the chi-squares test of the Kalman filter (Section 3.2.3) to correctly identify the VO measurement as an outlier and reject the measurement accordingly. If the internal uncertainty is high (F), only very few false measurements are rejected and the overall mean ATE is high. If no VO measurements are rejected, due to a disabled chi-squared test, then the mean ATE is even worse.

The masking of static humans can also decrease the localization performance. This is exemplified on the simulated dataset *sim-corr-s02* (detailed in Table 7.2). The dataset includes 9 simulated persons that are kept static and therefore do not introduce dynamics into the scene, as illustrated in Figure 7.5 (d, top). Though, *IPS-masked* still ignores all features on humans, see Figure 7.5 (d, bottom). Consequently, less features are used for the VO ego-motion estimation, specifically around 42 features in average instead of 64 features for this example. This results in a less accurate (higher mean RTE) and more uncertain (higher $\hat{\sigma}_x$ of $\Delta\mathbf{T}$) VO ego-motion estimation, and consequently in a worse localization result (higher mean ATE). Though, this loss in performance is rather insignificant in comparison to the error that results from moving objects, which dominates the results in Figure 7.4.

Comparing the results from simulation and real world, they show strong correlations despite different evaluation metrics. It is particularly striking that *IPS-accurate-masked* performs mostly similar at all frame rates, in simulation and real world respectively.

Table 7.2: Analysis of the influence of non-moving humans based on *IPS-fast* and simulated dataset *sim-corr-s02*.

method:	IPS-fast-normal		IPS-fast-masked	
calibration:	(I)	(F)	(I)	(F)
mATE [m]	0.0063	0.0071	0.0124	0.0113
mRTE [mm]	0.66	0.78	1.19	1.41
$\hat{\sigma}_x$ of $\Delta\mathbf{T}$ [mm]	0.46	0.85	0.75	1.44
used features	64.8	64.6	42.1	41.8

7.2.2 Combined Sensitivity Analysis

A limitation of the previous experiment is the limited versatility in the datasets with respect to other error sources. For instance, the camera calibration was assumed to be perfect and camera noise was reduced to a minimum, oriented on the observed noise from the real-world system in ideal conditions. Though, experiments under such ideal conditions are less meaningful for application in a first responder context. The objective of the following experiment is to introduce such error sources and to additionally weight their influence against each other. Therefore, the *combined sensitivity analysis* (Section 4.3.3) is applied in this section.

Experiment Setup

The experiment is based on the artificial corridor dataset with slow moving humans, which was introduced and visualized in Section 4.2.1. Table 7.3 summarizes the considered parameters that are sampled during each iteration in a Monte-Carlo manner. Four different types of parameters are considered (introduced in Section 4.3). Starting from the bottom, all geometric calibration parameters are varied based on their specified distribution from the first responder calibration set F . This includes intrinsic camera parameters κ and distortion parameters δ , and the extrinsic parameters for the stereo transformation \mathbf{T}_l^r and registration to the IMU \mathbf{T}_l^n . Further, image blur and image noise are considered to account for possible adverse conditions that may degrade camera properties. Both are sampled in a reasonable range, which was predetermined with a single parameter sensitivity analysis (Appendix B.1). Capture gain defines the noise ratio based on formulas 3.8 - 3.11 (Section 3.1.2), but does not change the image intensity in the simulation. Next, one design parameter is considered that describes the simulated size of the stereo baseline, sampled closely around the baseline value of the real-world system. Finally, two environmental parameters are considered, which are person speed and person height (Section 4.2.1). Person speed is sampled in the time domain and measures how much time both persons in front of the elevator need to move 1.5 m. Figure 7.7 exemplifies simulated images of four different randomly-selected parameter set samples.

The experiment follows the procedure of the combined sensitivity analysis, described in Section 4.3.3. 900 different parameter sets are sampled and used for simulation and

Table 7.3: Considered parameters and sample distributions for the conducted combined sensitivity analyzes. Considered are calibration parameters \mathbf{P}_C , camera properties \mathbf{P}_P , system design parameters \mathbf{P}_D , and environment parameters \mathbf{P}_E (Section 4.3).

name	type	sample distribution	allowed range
person speed (time) [s]	\mathbf{P}_E	$P_s \sim N(11.5, 2.5)$	$6.5 < P_s < 16.5$
person size [m]	\mathbf{P}_E	$P_h \sim N(1.56, 0.39)$	$0.78 < P_h < 2.34$
baseline [m]	\mathbf{P}_D	$B \sim N(0.2, 0.03)$	$0.05 < B < 0.35$
Gaussian blur [px]	\mathbf{P}_P	$\sigma_{blur} \sim N(0, 1)$	$0 < \sigma_{blur} < 5$
capture gain factor	\mathbf{P}_P	$C \sim N(0, 5)$	$0 < C < 20$
geometric calibration	\mathbf{P}_C	$\kappa, \delta, \mathbf{T}_l^r, \mathbf{T}_l^n$ of calibration F (Appendix A.3)	



Figure 7.7: Exemplary images from different samples of the combined sensitivity analyzes that show the variation in camera properties and environment parameters.

for application of *IPS-fast-normal-F* and *IPS-fast-masked-F*. The mean ATE is used for trajectory evaluation. The data is analyzed based on a correlation plot, scatter plots, and evaluation of the normalized error (Section 3.3.2).

Results

The correlation of the resulting mean ATE with the varied input parameters is presented in Figure 7.8. The calibration parameters are reduced in this plot to intrinsic parameters κ^l of the left camera for better visualization. Considering *IPS-normal* (based on *IPS-fast-normal-F*), the strongest correlation with the mean ATE can be observed for the environment parameters person size and person speed, which was expected due to the chosen setup of this experiment. For instance, the ratio of features on dynamic objects increases if the person is taller, which in turn increases the probability of corrupted or false VO estimations. The influence of other parameters show generally less correlation with the mean ATE. Considering *IPS-fast-masked-F*, the influence of the dynamic environment parameters vanishes since all persons are completely masked out during feature detection, using the GT-segmentation. Instead, the blurring shows to be a significant factor in this case. Also, the error of the principal point in horizontal direction u_0^l seems to be relatively significant, which is comprehensible since u_0^l directly affects the scale of the trajectory. Interestingly, the design parameter stereo baseline shows a negative correlation, which indicates that a larger baseline is generally more suited than a smaller one.

However, as mentioned in Section 4.3.3, the correlation analyzes is relatively limited due to high-nonlinearity in the transfer function, transferring the input parameters to the mean ATE. Therefore, the observations needs to be confirmed by directly observing the data in terms of a sensitivity analysis, for instance, using a scatter plot.

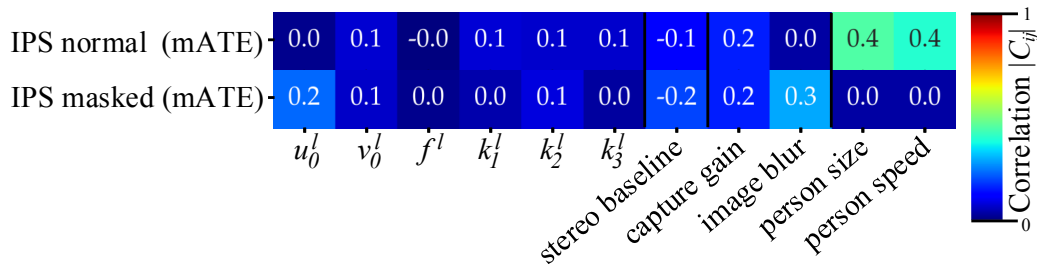


Figure 7.8: Correlation analysis for *corridor* with 9 out of 29 varied parameters.

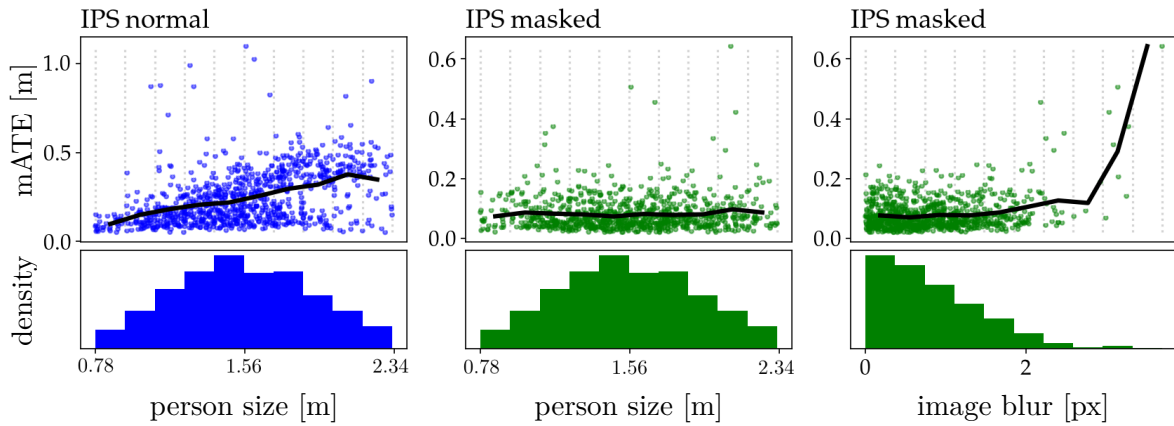


Figure 7.9: Sensitivity analysis for the *corridor* dataset. Points show the mean ATE of one sample with respect to the sampled parameter (top), whose sample distribution is shown (bottom). Lines connect the mean of samples that are divided into 10 bins.

Figure 7.9 (left) shows the resulting mean ATE for each sample with respect to the corresponding environment parameter person size. It shows that taller persons in the image lead to higher errors of the trajectory, which confirms the observation from the correlation analyzes. If the segmentation aid is applied, the influence of the person size vanishes, as shown in Figure 7.9 (middle). The observed influence of image blur is visualized in Figure 7.9 (right). It reveals that the observed correlation comes from a few samples with strong image blur. Even though this effect is comprehensible, this range of image blur is not densely sampled and should not necessarily dominate the correlation analysis. In future, an outlier rejection scheme could be considered to reduce the influence of single samples.

The overall results over all samples are noted in Table 7.4, where the improvement by the segmentation aid is quantified in terms of mean ATE for the final trajectory and mean RTE for VO estimations. The table additionally shows the SDs of the normalized error $\tilde{\sigma}$ (Section 3.3.2) for the translation components of $\Delta\mathbf{T}$, which was used in Section 5.3.1 for the geometric MCS and is also applicable for the combined sensitivity experiment. While *IPS-masked* shows SDs of the normalized error $\tilde{\sigma}$ that correctly tend to 1.0, *IPS normal* shows highly optimistic estimations with $\tilde{\sigma} \gg 1.0$. This is caused by corrupted or false VO estimations, where the error can not be correctly modeled by the estimated SD from error propagation. Figure 7.10 visualizes the distribution $(\tilde{\sigma}_x, \tilde{\mu}_x)$ of the normalized error \tilde{e}_x exemplary for the estimated translation in x -direction

Table 7.4: Results of the combined sensitivity analysis for *corridor* using *IPS-fast-F*.

method:	IPS-normal	IPS-masked
mATE [m]	0.246	0.081
mRTE [m] (VO)	0.0035	0.0026
$\tilde{\sigma}_x$	1.49	0.87
$\tilde{\sigma}_y$	1.51	0.96
$\tilde{\sigma}_z$	1.26	0.85

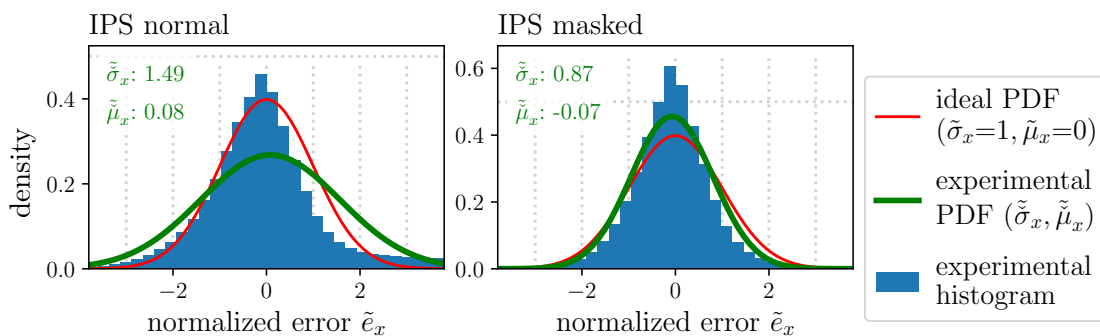


Figure 7.10: Evaluation of propagated uncertainties for VO ego-motion estimations.

of ΔT . For *IPS-normal* (left), false VO estimations introduce outliers, which lead to a non-Gaussian distribution. This experiment shows that VO errors that result from dynamic objects cannot be modeled by the propagated uncertainty.

7.3 Confirmation with Real World Data

In the previous section, the influence of dynamic objects was analyzed in simulation and based on real-world corridor datasets that were designed to bring IPS to its limits. The semantic segmentation aid has shown to contribute well to improve the robustness of IPS. In this section, IPS is tested on a large-scale real-world mall dataset to confirm the possible benefit of the segmentation aid in a realistic scenario. This dataset might be closest to a first responder scenario over all currently existing IPS datasets.

Experiment Setup

The dataset was recorded in 2014 for the indoor navigation competition at the international conference on Indoor Positioning and Indoor Navigation (IPIN, 2014). Exemplified in Figure 7.12, this dataset is challenging for visual localization due to the presence of densely crowded areas (a), people walking frequently ahead (b), strong light reflections (a,b), large homogeneous surfaces (c,d), numerous escalators scenes (c) between three floors, and strong camera motion (listed in Table A.6, Appendix A.2).

GT is provided in form of GCPs, for which the IPIN team provided latitude and longitude coordinates. 6 GCPs are selected for evaluation in this work, where the system was hold still for a few seconds. An additional barometer measurement of IPS was used to derive relative ground truth altitude information. This enables an

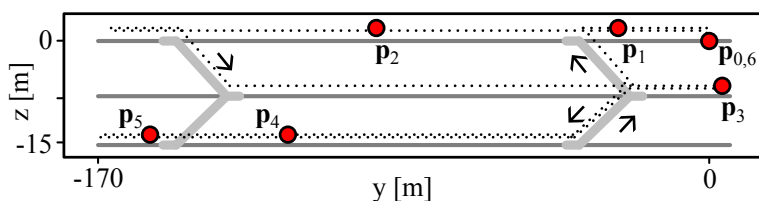


Figure 7.11: Illustration of the walked trail for the *mall* dataset (black dotted) and of the used GCPs (red points) for evaluation. Floor levels are connected via escalators.

evaluation based on the mean ATE. Figure 7.11 shows the trajectory of three floors and the distribution of the GCPs $\{\mathbf{p}\}_{i=0}^6$. The trajectory has a length of around 900 m and two sessions were recorded walking the same path, each with a different operator.

Images were recorded in half resolution (680×512 px) and at 10 Hz, which allows the application of *IPS-fast*. The parameter values of the calibration *ipin* are used in this experiment (Appendix A.3). They are based on a stand-alone and complete laboratory calibration, which was conducted directly before the IPIN challenge and can be assumed to be accurate and relatively uncertain. However, since such a calibration is rather unrealistic in the context of first responder localization, I still use the defined calibration uncertainty settings *I* and *F* for this setup. Though, the error in *ipin* is most likely not well represented by *I* and *F*.

Results

The results for the two runs of the same trail are listed in Table 7.5. First considering run *ipin-d2*, the results indicate that smaller uncertainties (*I*) generally show better results than larger uncertainties (*F*) in this setup. This might be attributed to the mentioned accurate geometric calibration of the system. The results of *ipin-d2* further indicate a general loss in performance of around 6 to 8% when applying the segmentation aid. This might be attributed to densely crowded areas, such as in Figure 7.12 (a), where people actually stand still during a conversation. They can provide additional static features that stabilize the localization, as considered with Table 7.2 in Section 7.2.1. In contrast, run *ipin-d1* shows a significant improvement of *IPS-masked-F* over *IPS-normal-F*, resulting in the overall best result for this experiment. This indicates a significant influence of dynamic objects on *IPS-normal-F* in this run. Considering calibration settings *I* for *ipin-d1*, the segmentation does not show an influence. This indicates that smaller system uncertainties in *IPS-normal-I* help to identify false VO measurements based on the chi-squared test of the Kalman filter, as considered with Table 7.1 in Section 7.2.1.

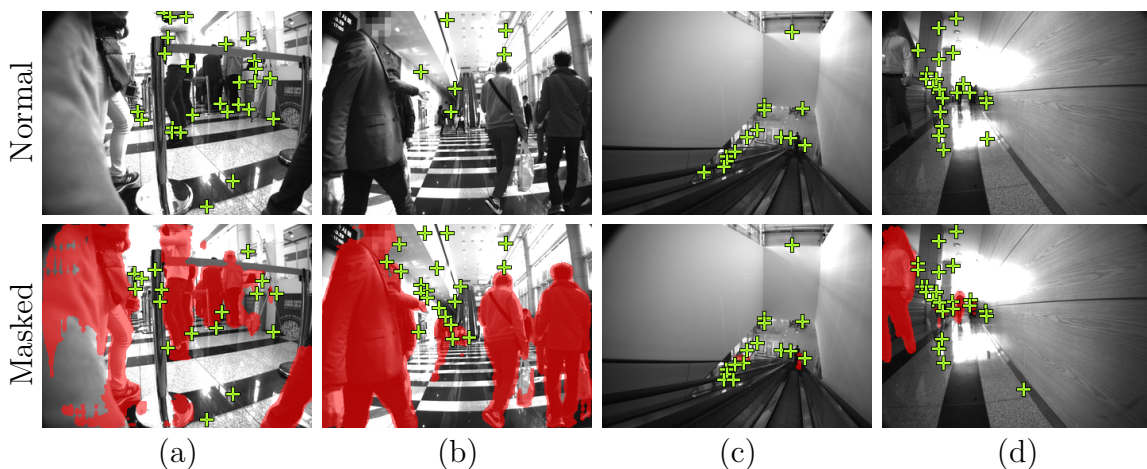


Figure 7.12: Examples from the *mall* dataset that show sets of selected features (green crosses) after RANSAC filtering with optional use of the segmentation aid (red area).

Table 7.5: Results of the application of *IPS-fast* on two runs of the IPIN dataset. Bold numbers mark the best result for each run. Particularities are marked in red.

method:	IPS normal		IPS masked	
calib.unc.:	(I)	(F)	(I)	(F)
ipin-d1	3.25	7.13	3.26	2.53
ipin-d2	2.61	3.23	2.81	3.44

The significant error for *IPS-normal-F* in run *ipin-d1* originates in an escalator scene. This scene is analyzed in Figure 7.13, exemplary for the determined height in local coordinates. First, the body-frame up-axis z^b of the IMU is shown to delimit the escalator scene. Before and after the escalator, the IMU measures the walking motion profile and only measures the gravity during the ride. Second, the estimated normalized relative translation of the VO estimation is shown. While the relative translation should be constantly high, the VO of *IPS-normal* frequently estimates zero-movement due to a high number of detected features on the person, as shown in Figure 7.1 (a, p.81). Using these VO estimations as aid in the navigation filter leads to wrongly estimated bias terms, exemplified for bias b_a on z^b in comparison with *IPS-masked*. As a result, the navigation solution fails to estimate the change in height for this scene, compared to a barometer measurement as reference. Contrary, *IPS-masked* is able to estimate the height almost similar to the reference.

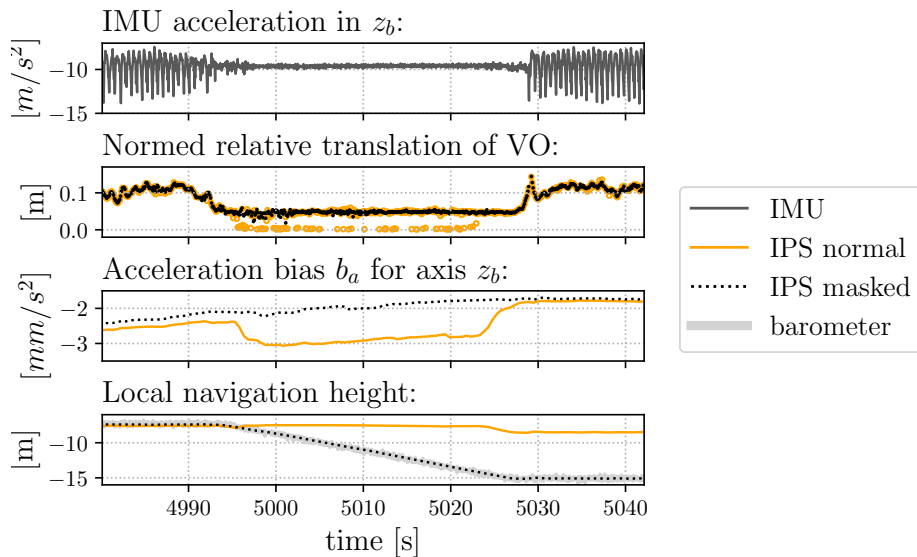


Figure 7.13: System- and localization parameters in local coordinates of *IPS-fast masked-F* on an escalator of dataset *ipin-d1*.

7.4 Summary

Moving objects can have a significant influence on visual localization, which was shown in this chapter by three experiments that considered persons as primary source of

dynamics in indoor environments. Three experiments were conducted that provide significant insights into the behavior of VIO systems in dynamic environments on the example of IPS and how it can be improved.

The experiment of Section 7.2.1 consisted of a sensitivity analysis for different parameters and was considered simultaneously in simulation and real world. The main insights are that the influence of dynamic objects can be reduced by (i) the segmentation aid, which benefit is limited if the considered objects are actually not moving, (ii) low image frequencies, (iii) high image resolution with stricter feature matching, and (iv) small system uncertainties. It further shows a well match of the results from simulation and real world.

The experiment of Section 7.2.2 consists of a combined Monte-Carlo based sensitivity analysis for one exemplary selected scene. The main insights are that (v) the influence of moving objects is more dominant than other error sources in this setup and that (vi) errors in VO from observed moving objects are not represented by the propagated SDs. It further demonstrates the applicability and also usefulness of this simulation strategy.

The experiment of Section 7.3 consists of a realistic large-scale real-world dataset for localization in a highly dynamic indoor environment. It confirms the benefit but also the limitations for (i) the segmentation aid and (iv) small system uncertainties. Most significant is the false localization in the escalator scene, which resulted in a false estimation of the floor level and is critical for localization of first responders. It could be resolved either based on (i) or (iv). This experiment further helped to explain how the error from false VO estimation influences the internal filter states and the final trajectory in VIO.

Chapter 8

A Sensor-AI Approach to Improve Visual Odometry in Adverse, Dynamic Environments

The application of self-localization by first responders is not restricted to environments with only common object types. Dynamic can be caused by various types of objects for which specialized pre-trained neural networks are not always available. Further, the influence of objects might depend on the used sensor system with special imaging hardware and processing methods, such as the used feature matching procedure. Training a dedicated DNN for application in different environments would require a vast amount of manual labeling to cover each possible dynamic object. This is not feasible for the high variety of different environments from first responder applications.

Motivated by Sensor-AI strategies (Börner et al., 2020), I propose an approach that targets to improve VO in adverse, dynamic environments with the help of specifically trained DNNs and without the need of extensive manual labeling. The related Sensor-AI strategy envisions a close interaction and combination of physical models, data-based models and classical approaches in one sensor system, primarily for applications that are defined by strict energy requirements. Similarly, the proposed approach com-

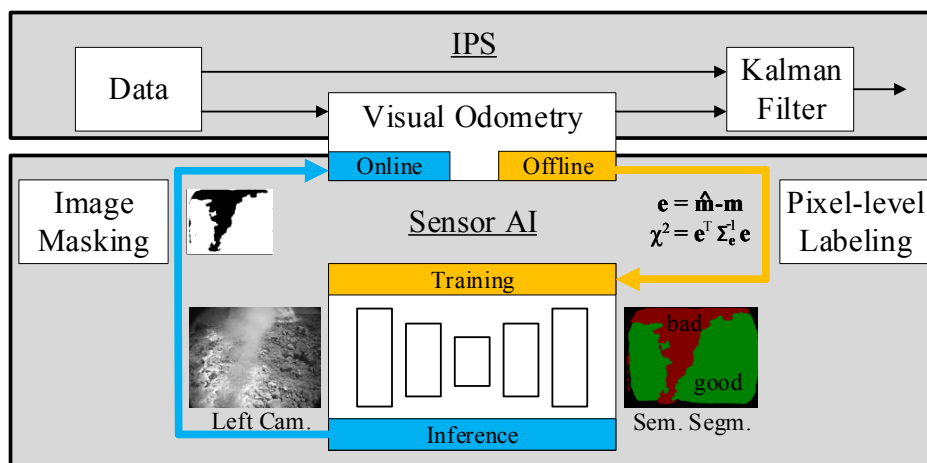


Figure 8.1: Proposed concept to improve VO-based sensor systems.

binizes classical feature-based VO, which relies on physical models, and analytical error propagation, with a data-based model in form of a DNN for semantic segmentation.

The proposed concept is visualized in Figure 8.1. It intends to use Sensor-AI with a DNN for semantic segmentation to automatically learn critical image areas from VO and use its prediction to improve the same VO method. Offline, training data is automatically generated based on semi-dense feature matching and statistical evaluation of reprojection errors in comparison to propagated uncertainties. The network is trained simultaneously for multiple environments and is trained on data that was recorded with the same sensor system. Online, the trained DNN is used to detect critical image areas to generate a mask for feature selection.

This chapter contains two contributions. The first contribution is a description and evaluation of the proposed Sensor-AI concept to improve VO-based sensor systems. The second contribution is an investigation of the influence of *smoke* and *water* on visual localization, on the example of IPS. This is done in simulation using the digital twin of IPS and in real world using the Sensor-AI concept. The investigations correspond to the motivated first responder scenarios *flood disaster* and *wildfire*.

The chapter is organized as follows. First, the automatic labeling procedure and a method to automatically select the most suited timestamps from the trajectory are introduced (Section 8.1). Second, the generated training, validation, and test datasets are explained and a DNN is trained and evaluated with focus on the environments *fumaroles*, *coast*, and *river* (Section 8.2). Third, the influence of smoke and water is first investigated in simulation with the strategy *combined sensitivity analysis* (Section 8.3). Fourth, real-world experiments are presented to investigate the influence of those elements in real world and to evaluate the improvement of VO (Section 8.4). Finally, a short summary ends this chapter.

8.1 Automatic Training Data Generation

In this section, methods are introduced to automatically generate training data. This includes an automatic pixel-level labeling procedure and an automatic timestamp selection to avoid redundant data labeling and ensure efficient training.

The presented procedures include multiple hyper parameters, which were determined manually in an empirical manner based on small reference datasets. The chosen values might not be optimal and can be improved in the future.

8.1.1 Pixel-level Labeling

The objective is to label an image without any manual work by an operator into good and bad areas with focus on observed static and dynamic objects. Therefore, the VO module itself is exploited and applied on a pair of stereo frames $\{q, a\}$. Reference stereo frame q contains the camera image to be labeled. a describes one stereo frame out of the local temporal neighborhood of q and consists of a left image a^l and a right image a^r . The overall pipeline is presented in Figure 8.2 and will be described in detail in the following. In summary, *(i,ii)* image features are densely detected and tracked between two stereo pairs. *(iii)* Their reprojection errors and related uncertainties are computed to *(iv)* separate them into the classes *good* and *bad* based on a chi-squared test, which

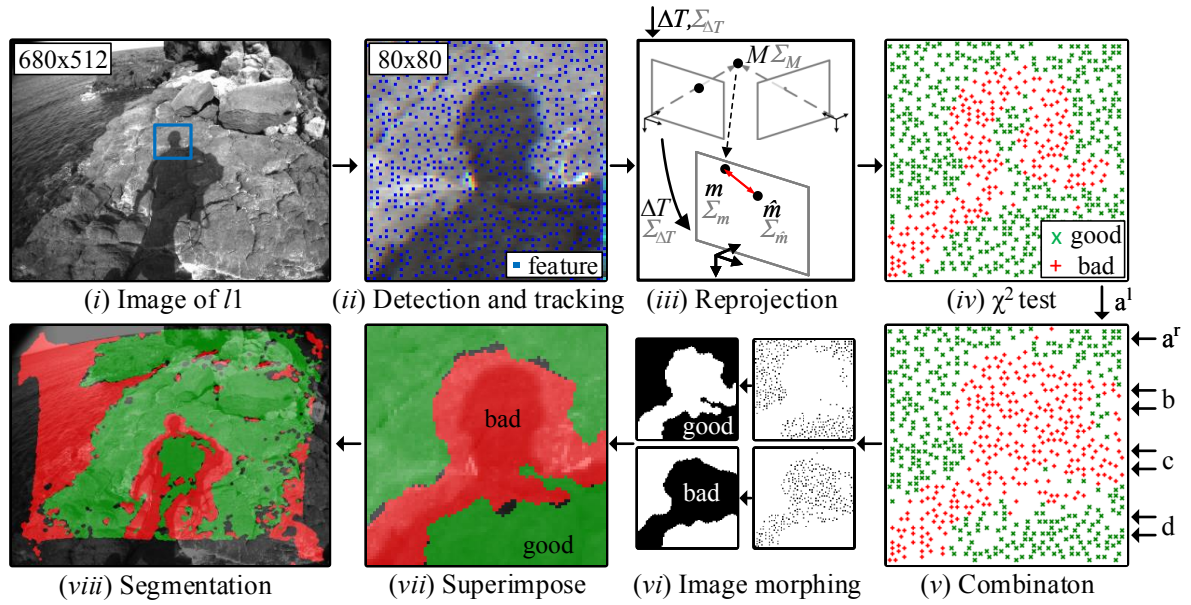


Figure 8.2: Illustration of the automatic pixel-level labeling procedure.

(v) is done for multiple neighboring stereo pairs. (vi) Image morphing is applied to close gaps and to (vii, viii) create the final reference segmentation image.

(i, ii) Image features are densely detected and tracked in the image. The used methods were introduced in Section 3.2.1. During feature detection, the AGAST intensity threshold is set to zero and the maximum number of detected features is set to infinite. This leads to a detection of feature candidates for almost every pixel, except in pure homogeneous areas, such as on a clear sky. Non-maximum suppression is kept activated and is applied with a radius of 1 px to bound the maximum required computing power. All constraints during intra- and inter-matching are relaxed in a similar way to ensure that every feature finds a match. This includes epipolar constraints, the search space from inertial constraints, and the thresholds for the matching metrics NCC and SAD. The result is a semi-dense set of tracked features, illustrated in Figure 8.2 (ii).

(iii) The next step is the computation of the error \mathbf{e} between the matched feature point $\hat{\mathbf{m}}^{l2}$ and the projection \mathbf{m}^{l2} of $\vec{\mathbf{M}}$ and the corresponding propagated covariance matrix $\Sigma_{\mathbf{e}}$. This error was formulated in Equation 5.21 (Section 5.2.3) with

$$\mathbf{e} = \hat{\mathbf{m}}^{l2} - \mathbf{m}^{l2}, \quad (8.1)$$

$$\Sigma_{\mathbf{e}} = \Sigma_{\hat{\mathbf{m}}^{l2}} + \Sigma_{\mathbf{m}^{l2}}. \quad (8.2)$$

The individual components of this computation are shown in Figure 8.2 (iii) and the methods of Griebach (2015) [OSLib] are used for uncertainty calculus. The required relative transformation $\Delta\mathbf{T}$ between the two stereo frames is computed in a parallel path based on VO in a standard configuration (*IPS-accurate*, Section 7.1.2).

(iv) A chi-squared test is applied to statistically assess whether the error \mathbf{e} fits to the propagated uncertainty $\Sigma_{\mathbf{e}}$. If the test fails, the feature might correspond to a homogeneous or repetitive object surface or to a moving object and should be

considered as *bad*. The applied chi-squared test is formulated based on the Mahalanobis distance with

$$\chi^2 = \mathbf{e}^T \Sigma_e^{-1} \mathbf{e}, \quad (8.3)$$

which follows a chi-squared distribution with two degrees of freedoms. The feature is valued as *good* if it is within the 99% interval and as *bad* if it is outside the 99.99% interval (empirically chosen), given with

$$\mathbf{m}^{l1} \text{ of } \{q, a^l\} \text{ is } \begin{cases} \text{good} & \text{if } \chi^2 < 9.2 \text{ and } |\mathbf{e}| < 2\text{px} \\ \text{bad} & \text{if } \chi^2 > 18.5 \\ \text{background} & \text{else} \end{cases}. \quad (8.4)$$

The class *background* is assigned to feature points if the chi-squared test is not clear. Further, the maximum error for the class *good* is bounded by a fixed threshold, such as 2px in this implementation. Pixels in the image that could not be matched are not valued and are assigned the class *background*.

(v) This procedure is done for multiple image pairs to account for the statistical nature of the chi-squared test. Specifically, reference frame q is evaluated based on the projection of the object point into the left and right image of four neighboring stereo frames (a, b, c, d) , denoted as $\{q, (a, b, c, d)\}$. A VO solution must exist for at least three stereo frame pairs to further consider q . This adds up to 8 valuations by the chi-squared test for each feature. The final classification is formulated as

$$\mathbf{m}^{l1} \text{ of } \{q, (a, b, c, d)\} \text{ is } \begin{cases} \text{bad} & \text{if } r_{bad} \geq 0.5 \text{ and } r_{valid} > 0.75 \\ \text{good} & \text{if } r_{good} \geq 0.75 \text{ and } r_{valid} > 0.75, \\ \text{background} & \text{else} \end{cases}, \quad (8.5)$$

where r_{valid} states the ratio of valid valuations for \mathbf{m}^{l1} in $\{q, (a, b, c, d)\}$, and r_{good} and r_{bad} the ratio of bad and good valuations in the same set. The result is a set of good and bad feature points, illustrated in Figure 8.2 (v). In future developments, this combination component could be replaced by a chi-squared test with 16 dimension, considering all 8 errors at once.

(vi) Image morphing (Gonzalez and Woods, 2018, p. 635) is applied in the next step to close the gaps between classified feature points. Therefore, two binary maps are generated for both good and bad features, which are visualized in Figure 8.2 (vi) in black. Considering the mask *good*, the sequence {dilation by 5 px, erosion by 3 px} is applied, which closes gaps and additionally marks areas as good that are close to good features. Considering the mask *bad*, the sequence {dilation by 5 px, erosion by 6 px, dilation by 2 px} is applied, which closes gaps and subsequently removes areas that result from isolated bad features. They are removed since the focus is on dynamic objects that are usually described by a dense set of *bad* features and not by isolated *bad* features points.

(vi, vii) the two masks are superimposed to build the final semantic segmentation map, exemplified in Figure 8.2 (vii). The label *good* is preferred over the label *bad* during class assignment of each pixel, because the information of good image areas is valuable and should not be lost. The class background is assigned to pixels that

are neither classified as *good* nor *bad*. The final result in Figure 8.2 (viii) shows that moving water and the moving shadow are marked as bad, while the stones are mostly classified as good image areas.

The current implementation of the complete automatic labeling procedure provides a throughput of one image per 19s on a [DellPrecision] and occupies up to 50% of the CPU capacity. This shows that processing all images of a dataset is not feasible.

8.1.2 Timestamp Selection

A preselection of the reference frames and corresponding stereo pairs is necessary to reduce the computational overhead and to select the most suited image frames for efficient training. This preselection needs to fulfill certain apparent criteria. First, the selected reference frames should not be duplicates of the same image content to provide a versatile training dataset. Second, the corresponding stereo pairs need to show a high image overlap with the reference frame to ensure wide image coverage of the reference segmentation. Third, each stereo pair must be apart in time to give the object time to move, but not too far to ensure that the object does not cover different parts of the static background.

Figure 8.3 summarizes the procedure to select a reference stereo frame q and corresponding frames (a, b, c, d) . Characteristics of the estimated trajectory are investigated to select the best set of stereo pairs $\{q, (a, b, c, d)\}$ for each timestamp based on several constrains and a score function. Then, the timestamps with the highest score in a local temporal neighborhood are selected to apply the automatic labeling procedure.

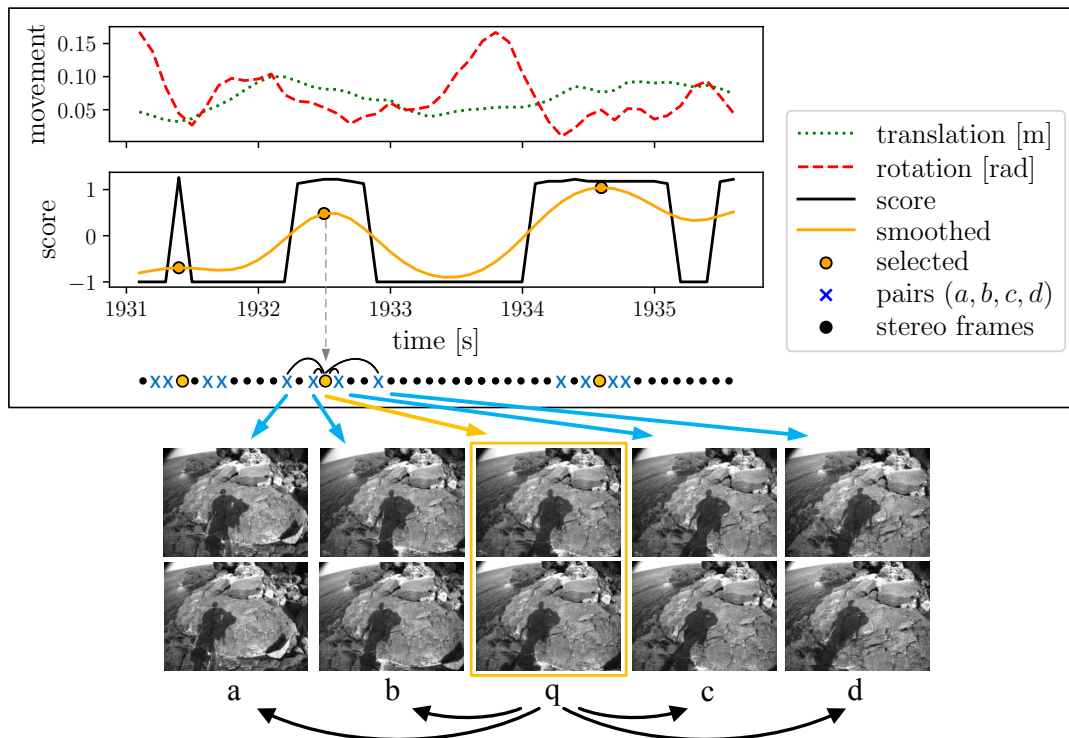


Figure 8.3: Illustration of the automatic timestamp selection procedure.

The identification of reference frames with corresponding set of pairs $\{q, (a, b, c, d)\}$ is based on the investigation of characteristics of an estimated trajectory. The trajectory was computed beforehand with *IPS-fast* (Section 7.1.2). Each stereo frame of the trajectory is considered as a reference frame candidate q_c with timestamp t_{q_c} and a list of all possible sets $\{\{q_c, (a, b, c, d)\}_i\}_{i=0}^M$ of some length M is generated. Each set must fulfill the constraints of Table 8.1 with respect to maximum differences in time, rotation, translation, and exposure time. The exposure time constraint is required since the applied matching-metric SAD is not invariant to image intensity changes. Further, the pairs a and d must be outside the time interval $[t_b, t_c]$ of the pairs b and c . Each set of $\{\{q_c, (a, b, c, d)\}_i\}_{i=0}^{M-1}$ is then assessed by the score s_{q_c} with

$$s_{q_c} = 0.5 \cdot \sqrt{t_{q_c} - t_a} + \sqrt{t_{q_c} - t_b} + \sqrt{t_c - t_{q_c}} + 0.5 \cdot \sqrt{t_d - t_{q_c}} \quad (8.6)$$

and the candidate with the highest score is selected as set for q_c . The score function is designed to favor sets whose individual pairs are distant to q_c . The factor of 0.5 is added for pairs a and d to prioritize distant inner pairs b and c . If no valid candidate set exists for q_c , s_{q_c} is set to -1.

The same score is used to identify the most promising sets $\{q, (a, b, c, d)\}$ of the trajectory. Exemplified in Figure 8.3, the score function is smoothed and its peaks identify the final reference frame q and corresponding pairs (a, b, c, d) . Additionally, a non-maximum suppression is applied with a radius of 0.2 m to ensure that individual reference frames are apart from each other in space, which helps to reduce the amount of duplicates. The remaining sets are then processed by the automatic pixel-level labeling procedure of Section 8.1.1.

Table 8.1: Constraints that must be fulfilled by each set of pairs.

constraint	inner pairs	outer pairs
max. change in	$\{q, b\}, \{q, c\}$	$\{q, a\}, \{q, d\}$
time	< 0.3 s	< 0.5 s
rotation	< 4°	< 12°
translation	< 0.1 m	< 0.3 m
exposure time	< 10 %	< 10 %

8.2 Training for Semantic Segmentation

The automatic labeling procedure is used in this section to generate reference datasets and to train and evaluate DNNs. In the following, the training procedure and the used DNN are introduced first. Then, the composition of the training, validation and test datasets are explained. Finally, a selection of trained DNNs are evaluated with focus on seen and unseen environments.

8.2.1 Technical Notes

The DNN Deeplabv3+ (Chen et al., 2018b) for semantic segmentation with a Mobilenetv2 (Sandler et al., 2018) backbone structure (Section 6.1) is used as basis in this theses. The network is selected in this work due to its lightweight architecture that was specifically designed for mobile applications. The network requires around 15ms to process one image with a resolution of 680×512 px on a RTX 6000 in the presented setup with three classes.

The training strategy *transfer learning* is applied. It describes the process of reusing trained model parameters from another task to initialize model parameters for the current task. Hidden units usually learn representations or features that are useful for multiple tasks. Therefore, I use pre-trained weights from [DeepLab] for Deeplabv3+ Mobilenetv2 that was trained on the public Ade20k dataset (Zhou et al., 2017). The pre-trained network is able to distinguish between 80 different objects and should provide generalized representations.

The overall training procedure mostly follows the suggestions by the authors of Chen et al. (2018a,b). They also provide an open source implementation [DeepLab] and scripts for training. A relative high ratio of the input image size to the output feature map size of 16 is chosen, which allows fast training at expense of reduced accuracy due to coarser feature maps (Chen et al., 2017). A crop size of 689×513 px is used that covers the targeted image size of half resolution. All considered networks in this thesis are trained on a RTX 8000 GPU with 48 GB memory. This allows to use a relatively large batch size of 24, which generally provides more accurate gradient estimates (Goodfellow et al., 2016, p. 272). Batch normalization is activated that normalizes hidden units and speeds up training (Ioffe and Szegedy, 2015). Stochastic gradient descent optimizer with momentum is used as optimizer.

The learning rate is set based on the “poly” learning rate policy that gradually decreases the base learning rate. I found empirically that the relatively high base learning rate of 0.1 worked best for the considered dataset. This value was determined by an initial grid-based hyper parameter optimization with 5 steps to find a good base learning rate, which was then used for all experiments. The number of learning steps is set to 50000 and the end learning rate is set to 0.001. Based on this configuration, the training of a single network takes about 16 h on a RTX 8000.

8.2.2 Datasets

A fundamental approach of Sensor-AI is to keep the neural network as close as possible to the sensor system. Therefore, only data that was recorded with the considered hand-held sensor system is used to generate the training-, validation- and test datasets.

The data that is used in this chapter was mostly recorded in the course of this thesis. Most of the data was recorded during the Vulcano Summer School 2019 (Unnithan et al., 2019) in harsh environments, such as a volcanic fumarole field, a coast environment, a martian analogue site, a volcanic mud field, and at Valle dei Mostri. An advantageous peculiarity of this data is that they were recorded for thermal inspections and therefore, the system was frequently hold still for a short moment to trigger a thermal camera image. This recording procedure is an excellent prerequisite for the developed timestamp selection method. The other part of the data was

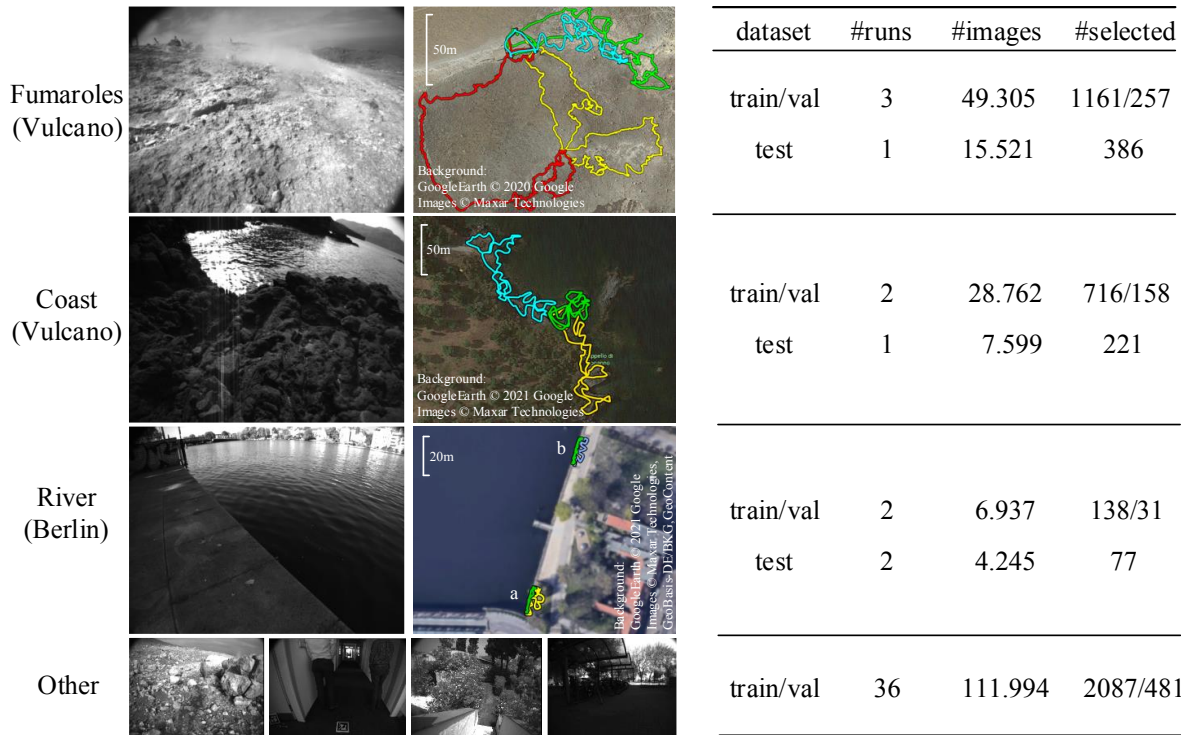


Figure 8.4: Visualization and statistics of training, validation and test datasets. Colorful lines illustrate trajectories of recorded runs. Green lines represent the test runs.

recorded in Berlin in urban environments, such as a river site and several indoor and outdoor environments.

Figure 8.4 provides data statistics and shows trajectories of the three targeted environments. Altogether 5630 stereo image frames were automatically selected out of nearly 200.000 stereo frames that correspond to 48 different trajectories. This dataset is splitted into training, validation and test datasets. The training and validation dataset are based on the same runs and were split randomly and image-wise. All test datasets are based on separate runs. For the fumarole and coast environments, the test run was recorded in the same environment but in a different place, different except for the start and end of each run. The training and test runs for the river dataset are from the same places, because comparatively little data is available for the river dataset.

The described training dataset was generated based on the calibration I . A similar dataset was generated for calibration F and shows to be consistent in terms of the amount of data with 4125 training and 934 validation images. The different calibration settings were introduced in Section 5.1 and are listed in Appendix A.3.

8.2.3 Evaluation

This section presents an evaluation of selected trained DNNs. The objective is to provide a broad impression about their capabilities to segment the targeted dynamic environmental elements. The interested reader might be referred to Appendix B.5 that provides additional results and shows exemplary predictions in other environments.

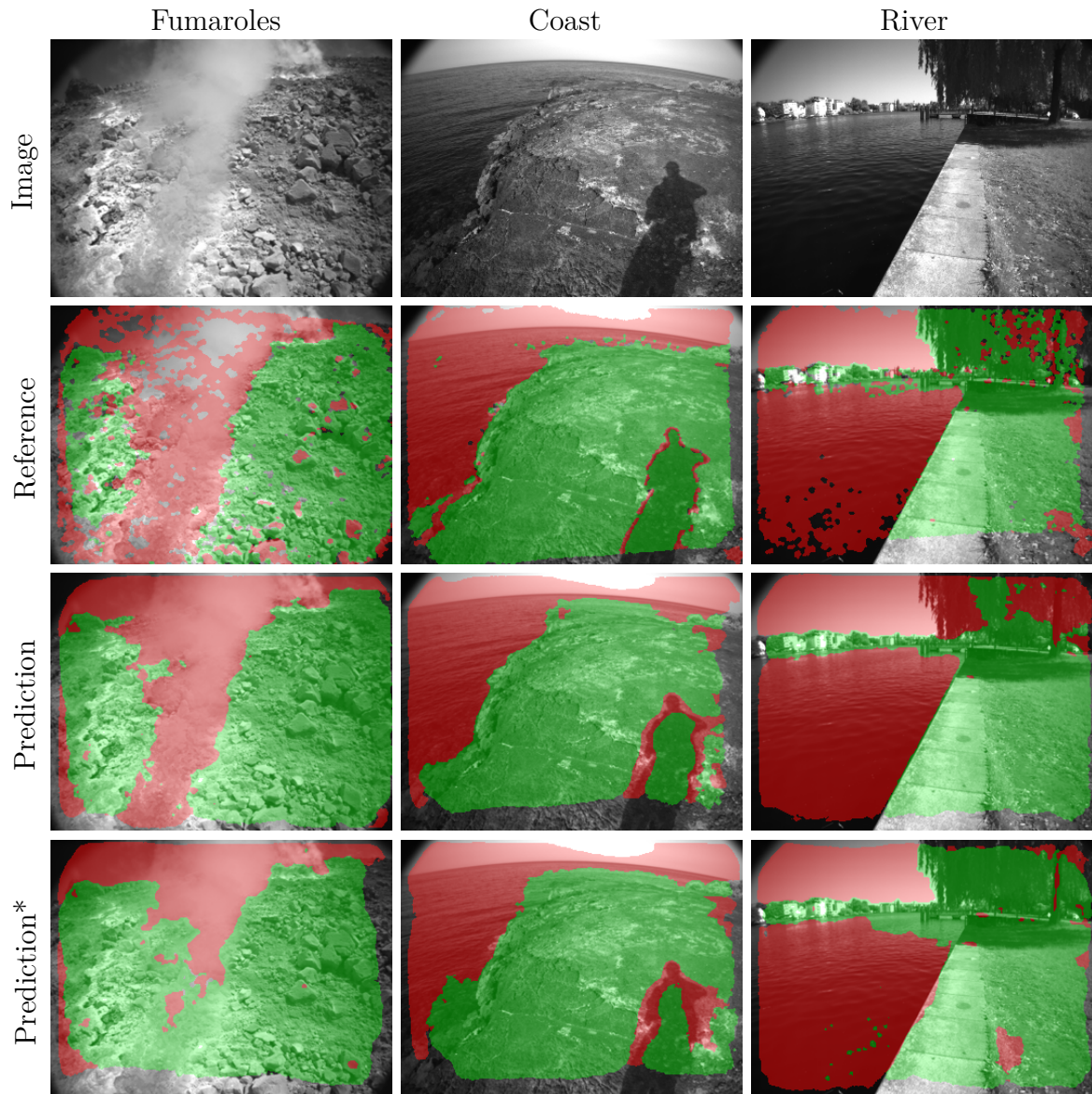


Figure 8.5: Visualization of semantic segmentations from the test dataset. (*) shows the result of three DNNs that have not seen the individual environment during training.

First, two networks *all-I* and *all-F* are considered that were trained on the full training datasets, which however were generated using the two different calibration settings *I* and *F*. Listed in Table 8.2, their mIOU score (Section 6.2) shows that the procedure equally works well for different calibration settings.

The mean scores are relatively low (e.g., 0.57 for *all-I*). This is most likely caused by a high proportion of pixel-level label noise in the reference data, such as visualized in Figure 8.5. While the reference shows many incomplete or disturbed segments, the predicted segmentation looks smooth and intuitive. On the one hand, this could suggest a good generalization capability of the network for the presented segmentation problem. On the other hand, this could be caused by relatively coarse feature maps of the last layers of the used DNN (Section 8.2.1).

Table 8.2: Evaluation of DNNs based on the mean IoU. Crossed out words indicate unseen environments. Particularities are marked in red.

network	fumaroles	coast	river ⁺	mean
all (<i>F</i>)	0.56	0.58	0.63	0.59
all (<i>I</i>)	0.55	0.57	0.59	0.57
smoke (<i>I</i>)	0.51	0.56	0.59	0.55
coast (<i>I</i>)	0.55	0.51	0.59	0.55
river (<i>I</i>)	0.55	0.57	0.54	0.55

(*) the place was seen during training in another run

Three more trained DNNs are evaluated in Table 8.2 that are trained on three reduced datasets. In each dataset, one of the environments *fumaroles*, *coast*, *river* is not included. This leads to a decrease of the mean IoU by 5 percent in each unseen environment (marked in red). Figure 8.5 shows each one prediction for the three environments by the individual network that has not seen this environment during training. Most dynamic object areas are segmented correctly, which suggests that similar clues can be learned from different environments. For instance, smoke, clouds, and homogeneous or blurry image areas might share similar clues. Or, the coast and river both consist of water. However, some specific and unseen dynamic object elements could not be segmented correctly, such as the flotsam on the river in the bottom right image that were marked as *good* by the DNN.

8.3 Combined Sensitivity Analysis for Coast and Fumarole Environments

The presented digital twin of Chapter 4 is deployed to analyze the effect of smoke and water on IPS based on synthetic clones that provide complete GT data. The simulation strategy *combined sensitivity analysis* (Section 4.3.3) is used that allows high variability in the observed environmental elements. The synthetic dataset and their specific dynamic parameters were introduced in Section 4.2.

Table 8.3: Parameters of the combined sensitivity analysis for *coast* and *fumaroles*.

name	type	sample distribution	allowed range
geometric calibration	P_C	$\boldsymbol{\kappa}, \boldsymbol{\delta}, \mathbf{T}_l^r, \mathbf{T}_l^n$ of calibration	F (Appendix A.3)
Gaussian blur [px]	P_P	$\sigma_{blur} \sim N(0, 1)$	$0 < \sigma_{blur} < 5$
capture gain (factor)	P_P	$C \sim N(0, 5)$	$0 < C < 20$
water scale (factor)	P_E	$W_{scale} \sim 1/N(7, 5)$	$1/1 < W_{scale} < 1/15$
water ripple (factor)	P_E	$W_{ripple} \sim N(1.0, 0.5)$	$0.1 < W_{ripple} < 1.9$
water flow (factor)	P_E	$W_{flow} \sim N(1.0, 0.5)$	$0.1 < W_{flow} < 1.9$
smoke density (factor)	P_E	$S_N \sim N(0.9, 0.6)$	$0.05 < S_N < 2$
smoke speed (factor)	P_E	$S_S \sim N(0.9, 0.6)$	$0.2 < S_S < 2$
texture blur [px]	P_E	$\sigma_{tex} \sim N(9, 1)$	$0 < \sigma_{tex} < 5$

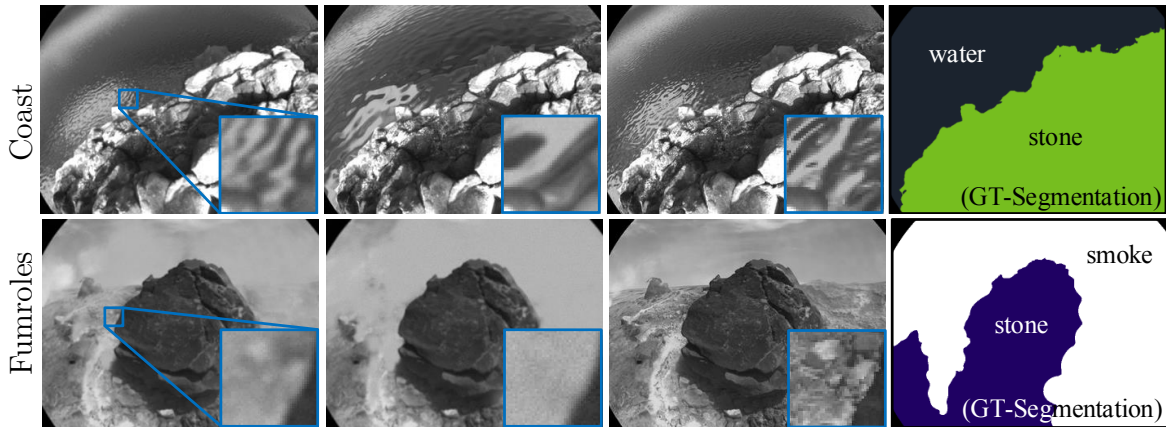


Figure 8.6: Images of selected samples and GT segmentation for the third column.

Listed in Table 8.3, both experiments consider calibration parameters and the camera properties blur and noise (via capture gain), which ranges were predetermined with a single parameter sensitivity analysis (Appendix B.1). Considered dynamic environment parameters are W_{flow} , W_{ripple} , and W_{scale} for the coast dataset and $S_{density}$ and S_{speed} for the fumarole dataset. These parameters and their sample ranges are selected in such way that the simulation mostly looks realistic for the human eye.

Both datasets show rocky environments that generally provide rich sets of image features. To reduce this richness, the fumarole dataset additionally considers the static environment parameter σ_{tex} . All object textures are initially subjected to Gaussian blur based on σ_{tex} , before the current sample is simulated.

8.3.1 Coast

The analysis for the *coast* dataset is based on 900 samples and investigates *IPS-fast-F*. Figure 8.6 (top) shows three samples with high diversity in the appearance of water.

The correlation analyzes in Figure 8.7 indicates a significant influence of water flow and scale on *IPS-normal*. Both parameters define the speed of features that are tracked on the water. These influences are eliminated for *IPS-masked* that masks out water based on the GT semantic segmentation. The new most influential parameter shows to be the error in u_0 , which directly affects the estimated scale of the trajectory.

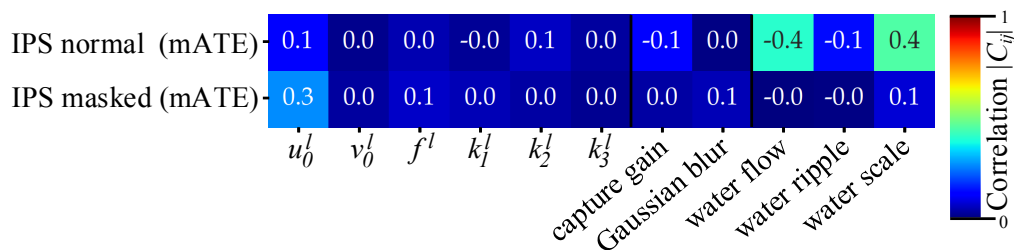


Figure 8.7: Correlation analysis for *coast* with 9 out of 29 varied parameters.

The sensitivity analyzes of Figure 8.8 visualize the data for three outlined parameters. Water flow shows a peak at value of 0.5, which can be interpreted as follows. Below this value, tracked features are very slow and might only introduce small errors.

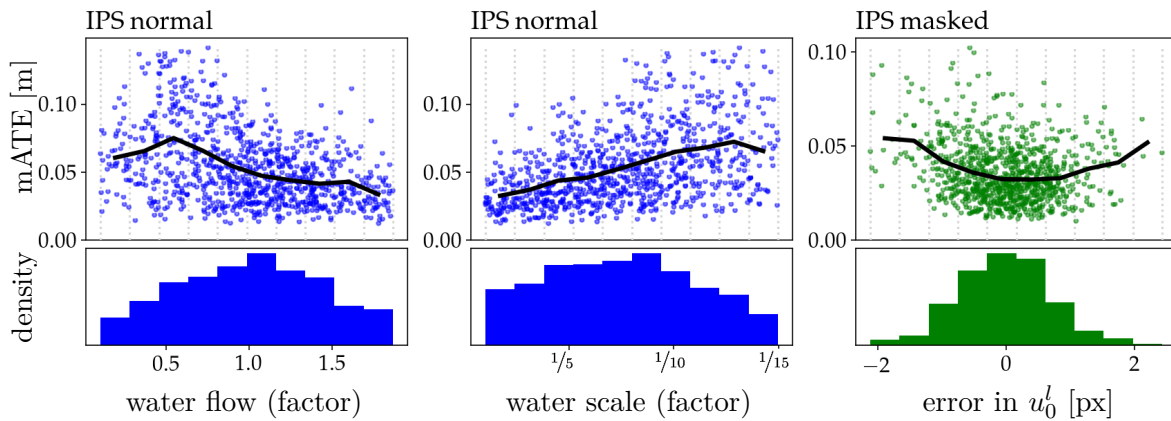


Figure 8.8: Sensitivity analysis for the *coast* dataset. Points show the mean ATE of one sample with respect to the sampled parameter (top), whose sample distribution is shown (bottom). Lines connect the mean of samples that are divided into 10 bins.

Above this value, tracked features are fast and are either possibly easier to filter out by RANSAC, change their appearance to fast, or cannot be tracked by the inertial constraint tracking method. A similar behavior can be observed for the water scale, where a smaller texture size (large: $1/5$, small: $1/10$) leads to smaller object movements and consequently leads to larger trajectory errors.

This experiment suggests that water can decrease the navigation result not only by its presence from covering large parts of the image, but also by its inherent dynamics that lead to non-static features. Though, explicit realism can not be guaranteed with the technology used and an investigation in real world is mandatory.

8.3.2 Fumaroles

The analysis of the *fumaroles* dataset is based on 450 samples and is conducted for *IPS-fast-F*. The exemplified synthetic GT segmentation of Figure 8.6 (bottom) shows a very high sensitivity of the GT smoke extraction method. This indicates that *IPS-masked* is not applicable based on the synthetic GT segmentation, because disproportionately many areas would be masked out. Instead, *IPS-normal* is investigated in more detail with evaluation of the total number of used features (after RANSAC filtering) for each sample described as feature count.

The correlation analyzes in Figure 8.9 only indicates a moderate influence of smoke density on the mean ATE. The correlation with feature count on the other hand indi-

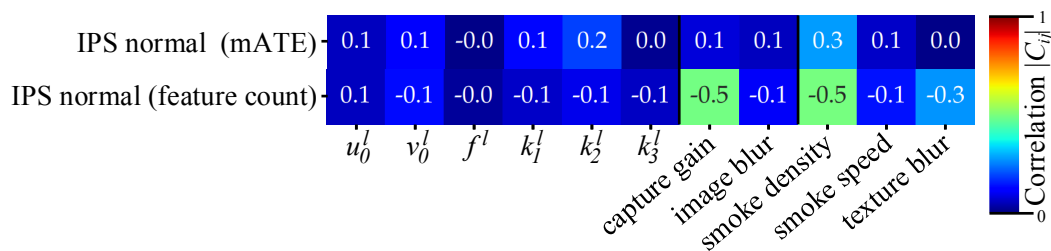


Figure 8.9: Correlation analysis for *fumaroles* with 9 out of 29 varied parameters.

cates a significant influence of smoke density and capture gain. Also a higher texture blur seems to gradually decrease the total number of used features.

Figure 8.10 (middle, right) shows that smoke density and capture gain decrease the feature count with similar severity. Though, only smoke density shows to moderately affect the mean ATE. This can be because capture gain deteriorates all image frames equally and the lack of features is distributed over the whole dataset. In contrast, dense smoke reduces the number of features drastically for a few frames where smoke is present. This can cause complete failures of VO for short periods, which leads to a strong drift of the inertial navigation solution and consequently in a high mean ATE.

This experiment suggests that smoke affects *IPS-fast-F* by its mere presence, which covers and blurs large parts of the image, and not by its motion. Again, explicit realism can not be guaranteed with the technology used and real world experiments are mandatory.

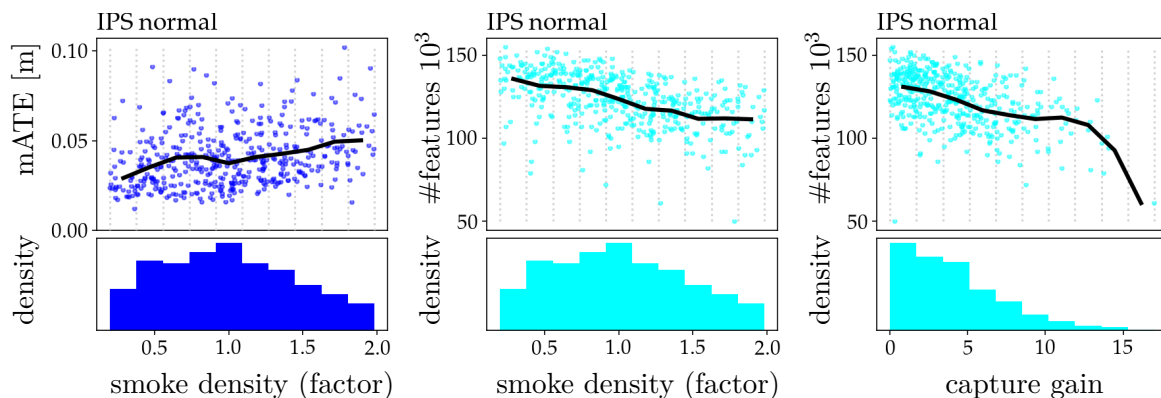


Figure 8.10: Sensitivity analysis for the *fumaroles* dataset. Points show the mean ATE of one sample with respect to the sampled parameter (top), whose sample distribution is shown (bottom). Lines connect the mean of samples that are divided into 10 bins.

8.4 Confirmation with Real World Data

The previous experiments have shown that water can deteriorate the navigation solution due to its inherent dynamics. Smoke has shown a negative impact by covering and blurring parts of the image. However, the realism of the experiments based on synthetic data are limited by the technology used. Therefore, real-world data is exploited in this section to validate made observations.

IPS is considered without (*normal*) and with (*masked*) the segmentation aid for the two calibration settings I and F (Section 5.1), and is based on *IPS-fast* that was introduced in Section 7.1.2. *IPS-masked* is applied with two networks that have seen all environments during training and only differ in the used calibration settings (I , F) that was also used to generate their training data. *IPS-masked** is applied with three different networks, which each have not seen one of the considered environments (*fumaroles*, *coast*, *smoke*) during training. The mask is generated based on the segmented class *bad* from the Sensor-AI approach and is applied in IPS similarly as presented in Section 7.1.1, but without subsequent image morphing.

Table 8.4: Evaluation of IPS in dynamic real world environments using the nCLE [%] based on closed loops (CL). Crossed out words indicate unseen environments. Bold numbers show apparent differences between corresponding *IPS-normal* and *-masked*. Calibration partly differs between data generation (train) and IPS application (apply). Calibration partly differs between data generation (train) and IPS application (apply).

	IPS: normal		masked		masked*			dataset-properties	
calib. (train):	I	I	F	F	I	I	I		
calib. (apply):	I	I	F	F	F	F	F		
training data:	-	all	-	all	smoke	coast	river	d[m]	CLs
fumaroles-d1 ⁺⁺	1.30	1.27	1.14	1.11	1.13	1.11	1.11	360	5
fumaroles-d2	1.48	1.43	0.90	0.87	0.90	0.87	0.87	549	3
coast-d1 ⁺⁺	0.71	0.71	0.58	0.57	0.57	0.57	0.57	656	6
coast-d2	0.55	0.55	0.47	0.49	0.47	0.48	0.48	268	3
river-a-d1 ⁺	0.78	0.61	0.46	0.24	0.27	0.26	0.40	44	1
river-a-d2 ⁺	1.57	0.50	0.99	0.09	0.06	0.31	0.96	43	1
river-b-d3 ⁺	0.79	0.80	0.68	0.77	0.74	0.78	0.71	41	1
river-b-s1 ⁺	0.17	0.15	0.25	0.26	0.27	0.29	0.27	36	1

(*) the place was seen during training in another run, (**) the run was used to generate the training dataset

Table 8.4 presents the results for a fair selection of datasets from the different environments. Dataset characteristics are listed in Table A.7 (Appendix A.2). The nCLE (Section 3.3.1) is used for evaluation since no other GT information are available.

Considering the environments *fumaroles* and *coast*, the results show no significant differences between *IPS-normal* and *-masked*. This indicates that the inherent dynamics of *smoke* and *water* do not deteriorate the localization solution for the considered datasets. The results for the fumarole dataset are generally rather poor, which might be attributed to the high presence of smoke in the images that severely reduces the amount of tracked features. In general, *IPS-normal* is capable of selecting mostly only features that are on the static background. Exemplified in Figure 8.11 for *fumaroles* and *coast*, the feature sets after RANSAC filtering are very similar for *IPS-normal* and *-masked*. However, the considered datasets were recorded for the purpose of inspection and the operator actively tried to reduce the amount of observed water and smoke.

The datasets of the river environment were recorded with a first responder application in mind (Section 1.2), in which the observed presence of dynamic objects cannot be actively avoided. Table 8.4 shows mixed results for the comparison of *IPS-normal* and *-masked* for the different datasets. Significant improvements of *IPS-masked* over *-normal* can be observed at river site *a*, for instance, by a factor of 10 on the dataset *river-a-d2* and using calibration *F*. The reason for this improvement is shown in Figure 8.11 (right). Slowly moving flotsam on calm water frequently leads to used features on the water surface in *IPS-normal*, which are correctly masked out in *IPS-masked*. Flotsam is not present at river site *b*. Dataset *river-b-d3* shows a slight deterioration for *IPS-masked-F*, which however could not be clearly assigned to the mask and rather shows the limitations of the used metric nCLE. Dataset *river-b-s1* presents a reference dataset that was recorded near the river sites without the presence of water in the images and shows no significant difference for the different methods.

*IPS-masked** describes the application of the mask-approach in environments that have not been seen during training by the DNN. Table 8.4 shows that the localization based on *masked** does not worsen against *normal*, except again for dataset *river-b-s1*. Though, strong improvement of *masked* of river site *a* cannot be achieved, if the river environment was not seen during training. Figure 8.11 (bottom, right) shows that the water is only partly segmented and especially the flotsam could not be masked out.

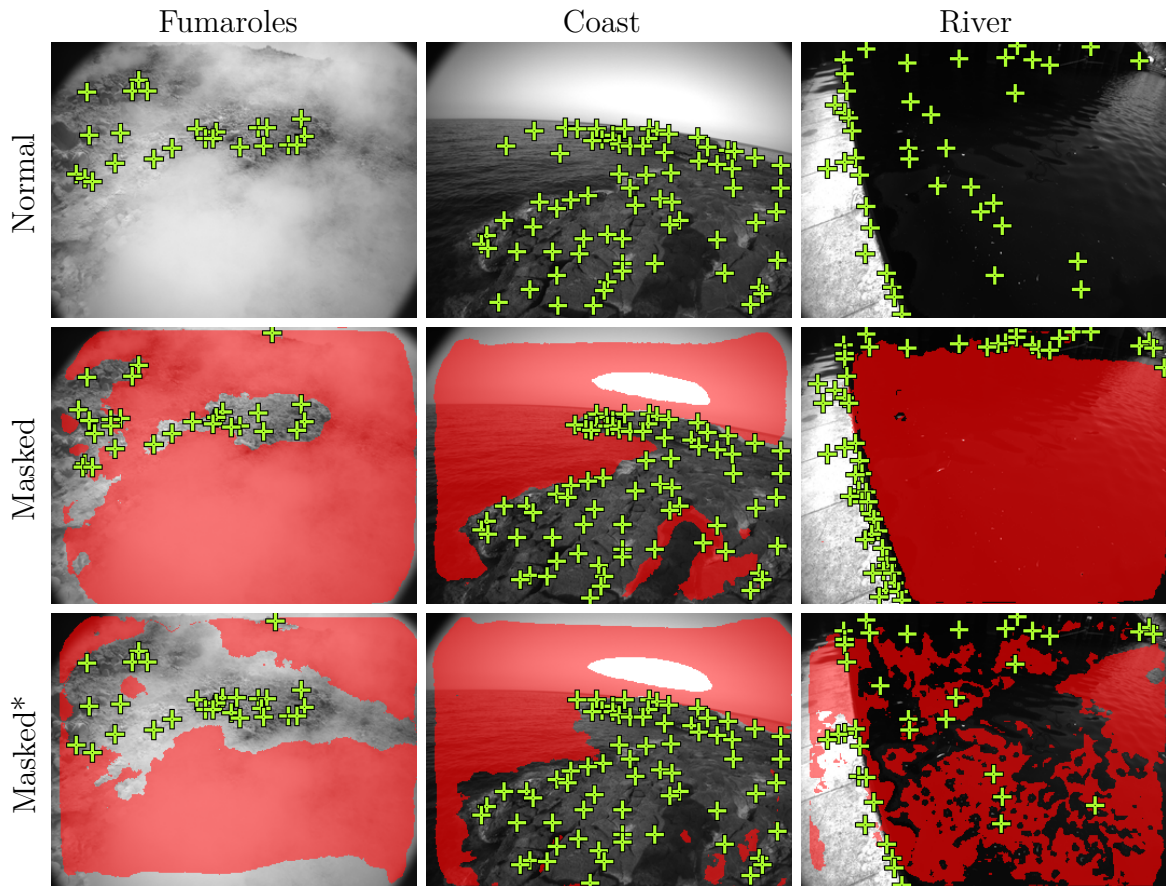


Figure 8.11: Illustration of used features in green and the generated mask in red. The three DNNs of *masked** have not seen the considered environment during training.

8.5 Summary

In this chapter, a method was presented that learns to segment image areas that are critical for a specific VO system and use this knowledge to improve the same VO method. This method enables to analyze the influence of smoke and water on visual odometry using real-world data. In line with the general approach of this thesis, the analyzes was first conducted in simulation using GT segmentation and then in real world using the Sensor-AI approach.

The experiments have shown that water and smoke can have a strong influence on the localization result. The influence of water is caused by its inherent dynamics and could be confirmed in simulation and real world. In real world, however, the

localization was mainly influenced by flotsam, which still can be attributed to water dynamics. The influence of smoke is caused by the presence of smoke itself that covers and blurs large parts of the image and leads to relatively small feature sets. This effect could be shown in detail in the simulation.

Using the Sensor-AI approach, the dynamic elements smoke and water could be segmented successfully in real-world datasets. It further lead to a strong improvement of the localization solution in a river environment in the presence of flotsam. In other environments, it has neither shown an improvement nor a deterioration. If this approach is applied in a new environment, which was not seen during training of the DNN, the results still neither improved nor deteriorated in comparison to the base method. This might be because the seen and unseen environments often share a similar static background, such as a rocky ground, which might have helped to reliably segment good image areas in unseen environments.

In future, the semantic predictions themselves can further be used to provide visual feedback to the user about which areas are well suited for localization and might guide his way of proceeding to get an optimal localization solution. This approach is not only restricted to dynamic elements, but should be applicable to all visual observable factors that might disturb the localization, such as image areas with motion blur, defocus blur or strong image noise.

Chapter 9

Discussion

This chapter discusses the results of the previous chapters with regard to the research questions, which will be addressed directly in Section 10.1, and puts them into context of the application for first responders. The first section focuses on the influence of dynamic objects in the defined first responder scenarios. The following two sections discuss the more technical aspects of the proposed Sensor-AI module and the digital twin. Finally, a short discussion follows about the application of IPS for first responder applications.

9.1 The Influence of Dynamic Objects

Three first responder scenarios were introduced in Chapter 1. Related datasets were abstracted from inspection data and were recreated with synthetic video clones within this thesis. The scenarios are (i) *indoor rescue*, (ii) *flood disaster* and (iii) *wildfire*.

(i) Indoor Rescue

The rescue scenario was considered separately in Chapter 7 based on combined real-world and synthetic *corridor* datasets and a large-scale *mall* dataset.

Starting with the closest replica of a first responder scenario of this thesis, the *mall* dataset (Section 7.3) has shown the great potential of visual localization, but also its limitations. The results have shown a relatively accurate localization when navigating through large crowds of pedestrians over three floors. However, one extreme case was observed that resulted in a floor-level deviation to GT. This is a safety critical failure for first responder applications that require at least reliable floor-level accuracy (Rantakokko et al., 2010). This error was caused by a person standing close in front of the camera while going down an escalator. This resulted in the occurrence of consensus inversion (Bojko et al., 2021), i.e., the observation of more features on moving than on static objects, which repeatably resulted in a false ego-motion estimation. By using the introduced DNN-based segmentation aid, all features on the person could be excluded from ego-motion estimation, which resulted in a correct and robust localization. This scene highlights the importance of semantic understanding for visual localization, which also has been noted in related research (Bescos et al., 2018; Kaneko et al., 2018).

Other solutions to overcome this limitation of visual localization in such a scenario exist and are conceivable for first responder operations. For instance, signals from pre-installed or pre-deployed infrastructure could be considered in terms of data fusion, such as in Fischer et al. (2008) with UWB beacons. Furthermore, a UWB-based, collaborative approach based on relative distance measurements between first responder (Rantakokko et al., 2011) might easily identify such rare outliers and correct the localization solution.

The experiments in the *corridor* environment (Section 7.2) allowed a more detailed analysis of the influence of the object *person*, due to a more controllable environment and generated synthetic video clones. The main challenges that could be replicated were slow object motions and dependent motion (Saputra et al., 2018), i.e., two people moving together. The observed main problems were a low number of static features due to the homogeneous corridor environment and presence of moving objects, a difficult clear separation of static and dynamic features, and consensus inversion. The segmentation aid reduced the impact of dynamic objects to their mere presence in the image. Though, due to the basic mask approach, features were also ignored on non-moving persons, which resulted in reduced accuracy and higher uncertainty in scenes with many non-moving people (Section 7.2.1). A similar scenario was, for instance, described in Bescos et al. (2018) for vehicle applications with lots of parking cars. This problem was targeted by Schorghuber et al. (2019) by using a continuously estimated confidence factor for classification of features in static, static-dynamic and dynamic.

Further experiments in the corridor environment have shown the potential of advanced geometric approaches (Section 7.2.1). A higher image resolution with stricter feature matching improved the results in dynamic environments, indicating that robustness can be bought with computational power. A lower image frame rate allowed to better distinguish between static and dynamic features since object movements are more noticeable. This indicates that feature point tracking and multi-view geometry can significantly improve the results, which was, for instance, shown by Migliore et al. (2009). Further, it highlights the need of keyframe-based techniques, which is the “gold standard” in V-SLAM approaches (Campos et al., 2021). Besides that, smaller system uncertainties helped to identify wrong vision-based ego-motion estimations based on a chi-squared test in the filter. This underlines the need for accurate geometric calibration, whose results are stable during operation even under harsh conditions with potentially high physical and temperature stress. This requires special sensor designs for first responder applications. In summary, the results indicate that geometric approaches reduce the likelihood of dynamic objects to corrupt localization, but cannot eliminate their influence completely due to a lack of semantic understanding.

(ii) Flood Disaster

The influence of the dynamic element *water* was considered based on real-world *coast* and *river* datasets (Section 8.4) and in simulation (Section 8.3.1). In direct comparison to the element *person*, the influence of its dynamics was mostly not measurable in real-world and was also less significant in simulation. This might be attributed to the high non-rigidity of water and rapid change of its appearance. Though, flotsam lead to a visible distraction and measurable deterioration at one river sight, which can be

accounted to a dependent motion of individual floating particles and a high fill factor of water in the image. Again, the segmentation aid could prevent this distraction.

Further improvements are conceivable from a geometric point of view. For instance, cameras with a larger field-of-view, such as fisheye cameras, could help to capture more of the surrounding static background. If the water then still covers large parts of the image, it would mean that the first responder is in a large open area and sensor fusion with a GNSS-receiver would be the ultimate choice.

(iii) Wildfire

The *wildfire* scenario was considered based on *fumarole* datasets, in which the dynamic element *smoke* was represented by vapor. The localization accuracy on the real-world datasets were generally rather moderate (Section 8.4). Based on the real-world experiments with and without using a mask for feature selection, the results indicate that the inaccuracy did not result from smoke movements. This implies that the main influence of smoke is by its mere presence, which covers and blurs large images areas. This observation is supported by the simulated experiments (Section 8.3.2), which showed a steady descent of the number of good features with increasing smoke density. The consideration of vapor as a representative for smoke limits the range of these findings for the scenario wildfire, since vapor generally shows less striking structures in its appearance than dense particle-rich smoke.

These experiments show the limited applicability of cameras, which are sensitive in the visual light spectrum, in environments with dense smoke. Though, as motivated in the introduction, the considered application of ground-based fire fighters foresees the use of thermal cameras for ember detection and mapping. These infrared cameras are less affected by smoke (Starr and Lattimer, 2014) and could be incorporated into the localization procedure (Brunner et al., 2013).

The real-world experiments were further severely subjected to other strong influences. This includes strong physical stress, which was caused by unusually high system temperatures and ruthless use of the IPS. This might have led to a decreased calibration accuracy, as discussed in detail in Section 5.1.2, which advocates the integration of online calibration functionality to the used sensor system. Furthermore, the fumarole outdoor inspection dataset shows frequent changes of pixel intensities due to automatic camera exposure time adjustments, caused by irregular partial viewing of the sky during the inspection run. Feature matching based on SAD (Section 3.2.1), which is often used in IPS due to an existing subpixel matching approach and implemented error propagation from noise, is therewith not a good choice since it is not invariant to intensity changes. As a consequence, a more sophisticated feature matching approach will easily improve the results. Obvious candidates are NCC or descriptor-based feature matchers such as ORB, which, for instance, is the basis of the state-of-the-art feature-based SLAM approach ORBSLAM3 (Campos et al., 2021).

9.2 The Deep Learning Module

The proposed Sensor-AI approach (Chapter 8) learns critical image areas from VO offline and uses this knowledge to improve the same VO method online. As discussed,

this approach helped to investigate the influence of smoke and water and could significantly improve IPS at a river site. Generated reference data has shown a relatively high amount of label noise, which might be accountable to the used relatively weak feature tracker based on small patches and SAD. Though, the predicted results of the DNN looked smooth (Section 8.2.3), which can be attributed to a good generalization capability or to small internal feature resolution of the used DNN. Furthermore, the approach has shown first signs of good generalization to unseen environments (Section 8.2.3). Though, it could not identify water with flotsam, if it has not seen this element before (Section 8.2.3, 8.4). Therefore, this approach can only be seen as a first attempt and is currently limited to seen environments, which can possibly be accounted to the high variability in appearance of smoke and water in different environments. Further, the DNN is only applied in offline processing, since it is not yet integrated into the sensor system.

A pre-trained DNN was applied for *person* detection in indoor environments (Chapter 7). The Sensor-AI approach was not required for this element, since the pre-trained DNN mostly segmented persons accurately. The application of the presented Sensor-AI approach for learning to detect persons might be limited. Lots of data would be required from the same sensor system with a diverse set of moving persons, which would raise privacy issues. Furthermore, a person can be both static and dynamic, in different scenes. This would lead to high data uncertainty in the training data and a DNN that only works on appearance-based clues might not be suited for this task. In this case, a more suited approach would be to simultaneously consider motion and appearance clues for motion detection, such as proposed by Siam et al. (2018). The Sensor-AI approach is principally not limited to semantic segmentation and can be extended.

9.3 The Digital Twin

The developed digital twin accompanied all experiments with synthetic video clones, which were used for in-depth analysis based on three simulation strategies. The use of semantic object segmentation from GT data allowed the analyses of object influences in simulation without possibly inaccurate segmentation from DNNs.

Datasets were generated that closely resemble the real-world datasets (Section 4.2). The corridor dataset was created manually with a measurement tape. This is a time-consuming procedure and only works for structured human-made objects. In contrast, the unstructured coast and fumarole environments were recreated using a professional photogrammetry tool. This is an efficient way to generate 3D worlds for a specific trajectory, but it is not applicable in environments with homogeneous object surfaces. Generally, if the requirement for a real-world-replicating video clone can be relaxed, then other options exist. For instance, the motion profile can be transferred into any synthetic environment to simulate image sensor data, such as done by Sayre-McCord et al. (2018) in real-time. Furthermore, this allows to exploit the large synthetic worlds and capabilities of game engines (e.g., Shah et al., 2017, Dosovitskiy et al., 2017).

Different simulation strategies were used to enable special in-depth analysis (Section 4.3). First, the standard *sensitivity analysis* allowed to analyze the influence of single parameters that can not be analyzed in real-world, such as motion blur. However, this procedure is laborious to conduct for many parameters (e.g., Section 7.2.1). Second,

the *geometric MCS* allowed to analyze the influence of calibration errors. It has shown the potential of the current error propagation concept of IPS, but also its limitations, as discussed in Chapter 5. Third, the *combined sensitivity analyses* allowed to investigate the influence of multiple parameters at once and to weight their influence on the trajectory error based on a correlation coefficient. Considered parameters originated from the environment, system design, sensor property and calibration errors. The experiments showed that dynamic environmental parameters are more influential than other considered parameters (Section 7.2.2, 8.3). This approach further includes the capabilities of the geometric MCS, which was deployed to proof that dynamic elements are not described by the propagated uncertainty in VO (Section 7.2.2). One restriction of this method is the required computational power and disk space, which severely limits the length of the dataset and the number of MCS samples. One contributing factor is the required initialization phase (Section 3.2.3) that, for instance, takes 45s in the simulated corridor dataset, which is one half of the simulation time. The integration of state-of-the-art approaches such as of Campos et al. (2020) might reduce this overhead.

An important point to consider is the gap between simulation and real world. All simulated dynamic elements in this thesis (Section 4.2) were designed to “look” realistic, but they are severely limited in their realism at a closer look. Persons are only based on one model, which is further limited to straight walking. Water is simulated based on texture variation on one plane. Smoke is based on partly transparent, one-intensity particles, which removes all structure from dense smoke. Other limitations that increase the gap are, for instance, missing motion blur (which is only applied for the standard sensitivity analysis) and changes in pixel intensity over time (Vaudrey et al., 2008).

Despite this gap, the gained insights matched mostly well between both domains. First, a sensitivity analysis was conducted for the corridor experiments in simulation and real world (Section 7.2.1). The observed localization results from both domains matched well, even though the real-world experiments were severely limited in GT data. Second, the water experiments did show a recognizable gap (Section 8.3.1, 8.4). The flowing water has shown an erroneous influence on IPS in simulation, which could only be shown in real-world in the presence of flotsam. Third, the results from the smoke experiments in both domains agreed that the influence of smoke is caused only by its presence itself (Section 8.3.2, 8.4).

9.4 IPS for Self-Localization of First Responders

IPS is a well-suited platform for the development of (visual) localization methods for diverse applications, such as inspection, vehicle navigation, geological mapping and also first responders. The advantage of IPS is that it is a complete sensor system, consisting of different hardware prototypes, the localization software and a digital twin. This allows a high degree of flexibility and favors a wide range of research applications, as all components can be studied, replaced and extended.

The existing IPS prototypes, which are designed primarily for research purposes, might currently not be operational enough for real emergency response operations, but they could be specifically engineered for this application. This limitation for IPS arises, for instance, due to possible adverse conditions such as dense smoke or its susceptibility to dynamic objects in extreme scenarios without the segmentation aid.

The latter is currently not integrated into the sensor system and is only beneficial for known objects or in known environments, but first responders are often confronted with unpredictable situations. Further, the sensors of the current IPS are currently relatively expensive, exceeding 1000\$ (Rantakokko et al., 2010), and it is relatively bulky due to the additional required computer. Though, IPS and its technologies have the necessary potential and further improvements are conceivable, for example, by integrating a keyframe- or a SLAM-approach, sensor fusion (e.g., LiDAR, thermal or event camera) or data fusion with external signals (e.g., UWB-beacons), besides the existing GPS-option (Baumbach et al., 2018; Zuev et al., 2019).

Finally, it is important to highlight that all results of this thesis are limited by the range of the conducted experiments. Most importantly, only one hardware system was considered with mostly only one methodical base configuration for localization. A higher number of features or a better feature distribution over the image might also reduce the likelihood of dynamic objects to distract IPS. For instance, Zhang (2018, p.50) considered 24 different configurations at once for each considered approach in his experiments. Such a variation of methodical parameters could easily be integrated into the combined sensitivity analysis. Besides, all real-world experiments were severely limited in GT and simulation experiments were restricted to relatively small datasets.

Chapter 10

Conclusion

This chapter concludes with a concise summary of the main findings and brief outlook to coming developments on the basis of this work and for IPS. The following summary starts with a brief wrap up of the focus of this thesis and conducted experiments, before the formulated research questions are answered. It continues with the main limitations of this work and ends with a concise conclusion for this thesis.

10.1 Summary

This thesis investigated the application of visual localization in dynamic, adverse environments to identify challenges and accordingly increase the robustness of the visual localization system. The main motivation was the application for self-localization by first responders, where challenges arise such as visual distractions from dynamic objects and sensor degradation due to adverse conditions and high physical stress on the system. The investigations were based on the sensor system IPS, a stereo-VINS, which was extended by a digital twin in this work. Experiments were based on recorded datasets from *corridor*, *mall*, *coast*, *river* and *fumarole* environments and targeted the analysis of the dynamic elements *person*, *water* and *smoke*. Simulated datasets were used to support these experiments using synthetic video clones. In simulation, sensor degradations were further considered in terms of geometric calibration errors and deteriorated camera properties to account for possible adverse conditions. Altogether, four research questions were formulated, which are considered in the following paragraphs.

The first research question addresses the influence of dynamic objects on visual localization. Based on the experiments, three different influences could be observed.

- First, moving objects can obscure the view and prevent the detection of static features, which slightly reduces localization accuracy. This was mostly observed for the element *smoke* that partly covered and blurred the full image.
- Second, features on moving and static objects might be not separable due to slow object motions, which can result in a moderate drift. This was observed for the element *water* in the presence of flotsam and for slow moving *persons*.
- Third, a set of moving features might be mistaken for the static background, which can lead to severe drifts. The cause is consensus inversion, which was observed for the element *person*. It is favored by homogeneous surfaces in indoor environments, persons walking close in front of the camera and dependent motion.

It was also shown that the influence of dynamic objects depends on the geometric properties of the localization method. For instance, the probability of an erroneous influence by a moving object decreases with lower image frequency, higher image resolution with stricter feature matching, or higher system uncertainties.

The second research question concerns the application of a DNN to identify critical image areas that belong to dynamic objects. The detection of the object *person* was realized simply with a fully trained DNN for semantic segmentation, which was provided by the DL community. For the detection of *water* and *smoke*, a Sensor-AI approach was developed that generates class-agnostic training data for semantic segmentation by directly exploiting the VO module. The segmentation results were promising for environments seen during training. In summary, all considered dynamic elements could be segmented and a successful application for other distracting elements can be deduced.

The third research question addresses the applicability of the DNN to improve visual localization. The basic mask approach was used to prevent the detection of features on potentially moving objects by masking specific object classes. Significant improvements were observed in the presence of moving persons and water with flotsam. However, small decreases in localization accuracy were observed in cases where the basic mask approach prevented the detection of features on non-moving persons. The mask approach neither lead to an improvement nor a deterioration for smoke or water without flotsam in the considered real-world experiments. In summary, the used segmentation aid mainly contributes in terms of robustness as it is able to prevent rare but significant localization failures.

The fourth research question aims to analyze various error sources in order to assess their different degrees of influence and to set the focus of future developments. This question is interconnected with the investigation of the previous questions and points to the used simulation strategy *combined sensitivity analysis*. This strategy allows to jointly consider environment, system design, sensor property and calibration error parameters in one Monte-Carlo-based sensitivity analyses. It was shown that the dynamic elements have the strongest influence in the selected scenarios. If their influence is eliminated, then calibration errors have shown a relatively big influence. This indicates that high effort should be put into the geometrical sensor calibration.

The results are subjected to several limitations. For instance, only one visual localization system was considered that comes with its own strengths and weaknesses. Further, only one methodical base configuration of IPS was considered in most experiments. In addition, available GT information was severely limited in real datasets and synthetic datasets were limited in realism due to limited rendering capabilities.

To conclude, visual localization shows great potential for the use in challenging environments and demanding applications, such as self-localization by first responders. This thesis specifically pointed out the high potential of hybrid approaches, combining model- and learning-based methods, on the example of using semantic segmentation for feature selection. Though, this approach understandably could not compensate for the lack of visible static backgrounds and further improvements are required to constantly guarantee a reliable and accurate localization solution. Future developments will consider improvements from versatile directions, including geometric approaches, learning-based methods and novel sensor technologies, to improve the visual localization system IPS in challenging environments.

10.2 Outlook

During this research, I came across promising ideas, approaches and technologies that I believe will shape future visual localization systems. In the following, I will resume some of those with specific focus on IPS technologies and applications.

Geometric approaches show a high level of development in related literature. IPS will benefit from an integration of long-term feature tracking, keyframe and SLAM approaches. Though, a specific focus should be the improvement of the propagation of uncertainties from feature matching and calibration parameters to fully exploit the potential of the methods developed by Griebßbach (2015).

Hybrid approaches that expand geometric methods with learning-capabilities will be key for robust visual localization in dynamic environments. The presented Sensor-AI approach, currently restricted to appearance clues (semantic segmentation), will benefit from an extension to use motion clues (optical flow) and depth clues. Furthermore, it would be interesting to explore machine learning techniques to estimate feature matching uncertainties or even for the error propagation itself.

Novel camera sensor technologies show great potential for localization in adverse environments. For instance, SWIR or thermal cameras could be explored in IPS for reliable localization in dense smoke areas. The event camera will be considered in applications that show fast camera movements or require high dynamic range cameras. I further recommend to explore the solid-state LiDAR for localization in indoor or subsurface environments to cope for homogeneous walls or low-light conditions.

Realistic simulation is important to prepare visual localization methods for use in challenging and safety-critical scenarios. The digital twin will benefit from an extension to state-of-the-art graphic tools, including game engines with their large-scale virtual worlds or animation tools with realistic rendering based on extensive raytracing. The strategy *combined sensitivity analysis* could hold promise for future use in other applications, such as in on-orbit-servicing (e.g., Benninghoff et al., 2018) to assess the influence of strong light reflections from satellite surfaces on visual localization.

Finally, visual localization is and will be exploited in most different and most challenging environments, such as motivated in Figure 10.1 in the context of geological and planetary exploration. In this sense, “Quick and accurate planetary exploration requires tools which are able to survive extremes in environmental conditions, working both on the surface and sub-surface. Sub-surface exploration is critical for example on Mars or the Moon, since signs of extant or extinct life is likely to be found in sub-surface caves (Carrier et al., 2020)” (Irmisch et al., 2021).



Figure 10.1: Exploration of a fumarole field with IPS, Vulcano Summer School 2019.

References

- Abadi, Martín et al. (2016). “TensorFlow: A System for Large-Scale Machine Learning”. In: *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*. Savannah, GA: USENIX Association, pp. 265–283. ISBN: 978-1-931971-33-1.
- Akenine-Möller, Tomas, Eric Haines, and Naty Hoffman (2008). *Real-time rendering*. 3. ed. Wellesley, Mass.: Peters. ISBN: 978-1-56881-424-7.
- Alcantarilla, Pablo F. et al. (2012). “On combining visual SLAM and dense scene flow to increase the robustness of localization and mapping in dynamic environments”. In: *2012 (ICRA) IEEE International Conference on Robotics and Automation*. IEEE, pp. 1290–1297. ISBN: 978-1-4673-1405-3. DOI: 10.1109/ICRA.2012.6224690.
- Alkendi, Yusra, Lakmal Seneviratne, and Yahya Zweiri (2021). “State of the Art in Vision-Based Localization Techniques for Autonomous Navigation Systems”. In: *IEEE Access* 9, pp. 76847–76874. DOI: 10.1109/ACCESS.2021.3082778.
- Anderson, Michael L., Kevin M. Brink, and Andrew R. Willis (2019). “Real-Time Visual Odometry Covariance Estimation for Unmanned Air Vehicle Navigation”. In: *Journal of Guidance, Control, and Dynamics* 42.6, pp. 1272–1288. ISSN: 0731-5090. DOI: 10.2514/1.G004000.
- Aqel, Mohammad O. A. et al. (2016). “Review of visual odometry: types, approaches, challenges, and applications”. In: *SpringerPlus* 5.1, p. 1897. ISSN: 2193-1801. DOI: 10.1186/s40064-016-3573-7.
- Arras, Kai O. (1998). *An Introduction To Error Propagation: Derivation, Meaning and Examples of Equation $Cy = Fx Cx FxT$* . DOI: 10.3929/ETHZ-A-010113668.
- Auf dem Kampe, Jörn (2021). *Mount St. Helens: Vorstoß ins Innere eines Unruheherds: Vulkan-Expedition*. Ed. by Geo. URL: <https://www.geo.de/p/plus/natur-und-nachhaltigkeit/vulkan-mount-st--helens--vorstoss-ins-innere-eines-unruheherds-31439832.html>.
- Barnes, Dan et al. (2018). “Driven to Distraction: Self-Supervised Distractor Learning for Robust Monocular Visual Odometry in Urban Environments”. In: *(ICRA) IEEE International Conference on Robotics and Automation*. IEEE. ISBN: 978-1-4503-6584-0.
- Baumbach, Dirk et al. (2018). “GPS and IMU Require Visual Odometry for Elevation Accuracy”. In: *2018 (AVSS) International Conference on Advanced Video and Signal Based Surveillance*. IEEE, pp. 1–6. ISBN: 978-1-5386-9294-3. DOI: 10.1109/AVSS.2018.8639138.
- Becker, Frank-Micheal et al. (2003). *Formelsammlung: Formeln, Tabellen, Daten ; Mathematik, Physik, Astronomie, Chemie, Biologie, Informatik*. 1. Aufl. Berlin: Paetec. ISBN: 3-89818-700-4.

- Benninghoff, Heike et al. (2018). “RICADOS - Rendezvous, Inspection, Capturing and Detumbling by Orbital Servicing”. In: *7th International Conference on Astrodynamic Tools and Techniques*.
- Bescos, Berta et al. (2018). “DynaSLAM: Tracking, Mapping and Inpainting in Dynamic Scenes”. In: *IEEE Robotics and Automation Letters* 3, pp. 4076–4083. ISSN: 2377-3766. DOI: 10.1109/LRA.2018.2860039.
- Bescos, Berta et al. (2021). “DynaSLAM II: Tightly-Coupled Multi-Object Tracking and SLAM”. In: *IEEE Robotics and Automation Letters* 6.3, pp. 5191–5198. ISSN: 2377-3766. DOI: 10.1109/LRA.2021.3068640.
- Bojko, Adrian et al. (2021). “Learning to Segment Dynamic Objects using SLAM Outliers”. In: *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, pp. 9780–9787. ISBN: 978-1-7281-8808-9. DOI: 10.1109/ICPR48806.2021.9412341.
- Börner, Anko et al. (2017). “IPS – a vision aided navigation system”. In: *Advanced Optical Technologies* 6.2, pp. 121–130. ISSN: 2192-8576. DOI: 10.1515/aot-2016-0067.
- Börner, Anko et al. (2020). *Sensor Artificial Intelligence and its Application to Space Systems – A White Paper*. DOI: 10.14279/DEPOSITONCE-10185.
- Brown, Duane C. (1971). “Close-range camera calibration”. In: *Photogrammetric Engineering* 37.8, pp. 855–866.
- Brunner, Christopher et al. (2013). “Selective Combination of Visual and Thermal Imaging for Resilient Localization in Adverse Conditions: Day and Night, Smoke and Fire”. In: *Journal of Field Robotics* 30.4, pp. 641–666. ISSN: 15564959. DOI: 10.1002/rob.21464.
- Cadena, Cesar et al. (2016). “Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age”. In: *IEEE Transactions on Robotics* 32.6, pp. 1309–1332. ISSN: 1552-3098. DOI: 10.1109/TR0.2016.2624754.
- Campos, Carlos, Jose M.M. Montiel, and Juan D. Tardos (2020). “Inertial-Only Optimization for Visual-Inertial Initialization”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 51–57. ISBN: 978-1-7281-7395-5. DOI: 10.1109/ICRA40945.2020.9197334.
- Campos, Carlos et al. (2021). “ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial, and Multimap SLAM”. In: *IEEE Transactions on Robotics*, pp. 1–17. ISSN: 1552-3098. DOI: 10.1109/TR0.2021.3075644.
- Carrier, B. L. et al. (2020). “Mars Extant Life: What’s Next? Conference Report”. In: *Astrobiology* 20.6, pp. 785–814. DOI: 10.1089/ast.2020.2237.
- Chen, Changhao et al. (2020). “A Survey on Deep Learning for Localization and Mapping: Towards the Age of Spatial Machine Intelligence”. In: *CoRR*. URL: <http://arxiv.org/pdf/2006.12567v2>.
- Chen, Liang-Chieh et al. (2014). “Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs”. In: *CoRR*. URL: <http://arxiv.org/pdf/1412.7062v4>.
- Chen, Liang-Chieh et al. (2017). “Rethinking Atrous Convolution for Semantic Image Segmentation”. In: *CoRR*. URL: <http://arxiv.org/pdf/1706.05587v3>.
- Chen, Liang-Chieh et al. (2018a). “DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs”. In: *IEEE*

- transactions on pattern analysis and machine intelligence* 40.4, pp. 834–848. DOI: 10.1109/TPAMI.2017.2699184.
- Chen, Liang-Chieh et al. (2018b). “Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation”. In: *Computer vision - ECCV 2018*. Ed. by Vittorio Ferrari et al. Vol. 11211. Lecture Notes in Computer Science. Cham: Springer, pp. 833–851. ISBN: 978-3-030-01233-5. DOI: 10.1007/978-3-030-01234-2_49.
- Choinowski, André et al. (2019). “Automatic Calibration and Co-Registration for a Stereo Camera System and a Thermal Imaging Sensor using a Chessboard”. In: *(ISPRS) International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol. XLII-2/W13, pp. 1631–1635. DOI: 10.5194/isprs-archives-XLII-2-W13-1631-2019.
- Clark, Ronald et al. (2017). “VINet: Visual-Inertial Odometry as a Sequence-to-Sequence Learning Problem”. In.
- Dave, Achal, Pavel Tokmakov, and Deva Ramanan (2019). “Towards Segmenting Anything That Moves”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*.
- Davison, Andrew J. et al. (2007). “MonoSLAM: real-time single camera SLAM”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.6, pp. 1052–1067. ISSN: 01628828. DOI: 10.1109/TPAMI.2007.1049.
- Denninger, Maximilian et al. (2019). “BlenderProc”. In: *CoRR*. URL: <http://arxiv.org/pdf/1911.01911v1>.
- Dosovitskiy, Alexey et al. (2017). “CARLA: An Open Urban Driving Simulator”. In: *CoRR* abs/1711.03938.
- Dubbelman, Gijs, Peter Hansen, and Brett Browning (2012). “Bias compensation in visual odometry”. In: *(IROS) International Conference on Intelligent Robots and Systems*. IEEE/RSJ, pp. 2828–2835. ISBN: 978-1-4673-1736-8. DOI: 10.1109/IROS.2012.6385713.
- Durrant-Whyte, H. and T. Bailey (2006). “Simultaneous localization and mapping: part I”. In: *IEEE Robotics & Automation Magazine* 13.2, pp. 99–110. ISSN: 1070-9932. DOI: 10.1109/MRA.2006.1638022.
- El-Sheimy, Naser and You Li (2021). “Indoor navigation: state of the art and future trends”. In: *Satellite Navigation* 2.1. DOI: 10.1186/s43020-021-00041-3.
- Elvira, Richard, Juan D. Tardos, and J.M.M. Montiel (2019). “ORBSLAM-Atlas: a robust and accurate multi-map system”. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 6253–6259. ISBN: 978-1-7281-4004-9. DOI: 10.1109/IROS40897.2019.8967572.
- Engel, Jakob, Vladlen Koltun, and Daniel Cremers (2016). “Direct Sparse Odometry”. In: *CoRR* abs/1607.02565.
- Ernst, Ines and Heiko Hirschmüller (2008). “Mutual Information Based Semi-Global Stereo Matching on the GPU”. In: *Advances in Visual Computing*. Vol. 5358. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 228–239. ISBN: 978-3-540-89638-8. DOI: 10.1007/978-3-540-89639-5_22.
- Ernst, Ines et al. (2018). “Large-Scale 3D Roadside Modelling with Road Geometry Analysis: Digital Roads New Zealand”. In: *(I-SPAN) Symposium on Pervasive Sys-*

- tems, Algorithms and Networks*. IEEE, pp. 15–22. ISBN: 978-1-5386-8534-1. DOI: 10.1109/I-SPAN.2018.00013.
- Fekete, Alexander and Simone Sandholz (2021). “Here Comes the Flood, but Not Failure? Lessons to Learn after the Heavy Rain and Pluvial Floods in Germany 2021”. In: *Water* 13.21, p. 3016. DOI: 10.3390/w13213016.
- Ferreira, Andre Filipe Goncalves et al. (2017). “Localization and Positioning Systems for Emergency Responders: A Survey”. In: *IEEE Communications Surveys & Tutorials* 19.4, pp. 2836–2870. DOI: 10.1109/COMST.2017.2703620.
- Fischer, Carl et al. (2008). “Ultrasound-aided pedestrian dead reckoning for indoor navigation”. In: *Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*. Ed. by Ying Zhang. ACM Conferences. New York, NY: ACM, p. 31. ISBN: 9781605581897. DOI: 10.1145/1410012.1410020.
- Fischler, Martin A. and Robert C. Bolles (1981). “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography”. In: *Communications of the ACM* 24.6, pp. 381–395. ISSN: 00010782. DOI: 10.1145/358669.358692.
- Fraundorfer, Friedrich and Davide Scaramuzza (2012). “Visual Odometry : Part II: Matching, Robustness, Optimization, and Applications”. In: *IEEE Robotics & Automation Magazine* 19.2, pp. 78–90. ISSN: 1070-9932. DOI: 10.1109/MRA.2012.2182810.
- Gaidon, Adrien et al. (2016). “Virtual Worlds as Proxy for Multi-Object Tracking Analysis”. In: *(CVPR) Conf. on Computer Vision and Pattern Recognition*. IEEE, pp. 4340–4349. ISBN: 978-1-4673-8851-1. DOI: 10.1109/CVPR.2016.470.
- Galvez-López, D. and J. D. Tardos (2012). “Bags of Binary Words for Fast Place Recognition in Image Sequences”. In: *IEEE Transactions on Robotics* 28.5, pp. 1188–1197. ISSN: 1552-3098. DOI: 10.1109/TR0.2012.2197158.
- Geiger, Andreas, Julius Ziegler, and Christoph Stiller (2011). “StereoScan: Dense 3d reconstruction in real-time”. In: *2011 IEEE Intelligent Vehicles Symposium (IV 2011) ; Baden-Baden, Germany, 5 - 9 June 2011*. Piscataway, NJ: IEEE, pp. 963–968. ISBN: 978-1-4577-0890-9. DOI: 10.1109/IVS.2011.5940405.
- Gonzalez, Rafael C. and Richard E. Woods (2018). *Digital image processing*. Fourth edition, global edition. New York, NY: Pearson. ISBN: 1-292-22304-9.
- Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep learning*. Cambridge, Massachusetts and London, England: MIT Press. ISBN: 9780262035613. URL: <http://www.deeplearningbook.org/>.
- Grießbach, Denis (2015). “Stereo-Vision-Aided Inertial Navigation”. Ph.D. thesis. Freie Universität Berlin. URL: <https://elib.dlr.de/97245/>.
- Grießbach, Denis, Dirk Baumbach, and Sergey Zuev (2014). “Stereo-vision-aided inertial navigation for unknown indoor and outdoor environments”. In: *(IPIN) International Conference on Indoor Positioning and Indoor Navigation*. IEEE, pp. 709–716. ISBN: 978-1-4673-8054-6. DOI: 10.1109/IPIN.2014.7275548.
- Haarbach, Adrian, Tolga Birdal, and Slobodan Ilic (2018). “Survey of Higher Order Rigid Body Motion Interpolation Methods for Keyframe Animation and Continuous-Time Trajectory Estimation”. In: *2018 International Conference on 3D Vision*.

- Piscataway, NJ: IEEE, pp. 381–389. ISBN: 978-1-5386-8425-2. DOI: 10.1109/3DV.2018.00051.
- Harris, C. G. and M. Stephens (1988). “A Combined Corner and Edge Detector”. In: *Alvey Vision Conference*, pp. 147–151.
- Hartley, Richard and Andrew Zisserman (2003). *Multiple view geometry in computer vision*. Second edition. Cambridge: Cambridge University Press. ISBN: 978-0521540513. DOI: 10.1017/CB09780511811685.
- Hein, Daniel et al. (2019). “Integrated UAV-Based Real-Time Mapping for Security Applications”. In: *ISPRS International Journal of Geo-Information* 8.5, p. 219. DOI: 10.3390/ijgi8050219.
- Huang, Guoquan (2019). “Visual-Inertial Navigation: A Concise Review”. In: *2019 (ICRA) International Conference on Robotics and Automation*, pp. 9572–9582.
- Hwangbo, Myung, Jun-Sik Kim, and Takeo Kanade (2011). “Gyro-aided feature tracking for a moving camera: fusion, auto-calibration and GPU implementation”. In: *The International Journal of Robotics Research* 30.14, pp. 1755–1774. ISSN: 0278-3649. DOI: 10.1177/0278364911416391.
- IFAFRI (2022). *the International Forum to Advance First Responder Innovation: Capability Gaps*. URL: <https://www.internationalresponderforum.org/capability-gaps-overview>.
- INGENIOUS (2019). *The first responder of the future: a next generation integrated toolkit for collaborative response, increasing protection and augmenting operational capacity*. GD number 833435. URL: <https://ingenious-first-responders.eu/>.
- Ioffe, Sergey and Christian Szegedy (2015). “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”. In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Francis Bach and David Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, pp. 448–456.
- IPIN (2014). *IPIN International Conference on Indoor 2014*. URL: <http://ipin-conference.org/2014/>.
- Irmisch, Patrick (2017). “Camera-based distance estimation for autonomous vehicles”. Master Thesis. Berlin: Technical University Berlin. URL: <https://elib.dlr.de/116211/>.
- Irmisch, Patrick, Dirk Baumbach, and Ines Ernst (2020). “Robust Visual-Inertial Odometry in Dynamic Environments using Semantic Segmentation for Feature Selection”. In: *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences V-2-2020*, pp. 435–442. DOI: 10.5194/isprs-annals-V-2-2020-435-2020.
- Irmisch, Patrick et al. (2019). “Simulation Framework for a Visual-Inertial Navigation System”. In: *(ICIP) International Conference on Image Processing*, pp. 1995–1999. DOI: 10.1109/ICIP.2019.8803187.
- Irmisch, Patrick et al. (2021). “A Hand-Held Sensor System for Exploration and Thermal Mapping of Volcanic Fumarole Fields”. In: *Geometry and Vision*. Ed. by Minh Nguyen, Wei Qi Yan, and Harvey Ho. Vol. 1386. Communications in Computer and Information Science. [S.l.]: Springer, pp. 68–84. ISBN: 978-3-030-72072-8. DOI: 10.1007/978-3-030-72073-5_6.

- JCGM 101 (2008). “Evaluation of Measurement Data: Supplement 1 to the "Guide to the Expression of Uncertainty in Measurement" - Propagation of Distributions Using a Monte Carlo Method”. In: Joint Committee for Guides in Metrology.
- JCGM 102 (2011). “Evaluation of Measurement Data: Supplement 2 to the "Guide to the Expression of Uncertainty in Measurement" - Extension to Any Number of Output Quantities”. In: *Joint Committee for Guides in Metrology*.
- Jeon, Hae-Gon et al. (2019). “DISC: A Large-scale Virtual Dataset for Simulating Disaster Scenarios”. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 187–194. ISBN: 978-1-7281-4004-9. DOI: 10.1109/IROS40897.2019.8967839.
- Jiang, Ruyi, Reinhard Klette, and Shigang Wang (2010). “Statistical Modeling of Long-Range Drift in Visual Odometry”. In: vol. 6469, pp. 214–224. DOI: 10.1007/978-3-642-22819-3_22.
- Kachurka, Viachaslau et al. (2021). “WeCo-SLAM: Wearable Cooperative SLAM System for Real-time Indoor Localization Under Challenging Conditions”. In: *IEEE Sensors Journal*, p. 1. ISSN: 1530-437X. DOI: 10.1109/JSEN.2021.3101121.
- Kaneko, Masaya et al. (2018). “Mask-SLAM: Robust Feature-Based Monocular SLAM by Masking Using Semantic Segmentation”. In: *(CVPRW) Conference on Computer Vision and Pattern Recognition Workshops*. IEEE/CVF, pp. 3710–3718.
- Kang, I. G. and F. C. Park (1999). “Cubic spline algorithms for orientation interpolation”. In: *International Journal for Numerical Methods in Engineering* 46.1, pp. 45–64. ISSN: 0029-5981. DOI: 10.1002/(SICI)1097-0207(19990910)46:1<45::AID-NME662>3.0.CO;2-K.
- Koenig, N. and A. Howard (2004). “Design and use paradigms for Gazebo, an open-source multi-robot simulator”. In: *2004(IROS) IEEE/RSJ International Conference on Intelligent Robots and Systems (IEEE Cat. No.04CH37566)*. Vol. 3, 2149–2154 vol.3. DOI: 10.1109/IROS.2004.1389727.
- Kreienkamp, Frank et al. (2021). “Rapid attribution of heavy rainfall events leading to the severe flooding in Western Europe during July 2021”. In: *World Weather Attribution*.
- Kriechbaumer, Thomas et al. (2015). “Quantitative Evaluation of Stereo Visual Odometry for Autonomous Vessel Localisation in Inland Waterway Sensing Applications”. In: *Sensors* 15.12, pp. 31869–31887. ISSN: 1424-8220. DOI: 10.3390/s151229892.
- Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton (2012). “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems 25*. Ed. by F. Pereira et al. Curran Associates, Inc, pp. 1097–1105.
- Lehmann, Florian (2015). “Implementierung einer virtuellen Kamera mit Verzeichnung”. Internship report. Germany: German Aerospace Center.
- (2016). “Implementierung einer virtuellen Stereokamera”. Study report. German Aerospace Center.
- Ley, Andreas, Ronny Hänsch, and Olaf Hellwich (2016). “SyB3R: A Realistic Synthetic Benchmark for 3D Reconstruction from Images”. In: *2016 (ECCV) European Conference on Computer Vision*. Vol. 9911, pp. 236–251. DOI: 10.1007/978-3-319-46478-7_15.

- Lin, Tsung-Yi et al. (2014). “Microsoft COCO: Common Objects in Context”. In: *Computer Vision – ECCV 2014*. Ed. by David Fleet et al. Cham: Springer International Publishing, pp. 740–755. ISBN: 978-3-319-10602-1.
- Liu, Yuanzhi et al. (2021). “Datasets and Evaluation for Simultaneous Localization and Mapping Related Problems: A Comprehensive Survey”. In: *CoRR* abs/2102.04036.
- Long, Jonathan, Evan Shelhamer, and Trevor Darrell (2015). “Fully convolutional networks for semantic segmentation”. In: *2015 (CVPR) IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 3431–3440. ISBN: 978-1-4673-6964-0. DOI: 10.1109/CVPR.2015.7298965.
- Lopez-Fuentes, Laura, Claudio Rossi, and Harald Skinnemoen (2017). “River segmentation for flood monitoring”. In: *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, pp. 3746–3749. ISBN: 978-1-5386-2715-0. DOI: 10.1109/BigData.2017.8258373.
- Lowe, David G. (2004). “Distinctive Image Features from Scale-Invariant Keypoints”. In: *Int. J. Comput. Vision* 60.2, pp. 91–110. ISSN: 0920-5691. DOI: 10.1023/B:VISI.0000029664.99615.94.
- Mair, Elmar et al. (2010). “Adaptive and Generic Corner Detection Based on the Accelerated Segment Test”. In: *Computer Vision – ECCV 2010*. Vol. 6312. Lecture Notes in Computer Science. Springer, pp. 183–196. ISBN: 978-3-642-15551-2. DOI: 10.1007/978-3-642-15552-9_14.
- McDowell, Perry et al. (2006). “Delta3D: A Complete Open Source Game and Simulation Engine for Building Military Training Systems”. In: *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology* 3, pp. 143–154. DOI: 10.1177/154851290600300302.
- Migliore, Davide et al. (2009). “Use a Single Camera for Simultaneous Localization And Mapping with Mobile Object Tracking in dynamic environments”. In: *ICRA 2009*.
- Minaee, Shervin et al. (2021). “Image Segmentation Using Deep Learning: A Survey”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP, p. 1. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2021.3059968.
- Mourikis, Anastasios I. and Stergios I. Roumeliotis (2007). “A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation”. In: *2007 (ICRA) IEEE International Conference on Robotics and Automation*. IEEE, pp. 3565–3572. ISBN: 1-4244-0602-1. DOI: 10.1109/ROBOT.2007.364024.
- Mur-Artal, Raul and Juan D. Tardós (2016). “ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras”. In: *IEEE Transactions on Robotics* 33.5, pp. 1255–1262. DOI: 10.1109/TR0.2017.2705103.
- Mur-Artal, Raul and Juan D. Tardos (2017). “Visual-Inertial Monocular SLAM With Map Reuse”. In: *IEEE Robotics and Automation Letters* 2.2, pp. 796–803. ISSN: 2377-3766. DOI: 10.1109/LRA.2017.2653359.
- Murphy, Robin R. (2021). “Responsible Robotics Innovation for Disaster Response”. In: *Building and Evaluating Ethical Robotic Systems*. URL: <https://www.ers-workshop.com/pdf/MMers21.pdf>.
- Navarro, Fernando, Francisco J. Serón, and Diego Gutierrez (2011). “Motion Blur Rendering: State of the Art”. In: *Computer Graphics Forum* 30.1, pp. 3–26. ISSN: 01677055. DOI: 10.1111/j.1467-8659.2010.01840.x.

- Nister, D., O. Naroditsky, and J. Bergen (2004). “Visual odometry”. In: *(CVPR) Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 652–659. ISBN: 0-7695-2158-4. DOI: 10.1109/CVPR.2004.1315094.
- Noh, Hyeonwoo, Seunghoon Hong, and Bohyung Han (2015). “Learning Deconvolution Network for Semantic Segmentation”. In: *CoRR*. URL: <http://arxiv.org/pdf/1505.04366v1>.
- OpenGL (2012). *OpenGL / FAQ 2 - Depth Buffer Precision*. Ed. by The OpenGL Organization. URL: https://www.khronos.org/opengl/wiki/Depth_Buffer_Precision#Why_is_my_depth_buffer_precision_so_poor.3F.
- Pham, Thanh Tuan and Young Soo Suh (2018). “Spline Function Simulation Data Generation for Walking Motion Using Foot-Mounted Inertial Sensors”. In: *Electronics* 8, p. 18. DOI: 10.3390/electronics8010018.
- Poddar, Shashi, Rahul Kottath, and Vinod Karar (2018). “Evolution of Visual Odometry Techniques”. In: *CoRR*. URL: <http://arxiv.org/pdf/1804.11142v1>.
- Rantakokko, J. et al. (2010). “User requirements for localization and tracking technology: A survey of mission-specific needs and constraints”. In: *2010 International Conference on Indoor Positioning and Indoor Navigation*. Ed. by Rainer Mautz. Piscataway, NJ: IEEE, pp. 1–9. ISBN: 978-1-4244-5862-2. DOI: 10.1109/IPIN.2010.5646765.
- Rantakokko, Jouni et al. (2011). “Accurate and reliable soldier and first responder indoor positioning: multisensor systems and cooperative localization”. In: *IEEE Wireless Communications* 18.2, pp. 10–18. ISSN: 1536-1284. DOI: 10.1109/MWC.2011.5751291.
- Rauch, H. E., F. Tung, and C. T. Striebel (1965). “Maximum likelihood estimates of linear dynamic systems”. In: *AIAA Journal* 3.8, pp. 1445–1450. ISSN: 0001-1452. DOI: 10.2514/3.3166.
- RESCUER (2018). *First RESponder-Centred support toolkit for operating in adverse and infrastrUcture-less EnviRonments: GD number 847912*. URL: <https://www.rescuer.uio.no/>.
- Ronneberger, O., P. Fischer, and T. Brox (2015). “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Vol. 9351. LNCS. Springer, pp. 234–241.
- Rosten, E. and T. Drummond (2005). “Fusing points and lines for high performance tracking”. In: *Proceedings / Tenth IEEE International Conference on Computer Vision*. Los Alamitos, Calif.: IEEE Computer Society, 1508–1515 Vol. 2. ISBN: 0-7695-2334-X. DOI: 10.1109/ICCV.2005.104.
- Roumeliotis, S. I. and J. W. Burdick (2002). “Stochastic cloning: a generalized framework for processing relative state measurements”. In: *2002 IEEE International Conference on Robotics and Automation*. Piscataway, NJ: IEEE Service Center, pp. 1788–1795. ISBN: 0-7803-7272-7. DOI: 10.1109/ROBOT.2002.1014801.
- Sandler, Mark et al. (2018). “MobileNetV2: Inverted Residuals and Linear Bottle-necks”. In: *(CVPR) Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 4510–4520. ISBN: 978-1-5386-6420-9. DOI: 10.1109/CVPR.2018.00474.
- Saputra, Muhamad Risqi U., Andrew Markham, and Niki Trigoni (2018). “Visual SLAM and Structure from Motion in Dynamic Environments”. In: *ACM Computing Surveys* 51.2, pp. 1–36. ISSN: 03600300. DOI: 10.1145/3177853.

- Sayre-McCord, T. et al. (2018). “Visual-Inertial Navigation Algorithm Development Using Photorealistic Camera Simulation in the Loop”. In: *(ICRA) IEEE International Conference on Robotics and Automation*. IEEE, pp. 2566–2573. ISBN: 978-1-4503-6584-0. DOI: 10.1109/ICRA.2018.8460692.
- Scaramuzza, Davide and Friedrich Fraundorfer (2011). “Visual Odometry [Tutorial]”. In: *IEEE Robotics & Automation Magazine* 18.4, pp. 80–92. ISSN: 1070-9932. DOI: 10.1109/MRA.2011.943233.
- Scherer, Sebastian et al. (2012). “River mapping from a flying robot: state estimation, river detection, and obstacle mapping”. In: *Autonomous Robots* 33.1-2, pp. 189–214. ISSN: 0929-5593. DOI: 10.1007/s10514-012-9293-0.
- Schorghuber, Matthias et al. (2019). “SLAMANTIC - Leveraging Semantics to Improve VSLAM in Dynamic Environments”. In: *(ICCV) International Conference on Computer Vision*, pp. 3759–3768.
- Schreer, Oliver (2005). *Stereoanalyse und Bildsynthese: Mit 6 Tabellen*. Berlin: Springer. ISBN: 3-540-23439-X.
- Shah, Shital et al. (2017). “AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles”. In: *CoRR* abs/1705.05065.
- Siam, Mennatullah et al. (2018). “MODNet: Motion and Appearance based Moving Object Detection Network for Autonomous Driving”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 2859–2864. ISBN: 978-1-7281-0321-1. DOI: 10.1109/ITSC.2018.8569744.
- Smit, B.-P., R. Voûte, and E. Verbree (2021). “Creating 3D Indoor First Responder Situation Awareness in Real-Time through a Head-Mounted AR Device”. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences V-4-2021*, pp. 209–216. DOI: 10.5194/isprs-annals-v-4-2021-209-2021.
- Smith, Stephen M. and J. Michael Brady (1997). “SUSAN—A New Approach to Low Level Image Processing”. In: *Int. J. Comput. Vision* 23.1, pp. 45–78. ISSN: 0920-5691. DOI: 10.1023/A:1007963824710.
- Starr, Joseph W. and B. Y. Lattimer (2014). “Evaluation of Navigation Sensors in Fire Smoke Environments”. In: *Fire Technology* 50.6, pp. 1459–1481. ISSN: 0015-2684. DOI: 10.1007/s10694-013-0356-3.
- Strutz, Tilo (2016). *Data fitting and uncertainty: A practical introduction to weighted least squares and beyond*. 2nd, revised and extended edition. Wiesbaden: Springer Vieweg. ISBN: 9783658114558.
- Stumberg, Lukas von, Vladyslav Usenko, and Daniel Cremers (2018). “Direct Sparse Visual-Inertial Odometry using Dynamic Marginalization”. In: *(ICRA) IEEE International Conference on Robotics and Automation*. IEEE, pp. 2510–2517. ISBN: 978-1-4503-6584-0.
- Sturm, Jrgen et al. (2012). “A benchmark for the evaluation of RGB-D SLAM systems”. In: *(IROS) International Conference on Intelligent Robots and Systems*. IEEE/RSJ, pp. 573–580. ISBN: 978-1-4673-1736-8. DOI: 10.1109/IROS.2012.6385773.
- Thiele, Tom (2015). “Automatic Tree Analysis by Means of Stereo Vision”. Master thesis. Hochschule für nachhaltige Entwicklung Eberswalde, Warsaw University of Life Sciences. URL: <https://elib.dlr.de/98293/>.

- Titterton, David H. and John L. Weston (2004). *Strapdown inertial navigation technology*. 2. ed. Vol. 17. IET radar, sonar, navigation and avionics series. Stevenage: The Inst. of Engineering and Technology. ISBN: 9780863413582.
- Tölgyessy, Michal et al. (2021). “Evaluation of the Azure Kinect and Its Comparison to Kinect V1 and Kinect V2”. In: *Sensors* 21.2, p. 413. ISSN: 1424-8220. DOI: 10.3390/s21020413.
- Umeyama, S. (1991). “Least-squares estimation of transformation parameters between two point patterns”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13.4, pp. 376–380. ISSN: 01628828. DOI: 10.1109/34.88573.
- Unnithan, V. et al. (2019). “Vulcano Summer School 2019”. In: *EPSC-DPS Joint Meeting 2019*. Vol. 13. EPSC Abstracts, p. 2051.
- Vaudrey, Tobi et al. (2008). “Differences between stereo and motion behaviour on synthetic and real-world stereo sequences”. In: *2008 23rd International Conference Image and Vision Computing New Zealand*. Ed. by Kenji Irie. Piscataway, NJ: IEEE, pp. 1–6. ISBN: 978-1-4244-2582-2. DOI: 10.1109/IVCNZ.2008.4762133.
- Wang, Han, Chen Wang, and Lihua Xie (2021). “Lightweight 3-D Localization and Mapping for Solid-State LiDAR”. In: *IEEE Robotics and Automation Letters* 6.2, pp. 1801–1807. ISSN: 2377-3766. DOI: 10.1109/LRA.2021.3060392.
- Wang, Rui and Xuelei Qian (2012). *OpenSceneGraph 3 Cookbook: Over 80 recipes to show advanced 3D programming techniques with the OpenSceneGraph API*. Packt open source. Birmingham: Packt Publ. ISBN: 978-1-84951-688-4.
- Wang, Sen et al. (2017). “DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks”. In: *2017 (ICRA) IEEE International Conference on Robotics and Automation*. IEEE, pp. 2043–2050. ISBN: 978-1-5090-4633-1. DOI: 10.1109/ICRA.2017.7989236.
- Wang, Wenshan et al. (2020). “TartanAir: A Dataset to Push the Limits of Visual SLAM”. In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 4909–4916. ISBN: 978-1-7281-6212-6. DOI: 10.1109/IROS45743.2020.9341801.
- Welch, Greg and Gary Bishop (1995). “An Introduction to the Kalman Filter”. In: *Tech. rep. Chapel Hill, NC, USA: University of North Carolina at Chapel Hill*.
- Wendel, Jan (2011). *Integrierte Navigationssysteme: Sensordatenfusion, GPS und inertielle Navigation*. 2., überarbeitete Auflage. München: Oldenbourg. ISBN: 9783486704396. DOI: 10.1524/9783486705720.
- Wilken, Mark et al. (2015). “IRIS - An Innovative Inspection System for Maritime Hull Structures”. In: *International Conference on Computer Applications in Shipbuilding, ICCAS 2015*. Vol. 2, pp. 219–226. URL: <https://elib.dlr.de/103868/>.
- Wohlfeil, J. et al. (2019). “Automatic Camera System Calibration with a Chessboard enabling Full Image Coverage”. In: *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 1715–1722. DOI: 10.5194/isprs-archives-XLII-2-W13-1715-2019.
- Woodman, Oliver J. (2007). “An introduction to inertial navigation”. In: *Tech. rep. UCAM-CLTR-696*. URL: <https://www.cl.cam.ac.uk/techreports/UCAM-CLTR-696.pdf>.

- Yang, Junho et al. (2017). “Vision-based Localization and Robot-centric Mapping in Riverine Environments”. In: *Journal of Field Robotics* 34.3, pp. 429–450. ISSN: 15564959. DOI: 10.1002/rob.21606.
- Yuan, Feiniu et al. (2019). “Deep smoke segmentation”. In: *Neurocomputing* 357, pp. 248–260. ISSN: 09252312. DOI: 10.1016/j.neucom.2019.05.011.
- Zhan, Huangying et al. (2020). “Visual Odometry Revisited: What Should Be Learnt?” In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 4203–4210. ISBN: 978-1-7281-7395-5. DOI: 10.1109/ICRA40945.2020.9197374.
- Zhang, Chaofan et al. (2018). “VINS-MKF: A Tightly-Coupled Multi-Keyframe Visual-Inertial Odometry for Accurate and Robust State Estimation”. In: *Sensors* 18.11, p. 4036. ISSN: 1424-8220. DOI: 10.3390/s18114036.
- Zhang, Hongmou (2018). “Optical Navigation for Mobile Platforms Based on Camera Data”. PhD thesis. Technischen Universität Berlin. URL: <https://elib.dlr.de/120268/>.
- Zhang, Zichao and Davide Scaramuzza (2018). “A Tutorial on Quantitative Trajectory Evaluation for Visual(-Inertial) Odometry”. In: *(IROS) International Convergence on Intelligent Robots and Systems*. IEEE/RSL, pp. 7244–7251. DOI: 10.1109/IROS.2018.8593941.
- Zhou, Bolei et al. (2017). “Scene Parsing through ADE20K Dataset”. In: *2017 (CVPR) IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 5122–5130. ISBN: 978-1-5386-0457-1. DOI: 10.1109/CVPR.2017.544.
- Zou, Danping and Ping Tan (2013). “CoSLAM: collaborative visual SLAM in dynamic environments”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.2, pp. 354–366. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2012.104.
- Zuev, Sergey et al. (2019). “Mobile system for road inspection and 3D modelling”. In: *Internationales Verkehrswesen* 71.1, pp. 18–21. URL: <https://elib.dlr.de/127720/>.

Technology List

- [OSG] OpenSceneGraph-3.4.1. 2017. *OpenSceneGraph is an OpenGL-based high performance 3D graphics toolkit for visual simulation, games, virtual reality, scientific visualization, and modeling.* <http://www.openscenegraph.org>. Last downloaded 2020-03-02.
- [OpenGL] OpenGL. *The Industry's Foundation of High Performance Graphics.* <https://www.opengl.org/>. Last downloaded 2020-03-02. Embedded in OpenSceneGraph.
- [OpenCV] OpenCV-3.1 2015. *Open Source Computer Vision Library.* <http://opencv.org/>. Last downloaded 2016.
- [OSLib] OSLib. DLR Intern. *C++ Software Libraries for Image Processing.* Implement basic structures, classes and algorithms from computer vision and data fusion. Last downloaded 2021-04-24.
- [Deeplab] Deeplab. 2018. *Deep Labelling for Semantic Image Segmentation.* <https://github.com/tensorflow/models/tree/master/research/deeplab>. Last downloaded 2020-03-29.
- [Pybind11] Pybind11. 2016. *Seamless operability between C++11 and Python.* <https://github.com/pybind/pybind11>. Last downloaded 2018-05-22.
- [DellPrecision] Dell Precision 5820 Tower. Processor: Intel(R) Xeon(R) W-2145 CPU @ 3.7 GHz 8 Cores 16 Threads. Graphic Card: NVIDIA Quadro RTX 6000 24 GB GDDR6 4608/576/72 Cuda/Tensor/RT Cores. 64 GB RAM.
- [pyins] pyins-0.1. *Data processing commonly done in Strapdown Inertial Navigation Systems integrated with other aiding sensors..* <https://github.com/nmayorov/pyins/>.
- [SciPy] SciPy-1.5.4. *Fundamental algorithms for scientific computing in Python.* <https://scipy.org/>. Last downloaded 2018.
- [Blender] Blender-2.79. 2017. *3D modeling and animation tool.* <https://www.blender.org/>. Last downloaded 2018.
- [XNA] Microsoft XNA. 2010. *Human skinned model.*
- [Pix4D] Pix4Dmapper. 2020. Version 4.5.6. *Photogrammetry-Tool.* <https://www.pix4d.com/>.

Appendix A

Supplementary Material

This appendix provides additional material to enable a better understanding of different aspects. First, the idea behind the least-squares problem is explained, which is the mathematical basis of the considered VO approach of Section 3.2.2. Second, information about characteristics of the used datasets are listed and are linked to the individual corresponding sections. Third, calibration parameters and the respective uncertainties are listed, which complements Section 5.1.

A.1 The Least-Squares Problem

The least squares method (LS) belongs to the mathematical field of data fitting. This section follows the detailed introduction of (Strutz, 2016). Data fitting aims at finding model-parameters \mathbf{a} of a chosen model function f that best describe the connection between a set of pairs with conditions \mathbf{x}_i and experimental scalar observations y_i that are subjected to random errors ϵ_i . The data fitting problem is formulated as

$$y_i = f(\mathbf{x}_i|\mathbf{a}) + \epsilon_i. \quad (\text{A.1})$$

Solving this problem requires establishing linearly independent equations based on pairs of conditions and observations. Due to the presence of observation noise, it is beneficial to have more linearly independent equations than unknown variables in \mathbf{a} . A common solution to this over-determined problem is the least square method.

LS can be formulated based on maximum likelihood estimation (MLE), that aims at finding values of \mathbf{a} that maximize the likelihood of making the observations y_i based on the given conditions and parameter values. In this context, it is assumed that each observation y_i is drawn from an independent, uncorrelated normal distribution with mean equal to $f(\mathbf{x}_i|\mathbf{a})$ and SD σ_i , and that the error $\epsilon_i = y_i - f(\mathbf{x}_i|\mathbf{a})$ is normally distributed (Strutz, 2016, p.158). The probability for making the observation y_i is defined as

$$P(y_i|\mathbf{a}) = \frac{1}{\sigma_i\sqrt{2\pi}} \cdot \exp \left\{ -\frac{1}{2} \cdot \left[\frac{y_i - f(\mathbf{x}_i|\mathbf{a})}{\sigma_i} \right]^2 \right\}. \quad (\text{A.2})$$

This probability needs to be maximized for all observations in dependence of the model parameters. A joint optimization is achieved by a multiplication of all single probabilities, resulting in the observation probability $P(\mathbf{a})$, which is to be maximized:

$$P(\mathbf{a}) = \prod_{i=1}^N P(y_i|\mathbf{a}) = \prod_{i=1}^N \frac{1}{\sigma_i\sqrt{2\pi}} \cdot \exp\left\{-\frac{1}{2}\cdot\chi^2(\mathbf{a})\right\} \longrightarrow \text{Max.} \quad (\text{A.3})$$

This maximization equals a minimization of χ^2 in the exponential part. χ^2 describes the squared residual error between the observations and estimated value based on the model function. Thus, the least square problem can be derived with

$$\chi^2(\mathbf{a}) = \sum_{i=1}^N w_i \cdot [y_i - f(\mathbf{x}_i|\mathbf{a})]^2 \longrightarrow \text{Min.} \quad (\text{A.4})$$

Weight $w_i = 1/\sigma_i^2$ is based of the observation uncertainty of y_i and can be used for the weighted least square method (WLS), if the uncertainty of y_i is known.

The solution of WLS depends on whether the relation between the observations y and model parameters \mathbf{a} is linear or nonlinear. For the former, the optimal parameters for \mathbf{a} can be estimated directly. For the latter, an iterative approach is required that solves

$$\mathbf{a}_{k+1} = \mathbf{a}_k + \Delta\mathbf{a}. \quad (\text{A.5})$$

The model parameter update $\Delta\mathbf{a}$ can be estimated using the Gauss-Newton algorithm that approximates the target function by a Taylor series of second order (Strutz, 2016, p.164). The solution is

$$\Delta\mathbf{a} = (\mathbf{J}^T \cdot \mathbf{W} \cdot \mathbf{J})^{-1} \cdot \mathbf{J}^T \cdot \mathbf{W} \cdot \mathbf{r}, \quad (\text{A.6})$$

which is composed of the residual vector \mathbf{r} with $r_i = y_i - f(\mathbf{x}_i|\mathbf{a})$, the diagonal weight matrix \mathbf{W} and the Jacobian matrix \mathbf{J} . The latter contains partial derivatives $J_{ij} = \delta f(\mathbf{x}_i|\mathbf{a})/\delta a_j$, with a matrix size of (N, M) given N observations and M model parameters. The advantage of Gauss-Newton is a general fast convergence to the closest minimum, but it requires an initial estimate that is close to the optimum.

An appropriate alternative is Levenberg-Marquart if a good initial estimate can not be guaranteed. It combines the Gauss-Newton method with the gradient descent method that is constraint by a dynamic damping factor. This factor ensures a dominant application of gradient descent, if the estimate is still far from the optimum, and of Gauss-Newton, if it is relatively close.

A.2 Datasets

This appendix provides additional information on datasets used in this thesis. This includes exemplary images, the trajectory lengths in 3D, information about camera motion dynamics and information about GT. Note that brightness of most of the images was increased for better visualization.

Table A.1: Characteristics of simulated datasets used for *geometric MCS* (Section 5.3.1).

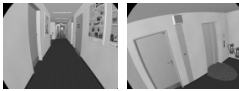
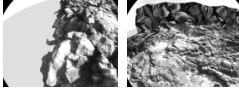

dataset	representative images	dist. [m] 3D	velocity [m/s]		angular rate [°/s]		Ground Truth
			mean	max	mean	max	
corridor (static)		24.81	0.62	1.34	20.03	86.32	yes
coast (static)		26.15	0.46	1.68	32.19	159.32	yes
fumaroles (static)		48.61	0.4	1.49	32.35	172.21	yes

Table A.2: Characteristics of simulated corridor datasets (Section 7.2.1).




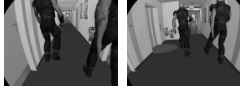
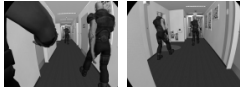

dataset	representative images	dist. [m] 3D	velocity [m/s]		angular rate [°/s]		Ground Truth
			mean	max	mean	max	
sim-corr-s01 (strong motion)		18.89	0.74	1.41	34.59	252.53	yes
sim-corr-s02 (static humans)		24.81	0.62	1.34	20.05	85.99	yes
sim-corr-d01		17.51	0.87	1.62	13.23	48.71	yes
sim-corr-d02		17.6	0.84	1.51	13.03	54.66	yes
sim-corr-d03		17.74	0.84	1.34	12.93	44.88	yes
sim-corr-d04 (slow humans)		24.81	0.62	1.34	20.03	86.32	yes

Table A.3: Characteristics of real datasets without dynamic elements (Section 5.3.2).











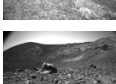

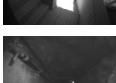


dataset	representative images	dist. [m] 3D	velocity [m/s]		angular rate [$^{\circ}/s$]		Ground Truth (GCPs)
			mean	max	mean	max	
corridor-1		33.49	1.0	1.57	29.78	97.49	1
corridor-2		37.56	0.74	1.44	25.63	85.15	2
corridor-3		37.74	0.81	1.53	27.0	81.58	2
(corridor-4)		24.94	0.54	1.35	18.05	86.97	2
basement-1		215.26	0.54	1.2	10.92	90.89	6
park-area-1		364.02	0.73	1.46	12.13	93.43	15
park-area-2		30.44	0.87	1.3	19.65	69.52	1
coast-1		49.9	0.67	1.33	22.12	97.73	1
crater-rim-1		67.64	0.41	1.77	15.6	135.65	1
crater-rim-2		42.24	0.66	1.7	36.75	158.35	1
mars-1		67.66	0.72	1.54	32.4	107.62	1
hotel-1		43.89	0.66	1.94	37.47	195.88	1
mine-1		122.74	0.41	0.83	14.57	83.16	3
mine-2		134.34	0.36	0.98	15.56	100.89	3
park-stairs-1		36.35	0.71	1.1	22.89	99.61	1

Table A.4: Characteristics of real datasets from the corridor environment (Section 7.2.1).


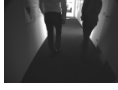




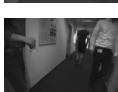
dataset	representative images	dist. [m] 3D	velocity [m/s]		angular rate [$^{\circ}$ /s]		Ground Truth (GCPs)
			mean	max	mean	max	
corridor-d1		17.96	0.7	1.18	13.73	52.76	2
corridor-d2		17.85	0.49	0.91	12.87	46.02	2
corridor-d3		18.06	0.47	0.84	13.24	39.96	2
corridor-d4		18.71	0.53	1.21	15.98	72.38	2
corridor-d5		18.79	0.51	1.05	17.65	86.88	2
corridor-d6		20.09	0.55	1.02	17.91	57.21	2
corridor-d7		144.93	0.72	1.61	24.0	148.81	2

Table A.5: Characteristics of simulated dataset used for *combined sensitivity analysis* (Sections 7.2.2, 8.3.1, 8.3.2).


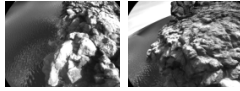
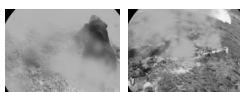
dataset	representative images	dist. [m] 3D	velocity [m/s]		angular rate [$^{\circ}$ /s]		Ground Truth
			mean	max	mean	max	
corridor (slow humans)		24.81	0.62	1.34	20.03	86.32	yes
coast (dynamic water)		26.15	0.46	1.68	32.19	159.32	yes
fumaroles (dynamic smoke)		48.61	0.4	1.49	32.35	172.21	yes

Table A.6: Characteristics of real datasets from the mall environments (Section 7.3).



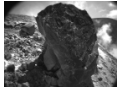


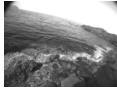




dataset	representative images	dist. [m] 3D	velocity [m/s]		angular rate [°/s]		Ground Truth (GCPs)
			mean	max	mean	max	
ipin-1		902	0.9	1.63	18.3	129	6
ipin-2		874	1.06	1.92	22	156	6

Table A.7: Characteristics of real datasets from fumaroles, coast, river environments (Section 8.4).

dataset	representative images	dist. [m] 3D	velocity [m/s]		angular rate [°/s]		Ground Truth (CLs)
			mean	max	mean	max	
fumaroles-d1		359.81	0.46	1.65	32.1	175.69	5
fumaroles-d2		549.02	0.43	2.08	25.15	208.9	3
coast-d1		656.32	0.48	1.66	35.74	200.27	6
coast-d2		267.66	0.49	1.67	32.48	191.24	3
river-a-d1		44.3	0.54	0.97	18.8	78.37	1
river-a-d2		43.03	0.34	1.37	13.54	99.07	1
river-b-d3		40.74	0.61	0.92	19.19	85.23	1
river-b-s1		36.35	0.71	1.1	22.89	99.61	1

A.3 Geometric Calibration Parameters

Different calibration settings are considered in the experiments of this thesis. The parameter values are listed in A.8 and their corresponding applied uncertainties are listed in Table A.9.

Three different base calibration sets are considered. The laboratory calibration describes a camera calibration that was conducted in a laboratory (see Section 5.1.1). The in-situ calibration describes a camera calibration that was conducted during field operations (see Section 5.1.1). Additionally, the calibration *ipin-2014* was conducted in 2014 for the IPIN challenge (IPIN, 2014). The calibration for inspections (I) is based on the laboratory calibration parameters, which uncertainties were scaled and listed in Table A.9. Similarly, the calibration for first responders (F) is based on the in-situ calibration parameters, which uncertainties were scaled and listed in Table A.9. A detailed discussion is presented in Section 5.1.

Additionally, the calibration set *prior* is used in Section 5.3.2, which is based on prior uncertainties for the intrinsic camera parameters based on expert knowledge and uncertainties from the laboratory calibration for the stereo transformation.

Table A.8: Geometric calibration parameters for intrinsic and extrinsic camera parameters and IMU registration. Three different calibration sets are considered.

(a) Camera calibration parameters, exemplary for camera left (κ^l, δ^l)						
calibration	f [px]	u_0 [px]	v_0 [px]	k_1	k_2	k_3
laboratory	775.23	710.03	546.07	-0.2593	0.1166	-0.0281
in-situ	775.34	712.63	546.26	-0.2634	0.1265	-0.0335
ipin 2014	776.05	711.58	547.78	-0.2591	0.1132	-0.0256

(b) Stereo camera calibration parameters \mathbf{T}_l^r						
calibration	α_x [°]	α_y [°]	α_z [°]	t_x [mm]	t_y [mm]	t_z [mm]
laboratory	0.0075	0.0067	0.0048	-200.54	-0.405	0.57
in-situ	0.0095	0.007	0.0039	-200.93	-0.172	-1.362
ipin 2014	0.0058	0.0081	0.0067	-200.89	-0.035	0.383

(c) IMU registration parameters \mathbf{T}_l^b						
calibration	α_x [°]	α_y [°]	α_z [°]	t_x [mm]	t_y [mm]	t_z [mm]
laboratory	-1.567	0.0028	1.581	2.3	22	32.5
in-situ	-1.567	0.0028	1.581	2.3	22	32.5
ipin 2014	1.5716	-0.0057	1.5609	-15.5	22	32.5

Table A.9: SDs of geometric calibration parameters for intrinsic and extrinsic camera parameters and IMU registration. Three different calibration sets are considered.

(a) Intrinsic camera calibration uncertainties, exemplary for camera left ($\boldsymbol{\kappa}, \boldsymbol{\delta}$)

calibration	f [px]	u_0 [px]	v_0 [px]	k_1	k_2	k_3
prior	0.2	0.3	0.3	0	0	0
laboratory	0.043	0.073	0.073	0.00014	0.00024	0.00012
<i>scale factor</i>	<i>4</i>	<i>4</i>	<i>4</i>	<i>4</i>	<i>4</i>	<i>4</i>
inspection (I)	0.172	0.291	0.29	0.00057	0.00097	0.00048
in-situ	0.274	0.538	0.416	0.00228	0.00611	0.00435
<i>scale factor</i>	<i>1.3</i>	<i>1.3</i>	<i>1.3</i>	<i>1.3</i>	<i>1.3</i>	<i>1.3</i>
first responder (F)	0.357	0.7	0.54	0.00297	0.00795	0.00566

(b) Stereo camera calibration uncertainties \mathbf{T}_l^r

calibration	α_x [°]	α_y [°]	α_z [°]	t_x [mm]	t_y [mm]	t_z [mm]
laboratory	0.00173	0.00335	0.00041	0.0083	0.0061	0.0243
<i>scale factor</i>	<i>5</i>	<i>5</i>	<i>5</i>	<i>20</i>	<i>5</i>	<i>5</i>
inspection (I)	0.01521	0.01935	0.00328	0.166	0.0303	0.1214
in-situ	0.00865	0.01677	0.00205	0.124	0.0903	0.26907
<i>scale factor</i>	<i>1.5</i>	<i>1.5</i>	<i>1.5</i>	<i>2</i>	<i>1.5</i>	<i>1.5</i>
first responder (F)	0.02281	0.02903	0.00492	0.248	0.1355	0.4036

(c) IMU registration uncertainties \mathbf{T}_l^b

calibration	α_x [°]	α_y [°]	α_z [°]	t_x [mm]	t_y [mm]	t_z [mm]
standard	0.01785	0.0334	0.01733	1.0	1.0	1.0
<i>scale factor</i>	<i>5</i>	<i>5</i>	<i>5</i>	<i>1</i>	<i>1</i>	<i>1</i>
inspection (I)	0.08923	0.167	0.08663	1.0	1.0	1.0
<i>scale factor</i>	<i>15</i>	<i>15</i>	<i>15</i>	<i>3</i>	<i>3</i>	<i>3</i>
first responder (F)	0.26769	0.501	0.25989	3.0	3.0	3.0

Appendix B

Supplementary Experiments

This appendix presents additional experiments to complement various investigations from this thesis. First, a sensitivity analysis is used for parameter range selection for image noise and Gaussian blur of the combined sensitivity analyses to complement sections 7.2.2 and 8.3. Second, feature matching is evaluated with a more extensive experiment to complement Section 5.2.1. Third, the error propagation for feature undistortion is briefly investigated to complement Section 5.2.2. Fourth, a MCS is conducted to verify the error propagation in VO of Griebach (2015) with the presented WLS approach to complement Section 5.2.3. Finally, the segmentation results of a few more DNNs is evaluated to complement Section 8.2.3.

B.1 Sensitivity Analysis

Figure B.1 presents two additional sensitivity analyses that were used to select sample distributions for the parameters (i) *capture gain* and (ii) *gaussian blur* to prepare the combined sensitivity analyses (Section 7.2.2, 8.3). Each box comprises of 50 repetitions.

Image noise results in a stable localization up to 20 db. Interestingly, the coast dataset is less affected by noise, which might be attributed to the higher contrast in intensity values, observable in Figure 4.10 (p. 47) in comparison to Figure 4.8 (p. 45) and Figure 4.9 (p.46). Oriented on this data, the maximum range of the parameter

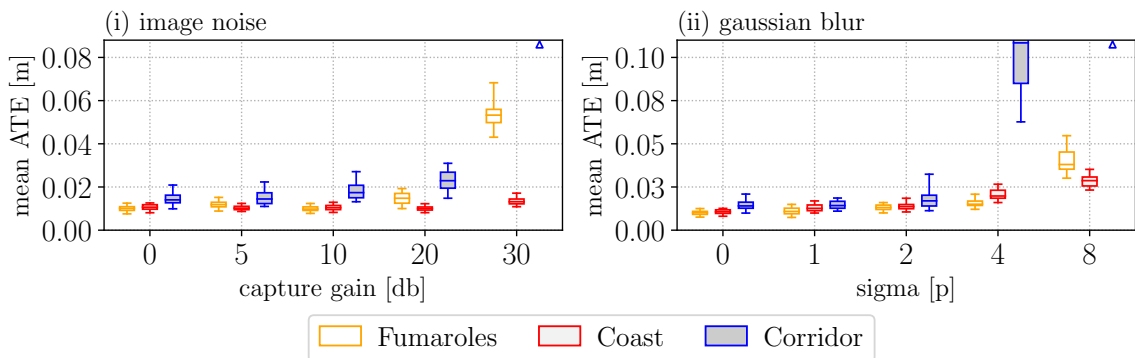


Figure B.1: Supplementary sensitivity analysis based on static simulated datasets.

capture gain for the combined sensitivity analyses was set to 20 db, its mean to 0 db and its SD to 5 db with the condition to be greater 0 db.

Gaussian blur results in a stable localization of up to 4 px. The corridor dataset is most affected, which can be attributed to the small number of detected features due to many existing homogeneous surfaces. Oriented on this data, the maximum range of the parameter Gaussian blur (sigma) was set to 5 px (which is optimistic for the corridor dataset), its mean to 0 px and its SD to 1 px with the condition to be > 0 px.

B.2 Feature Matching Evaluation

In this appendix, feature matching is evaluated with a more extensive experiment to complement Section 5.2.1. Table B.1 evaluates feature matching during intra $\hat{\mathbf{m}}^{r1\delta}$ and inter $\hat{\mathbf{m}}^{r2\delta}$ matching for different severity of motion blur (exposure time) and different severity of view angle changes (image frequency). The experiments are based on the synthetic datasets of Table A.1 without dynamic environment elements.

It can be observed that the error depends on the considered environment, increases with the level of motion blur and with the severity of view point changes. A particularity is the corridor dataset. It can be observed that the quality and number of features increase with the level of motion blur. The additional blurring seems to support the feature detection in the environment with fine repetitive structures on the carpet and wallpaper. This might explain the slightly improved localization solution for the corridor dataset, which can be observed in Figure 4.11 (right, p.48) at 20 ms.

Table B.1: Supplementary evaluation of feature matching errors (SD [px]) with respect to image frequency, synthetic datasets and motion blur. A white-to-red heat map visualizes the deteriorated matching error for each column. N denotes the number of features used to estimate the SD in each experiment.

		images at 10 Hz				images at 5 Hz					
		N	$\hat{m}_0^{r1\delta}$	$\hat{m}_1^{r1\delta}$	$\hat{m}_0^{r2\delta}$	$\hat{m}_1^{r2\delta}$	N	$\hat{m}_0^{r1\delta}$	$\hat{m}_1^{r1\delta}$	$\hat{m}_0^{r2\delta}$	$\hat{m}_1^{r2\delta}$
Fumarole	0 ms	102k	0.39	0.322	0.151	0.137	46k	0.38	0.32	0.176	0.166
	5 ms	101k	0.401	0.326	0.153	0.137	46k	0.388	0.321	0.181	0.169
	20ms	91k	0.438	0.344	0.287	0.295	38k	0.415	0.336	0.314	0.32
Coast	0 ms	42k	0.407	0.34	0.163	0.151	19k	0.386	0.333	0.192	0.183
	5 ms	42k	0.417	0.341	0.163	0.148	19k	0.395	0.334	0.197	0.185
	20ms	39k	0.454	0.349	0.292	0.286	16k	0.423	0.342	0.335	0.322
Corridor	0 ms	5.8k	0.479	0.358	0.199	0.189	2.8k	0.435	0.35	0.225	0.215
	5 ms	5.9k	0.465	0.357	0.193	0.18	2.8k	0.454	0.346	0.225	0.197
	20ms	6.2k	0.504	0.352	0.185	0.164	3.0k	0.472	0.347	0.232	0.188

B.3 Error Propagation for Feature Undistortion

This section complements Section 5.2.2 with an experiment to validate the use of 11×11 covariance matrices ($cov11x11$) instead of 2×2 covariance matrices ($cov2x2$)

during feature undistortion. Formulated in Equation 5.6, the three required steps are given with

$$\mathbf{m}^\delta \rightarrow \tilde{\mathbf{m}}^\delta \rightarrow \tilde{\mathbf{m}} \rightarrow \hat{\mathbf{m}} . \quad (\text{B.1})$$

The distorted image point \mathbf{m}^δ is transformed into normal camera coordinates $\tilde{\mathbf{m}}^\delta$, undistorted to $\tilde{\mathbf{m}}$ and transformed back into image coordinates $\hat{\mathbf{m}}$.

The notation for the error propagation is similar as in Section 5.2.2 with the addition of the distortion parameters δ . For instance, the first step of *cov2x2* is formulated as

$${}_{(2,2)}\Sigma_{\tilde{\mathbf{m}}^\delta} = {}_{(2,2)}\mathbf{J}_{\mathbf{m}^\delta} \cdot {}_{(2,2)}\Sigma_{\mathbf{m}^\delta} \cdot {}_{(2,2)}\mathbf{J}_{\mathbf{m}^\delta}^T + {}_{(2,4)}\mathbf{J}_{\boldsymbol{\kappa}} \cdot {}_{(4,4)}\Sigma_{\boldsymbol{\kappa}} \cdot {}_{(2,4)}\mathbf{J}_{\boldsymbol{\kappa}}^T . \quad (\text{B.2})$$

Related, the first step of *cov11x11* is formulated as

$${}_{(11,11)}\Sigma_{\tilde{\mathbf{m}}^\delta, \boldsymbol{\kappa}, \delta} = {}_{(11,11)}\mathbf{J}_{\mathbf{m}^\delta, \boldsymbol{\kappa}, \delta} \cdot {}_{(11,11)}\Sigma_{\mathbf{m}^\delta, \boldsymbol{\kappa}, \delta} \cdot {}_{(11,11)}\mathbf{J}_{\mathbf{m}^\delta, \boldsymbol{\kappa}, \delta}^T . \quad (\text{B.3})$$

The steps two and three are formulated accordingly.

Figure B.2 shows a MCS (Section 3.3.2) for the undistortion steps for one exemplary feature point. It is conducted to verify the analytically propagated uncertainties on the basis of the statistical propagation. It shows that after the first and the second step, all propagated uncertainty are the same. This is due to the assumption of independent calibration parameters, i.e., using only the SD of each parameters, which is also applied during the MCS. This topic is discussed in Section 5.4. Though, after the third step, the propagated uncertainties of *cov2x2* diver significantly to the MCS result. This is again due to the missing information that the uncertain camera model with parameters $\boldsymbol{\kappa}$ was already used in the first step, similar as explained in Section 5.2.2. Propagated uncertainties of *cov11x11* equal the MCS result and are therefore the better choice.

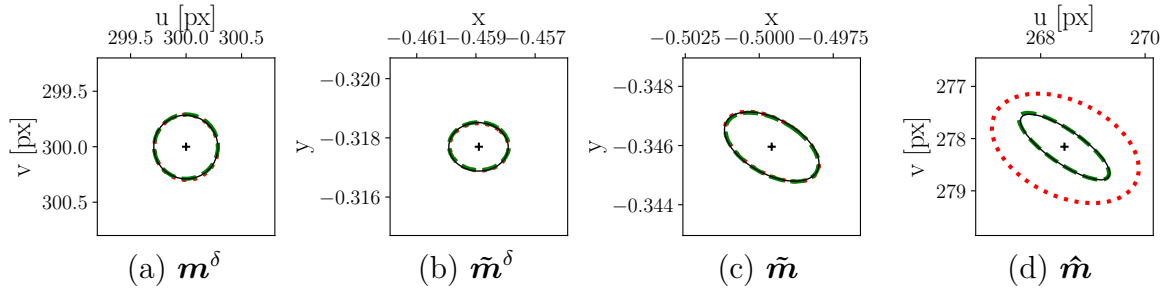


Figure B.2: Supplementary evaluation of error propagation for feature undistortion based on a MCS (black, solid), analytical method based on 2×2 covariance matrices (red, dotted), analytical method based on 11×11 covariance matrices (green, dashed).

B.4 Monte-Carlo-Simulation for Visual Odometry

In this section, the WLS approach (Section 5.2.3) is validated based on a MCS, in which noisy feature points are introduced. The experiment is based on a set of 20 GT object points $\{\vec{\mathbf{M}}_i\}_{i=0}^{19}$ and a GT transformation $\Delta \mathbf{T}$ with parameters $(t_x, t_y, t_z, \alpha_x, \alpha_y, \alpha_z) =$

(5cm, 5cm, 5cm, 5°, 5°, 5°). Their projections in all four camera images form the sets of GT feature points, given with $\{(\mathbf{m}_i^{l1}, \mathbf{m}_i^{r1}, \mathbf{m}_i^{l2}, \mathbf{m}_i^{r2})\}_{i=0}^{20}$.

During each iteration of the MCS, the parameters for the camera model are sampled based on their specific distributions (I, F , see Appendix A.3). Also, feature point values are noised with a SD of 0.2px during each iteration, which results in the observed feature sets $\{(\hat{\mathbf{m}}_i^{l1}, \hat{\mathbf{m}}_i^{r1}, \hat{\mathbf{m}}_i^{l2}, \hat{\mathbf{m}}_i^{r2})\}_{i=0}^{19}$. For a subset of feature sets (0%, 20%), the feature point values are noised with a SD of 0.8px to introduce noisy points. The relative transformation is estimated based on the observed set of feature points with knowledge of their error distribution. Camera distortion parameters δ are not considered.

Table B.2 shows the mean RTE over all 1000 iterations for the proposed WLS approach and the original least-squares (LS) method. As to be expected, the results do not differ for 0% noisy points. However, WLS outperforms LS in the presence of 20% noisy points. Besides, propagated uncertainties (based on methods of Grieffbach, 2015) of this experiments could be verified using this MCS, which is not mentioned in the table.

Table B.2: Supplementary evaluation of WLS. It shows the mean RTE [mm] and SD (\pm) from 1000 iterations of a MCS for one VO estimation in the presence of noisy points (0%, 20%). Bold numbers mark the best results of each comparison.

Calib. Setting:		(I)	(F)	
Noisy [%]	0	WLS	0.40±0.19	0.69±0.37
		LS	0.41±0.19	0.69±0.38
	20	WLS	0.52±0.25	0.87±0.47
		LS	0.76±0.38	0.94±0.5

B.5 Semantic Segmentation

This section provides a few more results to the training of DNN based on the Sensor-AI approach and complements Section 8.2.3. Table B.3 shows the mean IoU for the different classes of the network *all-I*, which was trained on the full dataset that was generated with calibration I . It shows that *good* achieves the highest score. This might be due to its dominant presence in the image. This might reduce the influence of pixel-level label noise, which was observed in the data (Section 8.2.3).

Table B.3: Supplementary evaluation of the trained DNN *all-I*.

class	fumaroles	coast	river	mean
good	0.79	0.77	0.73	0.76
bad	0.45	0.44	0.60	0.50
background	0.43	0.50	0.44	0.46
mean	0.55	0.57	0.59	0.57

Table B.4: Supplementary segmentation results (calib. I), showing the mean IoU.

name	images	network	fumaroles	coast	river	mean
all	4102	Mobilenetv2	0.55	0.57	0.59	0.57
all-1/2	2051	Mobilenetv2	0.55	0.55	0.58	0.56
all-1/4	1025	Mobilenetv2	0.53	0.54	0.56	0.54
all-xception	4102	Xception	0.55	0.57	0.58	0.57
all-mbv3	4102	Mobilenetv3	0.55	0.56	0.59	0.56
all-augmented	4102	Mobilenetv2	0.42	0.42	0.31	0.38

Table B.4 lists the results of some more exemplary selected trained DNNs. Networks $all-1/2$ and $all-1/4$ are trained with fractions of the dataset. This experiment shows that reducing the training dataset size decreases the mean IoU gradually, which in turn suggests that more data could increase the mean IoU. Networks $all-xception$ and $all-mbv3$ are based on different network structures. This experiment shows that training DNNs based on other structures does not change the outcome much, which strengthens the conclusion that the performance is limited by the training data, due to limited amount of reference images and relatively high pixel-level label noise in the data. Network $all-augmented$ was trained on an augmented version of the dataset, which was augmented in terms of image noise, image blur, distortion effects and similar. This experiment exemplifies that data augmentation can lead to a strong drop in the mean IoU score. This indicates that, for instance, blur or noise might provide important clues for certain objects, such as smoke or brushwood, within the Sensor-AI approach.

Figure B.3 shows exemplary predictions of $all-I$ in other environments. The corresponding runs were not included in the training dataset, but the same places were seen for (a,b) and the same environment was seen for (d). (a) shows dark sand at Valle dei Mostri that shows a similar structure as water and is partly labeled as bad by the DNN. (c) shows a river site that was not included in the data and is partly mislabeled, possibly due to water structures and appearances that are unknown to the DNN.

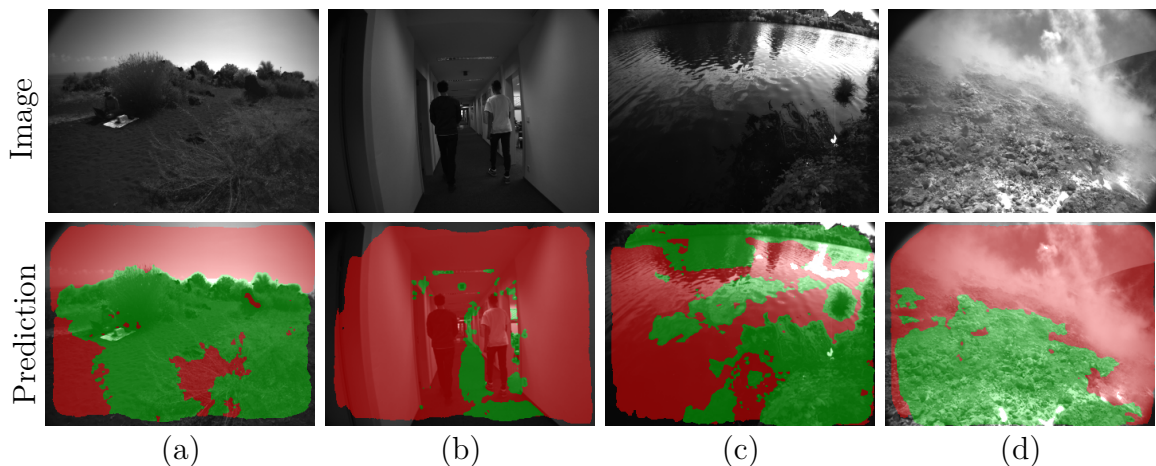


Figure B.3: Supplementary qualitative segmentation results based on $all-I$.