DLR-IB-RM-OP-2022-7

Development of a Front-End Module for a Decentralized Multi-Modal SLAM Framework in the Domain of Mobile Robotics

Master's Thesis

Xiaozhou Luo

Deutsches Zentrum
DLR für Luft- und Raumfahrt

Entwicklung eines Front-End Moduls für ein dezentrales multimodales SLAM Framework im Bereich der mobilen Robotik

Development of a Front-End Module for a Decentralized Multi-Modal SLAM Framework in the Domain of Mobile Robotics

MMVO

Scientific work for obtaining the academic degree
Master of Science (M.Sc.)
at the Department Mobility Systems Engineering
of the TUM School of Engineering and Design
at the Technical University of Munich

Supervised by

Prof. Dr.-Ing. Markus Lienkamp
Florian Sauerbeck, M.Sc.
Chair of Automotive Technology

Marco Sewtz, M.Sc.
Institute of Robotics and Mechatronics
German Aerospace Center (DLR)

Submitted by

Xiaozhou Luo, B.Sc.
Margarete-Schütte-Lihotzky-Str. 16a
80807 München

Submitted on November 23, 2022

Chair of Automotive Technology
TUM School of Engineering and Design
Technical University of Munich

Projektbeschreibung

Entwicklung eines Front-End Moduls für ein dezentrales multimodales SLAM Framework im Bereich der mobilen Robotik

Am Institut für Robotik und Mechatronik (RMC) des Deutschen Zentrum für Luft- und Raumfahrt (DLR) wird ein dezentraler SLAM (Simultaneous Localization and Mapping) Algorithmus entwickelt. Es handelt sich um ein System, das sowohl eine Vielzahl von Daten gleicher Sensormodalität als auch Messungen von verschiedenen Sensortypen fusionieren soll.

Im Rahmen dieser Arbeit soll ein Visual Odometry (VO) Front-End Modul für das System entwickelt werden, das verschiedene Modalitäten im Software- und Hardwarebereich berücksichtigen soll. Zunächst soll das Entwicklungspotential anhand der Roboterspezifikation analysiert werden. Hier gilt es in Zusammenhang mit dem aktuellen Stand der Technik die vielversprechendsten Methodiken und Entwicklungsrichtungen herauszuarbeiten. Anschließend wird ein multimodaler VO Algorithmus konzeptionalisiert und implementiert. Dabei liegt das Hauptaugenmerk auf dem Zusammenspiel unterschiedlicher Software Modalitäten im Bereich der Feature Extraction. Damit sollen Synergien zwischen verschiedenen Featuretypen geschaffen werden, um die Qualität der Datenpunkte für die anschließende Bewegungsschätzung zu verbessern. Abschließend sollen die neuen Methoden analysiert und quantifiziert werden. Die Ergebnisse sollen mit bestehenden Datensätzen validiert und anschließend dokumentiert werden.

Folgende Arbeitspakete umfasst die zu vergebende Studienarbeit:

- Einarbeitung in VO und visual SLAM
- Analyse des Entwicklungspotentials und der verfügbaren Methodiken
- Entwicklung von Methoden für die Kombination unterschiedlicher Modalitäten
- Implementierung des multimodalen VO Front-End Moduls
- Validierung und Dokumentation der Ergebnisse

Die Ausarbeitung soll die einzelnen Arbeitsschritte in übersichtlicher Form dokumentieren. Der Kandidat verpflichtet sich, die Studienarbeit selbständig durchzuführen und die von ihm verwendeten wissenschaftlichen Hilfsmittel anzugeben.

Die eingereichte Arbeit verbleibt als Prüfungsunterlage im Eigentum des Lehrstuhls.

Ausgabe: 23. Mai 2022

Abgabe: 23. November 2022

Prof. Dr.-Ing. Markus Lienkamp

Florian Sauerbeck, M.Sc.

Chair of Automotive Technology
TUM School of Engineering and Design
Technical University of Munich

Geheimhaltungsverpflichtung

Herr: **Luo, Xiaozhou**

Gegenstand der Geheimhaltungsverpflichtung sind alle mündlichen, schriftlichen und digitalen Informationen und Materialien die der Unterzeichner vom Lehrstuhl oder von Dritten im Rahmen seiner Tätigkeit am Lehrstuhl erhält. Dazu zählen vor allem Daten, Simulationswerkzeuge und Programmcode sowie Informationen zu Projekten, Prototypen und Produkten.

Der Unterzeichner verpflichtet sich, alle derartigen Informationen und Unterlagen, die ihm während seiner Tätigkeit am Lehrstuhl für Fahrzeugtechnik zugänglich werden, strikt vertraulich zu behandeln.

Er verpflichtet sich insbesondere:

- derartige Informationen betriebsintern zum Zwecke der Diskussion nur dann zu verwenden, wenn ein ihm erteilter Auftrag dies erfordert,
- keine derartigen Informationen ohne die vorherige schriftliche Zustimmung des Betreuers an Dritte weiterzuleiten,
- ohne Zustimmung eines Mitarbeiters keine Fotografien, Zeichnungen oder sonstige Darstellungen von Prototypen oder technischen Unterlagen hierzu anzufertigen,
- auf Anforderung des Lehrstuhls für Fahrzeugtechnik oder unaufgefordert spätestens bei seinem Ausscheiden aus dem Lehrstuhl für Fahrzeugtechnik alle Dokumente und Datenträger, die derartige Informationen enthalten, an den Lehrstuhl für Fahrzeugtechnik zurückzugeben.

Eine besondere Sorgfalt gilt im Umgang mit digitalen Daten:

- Für den Dateiaustausch dürfen keine Dienste verwendet werden, bei denen die Daten über einen Server im Ausland geleitet oder gespeichert werden (Es dürfen nur Dienste des LRZ genutzt werden (Lehrstuhllaufwerke, Sync&Share, GigaMove).
- Vertrauliche Informationen dürfen nur in verschlüsselter Form per E-Mail versendet werden.
- Nachrichten des geschäftlichen E-Mail Kontos, die vertrauliche Informationen enthalten, dürfen nicht an einen externen E-Mail Anbieter weitergeleitet werden.
- Die Kommunikation sollte nach Möglichkeit über die (my)TUM-Mailadresse erfolgen.

Die Verpflichtung zur Geheimhaltung endet nicht mit dem Ausscheiden aus dem Lehrstuhl für Fahrzeugtechnik, sondern bleibt 5 Jahre nach dem Zeitpunkt des Ausscheidens in vollem Umfang bestehen. Die eingereichte schriftliche Ausarbeitung darf der Unterzeichner nach Bekanntgabe der Note frei veröffentlichen.

Der Unterzeichner willigt ein, dass die Inhalte seiner Studienarbeit in darauf aufbauenden Studienarbeiten und Dissertationen mit der nötigen Kennzeichnung verwendet werden dürfen.

Datum: 23. November 2022

Unterschrift: _____

Chair of Automotive Technology
TUM School of Engineering and Design
Technical University of Munich

Erklärung

Ich versichere hiermit, dass ich die von mir eingereichte Abschlussarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Garching, den 23. November 2022

_____

Xiaozhou Luo, B.Sc.

Chair of Automotive Technology
TUM School of Engineering and Design
Technical University of Munich

Declaration of Consent, Open Source

Hereby I, Luo, Xiaozhou, born on April 15, 1997, make the software I developed during my Master Thesis available to the Institute of Automotive Technology under the terms of the license below.

Garching, November 23, 2022

_____

Xiaozhou Luo, B.Sc.

Contents

List of Abbreviations

ASIMO	Advanced Step in Innovative Mobility
aSLAM	acoustic SLAM
ASR	automatic speech recognition
BA	bundle adjustment
BRIEF	Binary Robust Independent Elementary Features
CenSurE	Center Surround Extrema
COTS	commercial off-the-shelf
CPU	central processing unit
DHCP	Dynamic Host Configuration Protocol
DLR	German Aerospace Center
DoA	directions of arrival
DoF	degrees of freedom
DoG	Difference of Gaussian
DSO	Direct Sparse Odometry
DTAM	Dense Tracking and Mapping
EKF	extended Kalman filter
EtherCAT	Ethernet for Control Automation Technology
FAST	Features from Accelerated Segment Test
FEF^2	Feature Entity Fusion and Filtering
FLANN	Fast Library for Approximate Nearest Neighbors
FoV	field of view
FPS	frames per second
GFTT	Good Features To Track
GNSS	global navigation satellite system
HCD	hand-held camera device
IGEL-SLAM	Indoor Optimized, Globally Consistent, Environment-Aware, Longlife-SLAM
IMU	inertial measurement units
IndoorMCD	Indoor Multi-Cam Dataset
IPC	interprocess communication
ISRU	In Situ Resource Utilization
JPL	Jet Propulsion Laboratory
LB	line band
LBD	Line Band Descriptor
LiDAR	light detection and ranging
LoG	Laplacian of Gaussian
LoT	loss of tracking
LSD	Line Segment Detector
LSD-SLAM	Large-Scale Direct monocular SLAM

LSR	line support region
LWR	Light Weight Robot
MARVIn	Mock-up Platform for Audio Research and Vision on Rollin' Justin
MER	Mars Exploration Rovers
ML	machine learning
MMVO	Multi-Modal Visual Odometry
MROSLAM	Multi-camera Robust ORB-SLAM
MSCKF	multi-state constrained Kalman filter
NASA	National Aeronautics and Space Administration
NTP	Network Time Protocol
oFAST	oriented FAST
OKVIS	Open Keyframe-based Visual-Inertial SLAM
ORB	Oriented FAST and Rotated BRIEF
PCB	printed circuit board
PnP	Perspective-n-Point
PoI	pixel of interest
PTAM	Parallel Tracking and Mapping
RANSAC	RANdom SAmple Consensus
rBRIEF	rotated BRIEF
RMSE	root mean squared error
ROVIO	Robust Visual Inertial Odometry
RPE	relative pose error
SDK	software development kit
SfM	structure from motion
SIFT	Scale-Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
SMiLE	Servicerobotik für Menschen in Lebenssituationen mit Einschränkungen
SSL	sound source localization
SSS	sound stream separation
SURF	Speeded-up Robust Features
SVO	Semidirect Visual Odometry
USB	Universal Serial Bus
VCSEL	vertical-cavity surface-emitting laser
VO	Visual Odometry
WLAN	wireless local area network
WVGA	Wide Video Graphics Array

Formula Symbols

Formula Symbols	Unit	Description
Δ	-	Time interval (the number of frames) in which the relative pose error (RPE) of a trajectory is measured
λ_i	-	Eigenvalue i of the auto-correlation matrix of the Shi-Tomasi criteria
m	-	Number of individual relative error parameters
n	-	Number of poses
$n_{\boldsymbol{P}_k}$	-	Number of estimated camera poses within the image sequence k
$n_{\boldsymbol{Q}_k}$	-	Number of time-synchronized ground truth poses within the image sequence k
n_d	-	Number of the features extracted by the detector d in the specific image region
$\boldsymbol{P}_i$	-	Estimated camera pose i
$\boldsymbol{Q}_i$	-	Ground truth pose i
$\widetilde{r_d}$	-	Scaling factor for features extracted by the detector d
R	-	Response value of the Shi-Tomasi criteria
RPE_i	m/s	Relative pose error at time step i
$S_{c,IACC}$	-	Cell-specific significance value generated by the intra-class feature collaboration
S_f	-	Significance value of feature f
$S_{f,IACC}$	-	Feature-specific significance value generated by the intra-class feature collaboration
$S_{f,IRCC}$	-	Feature-specific significance value generated by the inter-class feature collaboration
w_d	-	Weighting factor of the detector d

1 Introduction

1.1 Motivation

In recent times, human spaceflight beyond low earth orbit has gained more and more interest. Nearly 30 years after the last human being departed from another celestial body, the National Aeronautics and Space Administration (NASA) announced with their Artemis program the return of human beings to the Moon in 2024 [1]. As the first step with the ultimate goal of crewed exploration of Mars, the Earth-moon resembles the ideal proving ground for future planetary exploration missions. However, crewed spaceflight is a very complex and therefore expensive subject, as the utilized spacecraft and equipment have to obtain a human-rating certification. Even in recent times, together with these precautionary measures, it still bears a substantial amount of risks, as well as psychological and physiological challenges for the astronauts. Human-built exploration technology has not yet reached self-sustained operation capability and is still dependent on regular resupplies from the Earth. Since the turn of the millennium, there has been considerable interest in In Situ Resource Utilization (ISRU), which would guarantee independent operation by manufacturing necessary materials from resources at the mission site. Unfortunately, this technology is still in development and has not been implemented in a space-related mission for material production. While the first technology demonstrators are planned in the next few years on the Moon with the goal of producing water or breathable oxygen before 2025 [2], it would still take decades to reach mandatory technology readiness. Therefore, especially in cases of long-running missions and exploration of extraterrestrial bodies, the utilization of robots would significantly increase the possible range and duration of the operation. It boosts the overall exploration capability and still provides the best cost-to-scientific-benefit ratio in astronautics.

Apart from standalone research and exploration duties, robotic systems can also provide assistance to crewed missions in collaboration with human operators. In this case, tasks with higher uncertainties and risks could be assigned to them, as they are more robust against the hazardous environment. In addition, the loss of technical equipment is more acceptable than potential injuries to crew members up to the possibility of crew loss. While the capabilities of robotic systems are steadily growing with the first ones, e.g., Boston Dynamics' Spot, reaching commercial viability, they are still mainly dependent on a human operator due to their lack of autonomy. First steps in making them more independent can be seen with Boston Dynamics' latest iteration of their humanoid walking robot Atlas. In addition, Tesla's recent announcement of the development of a humanoid assistance robot further increased the attention on the public side. Aside from the fast-growing commercial sector, considerable research is also being conducted at universities and research facilities like the German Aerospace Center (DLR). At the Institute of Robotics and Mechatronics, Surface Avatar and SMiLE2gether, which is derived from the German expression "Servicerobotik für Menschen in Lebenssituationen mit Einschränkungen", are just two examples of projects in the field of humanoid service and assistance robots. In

the case of Surface Avatar, the focus is mainly on supervised teleoperation with autonomous robots and collaboration between different robotic applications in a martian surface exploration setting. On the contrary, SMiLE explores the possibility of utilizing robotic technology in the field of health- and elderly care. Its surroundings are pretty modest compared to the planetary exploration scenario in Surface Avatar and resemble a typical housing environment on Earth.

In order to improve autonomous robot operation, the ability to accurately perceive its surroundings is crucial for the reliability and robustness of the system. Therefore, one of the main requirements is to provide a robust and accurate working localization and perception framework to establish spatial awareness at all times. Especially in densely populated areas and in collaboration with on-site human operators and other participating robots, establishing situation awareness is essential to ensure operational safety. In recent years, machine perception algorithms have become more sophisticated and are already able to provide estimates with sufficient accuracy. Nevertheless, these applications are still under development and associated with many unresolved challenges. This includes the major issue surrounding the performance and robustness of these primarily image-based approaches since it is highly dependent on the overall environmental conditions and the texture of the surrounding surfaces. Alongside the qualitative aspect, these methods are still very computationally intensive, which poses an additional challenge, especially in the area of mobile systems without external processing capabilities.

1.2 Research Questions

In this work, a robust and efficient front-end module responsible for tracking and short-term localization tasks is developed as part of a novel perception framework. While doing so, the following research questions are central to the development and conceptualization process:

- How can different types of sensors and methods within the processing pipeline for a tracking system be effectively and efficiently combined to create a multi-modal perception module on both the hardware-specific area with a special focus on visual-inertial sensors and the software-related domain?
- What is the best strategy for optimizing and distributing the demanded computing power to meet the real-time processing capability requirements on mobile platforms? How can such a system be realized and implemented?
- How does the new approach's performance and robustness compare to other state-of-the-art methods in generalized and application-related environments?

1.3 Structure of the Work

At first, Chapter 2 focuses on the theoretical background of the thesis, which forms the foundations of the considerations and developments within the following chapters. Based on the available hardware setup and our targeted field of application introduced in the previous segment, the next chapter reflects on the conceptualization process within the development of our targeted front-end module.

While novel methods for establishing multiple modalities in the hardware and software-related domain are proposed in Chapter 3 within a theoretical context, these approaches are further elaborated in Chapter 4 and integrated into the processing pipeline of the tracking system.

Following the conceptualization and implementation tasks, an experimental evaluation is conducted in Chapter 5, in which the performance of the multi-modal front-end module is analyzed in greater detail. In the end, the thesis is rounded up by the critical assessment of the proposed and implemented methods in Chapter 6 and concluded in Chapter 7.

2 Theoretical Background

This chapter introduces the audience to the subject surrounding sensation and perception to create machine perception with a particular focus on tracking, localization, and mapping targeting the creation of spatial awareness. Starting from the fundamentals in Section 2.1, we clarify the terms of sensation and perception in the human context and technical realization. In the next step, perception is subdivided into different types based on the correlations utilized in human psychology, which also find use in machine perception. Section 2.2 introduces the robotic platform and available types of sensory, which serves as the hardware reference to the development within the thesis. Inspired by this human ability, researchers have transferred and reconstructed this skill into the man-made domain. At last, individual branches of machine perception and their current state of research is examined in Section 2.3.

2.1 Fundamentals of Machine Perception

Already since the beginning of engineering, researchers and scientists have drawn inspiration from nature. While evolutionary processes in biology are relatively slow, the final results are well adapted to the requirements and very frugal in the case of resource utilization. Thus, researchers have also adopted the human perception and sensation process, which results in robotic sensing and machine perception. While the general term and its technical realization have already been broached in the introduction, it remains still somehow unclear.

2.1.1 General Aspects

In general, machine perception is the comprehensive term for the capability of an artificial system to interpret data collected by sensors in a manner that is similar to the way human use their senses and the abilities to relate to the world around them. Before getting into details, it is necessary to clarify the differentiation between sensation and perception since there is a general misconception between them and their covered areas in psychology and neuroscience, as well as in their technical counterpart.

The likelihood of confusion between the terminologies already starts with the origin of the term perception, which is derived from the Latin word *perceptio*. Initially, it resembles the meaning of collection and can also be interpreted as comprehension figuratively. Merely from the linguistic perspective, it unites these two steps of information processing.

Returning to the fields of psychology and neuroscience, the human perceptual process consists of a sequence of individual processes that interact with each other in order to determine our experience of and reaction to stimuli in the environment [3]. Hereby, one would inevitably stumble upon perception and sensation, where the latter focuses on the process of receiving and gathering data through human senses. Environmental stimuli are captured by the internal

and external sensory organs and transduced into an electric signal with the help of receptor cells. With this, a first abstraction step of the natural world is made, and the collected stimuli are then forwarded to the central nervous system for further processing. In short terms, sensation resembles simple awareness due to the stimulation of a sensory organ [4]. Each of the individual fields is fully independent, and information is collected in a strictly separated manner. Thus, there is no interrelation between the gathered data at this point, and each branch contributes to the creation of awareness in a unique way.

On the contrary, perception resembles a subordinate process after sensation and is responsible for the processing and interpretation of the individual pieces of received data. In general, the human body can be classified as a centralized system in which external information is collected by individual interfaces and forwarded to the spinal cord and brain for further processing. The received information is organized, identified, and interpreted to form a mental representation [4]. As a result, a higher-level general view is generated, which contributes, among other things, to the creation of situation awareness and spatial orientation. Apart from the direct sensory inputs from the five human senses, other psychological processes and social aspects, e.g., speech and face perception, are also considered. Unlike the strict sensory separation in sensation, perception is not limited to utilizing information gathered by one sensory modality. By combining different sensation types and psychological processes, new multi-modal cognitive branches can be constructed. Chronoception, for example, is not directly related to a specific sensory branch. Instead, it is a complex involving different psychological processes and areas of the brain, with the target of perceiving duration and time.

2.1.2 Types of Sensory Perception

In the field of machine perception, the majority of classic approaches without consideration of machine learning and the deployment of neural networks has the target to recreate direct perception mechanisms. Therefore, each of the five individual branches of external sensation eventually results in a particular field of perception. However, these mechanisms are not equally treated by the human body, which is also reflected in the individual research branches within machine perception. In consultation with linguistic research, they can be ordered in a hierarchical arrangement of three levels [5, 6].

Visual Field

Starting from the top, vision is ranked as the primary human sense. External signals are collected via the eye and transduced into electrochemical signals by two types of photosensitive cells on the retina. With this, three different groups of cone neurons achieve the ability to receive color-related information. In contrary, rod cells are more light-sensitive and can only collect information in a monochromatic sense. For this reason, they are almost entirely responsible for vision under poor lighting conditions. At this stage, some of the gathered data is already pre-processed directly within the neurons. The generated neural impulses are then collected through the different retina layers and transmitted via optical nerves to the brain, where perception mainly occurs in the cerebral cortex [4]. While the sensory stimuli are caused by electromagnetic waves in the case of vision, the human eye can only sense a tiny spectrum. In the human case, it typically ranges from ultraviolet to infrared between $380\,\mathrm{nm}$ and $750\,\mathrm{nm}$ in wavelength.

Although the entire process of visual perception is a complex and comprehensive subject, it is also the most researched field based on its significance in the overall structure of perception. This trend is also continuing in its technical counterpart, and machine vision is the most mature research domain within the field of machine perception. Thus, visual images have long been utilized for several purposes, as it provides a significant amount of information. Similar to the human eye, cameras are also classified as passive sensing devices. Therefore, they do not suffer from interference often encountered with active sensors, e.g., ultrasonic or laser-based devices [7].

For this reason, robotics and computer vision researchers have targeted visual mobile robotic localization and perception for decades. Especially in recent times, there has been a growing interest in visual-based systems since it provides a robust and cost-efficient alternative to infrared sensors and laser scanners. While localization methods like multi-band high-precision global navigation satellite system (GNSS) systems are already able to achieve accurate position measurements with centimeter-level accuracy under open sky conditions [8], they depend on existing infrastructure. In addition, they are not available in various scenarios, particularly in the indoor domain. With the turn of the century, perception frameworks using information collected by passive imaging sensors have gained more interest in robotics. Thus, a cost-efficient system can be constructed using passive sensors instead of active ones, such as light detection and ranging (LiDAR) and laser sensors. In addition, those camera systems are lightweight and comparably simple in design. For this reason, they can be more easily integrated on mobile platforms, even in large quantities. As a result, machine vision is utilized as the primary system for perceptive and cognitive tasks on most robotic applications and the first choice to create a perceptive framework.

Auditory Perception

A step down the hierarchy, the second level contains the ability to perceive sound by detecting vibrations through the air. This mechanism is also known as auditory perception, where the ears are utilized as the sensation tool. Starting from the outer ear, it is responsible for collecting and preliminary filtration of incoming sound waves. The pressure waves are then translated into mechanical oscillation in the middle ear, where also impedance matching is performed. This step has great significance because the acoustic impedance between the ambient air and the fluid in the inner ear has to be resolved for optimal connection and transmission capabilities. In the inner ear, precisely the Cochlea, hair cells transduced incoming oscillations into neural signals. It is then forwarded to the auditory cortex within the brain's temporal lobe, where primary and higher functions in hearing take place. The signal is passed down to the cerebral cortex for further processing, especially the creation of auditory perception. Apart from basic tasks of receiving plane auditory information, the perception of sound also involves more complex tasks, e.g., the separation of superimposed input data, their identification, and the estimation of the distance and direction of their associated sources. This is realized by the arrangement of the hearing apparatus with two separate input sources and higher-level automatism. Typically, frequencies between $20\,\mathrm{Hz}$ and $20\,000\,\mathrm{Hz}$ are detectable for the human ear. With increasing age, the hair cells for higher frequencies tend to fade out in contrast to the lower boundary, which hardly shows any signs of wear and tear.

While the research community directed its main focus towards the visual branch in the past decades, developments were also made in the direction of other perception types. Especially with the progress in computer science since the turn of the millennium, there has been an increasing interest in auditory research. However, it is still a niche in scientific research and development, as the auditory branch plays a subordinate role in the human perception beneath vision.

In the case of humanoid robots, it is to be expected that robot audition, which represents the field of machine hearing in robotics, facilitates capabilities similar to human ones. Therefore, only passive systems for auditory perception without utilizing an active emitter, e.g., ultrasonic sensors, are considered here. While the research within this field was mainly focused on human speech processing and understanding in the past, the comprehension of auditory scenes, in general, is receiving increasing attention. Also referred to as auditory scene analysis, it consists of three different domains, which are comprised of sound source localization (SSL), sound stream separation (SSS), and automatic speech recognition (ASR) [9]. In terms of scientific research, one would inevitably stumble upon the Honda Research Institute, the Imperial College London, and the Institute of Robotics and Mechatronics at DLR. Here, the main focus is on techniques for SSL and SSS, including beamforming, separation of superimposed signals, and voice enhancement. Apart from the rising capabilities of theoretical research with stationary microphone setups, only a handful of robotic systems are equipped with the necessary hardware for auditory perception. A notable application is the Advanced Step in Innovative Mobility (ASIMO) humanoid robot developed by Honda at the beginning of the 21st century [10]. It is equipped with a total of eight microphones, which are evenly distributed on the left and right-hand side of the head, resembling the location of human ears [11]. The ability of auditory scene analysis is provided by the proprietary robot audition framework HARK [12], which is still the most popular and powerful open-source robot audition software. With its help, ASIMO is able to perceive and interact with its surroundings through the combination of SSL, SSS, ASR, and miscellaneous higher-level functions.

Haptic, Gustatory, and Olfactory Perception

The three remaining perception mechanisms are equally arranged at the bottom level, where no further division is made. Haptic perception is the recognition of objects through experienced forces. As a result, external stimuli are transduced by somatosensory receptor cells in the skin and then forwarded to the brain for further processing and creating perceptual awareness. Gustatory perception is the sensory system partially responsible for taste perception by utilizing the tongue and parts of the oral cavity as the primary sensation tool. The perception of taste can be explained as the reaction between arriving external stimulus and the gustatory receptor cells on the taste buds, which are concentrated at the upper side of the tongue. However, the gustatory process can only partly construct the cognition of taste since only five different types of flavor consisting of sweetness, bitterness, sourness, saltiness, and the recently added umami can be discovered. The remaining part is contributed by the olfactory system, which utilizes the nose and nasal cavity for sensing. In general, olfactory perception is the process of the absorption of volatile molecules through the nose. By surpassing the first layer of the nasal mucous membrane, the odor encounters many cilia, which are directly connected to the individual olfactory sensory neurons. The stimulus is then transduced into an electrochemical signal through the olfactory system and transmitted to the olfactory bulb in the vertebrate forebrain for perceptive tasks.

In terms of environmental modeling, localization, and navigation, these mechanisms are only of a subordinate role. To be more precise, only olfactory perception can contribute to the overall process in a meaningful way from a theoretical perspective. Following vision and audition, machine olfaction is another significant sensory perceptual system that bears great potential for future developments. While the earliest research can be dated back to the 1960s [13], the field of olfaction has been underrated in the past. Thus, it has not received much attention in the research community, and a great majority of the topic remains unexplored.

The sensory device for this type of machine perception can be summarized as the term electric nose, which consists of a tool for primary sensing duties followed by an intelligent recognition module [14]. Unlike the sensation step, which is typically accomplished by an array of gas sensors, the subsequent processing steps are the more challenging. At this point, a considerable number of methods from the fields of statistical pattern recognition, neural networks, chemometrics, machine learning (ML), and biological cybernetics has to be utilized for processing incoming data from the sensor array [15].

In the case of olfactory perception, the solitary analysis of the air composition in the surroundings is insufficient. The interpretation and recognition of the odor provide the essential foundations in which the recognition framework has to be trained with carefully selected training samples. However, the entire system is still very much in research, where the major efforts have been directed towards classifying and recognizing gasses and odors so far. In contrast, the field of odor characterization is left unattended. This is a limiting factor for the development of qualification and quantification of odor properties. There is no universal agreement about a general theory that would be sufficient to depict the relation between odorants and odor quantities [16]. Unlike in the visual field, where the entire visible spectrum can be represented as a combination of three elementary colors, there is still no theory that can describe odors in a more general way.

Up to now, machine olfaction has already been applied to different fields within the foodstuff industry [17, 18] and environmental detection [19]. As a result, the primary advantage is the ability to sense odorants and odorless volatile chemicals without linguistic interference. Moreover, the entire assembly is very compact and can provide an instantaneous response.

2.2 Robotic Platform and Hardware Architecture

Apart from theoretical considerations, technical platforms must be equipped with suitable hardware to explore and utilize characteristics in their surroundings for tracking, localization, and mapping. Thus, a robotic system, ideally equipped with multiple classes of perceptual sensors and sufficiently powerful computation hardware, has to be selected as a reference for the development of an environmental modeling and perception system. At the Institute of Robotics and Mechatronics, a large number of robotic systems are equipped with state-of-the-art perception sensors. Since our field of research is primarily directed towards service robotics in the indoor domain, the Rollin' Justin [20] platform is the ideal hardware reference for the following development within the thesis.

2.2.1 General Robotic Architecture

As illustrated in Figure 2.1, Rollin' Justin is a research platform in service robotics, which was first introduced to the public in 2008. Resembling a human-like shape, it has roughly the size of a human adult, with $1.91\,\mathrm{m}$ in height and approximately $200\,\mathrm{kg}$ in weight. It is equipped with many individually controllable joints, resulting in 51 actuated degrees of freedom (DoF). Thus, it allows the robotic system to pursue several goals simultaneously while complying with a given task hierarchy.

In general, Rollin' Justin can be divided into two sections: the mobile platform at the bottom and the upper body system. Justin's base platform contains most of the computation hardware and is equipped with four retractable legs, at which wheels with individual hub motors are attached to omnidirectional turnable hinges. It provides a sturdy foundation for the tasks carried out by the torso while retaining the optional capability of reducing the areal footprint, in case it is necessary, at the same time. While doing so, the robot occupies an area of roughly $0.80\,\mathrm{m}^2$ in the extended state, which can be reduced to $0.35\,\mathrm{m}^2$ when entirely retracted [20]. Especially in an environmental setting where space is a limiting factor, and the necessity of overcoming narrow passages has to be considered, e.g., in the household working environment, it is a very convenient feature. Therefore, it significantly increases the robot's possible operation range and field of application. In addition, the mechanism is designed so that its state of operation, regardless of which expansion state it is currently situated, does not affect the base's general location, especially the height. The upper torso is mounted on top of the base platform. Apart from the humanoid head assembly, it contains a Light Weight Robot (LWR) in its third generation combined with a second-generation DLR hand (Hand II) on either side. Endowed with 43 DoF in total, the upper body provides the necessary flexibility, making him the ideal research platform for sensitive ambidextrous manipulation.

As an entire robotic platform, Rollin' Justin is able to complete complex assignments in various fields of application autonomously. Starting from basic household duties like floor-sweeping or window-cleaning, the complexity of the tasks ranges from relatively simple exercises with a small number of involved objects over more demanding duties, e.g., pouring water from a bottle, up to highly dynamic assignments of catching a ball mid-air or juggling exercises.

Designed as a humanoid service and assistance robot, it is outfitted with numerous collision sensors on the outer edges of the chassis and torque sensors in almost all controllable joints. Therefore, Rollin' Justin provides the necessary hardware to be safely manipulated by an operator nearby and minimizes potential risks of injuries in collaboration with a human or another robot.

Figure 2.1: Humanoid robot Rollin' Justin.

Over the years, Rollin' Justin has evolved into a universal multipurpose platform on which a large variety of scientific research is conducted. This includes, among others, research efforts in the field of human-machine interaction, teleoperation, perception and 3-D-scenery recreation, and autonomous robot operation, including navigation, path planning, and collision avoidance. The robotic platform has been refined with each development cycle during its persisting lifespan, including subsequent upgrades with state-of-the-art technologies. Especially in the field of sensation and environmental modeling, significant advancements have been made since the initial rollout. As a result, it is at the current state also the research platform at the Institute of Robotics and Mechatronics that features the highest quantity and variety of perceptual sensors.

2.2.2 Perception Sensors for Localization and Mapping

In the case of Rollin' Justin, illustrated in Figure 2.1, not only the basic form is human-like shaped, but also the assembly and location of its sensors. In the current design iteration, the research platform is equipped with a large variety of sensors, which is able to provide sensory information from different types of perceptual modalities.

Visual Camera Systems

Unsurprisingly, Rollin' Justin's conceptual design for establishing a perceptual system does not deviate from the standard practice in robotics. Primarily relying on visual sensors, it is outfitted with seven camera systems, of which three are located in the head. In contrast, the remaining ones are mounted at each corner of the base platform.

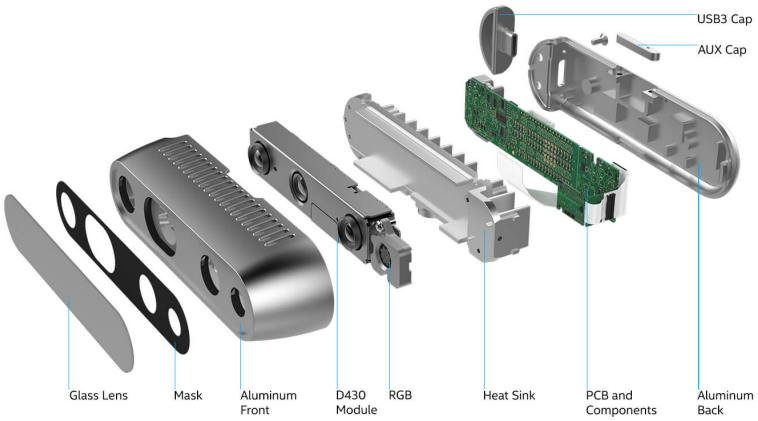


Figure 2.2: Exploded view of the Intel RealSense D435i camera system [23].

Orientated on the location of the human eye, the robot is equipped with a stereo camera pair. Operating in the visible spectrum, the cameras deliver a color image for object tracking tasks and as visual input for teleoperation. At the moment, they are not contributing to the perceptual system since the camera pair does not directly provide depth information. For the generation of a depth image, the stereo images have to be processed by block matching algorithms, which are very computationally intensive. In practice, the required computational power is not worthwhile since perceptual tasks can be economically taken over by the other camera system on the forehead.

In the course of the latest modernization measures, the robotic system was outfitted with five Intel RealSense D435i cameras, one of which is integrated into the forehead. In contrast, the other four are installed at each corner of the base platform. As shown in its components in Figure 2.2, this visual system was developed as a state-of-the-art stereo vision depth camera system for various fields of application, which also includes the area of autonomous mobile robots [21]. The imaging assembly can generally be divided into a color sensor and the D430 depth module. The latter component comprises primarily two OmniVision OV9282 infrared sensors responsible for collecting visual information for the subsequent depth image generation. For further improvements in the depth image quality, the infrared cameras can be supported by a vertical-cavity surface-emitting laser (VCSEL) pattern generator. It is mounted between the imaging sensors on the depth module and projects, if required, a predefined dot pattern

Table 2.1: Overview of most relevant properties of the Intel RealSense D435i camera system [21, 22].

Parameter	**Specification**
Infrared Image Sensor	OmniVision OV9282
Max. Resolution	1280 × 800 pixels
Recommended Resolution	848 × 480 pixels (WVGA)
Max. Frame Rate	90 frames per second (FPS)
Shutter Type	Global Shutter
Max. field of view (FoV) (H/V/D)	91.2°/65.5°/100.6°
Max. FoV at recom. resolution (H/V/D)	75.0°/62.0°/89.0°
RGB Image Sensor	OmniVision OV2740
Max. Resolution	1920 × 1080 pixels
Max. Frame Rate	90 FPS
Imager Shutter Type	Rolling Shutter
Maximum FPS	90 FPS
FoV at Max. Resolution (H/V/D)	69.4°/42.5°/77.0°

in the infrared spectrum unto the front-facing scenery. Especially in low lighting conditions and poorly textured surfaces, the projected static point pattern creates valuable references for infrared sensors. Apart from the module for true depth estimation, the camera is accomplished by an OmniVision OV2740 color sensor. Table 2.1 features a selection of relevant properties of the imaging assembly of the camera system. Although the depth cameras can provide a maximum resolution of 1280 $\times$ 800 pixels, the resolution should be adjusted to the Wide Video Graphics Array (WVGA) standard according to the manufacturer to achieve the best depth-sensing performance [22].

The recorded stereo image pairs from the D430 module are subsequently routed to the integrated D4 vision processor, where the depth map is calculated in real-time. In the following, all data, including the images from the three imaging sensors and the computed depth image, are forwarded via an external Universal Serial Bus (USB) 3.1 Gen1 interface with a USB-C connector.

Although the camera system has a relatively wide FoV, it nevertheless reaches its limits if the entire environment has to be covered. For this purpose, Intel has integrated an external sensor synchronization connector directly to the main printed circuit board (PCB) on the top side of the camera body. By this means, individual cameras can be synchronized for, e.g., image capturing at identical times while providing the same frame rates [21]. Unfortunately, Rollin' Justin does not provide the mandatory hardware wiring and necessary interfaces synchronizing its cameras.

Auditory Sensors

Besides the optical sensors, Rollin' Justin has also become a research platform for other, more exotic perceptual systems over its persisting lifespan. In order to have the capability for auditory scene analysis, a hardware assembly allocated across the robot's forehead was engineered in 2019 [24].

Following the broadband microphone sub-array approach, the audio spectrum is divided into three sub-bands responsible for receiving frequencies lower than $1\,\mathrm{kHz}$, between $1\,\mathrm{kHz}$ and $2\,\mathrm{kHz}$, and greater than $2\,\mathrm{kHz}$. For this reason, the sensor array consists of eight SPH0645LM4H microphones with Inter-IC Sound support, which are symmetrically arranged as shown in Figure 2.3b, and internally grouped into specific sub-arrays according to 2.3a. Since the spatial sensing of sound depends on the experienced signal, respectively time delay by the receivers, a suitable distance between the individual microphones has to be maintained. During the design process, their locations were carefully chosen and assigned to sub-bands based on the scaling effect of the required distance in between with the wavelength. Therefore, the outermost sensors are responsible for lower frequencies with longer wavelengths to utilize the maximal available distance based on the given hardware.

In contrast, the necessary gap for shorter waves scales accordingly and can be handled by inner sensors. Further improving the received speech quality, we increased the sampling rate to $16\,\mathrm{kHz}$ instead of the usual $8\,\mathrm{kHz}$. The microphones were soldered onto a flexible PCB to create an auditory scene analysis system. By choosing this manufacturing concept, lower electronic noise and a higher uniformity of electrical characteristics can be achieved compared to cable-based connections while maintaining structural flexibility in the integration process to Rollin' Justin's curved forehead.

The robot's auditory system is still not yet available in its current state since it is associated with further upgrades within the computation network. It is to be expected that the hardware is going to be fully integrated by early 2023.

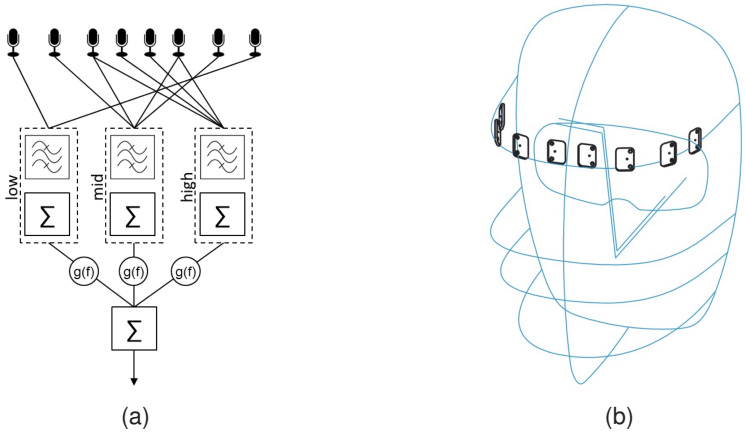


Figure 2.3: Microphone array design for Rollin' Justin [24]. (a) Illustrates the construction of the the sub-array approach, (b) shows the CAD drawings of the microphone positions on Rollin' Justin's forehead

Additional Sensors for Tracking and Localization

Apart from conventional sensation methods, Rollin' Justin features two additional types of sensors, which could be utilized for localization tasks.

Outfitted with rotary encoders on its wheels, a dead-reckoning location estimation system was established, also referred to as wheel odometry. Based on the measurement of the covered distance on all four wheels in combination with the mechanical boundary conditions of the robot's model, a reasonably reliable estimation of the change in position can be calculated. This process takes place in the Simulink model of Rollin' Justin, and the resulting information is then forwarded as a part of the robot's telemetry for further utilization. In general, a 3-D state vector is provided consisting of the robot's location in the x-y plane that is parallel to the ground, and the angle ϑ, which represents the orientation of the robot.

As is usual in mobile robotic systems, Rollin' Justin is equipped with inertial measurement units (IMU). Using a combination of accelerometers, gyroscopes, and magnetometers, it is able to obtain the specific force, angular rate, and orientation of a given body. Two sets of IMUs are integrated into the robot's inner framework, mainly for balancing tasks, which are currently unavailable for perception-related tasks. Nevertheless, each of the five RealSense D435i RGB-D camera systems does include an integrated BMI055 IMU. Comprising of a 16-bit triaxial gyroscope and a 12-bit accelerometer, also with detection abilities in three dimensions, it provides 6 DoF in total [25].

2.2.3 Computation Architecture

While the sensors, as mentioned above, are responsible for collecting information from the robot's surroundings, the computer network on the inside forms the backbone for higher-level perceptual tasks. In the case of Rollin' Justin, not only the outer appearance is oriented towards the human body, but also the computer architecture on the inside.

General Architecture

Table 2.2 summarizes the specifications of the individual processing nodes on Rollin' Justin. Starting from the bottom of the perception pyramid, peripheral sensors are not directly connected to high-level computers but are rather administrated by embedded computing boards. This layout resembles those in the human system, where sensory inputs are also organized and pre-processed in intermediate stages before being forwarded to the central nervous system. Administrative tasks are handled by three NVIDIA Jetson TX2 boards, each attached to an Auvidea J140 carrier. Two of them are located on the base platform, whereas the remaining one is included in the head assembly. In the case of the RealSense cameras, Jetson 1 and 2 are responsible for control and management duties in collaboration with Intel's RealSense software development kit (SDK) 2.0. At the same time, Jetson 3 takes charge of sensors in the head unit. As part of the subsequent development iteration containing the integration of the microphone array, the embedded computer in the head will be replaced by a more powerful Jetson Xavier board to satisfy the increased computation demands. Collected sensory information is then fed into the robot's internal Ethernet data bus, e.g., via the inter-process communication library SensorNet on the visual side. From there, the data streams are accessible by higher-level applications and will be distributed to their designated targets.

Apart from the peripherals, Rollin' Justin's high-level computation architecture is built up by three computers, two of which are real-time capable and in charge of controlling the robotic hardware. Communicating within the real-time Ethernet for Control Automation Technology (EtherCAT)

Table 2.2: Summary of the individual computation nodes and their specifications within Rollin' Justin.

Name	Specification	Task Description
Hannibal	Intel Core i7-7820EQ Quad-core at 3.0 GHz up to 3.7 GHz 32 GB DDR 4 SDRAM Debian 9	Head computer Administrative tasks for internal communications: File-, DHCP- and NTP-server
Face	Intel Core2 Quad Q9000 Quad-core at 2.0 GHz 4 GB RAM Debian 9 PREEMPT_RT	Robot kernel Tasks associated with motion and movement Simulink model of Rollin' Justin
Amit	QNX real-time OS	Tasks associated with DLR Hand II model LWR (arms)
Jetson 1 Jetson 2 Jetson 3	NVIDIA Jetson TX2 NVIDIA Tegra X2 SoC Dual- + Quad-core at 2.26 & 2.0 GHz 8GB LP-DDR 4 RAM Jetson OS	Administrative tasks for peripheral sensors Jetson 1: Cameras (left-hand side of the base) Jetson 2: Cameras (right-hand side of the base) Jetson 3: Head-mounted cameras
Jetson 3 (Upcom.)	NVIDIA Xavier NVIDIA Tegra Xavier SoC Hexa-core at 2.26 GHz 8 GB LP-DDR 4 RAM Jetson OS	Administrative tasks for head-mounted sensors Head-mounted cameras Microphone array
Decker	Intel Xeon E5-1620 Quad-core at 3.6 GHz up to 3.8 GHz 8 GB DDR 3 SDRAM Debian 9	External console

network, the robot kernel and the robot's Simulink control model integrated in *Face*. Therefore, this computer manages all tasks associated with motion and movement. The second computer for direct hardware control purposes is named *Amit*, and it is responsible for the LWR and the hand model. In the scope of the thesis, any IT systems related to the real-time branch are not of great interest to us since they do not contribute to the perceptual process in any sense. However, the most interesting computer for us is *Hannibal*, which is the backbone of perceptual processes. Also referred to as Rollin' Justin's brain, it is the head computer, where among other higher-level system tasks, machine perception-related duties are executed in conjunction with collected sensor data.

Furthermore, *Hannibal* is responsible for managing internal and external communication protocols, including the administration of file, Dynamic Host Configuration Protocol (DHCP), and the Network Time Protocol (NTP) server. Designed as a mobile platform, the robot can communicate with external hosts via a wired Ethernet connection and over its integrated wireless local area network (WLAN) interface. Connected to Rollin' Justin via the latter method, *Decker* is an external computer that oversees the entire robot operation. By providing an external console to its internal systems, operators can monitor and command the robot and its modules from the outside.

On the software side, the robot is equipped with DLR's in-house system deployment framework, *Links and Nodes*, which provides the basic communication scaffolding and control over individual modules in operation. From there, the entire robot is administrated and monitored by providing a structured view of the running processes and the way they are exchanging data.

Available Resources

As one can already assume from previous sections, the field of sensation and perception contains only a fraction of the tasks that have to be processed by the robot's internal hardware. For this reason, available computation resources must be shared between different processes relating to individual areas of responsibility. While administrative duties of peripheral sensing devices are already outsourced to individual nodes, *Hannibal* is the only available computer within the current hardware architecture for higher-level perceptual tasks.

However, due to the robot's general orientation restrictions as a mobile system, the already limited computation resources must be carefully distributed to individual modules according to their operational importance. Specifically for perception, only two of the eight central processing unit (CPU) cores are available, which is dwindling small for the given assignment size, given that comparable platforms utilize multiple onboard computers for perceptual and controlling duties [26].

2.3 Environmental Perception and Modeling in the Mobile Robotics

After the foundations of machine perception have been laid in the previous section, data collected by individual sensor systems have to be processed and interpreted to establish spatial awareness. Over the years, two major approaches in the field of tracking, mapping and localization have emerged in the scientific research community and are presented in the followings.

2.3.1 General Aspects and Terminology

According to the current state of research and technology, Visual Odometry (VO) and Simultaneous Localization and Mapping (SLAM) approaches are considered the most promising strategies for visual perception and creating spatial awareness for both computer vision and robotics [27]. In the early days, most of the research was motivated by the American mars exploration program to provide planetary rovers with the ability to estimate their relative motion with visual-based sensors. Compared with dead reckoning methods containing conventional odometry information from the wheels, visual data from an onboard camera is not affected by physical influences like wheel slippage, which is a common issue encountered on uneven and rough terrain.

With this, VO is the process of estimating the ego-motion of an agent (e.g., vehicle, human, and robot) using only the input of a single or multiple cameras attached to it [28]. As a particular case of the technique known as structure from motion (SfM), it provides an estimation of a camera's position and orientation by analyzing incoming image sequences [29]. Therefore, it mainly focuses on local consistency to calculate the camera's path incrementally, pose after pose, with potential local optimization steps to enhance the tracking accuracy and elaborate on minimizing the drift. On the other side, SLAM is a process in which an agent is required to localize itself in an unknown environment while incrementally constructing a map of its surroundings [28]. Thus, it focuses on establishing a globally consistent estimation of the robot's trajectory inside the simultaneously generated map. While the line between these approaches becomes increasingly blurred as development continues, this is precisely where the difference lies in their original definition. SLAM applications achieve global consistency by revisiting and recognizing already mapped regions at which loop closure events reduce the accumulated estimator error in both the map and camera trajectory [30]. In contrast, VO cannot provide an adequate method for the drift problem due to its design. Although these two approaches started as parallel but separate lines of research, SLAM can nowadays be viewed as an extension of the VO approach. In real-world applications, the selection of approach is based on the exact use case since SLAM methods can also be used for VO applications, in which only the trajectory is necessary. Here, the main aspects of the selection process are the trade-off between performance, consistency, and simplicity in implementation.

2.3.2 Design Principle and State of Research

From a structural perspective, modern SLAM systems can be divided into the front- and back-end. With this, the more locally focused concept of VO is used as the front-end component within the complete system to process the raw sensor data and recover the incremental motion of the camera.

Front-End – Visual Odometry

Generally speaking, visual SLAM algorithms can be divided into two types of strategies in terms of data association consisting of the direct and the feature-based approach. The ego-motion and camera relative pose estimation process in the former case is based on optical flow. With this, relative motion estimation is achieved by analyzing the variations in image intensities, and information about motion and structure can be reliably estimated by minimizing photometric error. On the other side, the feature-based method combines feature extraction and matching for calculating the relative motion in the image sequence. Therefore, reliable and invariant regions of interest have to be extracted from the original image for further processing and estimation.

While the direct method is best suited for, e.g., collision avoidance tasks by reconstructing the entire scene to generate a dense map, feature-based approaches are more advanced for perception and primary navigation duties. In the case of mobile robotic applications, the latter approach is auspicious since the necessary computation resources, especially memory consumption, can be relatively small. Due to the pre-processing and information compression to selected features, only a fraction of the input data has to be saved to generate an adequate surroundings model for localization and navigation purposes.

In terms of the feature-based approach, the VO pipeline usually incorporates the process of feature extraction, feature matching, motion estimation, and potential local optimization, as illustrated in Figure 2.4. Starting from the captured image, feature extraction is the combined process of generating an abstraction of the image by detecting distinctive regions of interest and assigning the individual areas with unique identifiers in compliance with their characteristic surroundings for the subsequent matching and recognition steps. Optionally, the processing pipeline is supplemented by intermediate stages for keyframe identification, in the case of an optimization-based approach, and outlier rejection within the feature matching step to achieve a more accurate camera motion estimation. The last component in the toolchain is responsible for the local optimization of the concatenated transformations estimated by the previous steps. Within this module, locally constrained and window-based spatial optimization can be performed to improve the algorithm's tracking accuracy further.

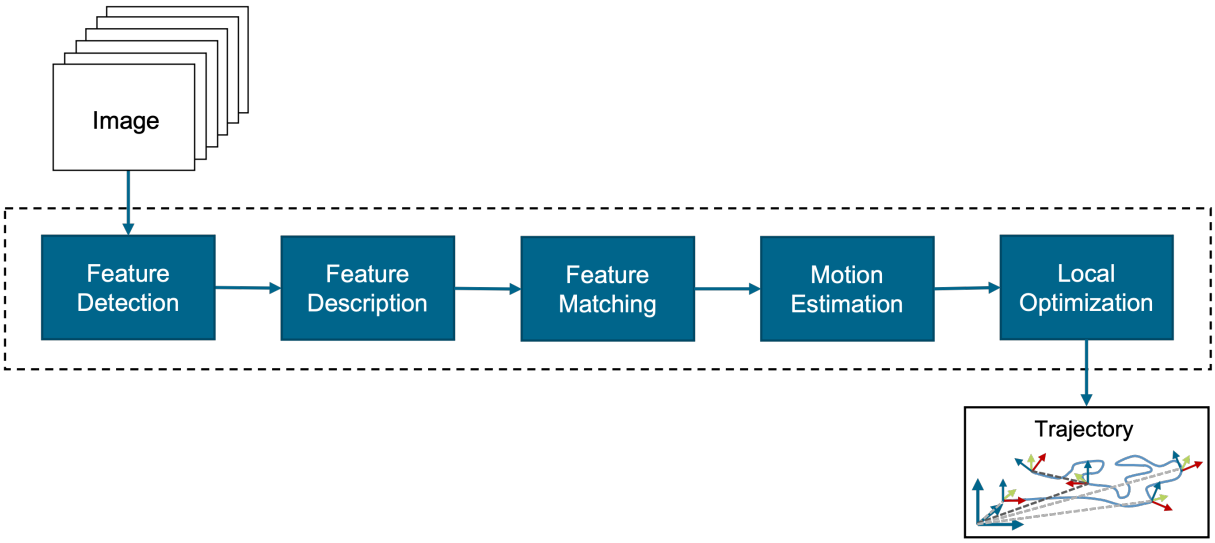


Figure 2.4: Basic architecture and components within the VO processing pipeline

Starting with the basic idea, Moravec proposed the possibility of estimating the ego-motion of an agent from visual-only inputs in the 1980s [31]. Although the author utilized a single camera in his so-called sliding stereo setup, it can already be classified as the first approach in the field of stereo vision since the motion estimation was based on 3-D points of interest. The location of the features is directly triangulated at each frame, resulting in the estimation of relative motion in a 3-D-to-3-D point alignment setup. It is accomplished by minimizing the L_2 or euclidean distance between corresponding 3-D feature sets in the source and target image, including the ability to estimate the absolute scale of the transformations. As an alternative, motion calculation in a stereo setup can also be handled using only 2-D information from both imager and the quadrifocal constraints instead [32]. Building on Moravec's approach, major contributions in the early ages are published in [33], [34], and [35]. With the growing confidence and technology maturity after the turn of the century, the NASA and Jet Propulsion Laboratory (JPL) successfully integrated the novel approach as a part of the navigation system onboard the Mars Exploration Rovers (MER) as one of the first major applications [36].

Nister et al. coined the generic term for this type of technology in their influential publication "Visual Odometry" [37] in relation to the concept of wheel odometry, which also similarly integrates increments to estimate the position of a robot. Furthermore, their proposed algorithm is conceived as the first real-time long-running implementation with a reliable outlier rejection scheme. With this, the commonly used feature tracking process is replaced by feature matching, which is more suitable when a significant motion or viewpoint change is expected [38]. In terms of motion estimation, they discovered that the 3-D-to-2-D approach delivers more accurate results than the 3-D-to-3-D method since it minimizes image reprojection instead of the 3-D feature position error. Especially in the case of triangulated 3-D points, they are equipped with much higher uncertainties in the depth direction, which significantly impacts the already very delicate motion estimation process and its achieved accuracy.

As mentioned above, it is possible to determine the relative movement using only 2-D information by direct comparison without the additional triangulation step. Therefore, the estimation is mainly based on the essential matrix and the epipolar constraint [28]. Especially in the case of a monocular setup, 2-D data from the sensor can be directly used for the calculation process. In comparison, the 3-D information for the 3-D-to-2-D approach has to be triangulated from two adjacent images and matched to 2-D points in a third image, involving a total of three frames [39]. It effectively utilizes only one sensor in a monocular setup with perspective and omnidirectional cameras. One of the main challenges with this approach is that the absolute scale of motion cannot be estimated and has to be determined from direct measurements or delivered by other sensors. Therefore, the common method is to normalize the distance between the first and second frames. The relative scale and camera pose with respect to the first frames are calculated either using the knowledge of 3-D structures or the trifocal tensor [40]. Particularly in large-scale environments, stereo vision degenerates to the monocular case when the distance to the scene is much larger than the stereo baseline. A rule of thumb would be 40 times the baseline.

Nevertheless, the 3-D-to-2-D method is more often used in practice than the 2-D-to-2-D approach since it is coupled with faster data association in terms of outlier detection and rejection. This has the background that the computing time is primarily influenced by the number of mandatory points for the motion estimation and, subsequently, the outlier rejection step. In the case of the former method, the minimal case requires a total of three corresponding points within the P3P algorithm [41]. In contrast, in the latter case, a minimum set of five correspondences is necessary for the five-point algorithm [37].

Back-End – Localization and Mapping

Further down the line, the back-end receives the intermediate representation and solves the underlying optimization problem behind long-term localization and mapping. Thus, it provides the estimation of a parameter set that describes the spatial pose of the previously extracted landmarks, from which the location and orientation of the robotic agent are eventually derived. An overview of the major visual SLAM algorithms is provided in Figure 2.5 in a chronological order.

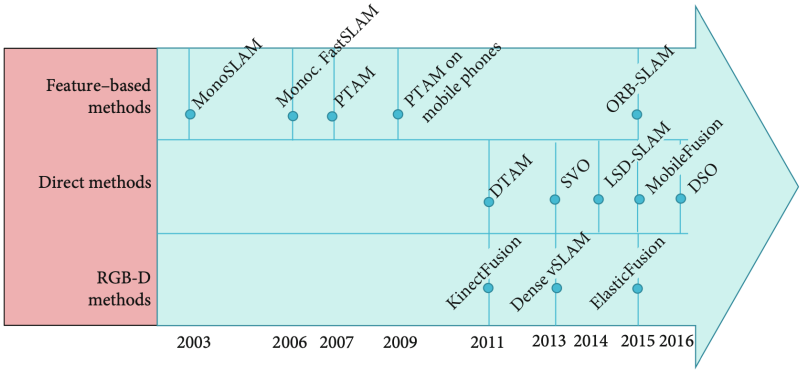


Figure 2.5: Overview of the most significant visual SLAM algorithms. Figure adapted from [39].

In the field of SLAM, there are two main design categories on which algorithms can rely. The initial approaches correspond to filter-based methods similar to those first used to solve the SLAM problem. In the early research phases, they are not explicitly targeted towards visual data. However, they are designed to fuse, e.g., odometry data from various sources and information from laser-based ranging sensors [39]. Therefore, algorithms, such as the EKF-SLAM proposed by Smith et al. [42], were based on the extended Kalman filter (EKF) for the associated tasks of tracking, localization, and mapping. The first real-time algorithm using the information from visual sensors was published by Davison et al. in MonoSLAM [43]. With their introduction of particle filters, they successfully tackled the issue of considering the initialization of new points by reducing the uncertainty on the field depth from newly detected visual landmarks. Although they are already capable of providing the first promising results, there are still significant issues concerning a large amount of data, especially in large-scale environments. Initial attempts to improve computation efficiency and scalability were based on windowed methods, which were relatively unsuccessful. A thriving remedy was created by FastSLAM [44], which solved the issue regarding the logarithmic scaling of mapped features. The utilization of an unscented Kalman filter was introduced by Chekhlov et al. in [45]. In the modern era, most methods are based on a multi-state constrained Kalman filter (MSCKF) [46], which deconstructs the classical state-vector into separate ones containing the camera pose and landmarks position separately. In addition, modern approaches, such as Robust Visual Inertial Odometry (ROVIO) [47], further reduced the system's complexity by implementing a restrictive culling of landmarks to only keep the most recent detected features in the state-vector.

The second design principle utilizes parallel methods for distributing necessary tasks inside the system's processing pipeline to different threads in the so-called optimization-based approach. Compared to the previous design, this method has an enhanced ability to reduce tracking drift by design. Also referred to as the keyframe-based method, distinctive and robust images are commonly arranged in a graph structure, providing a novel way of storing features without hardly any constraints in terms of scalability. Furthermore, the overall performance and accuracy

are enhanced using a global optimization process, which can also be arranged on a specific window of keyframes [39]. As a result, bundle adjustment (BA) can either be performed on poses between individual keyframes in pose graph optimization or with the focus on optimizing the map structure in a structure-only BA. On the other hand, this method also involves a very high computational cost, which forced the localization and mapping tasks mainly to be handled offline in early publications. Klein et al. introduced Parallel Tracking and Mapping (PTAM) [48] as one of the first real-time capable algorithms by dividing the tracking and mapping tasks into two separate threads running in parallel. Although this approach seems promising, it is not suitable for deployment in open environments, as it still has issues considering efficient feature storage, optimization strategy, and loop closure detection. The challenge of considering long-term robustness was addressed by Strasdat et al. in [49], and the innovative idea of double window optimization was proposed in [50]. Based on the latter approach, Lim et al. introduced an alternative method in [51], which allows for a more efficient feature extraction and description process at the cost of sacrificing invariance against rotation and scale change. This characteristic was reinstated by ORB-SLAM [52] utilizing the Oriented FAST and Rotated BRIEF (ORB) algorithm [53] for feature extraction purposes.

Apart from feature-based methods, the first meaningful contribution in the field of direct approach was introduced in Dense Tracking and Mapping (DTAM) [54], which uses the direct approach for tracking purposes but constructs a sparse map for easier processing. Further major contributions were made in the hybrid methods of Semidirect Visual Odometry (SVO) [55] and Direct Sparse Odometry (DSO) [56], taking advantage of both direct and indirect input searches. In terms of exploring large environments, the Large-Scale Direct monocular SLAM (LSD-SLAM) is one of the first direct methods which utilizes semi-dense mapping.

In recent years, the research community has been actively targeting RGB-D camera systems apart from the classic monocular and stereo setup. Newcombe et al. introduced the idea of integrating depth information in their KinectFusion [57] by combining RGB and depth data into a dense surface map. Although this approach lacks sufficient scalability for deployment in a large-scale environment and loop closure capabilities, it is still conceived as the first RGB-D SLAM system. The issues were then solved by [58], and the first feature-based approach was introduced by Endres et al. in [59]. Further meaningful contributions are the Dense vSLAM published by Kerl et al. [60] and ElasticFusion by Whelan et al. [61]. As an extension to the original ORB-SLAM, Mur-Artal et al. introduced the compatibility of RGB-D cameras, among several other enhancements, in their ORB-SLAM 2 [62]. With this, the depth information is used to extract a virtual stereo coordinate for each extracted feature to achieve a sparse reconstruction of the environment.

Especially in mobile robots, precise information about their position and spatial orientation is essential for the overall operation and serves as the basis for further manipulations. For this purpose, Rollin' Justin was equipped with a SLAM system based on the methods proposed and implemented in ORB-SLAM 2. The robot's unique features and hardware characteristics were taken into account, resulting in the development of the Multi-camera Robust ORB-SLAM (MROSLAM) [63]. As the name already suggests, MROSLAM is able to receive and fuse sensory information from different camera systems without any overlapping FoV. The final robot pose is estimated based on the extrinsic relations between the utilized camera systems. Since the integration of the system at the end of 2020, it has become an integral part of everyday robot operation and has already been used in current research projects. With this, the robot can explore and localize itself in a generic environment without pre-installed fiducials, as distinctive landmarks can be identified from the surrounding characteristics. While basic

tracking, localization, and mapping tasks can already be reliably carried out, the pose estimation accuracy and especially the robustness against perturbations could still be improved.

2.3.3 Multi-Modal Environmental Perception

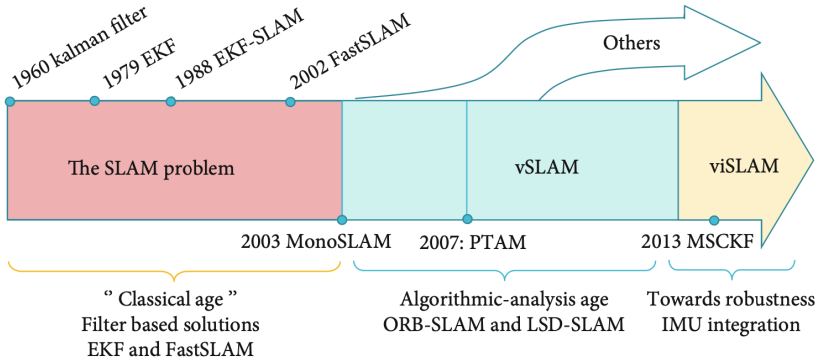


Figure 2.6: Overview of the historic milestones and direction of development in the domain of visual SLAM. Figure adapted from [39].

In general, many contributing factors can negatively affect the accuracy and reliability of a VO and SLAM system. On the one hand, reduced robustness may be caused by inaccuracies in the hardware-related domain or due to insufficient development and maturity of the proposed approaches within machine perception. On the other hand, technological constraints and deficiencies of the selected perception branch could also significantly impact the overall estimation quality. In order to further improve the capabilities and overcome those limitations, the scientific community has extended its research efforts increasingly beyond the visual branch. As a result of this, synergies through reasonable combination of different sensory and perception modalities should be utilized within a multi-modal system.

Figure 2.6 depicts the historic milestones and the current development trend in the field of visual SLAM algorithms. As a first step, researchers have directed their focus towards integrating inertial information from IMUs as an addition to the vision-based process. This sensory modality is comparable to the capacity of the human vestibular system responsible for the perception of balance, which is sensed by the inner ear. First evaluations have proven the effectiveness of the research direction, in which the shortcomings of individual approaches could be compensated.

While the previously presented visual SLAM methods are categorized by the camera system configuration and the input data type, visual-inertial approaches are classified using the level of coupling between the information sources [39]. In terms of loosely coupled approaches, the number of theoretical combinations is widely spread since it is possible to pair up all visual front-ends with a Kalman filter or with a pose graph optimization back-end like iSAM2 [64] in order to incorporate inertial measurements. Regarding the tightly coupled algorithms, major contributions are MSCKF [46] and ROVIO [47] in the domain of filter-based methods. On the other side, Open Keyframe-based Visual-Inertial SLAM (OKVIS) [65], VINS-Mono [66], and the newly introduced ORB-SLAM 3 [67] are representatives of the optimization-based design principle.

Apart from the visual-inertial field of research, multi-modal SLAM approaches are still a rarity. Nevertheless, there are some notable advancements in, e.g., the auditory domain. Regarding auditory research in robotics, most of the conducted research is focused on SSL, but also directed towards the construction of a independent SLAM system. While conventional approaches either

apply visual SLAM techniques to acoustic Times-of-Arrivals or perform localization tasks by actively probing the room, acoustic SLAM (aSLAM) [68] was developed as an independent auditory SLAM system from scratch. By utilizing directions of arrival (DoA) data as input, it can jointly estimate the unknown observer path and the position of multiple interfering sound sources with passive acoustic sensor arrays. This algorithm is equipped with sufficient robustness against reverberation, noise, and periods of source inactivity by design.

3 Conceptualization and Methodology

Based on the identified issues of state-of-the-art systems and the resulting research questions in the previous chapters, we aim to develop a novel VO approach as a part of an in-house developed SLAM system optimized for Rollin' Justin and other robots at the Institute of Robotics and Mechatronics. While still under construction, the Indoor Optimized, Globally Consistent, Environment-Aware, Longlife-SLAM (IGEL-SLAM) is designed as an application-oriented software framework for research on environmental perception and modeling with a particular focus on tracking, localization, and mapping.

During the conceptualization process, special attention is dedicated to the modular and decentralized design architecture by dividing the system into multiple independent components with well-defined interfaces to achieve the optimum adaption to the distributed hardware and computation architecture. Hence, the full potential of the otherwise unused computational resources allocated at different parts of the robot can be utilized and exploited for dedicated tasks. This approach is especially interesting for lightweight robots with minimal computing capabilities, as the onboard resources are insufficient for handling the entire machine perception system. Therefore, only the most essential real-time modules, e.g., the front-end component responsible for tracking, are executed locally on the robot. In contrast, other components of the SLAM framework responsible for localization and mapping tasks are carried out at a stationary workstation via a wireless connection. Taking it a step further, the tracking module itself should also have the freedom to be distributed to several computing nodes within a robot, on which the individual processing steps can be executed separately. For this purpose, the individual components within the toolchain are modularly structured and provided with the interfaces required for interprocess communication (IPC) at suitable locations. The major challenge here is to determine a balance between necessary IPC, especially the size of the transferred data as well as the resulting time-loss, while still managing a real-time capable application and preventing overloading of the communication bandwidth. In addition, the selected system structure implicitly results in good extensibility of the overall framework.

Apart from the structural aspects, great emphasis was placed on establishing multiple modalities at different system levels, which also highly affects the approaches utilized within the VO module. Thus, it provides a potential solution to the deteriorating performance and robustness of the primarily vision-based state-of-the-art perceptual system under adverse environmental conditions by exploiting potential synergies created by the collaboration between different methods. In the followings, this chapter introduces the audience to two possible domains in which multiple modalities can be established in the context of front-end applications responsible for tracking and short-term localization. At first, Section 3.1 introduces the hardware-specific area, in which the question of how and which type of sensor information can be reasonably utilized and combined within a front-end module is investigated. Following the sensor-related level, a multi-modal setup can also be realized in the software-related domain, which will be the central topic in Section 3.2.

3.1 Multiple Modalities in the Hardware Domain

Starting with the probably more intuitive sensory-related level, a multi-modal setup can be established in the hardware-specific domain based on two different concepts. In general, we distinguish between the homogeneous and heterogeneous approaches, characterized by the type of the considered sensors.

Intra-Class Sensor Collaboration

At first, it is achieved by combining sensor data from input devices with similar characteristics or at least within the same field of sensory perception. Specifically, information from multiple sensor systems with, e.g., a particular field of detection, could be consolidated and further processed to construct additional redundancy layers in the estimation toolchain. The primary target here is to enhance the quantitative aspects of the collected data in a homogeneous way, which may also positively impact the data's overall quality in terms of scenery composition and surface texture. By expanding the size of the database, which could be considered for the targeted purpose, it is more likely to obtain perceptual information with higher quality from a probabilistic point of view. As a result, the collaboration between similar sensors from the same class would also improve the overall estimation accuracy of state-of-the-art methods in case multi-sensor support is available. This particular approach was investigated in [63], in which a multi-camera visual SLAM algorithm was developed as an extension to ORB-SLAM 2. In addition, a study was conducted in which compelling results were achieved with real-world datasets. Particularly in applications with unfavorable environmental conditions, the addition of this type of redundancy significantly contributed to the system's enhanced robustness against external influences. Therefore, events which potentially lead to the failure of tracking and localization capabilities can be minimized. This includes occurrences both on the hardware- and implementation-related side, as well as in the operational and application-specific field, such as temporary loss of applicable sensor information caused by, e.g., obscured field of detection. Especially in the field of mobile service robots, robotic agents should be able to safely approach stationary objects of interest and navigate through narrow passageways, potentially in a less-textured environment.

For this reason, it has to be ensured that the failure of one sensor is absorbed by this type of redundancy and does not affect the overall stability of the superordinate system. In terms of Rollin' Justin, this kind of multiple modalities can be realized by combining input-data from the available Intel RealSense sensor systems. Since they are integrated into a circular arrangement with individual FoV, a comprehensive view of the robot's surroundings can be potentially created with an adequate environmental perception and modeling method.

From the conceptual perspective, the sensor fusion process can be integrated into the front-end and back-end modules. Apart from their general properties presented in Section 2.3.2, there is an additional difference in the field of task distribution, and its associated clock speed. While the front-end modules are responsible for tracking and short-term localization, these tasks are highly time critical. Thus, the relative motion estimation process has to be equipped with real-time capabilities. Considering the characteristics of the targeted mobile robot, continued tracking and localization updates with a minimum of $15\,\mathrm{Hz}$ have to be guaranteed.

On the contrary, the back-end is assigned to fulfill long-time localization and mapping duties. Even though these processes are an integral part of SLAM, which is reflected not only in the naming, they are mainly responsible for maintaining global consistency in the medium and long

term. Therefore, and especially with regard to mapping, the mandatory clock speed can be reduced to, e.g., 1 Hz. As a result of this, a reasonable balance between the required processing time, which scales with the available computation resources, and the effectiveness should be evaluated as part of the cost-benefit analysis in the conceptualization phase. For this reason, the module responsible for sensor fusion is pushed back into the back-end, which can operate at a lower clock speed since more complex calculations have to be made. In our current approach, the main objective of the front-end is to process raw sensor data and to provide the back-end with, among other parameters, a relative motion estimation with sufficient accuracy in a reasonable amount of time. For simplicity, these modules are limited to receiving information from only one sensor of the same class in the first design iteration. This decision was also made with for the best utilization of the distributed computing architecture. At least partially, individual front-end modules can be executed in the NVIDIA Jetson boards further upstream.

Inter-Class Sensor Collaboration

Apart from the homogeneous approach, sensor data from different sensory perception fields can be combined to further enhance the robustness and continuity of the perception module. Similar to the human perception system, potential synergies are created through a reasonable combination of different sensory modalities. With this, the strength of the utilized areas is bundled while their individual deficiencies are largely compensated from the system's perspective. In contrast to the previous method, in which the absolute quantity of the available information was enlarged, the focus within the heterogeneous approach is directed towards extending the spectral distribution of the collected environmental influences. By doing so, a multi-modal redundancy is established, and potential loss of tracking (LoT) events, which are sensor-specific and vary depending on the respective perceptual branch, can be averted.

As already introduced in Section 2.3.3, the combination of visual and inertial information is the most popular approach currently in the research community. While the visual domain is evidently one of the most potent areas in machine perception, the performance and reliability of state-of-the-art algorithms largely depend on the quality of the available images. This can significantly deteriorate during rapid motion sequences, in which the increased presence of motion blur impairs the resulting image quality. Especially in our selected feature-based approach, it poses a considerable challenge, as the texture of the recorded scene is blurred by this type of disturbance. Thus, detecting reliable and unique characteristics becomes more complex and negatively affects the application's operational performance. In the extreme case, this would result in a temporary LoT. Under adverse circumstances, however, this could also develop into a permanent LoT, which cannot be recovered. In these situations, the overall system would benefit from the additional IMU information, and tracking and short-term localization functions could be maintained until the visual method is reinstated. From the perspective of the inertial branch, it also benefits from this symbiosis. While inertial information in form of linear and angular acceleration make a valuable contribution in the previously stated situation, they are conceptually subject to a certain inaccuracy and cannot be used for establishing global consistency or for mapping purposes. In addition, the IMU model has to be precisely tuned in order to suppress background noise. For this reason, it is recommended to favor visual information over inertial data in these cases, which could also be marked as unreliable and neglected.

Regarding Rollin' Justin, information from various sensory modalities can be combined into a multi-modal framework. This includes perception data from the visual domain, auditory area, inertial measurements, and odometry estimations. However, from the perspective of the general

system architecture, front-end modules are typically very task-specific and sensor-oriented. For this reason, also to fulfill the targeted modular design principle and the mandatory real-time capabilities, a cost-benefit analysis should be conducted in order to determine the most suitable combination of the available sensory modalities.

Starting with the already acquainted combination between visual-based sensors and inertial measurements, they form the most popular and well-researched area since both sensor types are widely utilized in the engineering domain with a broad range of applications. From a theoretical perspective, this particular combination is worthwhile considering the resulting interaction between the realm of external sensory and internal perception. While the visual domain belongs is subordinate to the field of exteroception, measurements from an IMU are considered as a sub-area of proprioception that is referred to as the sense of self-movement, force, and body position. In contrast to external perception, which exclusively relies on sensory stimuli from the surrounding environment, the so-called "sixth sense" is practically independent of environmental conditions. Thus, these sensors give a more generalized view of the respective scenarios with a more static appearing spectrum of information. As for the practical application, IMUs are integrated as a low-cost extra, as they are very advantageous in terms of the cost-benefit ratio. In our targeted hardware platform, these measurement units have been integrated into the Intel RealSense sensor systems by default. Therefore, all the primarily vision-based sensors responsible for establishing spatial awareness are equipped with visual-inertial capabilities. Although Rollin' Justin is also fitted with a standalone IMU, which potentially provides measurements with better resolution, it is recommended to use the onboard sensors in this context, since they are integrated in a tightly coupled manner. Consequently, these sensors are entirely independent of each other, which also corresponds to our desired modular and distributed system architecture. A positive side effect of this constellation is that the time-consuming extrinsic calibration between the visual sensor and the IMU can be omitted since the manufacturer already provides the corresponding parameters.

In terms of the auditory perceptual branch, it is common practice to reconstruct the human ability of binaural audio localization by utilizing an array of microphones. With this, the tracking and localization process mainly relies on estimating the DoA of incoming signals. However, after careful examination of state-of-the-art technologies and available environmental stimuli in the targeted mission surroundings, the auditory branch has considerable deficiencies in terms of reliability and availability in the intended field of application. Due to the fact that only passive sensing devices are available, the overall approach relies on the presence of external acoustic sources. In a world designed by humans, active audio information is mainly used to announce a particular event. Permanent acoustic sources are tendentiously designed to be outside of the human audible range since it s mainly considered a disturbing factor unless it is explicitly desired, e.g., in the context of entertainment or by means of music. In contrast to light in the visual field, noise has to be, in most cases, actively produced by individual sources, which results in a somewhat sporadic behavior. Even in the day-to-day scenario portrayed in our targeted urban housing application, no reliable sound source can be utilized as a permanent feature for localization and mapping. Nevertheless, the addition of this, according to findings in the linguistic research, the second most essential type of sensory perception, would be a step in the right direction with the ultimate target of establishing a multi-modal perception framework similar to that of humans. In order to facilitate the previously stated boundary conditions of the front-end, the auditory modality is not integrated into the visual-inertial approach since methods capable of handling multiple sound sources are fairly computationally intensive. Thus, the second type of front-end module must be developed to utilize acoustic stimuli from the surroundings. The

intermediate representation and resulting estimations are then forwarded to the back-end, in which the usability of the data is evaluated and fused with information from other front-end modules as needed.

Last but not least, positional data from wheel odometry sensors are also available as an additional modality. Since the odometry information is provided in the form of a 3-D state vector consisting of a 2-D location and 1-D orientation representation, it is not worth the effort to reroute this type of data through the visual-inertial front-end module. Thus, it can be directly introduced into the back-end for further processing steps. Optionally, a dedicated front-end module can be constructed for odometry data to further enhance the reliability and robustness of the estimation.

While different sensory modalities are included in the consideration, it is evident that the visual-inertial branch forms the most important mainstay regarding tracking, localization, and mapping. Therefore, within the thesis, the target is to develop a novel approach for a visual-inertial front-end module within the context of the proposed multi-modal SLAM framework.

3.2 Multiple Modalities in the Software Domain

After the environmental stimuli are collected from the robot's surroundings, it has to be processed in order to create the ability of machine perception. Similar to the hardware-specific domain, the concept of utilizing multiple modalities can also be found on the software-related side. Building on the findings and conclusions from the previous sections, the following section concentrates on establishing multiple modalities by combining different image abstraction methods and the conceptualization of a novel motion estimation strategy in the visual-based domain.

3.2.1 Multi-Modal Feature Collaboration

From the perspective of computer vision with a particular reference to establishing tracking capabilities in the context of VO, the range of information in an image is far too extensive for the intended task. More importantly, vast amounts of computational resources are required in order to be able to process the sheer volume of data in a matter of milliseconds with onboard hardware on-line in real-time. At this point, it is recommended to distribute the already limited resources to a specific set of regions of interest with certain salient characteristics. For this purpose, it is common practice to insert a pre-processing and selection step to evaluate the available data since the value of the individual pieces of information contained within an image differs considerably in terms of uniqueness and the resulting recognizability. This process is summarized under feature extraction, in which the input data is analyzed and abstracted into a distinctive collection of landmarks. As a result, only a fraction of the input data has to be further processed in the optimization loop and saved to generate an adequate surroundings model for localization and navigation purposes. Especially in the case of mobile robotic applications, this approach is auspicious since the necessary computation resources, especially memory consumption, can be kept relatively low.

In computer vision and image processing, a ”feature” is defined as a piece of meaningful information within the content of an image. In general, features are not restricted by any geometric constraints. Particularly in the context of our targeted field of application, they should, above all, possess an adequate amount of characteristic properties to ensure reliable recognition of the already detected landmarks in repetitively.

Starting from the lowest level, interesting points are one of the fundamental and most popular features. Harris and Stephens [71] introduced the first reliable keypoint detection algorithm in the late 1980s to improve Moravec's corner detector [72]. Since then, it has been improved and adapted to many image processing algorithms, and researchers have proposed new approaches based on different detection techniques. Although it is not particularly difficult to find a reasonable number of characteristics that can be easily recognized by computer algorithms, identifying features that are invariant against photometric transformation presents a more significant challenge. This includes invariance against translation, rotation, change of scale, and covariance to geometric changes. In general, the group of point feature detectors can be divided into two overall approaches. The first group focuses on corner detection, which is defined as an intersection of two edges. Specifically, it is characterized as a point in which the direction of two edges unsteadily changes. Transferred to the image, a corner can be defined as a variation in the gradient in the associated image, which computer algorithms can easily identify. Although stable against rotation, conventional corner detectors are not scale-invariant. This issue is solved by the second group of detectors, which utilizes blob detection and a multi-scale representation in the form of an image pyramid. Unlike corners and edges, blobs are

(a) (b)

Figure 3.1: Illustration of a typical scene in the urban housing scenario. In specific, (a) the living room assembly is portrayed with (b) its contours consisting of line features extracted by a line segment detection algorithm.

characterized as a region of interest, and feature points are extracted by taking a supporting neighboring region into account. The rotation invariance was reinforced by assigning landmarks a specific orientation, which can be achieved, for instance, using the intensity centroid approach [53]. Here, it is assumed that a corner's intensity maximum does not overlap with its geometric center, and a robust orientation is attributed through the resulting vector.

Moving a step further, line features are suitable for describing the contours of human-made objects since such surroundings are built on a Cartesian grid, leading to regularities in the image edge gradient statistics [69]. Although line segment features are promising for localization and navigation purposes, most detection and description research has targeted point and region features in the past decade. In contrast to point detectors, line extraction algorithms are based on edge detection since most geometric lines and line segments are based on the outlines of actual physical occurrences. This forms a lower degree of feature generalization and directly connects this type of landmark with real-world attributes. Point detectors, on the other side, cannot be reliably equipped with this feature based on their design and a higher sensitivity against photometric disturbances. In terms of rotation invariance, a line segment is automatically equipped with a sense of direction. This type of orientation is exceptionally stable as it invokes actual physical occurrences, such as the contours of real-world objects. Even though some point extractors do feature a sense of orientation, as previously stated, it is not equivalent to the one in conjunction with a line feature. In this case, it is based on an approach that is not as sophisticated as the physical background and is significantly more susceptible to environmental influences. Figure 3.1 shows a typical scene in our urban housing scenario and individual line features extracted by a state-of-the-art line segment detector. It is worth mentioning how well the contours of the depicted objects are captured. Combined with appropriate filter settings, the line properties could be even more resilient than they already are.

After the extension from 0-D point to 1-D line features, the image abstraction process can also be handled by higher dimensional landmarks in the form of 2-D geometric shapes all the way up to an item-based concept. In the latter approach, the image is subjected to a semantic segmentation process, during which salient objects can be extracted through further processing steps and assigned as potential features. However, this method represents a very high degree of feature specialization, and thus it tends to lose the capability of being deployed in generic,

previously unknown environments. Therefore, a reasonable balance has to be found between the generalized and specialized approaches.

At this point, it is also worth taking a look at state-of-the-art feature-based approaches in which only one specific feature detector is utilized at once. The main idea behind this concept is to master all tasks that occur within a VO or SLAM algorithm with one feature type to reduce the framework's overall footprint. However, this negatively affects the overall robustness and especially the stability of the entire algorithm, especially in more dynamic situations. For this reason, in the following, we primarily address the two most generic landmark geometries and investigate possible multi-modal setups that can be formed based on them.

Intra-Class Feature Collaboration

As the name already implies, the first approach centers around the concept that detections from different landmark extraction algorithms of a specific geometry are able to collaborate with each other. By utilizing several feature detectors with different detection principles, algorithm-specific landmark collections are generated, and potential synergies between them could be formed to further improve the robustness of the image abstraction process. To make this possible, individual features are not compared one by one on an atomic level, as it would diminish the applicability and effectiveness of this approach. Due to the fact that the utilized feature detectors are designed to target various characteristics within the visual input data, a complete overlap between the properties of the detected features is rarely expected. For this reason, it would be more effective to take a step back and instate the categorization and ranking process at a superordinate layer, which assesses individual image regions according to their respective significance. This can be realized by dividing the image into individual sections and evaluating the resulting boxes based on the incorporated features. The evaluation process itself is configured with different emphases based on the specific use case. Therefore, it depends on the absolute quantity, the spatial distribution of the detected features, and their relative quality measure, e.g., based on their characteristic response value ranking within the associated feature collection. With this, the final target is to identify sections with a higher probability of containing features that are more likely to be robust and repeatedly detectable other than unstable phantom features caused by, e.g., photometric distortions.

Figure 3.2 illustrates the theory behind the intra-class feature collaboration by exemplarily depicting the spatial distribution of landmarks detected by three point feature detectors, each following a different extraction philosophy. In terms of a density-based classification approach, several areas with different properties can be identified in the schematics. The first category consists of sub-sections on which only landmarks from one feature detector can be found. At the same time, this type of area also represents the lowest level within the results of the classification process since any other utilized feature extraction methods do not confirm the detections. Therefore, they are considered to be relatively unstable, and features from this area category are more likely to contain a significant amount of outliers. As for the time being, these landmarks should be set aside and kept as a backup solution unless a sufficient number of features from more advanced levels cannot be identified in the image.

Taking it a step further, the following categories contain features generated from two or more different extraction algorithms, either in a small or large quantity. Especially in the latter case, it implies that the photometric properties within these respective regions are favorable in terms of their texture and visual composition, which computer algorithms can easily characterize.

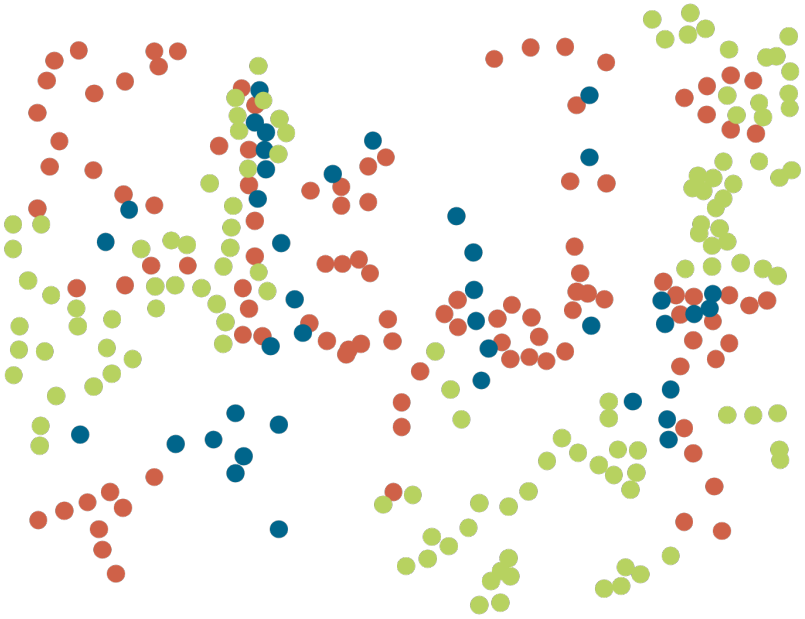


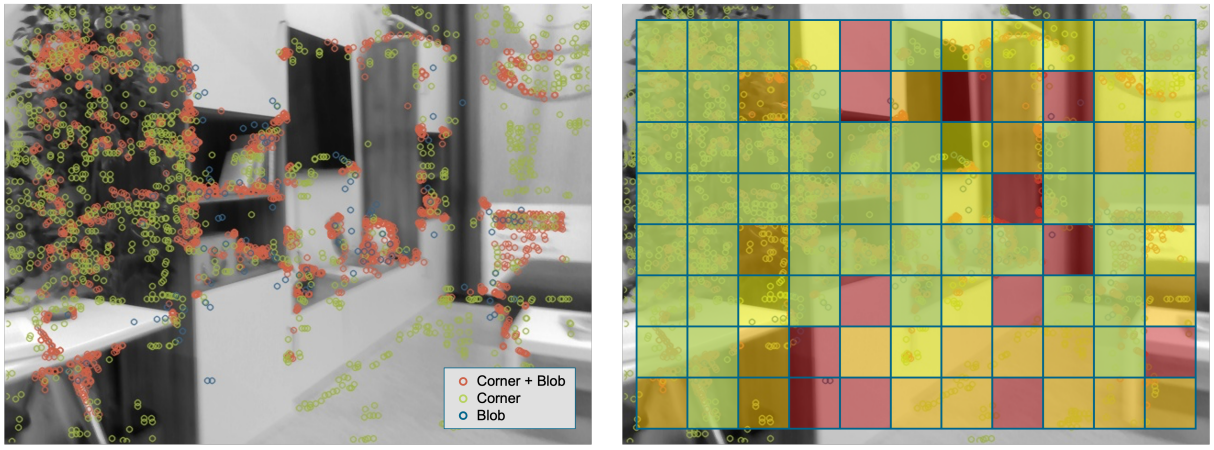
Figure 3.2: Schematic representation of the theoretical approach behind the intra-class feature collaboration.

Therefore, it can be assumed that landmarks within these areas are particularly distinctive, which would significantly improve the overall stability and robustness of the resulting algorithm.

In addition to the previously introduced levels, a further subdivision can be achieved, for instance, in a density-related approach. Here, a ranking is established within each category based on the absolute number of detected features within a particular area. As a result, an ordered list of preferences is generated, which can be utilized in the following steps within the processing pipeline of a VO algorithm. Furthermore, a distance-based nearest-neighbor concept can also be employed to be able to create the ability to provide an even more accurate assessment. At this point, a pre-defined distance to the nearest features of the same type is determined and applied to the overall assessment of the individual regions.

Supplementary to the previously presented theoretical illustration, a real-world example is depicted in Figure 3.3. In this particular instance, three state-of-the-art point feature detectors are deployed on the given image in Figure 3.3a, each with a different extraction technique. While the green and blue colored landmarks are identified by a respective corner and blob detection routine, the algorithm behind the orange ones has both paradigms in focus. In general, there are several noticeable areas when examining the resulting spatial distribution of the detected landmarks. On the left-hand side, an accumulation of features from all three detectors can be encountered on the indoor plant, as the individual leaves create a highly textured area with a large number of corners and potential blobs in the resulting image.

Further to the right, a similar arrangement can be identified in the TV setup, in which features are concentrated on the located objects. In contrast, only the corner detector can detect meaningful landmarks on the not exceptionally well-structured floor covering. In this case, no further information can be gained about their properties since these features cannot be combined with detections from other types of detectors in any way. Therefore, this particular area receives a lower evaluation, and these features should be disregarded for the time being. As a result, a valence ranking is generated by assessing each previously defined subsection in an ordered pattern, with which individual weights are assigned to the different areas for further



(a) (b)

Figure 3.3: Illustration of the evaluation and prioritization process within the intra-class feature collaboration. (a) The initial landmark collection for the valuation procedure is provided by three state-of-the-art point feature detectors and (b) the regions are assigned with their respective valuation score.

processing steps. Figure 3.3b exemplarily depicts the valuation and weighting process based on the previously used image. For illustration purposes, the image is divided into 88 sub-regions. In terms of practical application, the utilized grid would be much finer, and thus an adequate evaluation of the respective sub-areas can be achieved.

Apart from evaluating the detected features and identifying image sub-regions with potentially higher stability, invariance against the change of scale can be implicitly created as a secondary benefit. Since some of the state-of-the-art landmark detectors do not feature this type of invariance, they are not particularly suitable for long-term localization and mapping tasks. This is caused by the fact that the exact size of the feature and its surrounding region of interest cannot be reliably estimated. Although it might only have little effect on the performance of the targeted VO algorithm within the thesis, the impact on the overall SLAM framework is significantly higher. In the context of the intra-class feature collaboration, the landmark size can be implicitly estimated by considering nearby features' dimensions, which are invariant against the change of scale. Even though this approach cannot identify the exact size, it is a good start to give these features a sense of size and prepare them for mapping and localization tasks as a first step.

Inter-Class Feature Collaboration

Similar to the means of categorization in the hardware-specific domain, landmarks with different geometric properties are also able to collaborate with each other in order to enhance the robustness of the processing pipeline. Here, the basic idea behind this approach is to effectively combine given feature properties, which are unique to the respective detector class. By doing so, potential synergies are created with the target of maintaining and possibly extending the continuity of the resulting estimations.

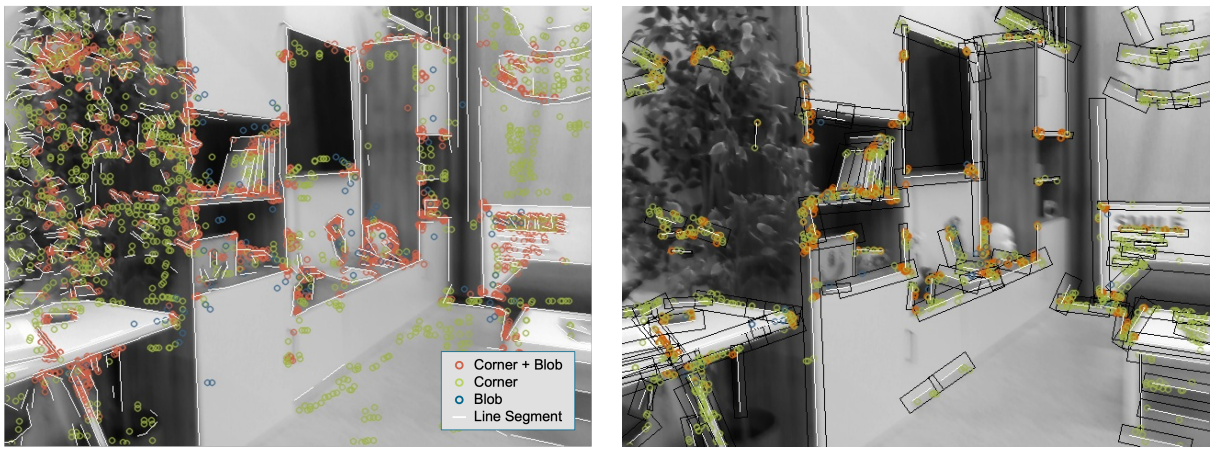
While the focus was mainly on the characteristics and possible correlations between point features within the feature extraction process in the previous approach, we will investigate possible methods surrounding the feature matching module. Although point features are the most commonly utilized landmarks in the context of VO and SLAM, they do not provide the best starting position in every use case. In parallel to unfavorable environmental conditions and deteriorating image quality, the feature extraction process is also negatively affected. Especially

Figure 3.4: Schematic representation of the theoretical approach behind the inter-class feature collaboration.

with increasing levels of blur distortion, the overall texture of the image and the distinctive characteristics, such as corners, are softened. Consequently, it diminishes both the quantity and quality, particularly stability, of the detected landmarks, which serves as the baseline for the subsequent processes in our feature-based method. Due to the fact that the targeted application is not purely theoretical, the matching process cannot be considered faultless, even under the most favorable conditions. Therefore, the motion estimation module always includes a separate outlier rejection routine in which incorrect matchings are identified and neglected for the subsequent optimization process.

In order to further improve the accuracy of feature matching, especially under the most unfavorable situations, an additional evaluation and filtering step has to be introduced to the processing pipeline. One possible solution is contained in the term of inter-class feature collaboration, in which characteristics of different landmark geometries are combined. Apart from the well-known point features, we are also targeting to utilize line segments, which are the most simple form of geometric features. Since human-made environments contain a significant amount of linear geometries, it provides a valuable supplement and is a reasonable choice in our first approach. In addition, line features are considered quite robust in terms of detection and matching capabilities, even in unfavorable situations, due to the fact that they are mostly related to real-world properties. From an experimental perspective, we already conducted a study in [70] in which line segments are identified as exceptionally reliable in terms of their repeatability and matching score, among other findings.

After the scope of available data has been determined in the previous selection, the next step is to connect the individual pieces of information, thus uniting the advantages of both feature geometries. Since the target is still to identify a collection of landmarks with higher stability and, therefore, robustness with a particular focus on feature matching, the process of geometric clustering is introduced as a potential solution. Basically, landmarks with different geometric properties are connected based on their location in the image, thus creating a tight correlation



(a) (b)

Figure 3.5: Illustration of the line clustering process within the inter-class feature collaboration. (a) The initial landmark collection for the valuation procedure is provided by three point feature detectors and one line segment detector. (b) Line clusters are assembled according to a predefined scheme.

in the form of a geometric cluster. Within the process, a bounding box is created around the higher-dimensional feature, and the remaining landmarks from other detectors are distributed in the respective collections based on their position. By assigning them into dedicated groups, an additional cross-check relation is established in terms of feature matching since the collective structure of a cluster remains consistent throughout the photometric transformation, as illustrated in Figure 3.2. At this point, the created geometric clusters from the source and target image are implicitly matched by evaluating the matching results of their members. Based on the assumption that the matching routine correctly identifies the majority of the feature matchings, this approach can be used to isolate possible misalignments and matching outliers. Similar to the previously stated intra-class collaboration, the final evaluation is determined based on a reasonable weighting of individual landmarks. As a result, the weights applied to the individual characteristics are not designed to be continuous but rather binary since this approach only allows the identification of whether a feature is considered an outlier. In the case that a particular matching is identified as correct, a static weighting coefficient with an appropriate magnitude is applied to the previously defined scoring system. Indeed, it is also possible to use additional ranking information generated in the associated feature collections to generate a variable weighting factor. However, in the first step, only binary weights are applied as the result of this collaboration method.

To further elaborate on the process of inter-class feature collaboration, the scenario that already appeared as a practical illustration in the context of intra-class collaboration is consulted again in Figure 3.5. In addition to the three illustrated types of point extractors from the previous example, Figure 3.5a includes the detections of a modern line segment detector in white. In the next step, a bounding box with a predefined size is formed around the line segments. All point features within the specific region are bundled into a geometric cluster, more precisely designated as a line cluster. Figure 3.5b illustrates the resulting line clusters. By taking a closer look, it is noticeable that not all detected lines are considered potential baselines and have a designated feature collection formed around them. This is because, in minor cases, the line detector also detects a line that cannot be classified as robust. Therefore, it is considered to be an incorrect detection instead. In order to identify and neglect such occurrences, a reasonable filter has to be

implemented. In the simplest case, line segments have to be longer than a predefined threshold value to be classified as a baseline.

3.2.2 Collaborated Motion Estimation

In order to obtain a high-quality result from the VO application, it is not sufficient enough to only provide the motion estimation modules with a well-selected and aligned set of input data but also to employ a reasonable optimization strategy for the calculation of the relative camera motion. While previous methods target the generation of an optimum feature collection, the focus is directed towards investigating potential multi-modal approaches within the motion estimation process. As mentioned in Section 2.3, the camera motion can be estimated from combinations of 2-D and 3-D correspondences. Depending on the selected data dimension, the resulting transformation can be derived from the essential matrix, obtained by minimizing the reprojection error or estimated by aligning the given point clouds. Even though the approaches above are fundamentally different in their mathematical background, they share a common aspect in the form of input data representation. The optimization strategies are all based on a collection of individual data points, which are considered the most atomic way in terms of information representation.

Although these approaches are considered state-of-the-art, the algorithms are still reasonably sensitive to remaining matching outliers, thus the overall distinctiveness and stability of the given point features. In order to further increase the robustness of the optimization process, one possible solution is to integrate valuable properties of line features into the motion estimation pipeline. The basic idea behind it is to utilize the additional orientation information in order to achieve a more robust calculation of the relative rotation. Alongside the already instated motion estimation approach, the directional information of the line segment features is established as an additional constraint, resulting in a more robust and accurate calculation of the relative movements. As a result, a multi-modal motion estimation process is formed, as illustrated in Figure 3.6.

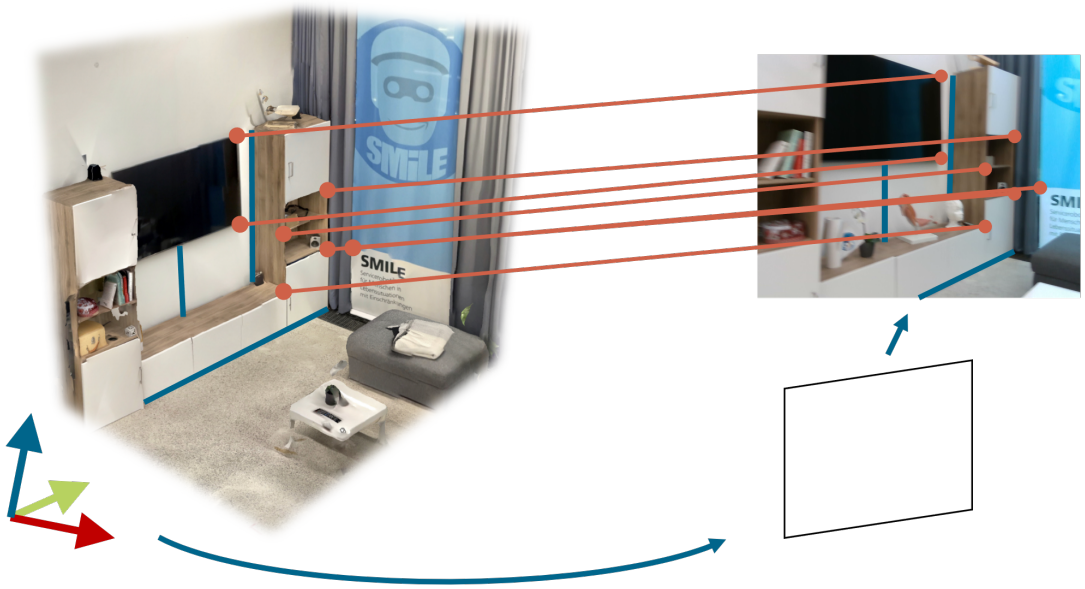


Figure 3.6: Schematic representation of the theoretical methodology behind the collaborated motion estimation.

On the practical side, the collaborated motion estimation can be realized in several ways. Starting from the conventional side, the first modality is to derive the relative transformation from a set of 2-D and/or 3-D correspondences. Although the minimization of 3-D-to-2-D reprojection error is more beneficial as it supports a 3-point minimal solution instead of a 5-point requirement, it is worthwhile also to investigate the potential of the 2-D approach because of the poor depth data quality of the utilized RealSense cameras. However, the 3-D rigid body transformation as well as the absolute scale cannot be calculated without further information in the third spatial dimension. For this reason, the missing parameters have to be calculated by a separate estimation routine incorporating the depth information as a supplement of the coordinates of individual point features on the image plan. By separating these processes from each other, only 2-D coordinates from the image plane, which can be considered highly accurate and otherwise the same magnitude of imprecision, are used for initial estimation of the relative motion. Thus, this process is not affected by the properties of the depth information at all. In this case, the variance in data accuracy does not influence the calculation of the initial motion estimation and can therefore be minimized. On the contrary, the resulting motion estimation strategy would be significantly more computation intensive. For this reason, the sequential 2-D-to-2-D approach is disregarded along with the disadvantages and stability issues of the 3-D-to-3-D method. Therefore, the 3-D rigid body transformation is estimated using the 3-D-to-2-D strategy.

In terms of utilizing the properties of line correspondences, there are two basic approaches with which this additional constraint can be incorporated. At first, line features can be seamlessly integrated into the previous optimization process. In this case, the given line segments are deconstructed into a line of point representatives, and correspondences between the feature collections are established accordingly. By assigning properly selected weighting parameters to these points within the optimization process, the more reliable properties of line features are implicitly embodied in the overall motion estimation module. Apart from the integrated approach, the directional information can also be taken into account by introducing a new optimization parameter. In addition to the 3-D and 2-D coordinates of the feature points, the relative rotation can be aligned by minimizing the overall line orientation error between 2-D line correspondences. By doing so, the associated optimization processes are carried out step by step in either a sequential or parallel manner. In general, the latter approach is considered the more advanced method, leading to more predictable results and eliminating the unpleasant fine-tuning process of the weighting parameters.

3.3 Multi-Modal Methods within the Context of Visual Odometry

After several multi-modal methods were proposed in the previous sections, these approaches are contextualized into the context of VO and SLAM in a more practically oriented manner in the followings. Figure 3.7 summarizes the previously proposed methods for establishing multiple modalities in the software-related domain. With this, it also provides a visual illustration of the potential operation site within the context of the processing pipeline of our targeted VO application.

Starting with the intra-class feature collaboration, it is utilized for sorting and prioritizing detected landmarks and the valuation of potential regions of interest. Findings obtained from the processing step are then used in the feature matching module for outlier rejection purposes. Furthermore, the resulting information can also be applied to obtain previously unknown properties of specific landmarks. With this, the size of features, which is not equipped with invariance against the change of scale, can be approximated based on properties of nearby size-invariant landmarks.

Further down the chronological order, the intra-class feature collaboration is based on the prioritization of features with the help of geometric clusters. As a result, the established correlations are utilized for outlier rejection after the feature matching process in the form of cross-check validation.

At last, the collaborated motion estimation remains the final approach to establish multiple modalities in the software-related domain. Within the context of the VO processing pipeline, it emerges entirely into the motion estimation module and has the task of entirely substituting the conventional methods.

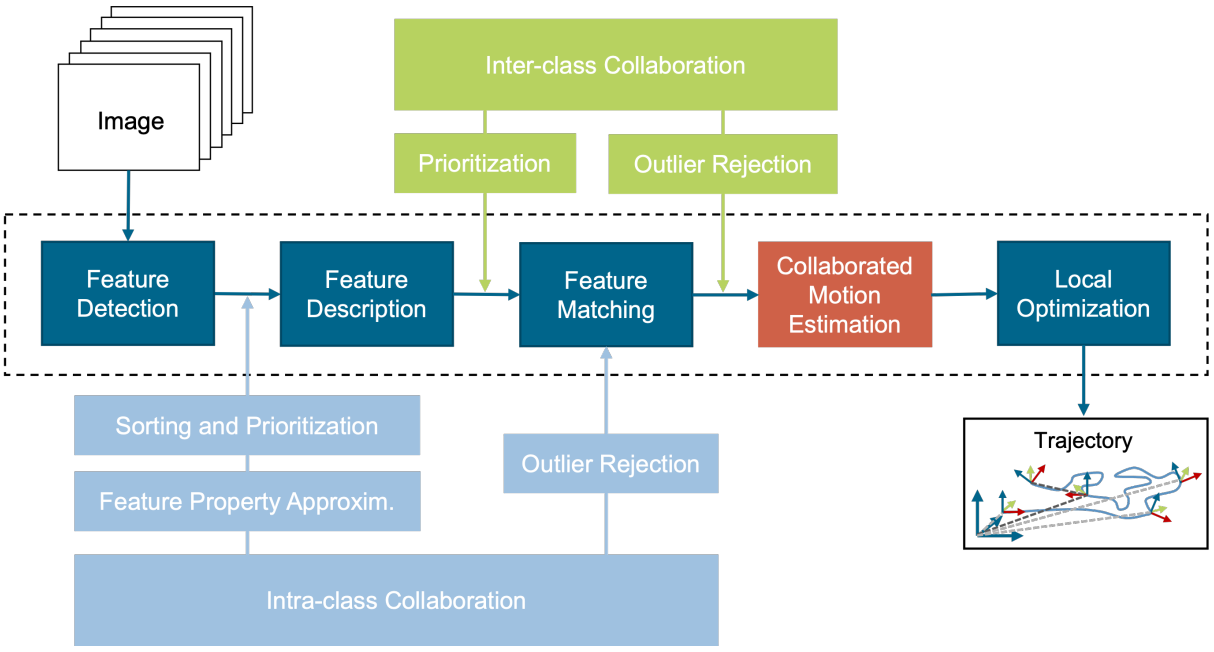


Figure 3.7: Multi-modal methods in the software-related domain within the context of a feature-based VO application's internal structure

4 Implementation of Multi-Modal Visual Odometry

Building on the considerations from the preceding chapter, we head one step further and practically apply the introduced methods to the development and implementation of our intended visual-inertial front-end module. While theoretical assessments and standalone experiments are good ways to identify the potential of novel approaches, the system-related behavior and performance characteristics can only be evaluated by integrating them into a working VO algorithm. However, there is no VO/SLAM framework to our knowledge that can provide the necessary structural flexibility and extendability to incorporate our targeted multi-modal setup. Therefore, the decision was made to construct a novel VO application from scratch.

This chapter introduces to the Multi-Modal Visual Odometry (MMVO) framework, which is the primary contribution of this thesis apart from the results of the theoretical assessment. After the general overview of the system level in Section 4.1, we will consolidate the task-related, more component-oriented domain in the following sections and address the specific implementation details of the individual modules in chronological order.

4.1 General System Overview

Following the basic architecture of a conventional feature-based VO algorithm, MMVO is constructed to incorporate the multi-modal methods presented in the previous chapter. Since a high level of modularity is considered one of the key design aspects, each main processing module is equipped with a well-defined data interface. This way, individual components within the processing pipeline can be easily interchanged in just few steps and replaced by another module based on a different, possibly more advanced concept for future developments. In addition, the intended structural division also allows utilizing the given system as a test bench for other state-of-the-art methods within the individual sections, which can be made available by providing a comprehensive library.

Figure 4.1 gives an overview of the framework's overall structure. From a more abstract perspective, the system's working principle can be summarized as follows: For every new frame, the relative displacement to the current keyframe, which serves as the source image during the motion estimation process, is estimated. Within this procedure, various types of point and line features are identified from the initial image. In the next step, the detected landmarks are equipped with a unique signature for the subsequent matching process. Here, reasonable correlations between the source and the target image are identified and forwarded to the motion estimation module. At the same time, the initial landmark collections are being further processed and fused into an adjusted set of features according to the previously introduced methods. Once all the requisite data are available, the relative displacement to the corresponding keyframe and

its true scale is calculated in combination with the pixel-wise aligned depth information. As an option, this process can be assisted by including estimations from the predictive state-space module, in which the IMU measurements are processed separately for pose tracking. The next component in the toolchain is responsible for the local optimization of the concatenated transformations computed by the previous steps. Within this module, locally constrained pose-graph optimization and windowed BA can be performed to improve the algorithm's tracking accuracy further.

Apart from that, the main motion estimation pipeline is supplemented by an intermediate stage for keyframe identification. Depending on a pre-defined set of criteria, a new keyframe is created and handed over to a temporary collection, which serves as the source image in the upcoming motion estimation iterations. Due to the required real-time processing capabilities, these tasks within the front-end are separated and distributed over several threads. As the last step, the estimated relative displacement is transformed into the respective frame of the currently used coordinate system and its origin, either set by the initialization process or automatically triggered after an LoT event occurs. In the end, the relative pose and the respective origin are forwarded to the back-end for mapping, long-term localization, place recognition functions, and sensor fusion purposes.

In order to better understand the system's individual components, the building blocks are explained in greater detail in the following sections. Since the conceptional design of MMVO strictly follows the internal structure of a feature-based VO application, it incorporates, among others, the central functions of landmark extraction, feature matching, motion estimation, and keyframe identification. Furthermore, methods surrounding the generation of potential synergies for evaluating and prioritizing the given feature collections are summarized under the term Feature Entity Fusion and Filtering (FEF^2). In contrast to the previously mentioned vital components, however, the implementation of reasonably configured modules for IMU data handling and optimization is associated with a high amount of workload, which cannot be ignored. For this reason, the realization of these modules would far extend the scope of the thesis. Therefore, these modules are constructed as inactive components at the corresponding locations and will not be elaborated in greater detail.

4.2 Landmark Extraction

In general, the landmark extraction terminology is composed of the sequential arranged processes of feature detection and description. At first, the visual information of an image is analyzed and abstracted into a collection of regions of interest, forming the foundations for subsequent processes. The selection is then passed to a description algorithm, which assigns the characteristic area with a distinctive mark considering its surroundings.

Although it is not difficult to find enough characteristics, which can be easily recognized and characterized by computer algorithms, but to identify features that are invariant against photometric transformations, i. e., translation, rotation, change of scale, and covariance to geometric changes. For this reason, we conducted a "Requirement Analysis for Perception On Assistant Robots in Multi-Modal Environment Condition" [70]. In addition to the analytical examination of promising state-of-the-art feature extraction algorithms, an experimental study was carried out based on real-world datasets from mission-related environments where Rollin' Justin is typically situated. With this, prevailing environmental properties are evaluated to identify the best-suited

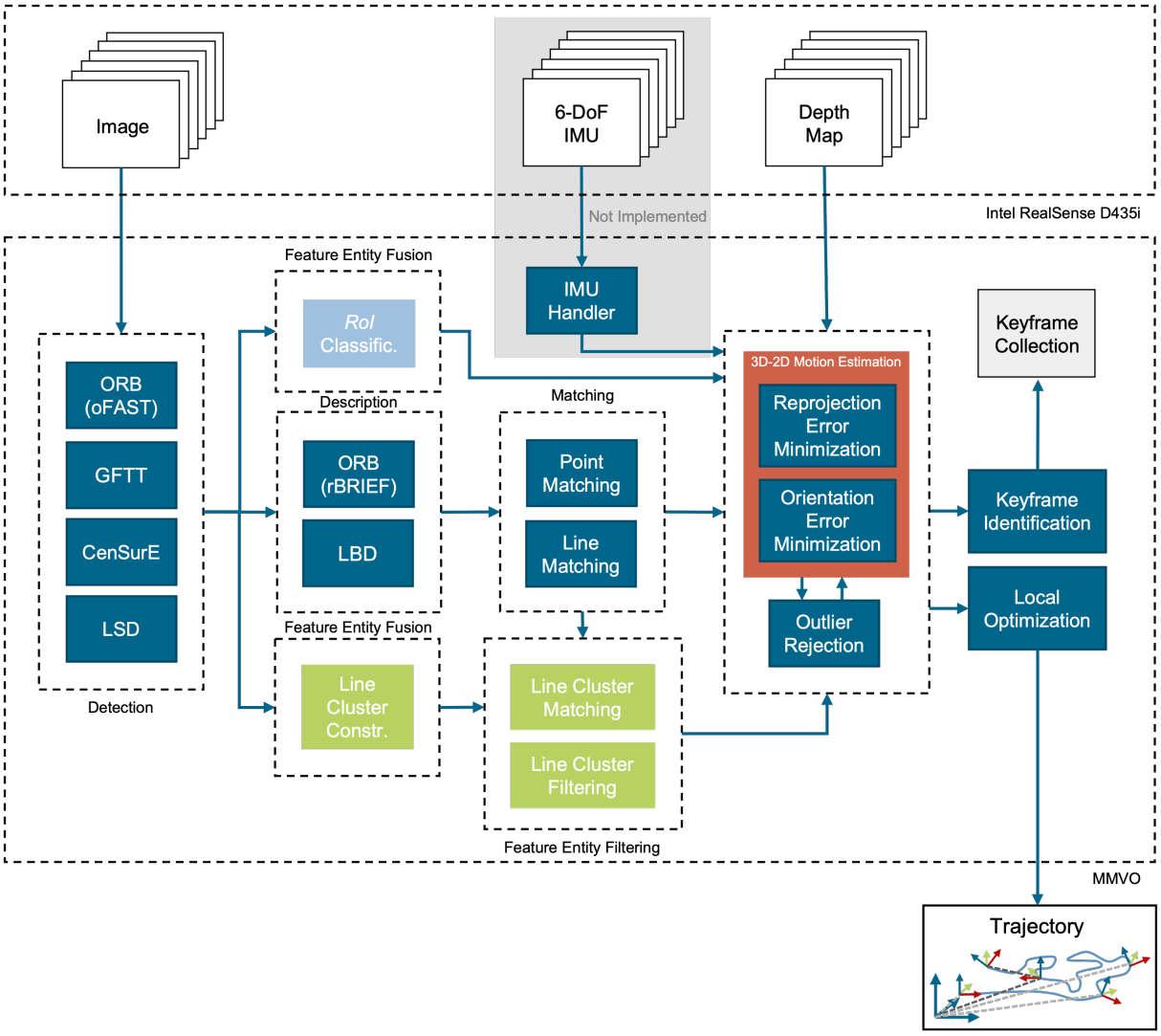


Figure 4.1: Overview on the general system architecture of the Multi-Modal Visual Odometry (MMVO)

visual abstraction and characterization frameworks. As a result, findings and realizations were summarized in several recommendations, including a proposed collection of feature extraction methods, which can be employed within the multi-modal image abstraction framework in the followings.

4.2.1 Feature Detection

Starting with the first step in the processing pipeline, the feature detection module is constructed with a particular emphasis on the capability to facilitate a variable number of individual detection algorithms. Therefore, this building block is divided into two specific types of components based on their associated functionalities and depth of integration. At first, the landmark detector manager acts as the superordinate module and is responsible for the general administration of the image abstraction process. Basically, it is in charge of converting and distributing the incoming visual information to the selected detection algorithms and managing the obtained landmark collection for subsequent processing steps. For this reason, it is directly embedded in the internal structure of MMVO. On the contrary, the second component is assigned to the actual image abstraction process. In order to fulfill the overall modular characteristic of the

front-end, different feature detection algorithms are implemented in individual function blocks with a generalized interface definition. In contrast to the management framework, these modules are easily interchangeable, and the collection of landmark detectors can be customized according to the specific use case. An external configuration file defines the exact composition and associated tuning parameters. Furthermore, a guardian mechanism has to be implemented to prevent regional accumulation of the detected landmarks since distinctive characteristics are not evenly spread throughout an image. To counteract these circumstances, a grid of 64 columns and 48 rows is applied to the image, in which the associated detectors are deployed separately within each cell. In the end, the provisional landmark collections are composed according to a relative threshold in order to ensure a homogeneous distribution over the image.

As the result of the previously introduced experimental evaluation, four different feature detection approaches, consisting of three point and one line feature detectors, are considered the most promising in the context of the anticipated field of application and the expected environmental conditions. Each of them is equipped with an individual focus on specific environmental circumstances, from which potential synergies can be generated by reasonably combining the characteristics of the landmarks. Therefore, in the first approach, these auspicious detection algorithms will be integrated into MMVO for our anticipated multi-modal setup. In order to provide an overview of the utilized methods, they are individually introduced in the followings.

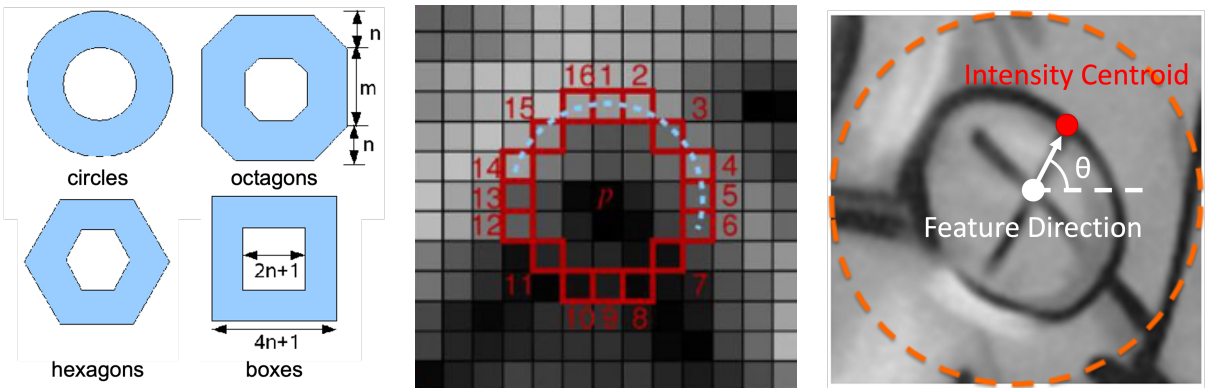
Good Features To Track

The first algorithm used in the proposed framework was introduced as a direct advancement of the Harris corner detector [71] in 1994. Inspired by Moravec's work [72], Harris and Stephens proposed the first meaningful corner feature detector [71] in 1988 by minimizing the auto-correlation function to compare an image patch against itself shifted for small displacements. Deducing from the mathematical model, they developed a measure for the qualitative quantification of corners. Depending on the magnitude of the response function, the examined region is classified as a corner feature if it surpasses a selected threshold value. Shi and Tomasi proposed their Good Features To Track (GFTT) [73] by redefining the decisive criterium as follows:

$$R = min(\lambda_1, \lambda_2), \tag{4.1}$$

in which λ_1, λ_2 are two eigenvalues of the auto-correlation matrix. Their research showed that the physical correspondence of feature points and the classifier's robustness could be further improved by an alternated definition of the response function.

In the experimental evaluation, GFTT achieved the best results based on the performance metrics. Especially in less-textured environments and more dynamic situations in which the increased presence of motion blur impairs the resulting image quality, this detection algorithm can still identify a sufficient number of distinctive features. Although GFTT cannot be assigned to SLAM-related tasks under normal circumstances in a standalone setup, it provides a valuable aid in our targeted multi-modal setup, particularly in challenging environmental conditions.



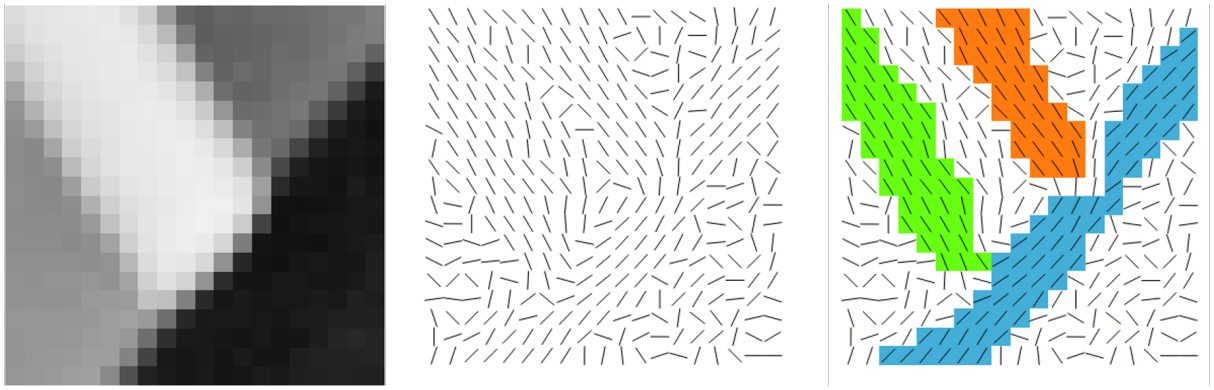
(a) (b) (c)

Figure 4.2: Illustration of (a) CenSurE bi-level filter geometries, (b) FAST test pattern, and (c) theory behind the intensity centroid. Images adapted from [74, 75].

Center Surround Extrema

Apart from the corner detectors, the second method uses the blob detection technique to identify suitable landmarks. Since it is based on a multi-scale representation in the form of an image pyramid, the location accuracy of the features detected by earlier approaches deteriorates with increasing octave and the consequent sub-sampling of the initial image. Agrawal et al. compensated the issue by introducing a new method for the approximation of Laplacian of Gaussian (LoG) in their proposed Center Surround Extrema (CenSurE) detector [74] in 2008. As illustrated in Figure 4.2a, the estimation is accomplished using bi-level center-surround filters, with which full spatial scale can be achieved at every image in the scale space. Multiple bi-level filter geometries can be used, ranging from rectangular box filter (quadragon) to circular filter as a polygon with infinite edges. After the filter responses are computed at each pixel in the image, potential feature points are identified by local extrema detection. At this point, a non-maximum suppression is performed over the scale space in a 3×3×3 neighborhood. They further enhanced their algorithm's robustness against noise by employing a threshold value-based selection process to reject candidates with low contrast. In addition, poorly localized keypoints along edges must be eliminated since they also negatively contribute to noise sensitivity. Therefore, the curvature of the surrounding area is analyzed by the calculation of a Hessian matrix at the keypoint's coordinates utilizing the proposed procedure by Harris and Stephens [71].

According to the results from the preliminary study, CenSurE performed very well, especially in the achieved matching score in the characterizing stand-alone objects in the evaluated scenarios. Although the number of detected correspondences is among the lowest in the participating detectors, it achieved the highest repeatability score. Based on the characteristics of the occurring pattern on the flooring, feature detection is naturally suppressed by the algorithm to a certain degree, where the number of detections is neglectable. For this reason, CenSurE would be a valuable complement to the multi-feature approach and is the ideal candidate for specific tasks in which features from the floor are undesired or have to be separated from others.



(a) (b) (c)

Figure 4.3: Illustration of (a) exemplary image section, (b) level-line field, and (c) identified line support regions. Images adapted from [76].

Oriented Features from Accelerated Segment Test

The last metric combines the advantages of the corner and blob detection approach. Although other feature detectors, such as the Difference of Gaussian (DoG) in the Scale-Invariant Feature Transform (SIFT) and Fast-Hessian in the Speeded-up Robust Features (SURF), have already generated satisfactory overall results, they are very opulent in the demand of necessary computational power and the subsequent high time consumption. Hence, Rosten and Drummon developed Features from Accelerated Segment Test (FAST) with the target of computation time minimization for real-time on-line applications in 2008. The authors developed an alternative metric in their publication [75] to identify keypoint candidates, in which the comparison is only happening in the image dimensions. As illustrated in Figure 4.2b, a circle of 16 pixels is built around the pixel of interest (PoI) without considering any information from the scale dimension. Whether it should be classified as a potential keypoint, respectively corner, is based on the deviation of surrounding pixels to the Intensity of the examined pixel. Drawbacks caused by the relaxation of the deciding factor are compensated using non-maximum suppression in the case adjacent feature pixels are detected. A decision tree is created to determine the statistically best choice of the evaluation starting point by utilizing machine learning principles, which also benefits the detection speed.

Rublee et al. robustified the detector by utilizing the Harris score [71] as an additional metric to improve the corner feature quality, resulting in oriented FAST (oFAST). Furthermore, they assigned an orientation to the detected keypoints through the intensity centroid theory in conjunction with developing their ORB algorithm [53]. The direction is represented by the vector between the corner's intensity maximum and its geometric center, as illustrated in Figure 4.2c.

In general, the preliminary study identified this detector as the all-rounder algorithm. Although it never achieved the best performance metrics, it was ranked as the method with the most comprehensive detection abilities in the evaluated scenarios. Therefore, it provides a solid foundation for our targeted multi-modal approach to constructing a basic detection structure.

Line Segment Detector

Moving to the field of one-dimensional representatives, one of the most popular algorithms was introduced by Gioi et al. in their Line Segment Detector (LSD) [76]. In particular, it was developed as a robust and self-adjusting framework where no manual parameter tuning is necessary. As illustrated in Figure 4.3, the line support regions are identified by examining the computed level-line field from the source image, which in turn consists of each pixel's level-line angle. The solution space is further downsized by a statistically based *a contrario* approach, and noise-related identifications are eliminated. A potential keyline candidate is classified as a line segment if the corresponding geometrical object, a rectangle, is associated with it. Those aligned points are identified by the directional correctness of their gradient orientation, which has to be within a tolerance region. At last, the number of false alarms is constructed as the final classification measure and compared to a selected threshold value ε. In case the number of false alarms value of the examined rectangle is smaller than ε, an ε-meaningful rectangle is identified, and thus, the line segment candidate is classified as a keyline.

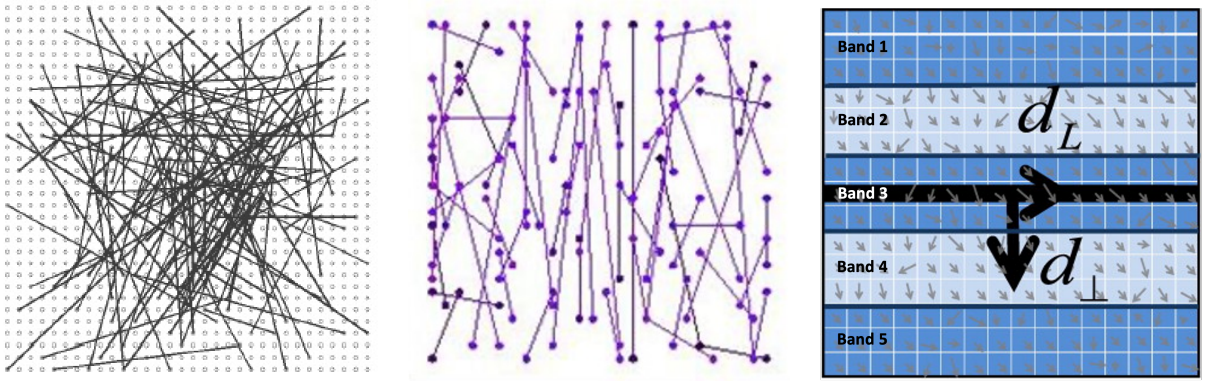
This detector shows great potential in the given environments should be included in a supporting role due to its high robustness against blur disturbance and the more solidified orientation information it includes. While this particular detector does not stand out from the performance parameters of the remaining detection algorithms, it nevertheless achieved similar results. This is mention-worthy because the selected benchmarking metrics are more demanding than those for point features since it implicitly inquires its orientation as a further evaluation criterion. All in all, it proves the outstanding potential of line features for their utilization in human-made surroundings.

4.2.2 Feature Description

After identifying stable and transformation-invariant key features, each element has to be equipped with a unique signature for comparison and recognition purposes. In this context, one can also mark it as the feature's fingerprint since it always contains information from its immediate neighborhood.

Similar to the construction on the detection side, this second building block within MMVO's main processing pipeline is divided between the landmark descriptor manager and subordinate components accommodating the individual description algorithms. The exact composition and the associated tuning parameters are also defined in an external configuration file and can be modified accordingly.

While feature detectors can be deployed in parallel in an extraction framework, feature descriptors are neither interchangeable nor designed to collaborate with others. This is because each descriptor characterizes the point of interest and its surroundings by a different pattern, which can only be interpreted by the same algorithm. As a result, only one description method can be chosen for the entire run-time of the framework. Since two different landmark geometries are utilized in our targeted multi-feature method, a two-pronged approach has to be applied in order to prepare the detections for the following processing steps. Therefore, based on the recommendations of the preliminary study, two different feature description concepts, one for the characterization of point features and the other for line segments, will be included in the first approach.



(a) (b) (c)

Figure 4.4: Illustration of (a) BRIEF sampling pattern, (b) ORB sampling pattern, and (c) characteristic line bands in LBD. Images adapted from [53, 77, 78].

Rotated Binary Robust Independent Elementary Features

From today's perspective, feature descriptors can be classified into two separate categories based on their associated data types. Even though distribution-based, therefore floating point, descriptors such as SIFT and SURF are well-constructed transformation-invariant algorithms, the feature signatures are not stored efficiently. In order to further decrease the descriptor size, the defined neighborhood around a keyfeature could also be classified by a relatively small number of pairwise intensity comparisons. Accordingly, a bit vector is incrementally composed as stated by a prescribed pattern. Calonder et al. proposed the Binary Robust Independent Elementary Features (BRIEF) as one of the first methods utilizing this kind of abstraction [77]. Figure 4.4a illustrates the sampling pattern of arbitrary point pairs from an isotropic Gaussian distribution with the spread of $\sigma^2 = \frac{1}{25}S^2$. This design is chosen for the sampling process within BRIEF, as it produces the best results in terms of recognition rate. However, BRIEF is not rotation-invariant, therefore, very sensitive to in-plane rotation. For this reason, Rublee et al. proposed the idea of rotated BRIEF (rBRIEF) as the descriptor for their ORB algorithm [53], targeting the addition of rotation awareness without compromising the computational and matching speed of the native BRIEF. Before the comparison tests, image patches are smoothed by using integral images. Each test point consists of a 5×5 sub-window of a 31×31 pixel patch, as depicted in Figure 4.4b. Basically, rotation-invariance is achieved by utilizing the provided orientation information and aligning the sampling pattern accordingly before the comparison process, therefore, constructing a rotated version of BRIEF. As a side effect, the variance decreases in each description string, making a feature less discriminative since it responds less distinctively to inputs. This is compensated by a greedy search algorithm, which iterates through all possible binary tests to find combinations with high variance and low correlation at once. As a progression to BRIEF, the authors have kept the number of elements at 256, thus corresponding to 32 bytes per keypoint.

Since only one point descriptor can be selected, it was decided to utilize rBRIEF as the feature description algorithm. Although the native BRIEF implementation is roughly three times faster than the "steered" version, the latter algorithm is still the more advanced choice, as it offers the best balance between reliability, robustness, and required computation time according to the preliminary analysis.

Line Band Descriptor

Apart from the point feature descriptors presented above, one-dimensional feature representatives have to be described differently. Zhang and Koch developed their Line Band Descriptor (LBD) [78] as an efficient and robust algorithm for the characterization of line segments and their line support region (LSR). As the name already implies, LBD splits the detected LSR into a set of bands parallel to the longitudinal side of the rectangle, all with identical lengths as the line itself. For each line band (LB), the orthogonal direction to the LB at the clockwise side is computed, and a reference frame centered in the middle point of the line is created. A Gaussian function is applied to the gradients of the pixels along the newly created direction with the subsequent assignment of global and local weights. After pre-processing, the algorithm calculates the individual band descriptors with respect to adjacent bands by accumulating gradients in the respective row. The final description matrix is constructed by stacking up all results together.

Each line band is represented by an eight-dimensional floating-point vector, comprising its standard deviation and mean vector. By default, the number of LB is set to 32, which yields 256 floating-point values, thus 1024 bytes in total. In the implementation process of MMVO, a runtime-optimized binary version of the native floating-point descriptor was integrated. Within this approach, an 8-bit string is generated by comparing each band descriptor and concatenating 32 comparison strings. As a result, each line segment is represented by 256 bits, thus 64 bytes in total.

4.3 Feature Matching

After the extracted landmarks are assigned with a unique signature regarding their characteristic surroundings, necessary correspondences between the feature collections of the source and target image have to be established. Similar to the construction of previous modules, the third building block within MMVO's main processing pipeline comprises of two different types of components, targeting a modular design. While the landmark matcher manager is responsible for administration and data distribution tasks, different matching algorithms are accommodated in individual subordinate modules. With this, the exact composition and the associated tuning parameters are also defined in an external configuration file and can be modified accordingly. Following the general structure of the feature description toolchain, a two-pronged approach must be applied within the feature matching module since landmark descriptions with different geometric properties have to be matched using the respective methods. Therefore, two different matching algorithms are integrated to facilitate both feature geometries in the first design iteration.

Although the utilized description approaches are fundamentally different in their theoretical principles, they both belong to the family of binary descriptors. As an alternative to the distribution-based methods, the respective support region can also be described by correlating specific properties of individual pixel pairs inside the region of interest. In pursuit of efficiency, the boolean value is incrementally sampled in a binary string to maximize the descriptiveness of every used bit. In contrast to the euclidean distance applied in floating-point descriptors, binary ones are based on the Hamming distance for comparison and matching purposes. This approach was inspired by the locality-sensitive hashing technique, which would minimize the calculation effort since only exclusive disjunctive operations followed by a bit count are necessary. Since both description algorithms are correlation-based, suitable binary matching methods have to be selected and integrated. In the case of point features, the matching method is based on the Fast

Library for Approximate Nearest Neighbors (FLANN) [79], which provides a quick approach to identify potential correspondences within the two sets of landmarks. In addition, a filter algorithm is implemented directly downstream of the matching procedure. Based on Lowe's distance ratio test [80], it is one of the first steps in the multi-staged filtering and outlier rejection routine, which specifically targets the identification and disposal of incorrect matchings. In terms of establishing correspondences between line features, the matching process is based on the characteristics of the associated line bands. With this, the most commonly applied method is incorporated into MMVO, which utilizes the multi-index hashing approach introduced in [81].

In order to reduce the computational effort and improve the real-time capabilities, MMVO utilizes the concept of keyframes, which are defined as a particularly selected image collection with specific properties. In general, it can be interpreted as the overall scaffolding of the associated tracking pipeline and is, therefore, essential for the motion estimation and short-term localization process. For simplicity, keyframes are utilized as the reference frame for the matching process, while the target image comprises the current frame. This matching procedure also meaningfully increases the stereo baseline between the source and target frame compared to identifying correspondences between successive image pairs. As a result, the estimation accuracy of the relative displacement between the selected frames is improved while enhancing the algorithm's tolerance against external disturbances.

4.4 Feature Entity Fusion and Filtering

After the initial image abstraction process, four different landmark collections are formed in our anticipated VO algorithm, which serves as the information basis for the subsequent steps in the motion estimation toolchain. Concurrent with the major components in a VO's primary processing pipeline, potential synergies between specific characteristics of these feature collections are formed for prioritization, property approximation, and outlier rejection purposes. This results in the composition of an adjusted landmark set containing more robust features for the actual relative motion estimation. As already briefly mentioned in the general system overview, the Feature Entity Fusion and Filtering (FEF^2) assembly is considered an accumulation of individual modules, in which our aspirations in the area of multi-modal feature collaboration are practically applied in the MMVO setup. While conventional filtering mechanisms in state-of-the-art applications are usually placed after the feature matching procedure, we want to extend these efforts and start one step earlier by integrating FEF^2 next to the central processing pipeline. With this, a multi-staged prioritization, filtering, and outlier rejection routine is established. First considerations and evaluations can be triggered in parallel to the feature description and matching pipeline once the initial landmark collection is assembled by the respective feature detectors. Therefore, the FEF^2 modules should be viewed as an additional component to the image abstraction and data provision toolchain. Basically, it is divided into the intra-class collaboration, contained in the fusion part, and the inter-class collaboration, which incorporates both components. Since the theoretical principles are already introduced in the conceptualization chapter, the following sections will highlight the implementation details. In general, a score-based prioritization and filtering system is created for the multi-modal feature collaboration. Here, each landmark f is assigned a particular valuation score S, consisting of the intra-class and inter-class contribution $S_{f,IACC}$ and $S_{f,IRCC}$. To express it more mathematically, it is defined as

$$S_f = S_{f,IACC} + S_{f,IRCC}. \tag{4.2}$$

4.4.1 Intra-Class Feature Collaboration

From a theoretical perspective, the intra-class feature collaboration concept could be applied to all kinds of landmark geometries once at least two feature types with compatible geometric properties are available. Based on the selected landmark detection algorithms within MMVO, the properties of three different point feature types consisting of oFAST, CenSurE, and GFTT are combined.

In general, the regions are evaluated by two different qualitative measures. Starting from the top, the image has to be divided into individual sections for evaluation. Since the feature detection process incorporates a similar procedure to ensure that the landmarks are evenly distributed throughout an image, the grid size has to be selected accordingly. In contrast to the previous application, a wider pattern has to be selected here since the primary objective is to identify correlations between different feature types based on their spatial distribution in the image. For this reason, the image is initially divided into a grid of 32 columns and 24 rows, forming a total of 768 cells. While conventional prioritization and filtering methods usually apply binary weights, we target to create a score-based valuation system. With this, the regions are individually assessed using a density-based approach, in which the absolute number of the contained features n is considered the primary qualitative measure for the evaluation. At this point, a first indication of the regions' content value and photometric condition can already be created based on the total number of landmarks in the individual cells. To further reinforce the informative value of the evaluation system, each feature type is assigned its weighting factor w since its associated significance depends on the specific characteristics of the respective algorithm. As an example, oFAST features are compared to those detected by the GFTT algorithm. It is evident that the first landmark type offers a higher value for VO and SLAM applications due to their invariance against rotation and change of scale. Therefore, these detections should be preferred over the ones extracted by the latter method, which can be achieved by giving them a higher valuation in the score-based prioritization system. For this reason, oFAST is given a weighting factor of 4, CenSurE of 2, and GFTT of 1, respectively. At last, the cell-specific score is refined by the second qualitative measure in the form of the feature-specific ranking position within the individual landmark collection. For this purpose, the characteristic response value of each feature type is normalized and summarized within the specific regions. The scaling factor $\widetilde{r}$ is then applied to the overall score of the respective cell as the arithmetic mean of the previously obtained value. In the end, the cell-specific significance value $S_{c,IACC}$ is defined as

$$S_{c,IACC} = \sum_{d}^{D} \widetilde{r}_d w_d n_d = S_{f,IACC}, \qquad D \in \textit{oFAST, CenSurE, GFTT.} \tag{4.3}$$

For the prioritization process on the feature-specific level, all landmarks contained in individual cells are assigned with the cell-specific significance value, as indicated in Equation (4.3). This way, a well-founded qualitative classification of landmarks within an image is achieved.

As a side effect of intra-class collaboration, previously unknown feature properties can be approximated by considering nearby landmarks detected by a detector of a similar approach. In this particular case, the size of GFTT landmarks can be approximated by adjacent oFAST features since they are both corner detectors. Therefore, for each cell, the arithmetic mean of the meaningful region size is calculated from individual oFAST features, and the expected block size is then applied to the GFTT detector to steer the resulting landmarks towards the targeted value actively. If a reasonable number of scale invariant features cannot be achieved in the specific cell, the eight neighboring regions' mean value is used. In the extreme case, where a sufficient

number of landmarks cannot be obtained within either neighbor, the mean region size from the previous iteration is taken. Although this decision would inevitably affect scale settings in the medium and long term, especially in the mapping domain, they are stiff sufficient for tracking tasks, leading to the minimization of LoT events. Depending on the given circumstances, the landmarks are assigned with a flag indicating that they are potentially not suited for mapping purposes in case a sufficient number of active mapping features is available.

4.4.2 Inter-Class Feature Collaboration

Building on the landmark prioritization method presented in the previous section, inter-class feature collaboration combines landmarks with different geometric properties for filtering and outlier rejection. Therefore, line clusters are created in MMVO using line segments detected by the LSD algorithm in association with the other three available point feature types.

Basically, this method is divided into two segments, which consist of the associated modules for the fusion and filtering procedures. Starting with the line cluster construction, the first step is identifying suitable baselines from the original feature collection. For this purpose, a filter is introduced to disregard line segments with a length less than a reasonably selected threshold. From a theoretical point of view, longer line features are considered more robust since the discriminative capability of edge contours improves with increasing size. In addition, they are more likely to be associated with real-world properties. Therefore, the threshold is selected based on the overall size distribution within the feature collection and the shortest $25\,\%$ of the line segments are disregarded in the followings. A bounding box is created around the associated baselines for the geometric clustering procedure. At this point, the width of the line cluster region is defined in relation to the detected width of the utilized line segment. In order to compensate for possible inaccuracies, the width of the meaningful line cluster region is magnified by the factor of five. On the other side, the length of the bounding box is based on the identified length of the respective line segment. Similar to the previous parameter, it is extended by the width of the line cluster region, increasing the boundary at each end by half of the respective width. As the last step of the fusion process, correspondences between the baseline and all other point features contained within the bounding box are created, resulting in the generation of line clusters.

The second step consists of the prioritization and filtering module, in which additional information from the feature matching process are considered. With this, correspondences between line clusters from the source and target image are implicitly created by the identified matchings between individual landmarks. A correspondence is formed in case more than $50\,\%$ of the line cluster members, including the baseline, are matched towards the same line cluster. As a result, a different approach for outlier rejection is established, in which binary weighting parameters are applied to the respective landmarks. It should be noted that the feature must be disregarded if it is identified as a matching outlier. On the contrary, the landmark's overall value should be elevated in case the correspondence is verified by the feature entity fusion and filtering routine. In summary, the feature-specific significance score $S_{f,IRCC}$ is defined as

$$S_{f,IRCC} = \begin{cases} +S_{f,IACC}, & \text{if } C_f^+ \in M^+. \\ -S_{f,IACC}, & \text{otherwise,} \end{cases} \tag{4.4}$$

where C_f represents the correspondence between the associated landmarks and M^+ the set of correct matchings.

4.5 Motion Estimation

Following the data acquisition and preparation toolchain, in which suitable landmarks and feature correspondences are identified, extracted, and refined from the image, the relative displacement between the source and target image is calculated in the following module. For this purpose, the position information of point features is combined with the directional properties of line segments to create a more robust motion estimation model.

In general, the presented toolchain can be divided into two segments consisting of the sequential arrangement of a first pose estimation step and the subsequent refinement process. At first, an initial estimation of the relative transformation is calculated based on the 3-D feature coordinates from the keyframe and their 2-D projection in the target image. Therefore, the multi-level data acquisition toolchain provides a prioritized and filtered collection of corresponding landmarks, from which a certain proportion of the most promising feature matchings is forwarded to the motion estimation pipeline. With this, only the highest valuated $50\,\%$ of the landmarks are considered, while the lower boundary is set to a minimum of 100 feature correspondences, if applicable. The relative pose displacement can be estimated by solving a set of equations surrounding the Perspective from N Points problem, referred to as Perspective-n-Point (PnP) [82] in the following. Based on the provided data collection, an optimum solution that minimizes the reprojection error from 3-D-to-2-D- point correspondences is found for the translatory and rotational components of the relative transformation in combination with intrinsic camera parameters and optional distortion coefficients. Within the collaborated motion estimation module, an initial solution is provided to the PnP solver in order to accelerate and stabilize the optimization process. For this purpose, MMVO assumes a constant velocity model between the previous two frames predicts the expected camera pose in the current iteration. Even though the feature correspondences are pre-selected through their significance score, a small proportion of outliers are expected to be included in the data collection Therefore, the PnP solver is combined with the RANdom SAmple Consensus (RANSAC) scheme [41] to reinforce the algorithm's stability and improve the estimation accuracy. If the quantity of the available correspondences falls below the minimum requirement of three, the camera pose cannot be computed based on the image pairing in the current iteration. In this particular case, a preliminary transformation is constructed either by the constant velocity model or the estimation provided in the IMU handler. However, these associated camera poses are considered significantly less accurate and robust than the visual-based method. Thus, an LoT event is triggered in case the image-based motion estimation cannot be recovered in a reasonable time frame. In addition to the resulting transformation matrix, the solver provides information about the identified inliers, which is transmitted to the keyframe identification module further downstream of the tracking toolchain. After the relative displacement was estimated in the previous process, the camera pose should be further refined by minimizing the associated reprojection error using a non-linear optimization algorithm. For this purpose, the orientation properties of the line segments are implicitly applied to the non-linear Levenberg-Marquardt minimization scheme.

4.6 Keyframe Identification

The last step in the MMVO's tracking pipeline is dedicated to the task of when and whether a keyframe should be introduced to the feature matching and motion estimation process. While on one end, the system's footprint and real-time capabilities could be adversely affected in case a profuse number of keyframes are instated, both types of divergences impair the pose estimation accuracy. Therefore, a reasonable balance has to be created within the keyframe identification procedure since it directly affects the framework's performance capacities on multiple levels. According to Lin et al. [83], appearance-based VO and SLAM applications commonly rely on five different types of keyframe selection methods. However, a closer inspection reveals that state-of-the-art approaches can be further generalized into two categories based on their respective selection procedures. At first, potential keyframes are identified and introduced to the associated collection using a threshold-based system. With this, a new keyframe is generated once the characteristic parameter falls below a respective threshold value. Starting from the most ordinary indicator, distance- and time-interval-based methods were integrated into earlier algorithms, e.g., PTAM [48] and SVO [55], but also modern ones, such as LSD-SLAM [84]. Further, the selection process can also be initiated by calculating the feature matching score between the source and target image in the feature matching process in OKVIS [65]. A seemingly more advanced method is introduced in VINS-Mono [66], in which the keyframe selection is based on parallax. In addition, this process can also be triggered by computing an image content index based on the clustering space and the feature distances between the associated landmarks within the source and target image. With this method, the question surrounding whether the current frame should be classified as a keyframe has to be answered instantly, and no further changes and modifications can be made in retrospect. Taking it a step further, the keyframe identification process can be supplemented by giving the possibility to be modified in retrospect by complying with, e.g., the survival of the fittest principle. This particular approach is applied in the ORB-SLAM [52, 62, 85] family, in which keyframes are selected in real-time more optimistically. After the initial decision, the keyframe collection is further adjusted, and individual candidates are neglected in the following iterations using suitable mechanisms. This method is generally considered the more promising one since the final decision is supplemented by additional information from the more extensive keyframe collection. On the contrary, it is also associated with a higher demand for computation resources, which is evidently outbalanced by the benefits of this approach. [52] [85]

Apart from the multi-modal feature collaboration and motion estimation methods, we also target to establish multiple modalities by a suitable combination of different parameters and threshold values. Starting from the approach of the identification module, the more advanced double-staged method is applied, in which the final keyframe is determined from a pre-selected collection. After the relative displacement is calculated in the motion estimation module, an overall rating of the target image, which is at the same time the currently received frame, is generated. Since tracking and localization in a global frame are based on the relation between subsequent keyframes, it is essential to ensure that a reasonable number of correspondences can be established in between. Further, selecting keyframes with many robust landmarks is also beneficial in the short-term since it serves as the reference for motion estimation purposes in the local frame. The main idea behind our approach is to consider *a posteriori* information collected since the last keyframe selection in order to create a more sophisticated identification procedure, with the target of selecting the best-suited frame. Basically, the general workflow of MMVO's keyframe selection process is inspired by the one introduced in ORB-SLAM and adapted to suit the peculiarities of a VO-only application. Therefore, the implemented system

works with three different thresholds defined in a relative frame. In contrast to the survival of the fittest principle, a different sampling method with a more advanced and lightweight post-selection algorithm is applied. As the first step within the keyframe identification module, an overall rating of each available image is generated. This score is based on the number of landmarks, which are classified as inliers during the motion estimation process and also referred to as stereo landmarks, in relation to the number of correspondences after the feature matching module. In case the number of stereo landmarks drops below $50\,\%$ of the initially identified quantity of robust features within the current keyframe in at least five successive frames, a temporary frame collection is sequentially constructed in the following iterations, including the images within the decision criterium. The second threshold comprises of the median value of the total number of feature correspondences between subsequent keyframes in the global frame within the current coordinate system. Once the number of stereo features falls short of either the previously stated figure or an absolute number of 100 verified matchings, a new keyframe is selected from the temporary collection. Since the procedure is primarily designed to minimize the effects of unpredictable photometric distortions, the temporary collection only contains the last $25\,\%$, or a minimum of three, of the frames between the decision point and the previous keyframe. For example, in case the last keyframe was inserted 21 frames ago from the decision point, only the last five frames are considered for the selection process. Especially in scenarios where the image quality can be drastically deteriorated from one frame to another and vary in an unpredictable manner, the general VO process would benefit from this approach. In the end, the selected keyframe is inserted into the processing pipeline. It serves as the reference for the successive iterations until a new keyframe is instated according to the same procedure. Although the estimation accuracy might be compromised in the short term in a more locally oriented frame since the keyframe is not instantly selected, the accuracy of the global pose estimation would be reinforced, as it is ensured that the reference frame is consistently chosen as the best possible for the following estimations.

5 Experimental Evaluation

Following the theoretical conceptualization and the implementation of MMVO presented in the previous chapters, we conducted an experimental evaluation concerning the performance characteristics of the multi-modal VO approach. Starting from the theoretical fundamentals of the selected evaluation metrics in Section 5.1, we focus on evaluating application-oriented real-world data from the urban housing scenario in which Rollin' Justin is usually situated. The first part of the chapter introduces the evaluation metrics and experimental setup. Section 5.3 explains the evaluation process in greater detail before the obtained results are presented and analyzed. With this, the impact of the individual multi-modal methods is assessed in a standalone manner at first. The achievements of the individual approaches are then compared to a performance baseline generated by the MMVO implementation that follows the multi-feature setup but does not include the collaboration methods. In the second step, a qualitative assessment of the overall framework, including all proposed multi-modal methods, is conducted. Hereby, the results are compared to the performances of MMVO in a single-feature setup and ORB-SLAM 2 [62]. On the hardware side, the experiments are conducted on a Dell Precision 5820 Workstation with an Intel Xeon W-2123 CPU running at $3.60\,\mathrm{GHz}$ and $16\,\mathrm{GB}$ of DDR 4 memory.

5.1 Evaluation Metrics

Before the performance of the novel approaches can be evaluated, a well-selected set of conclusive metrics has to be established. Overall, the evaluation is performed by analyzing the quality of the generated trajectory. The motion of rigid bodies can be expressed as a sequence in the specific Euclidean group $SE(3)$, which is provided by the transformations from the world to the body frame for each timestamp. As proposed by Sturm et al. [86], the assessment is achieved by measuring the differences between the camera poses $\boldsymbol{P_1, ..., P_n} \in SE(3)$ and the time-synchronized ground truth poses $\boldsymbol{Q_1, ..., Q_n} \in SE(3)$. At this point, only the relative quality measure is considered within the thesis since VO applications, as the name already suggests, are more focused on establishing local consistency. In contrast, global estimation accuracy, in case of continuous tracking, is depending on the selected back-end optimization strategy. Thus, the analysis in the global frame would be less conclusive in terms of MMVO. For this reason, the absolute measure will be neglected in the following examination. The trajectories are associated by the timestamps of the individual data points and aligned using Umeyama's method [87]. With this, the *evo* evaluation implementation [88] is utilized for the alignment of the pose estimates with ground truth information and the computation of performance metrics. In summary, three performance metrics are defined for the standalone assessments and the overall performance evaluation of the VO framework.

5.1.1 Relative Pose Error

At first, the relative pose error (RPE) measures the local accuracy of the estimated trajectory over a fixed time interval Δ. Therefore, this metric is often referred to as the key performance indicator for VO systems, as it corresponds to the drift of the given trajectory. According to Sturm et al. [86], the RPE at time step i is defined as

$$\boldsymbol{RPE}_i = (\boldsymbol{Q}_i^{-1}\boldsymbol{Q}_{i+\Delta})^{-1}(\boldsymbol{P}_i^{-1}\boldsymbol{P}_{i+\Delta}). \tag{5.1}$$

For the sequence of n camera poses $\boldsymbol{P_1}, ..., \boldsymbol{P_n}$, $m = n - \Delta$ individual relative error parameters are obtained along the resulting trajectory. In the end, the translational component of these error values, denoted as$\|trans(\boldsymbol{RPE}_i)\|$, are aggregated in the computation of the root mean squared error (RMSE) over the considered time frame defined as

$$RMSE(\boldsymbol{RPE}_{1:n}, \Delta) = (\frac{1}{m}\sum_{i=1}^{m}\|trans(\boldsymbol{RPE}_i)\|^2)^{0.5}. \tag{5.2}$$

Further, the rotational contributions can be evaluated by representing the individual rotation matrices as a 3-D vector. However, it has been found that it is sufficient to omit this particular evaluation step in standard applications since the rotation error is implicit in the translation measure. Alternatively, the performance of the VO and SLAM algorithms can be achieved by calculating the mean or median error instead of RMSE, which would further contribute to minimizing the influences of potential outliers. Nevertheless, the latter measure was deliberately chosen for the evaluation process, as it provides a clearer insight into the central tendencies of the given parameters.

5.1.2 Sequence Completeness Ratio

The second metric contributing to the system-level performance evaluation examines the tracking pipeline's robustness regarding the algorithm's stability against LoT. Since conventional VO approaches do not provide an adequate place recognition mechanism to recover from an LoT event, the completeness ratio can be considered an important steadiness measure of the respective approach. In general, this metric is defined as

$$\textit{Completeness Ratio} = \frac{n_{\boldsymbol{P}_k}}{n_{\boldsymbol{Q}_k}} \tag{5.3}$$

where $n_{\boldsymbol{P}_k}$ represents the number of estimated camera poses and $n_{\boldsymbol{Q}_k}$ the number of the time-synchronized ground truth information within the image sequence k.

5.1.3 Computation Time

The last metric aims at the statistical distribution of the necessary execution time of the overall system. For comparison reasons, time expenditure is normalized for the computation of a single pose estimation between the keyframe and the associated target. It is essential for on-line applications, e.g., construction of a VO, since the overall computation time is a determining factor for the real-time capability.

5.2 Dataset

Apart from the selected performance indicators, a suitable dataset must be generated as a prerequisite for the experimental examination. However, the selection and assembling process is challenging since the dataset has to be capable of assessing individual aspects of the associated VO and SLAM algorithm. Therefore, different real-world scenarios with varying environmental conditions, scene complexity, and, among others, realistic occurrences, such as distortion effects, have to be created.

5.2.1 Data Acquisition and General Structure

In this work, the Indoor Multi-Cam Dataset (IndoorMCD) introduced by Sewtz et al. [89] is used as the evaluation basis, which aims to establish a comprehensive benchmark in the field of visual-inertial VO and SLAM applications, with a particular focus on multi-sensor collaboration. It was recorded in the SMiLE Laboratory at the Institute of Robotics and Mechatronics. In general, a typical urban housing setting including a kitchen and living room assembly in its basic configuration is depicted, as illustrated in Figure 5.1. Since the environmental setup of the present dataset directly corresponds to the application-related surrounding in which our targeted hardware platform typically resides, it provides the ideal information base for the experimental evaluation. Within IndoorMCD, several scenarios have been recorded in varying setups. Therefore, three different environments are created in the laboratory, including a kitchen, a living room assembly, and an office area. To make the overall setup even more realistic, temporary walls, including a door, are used to create different room layouts between the individual scenarios with a total available area of $6.50\,\mathrm{m} \times 4.50\,\mathrm{m}$.

An overview is provided in Table 5.1. In more detail, the kitchen assembly consists of a typical countertop including an oven, a fridge, several electronic appliances, and ordinary things such as vegetables or a kitchen scale. Within this area, most structures are static, and their surface conditions do not offer a large number of textures. Arriving in the living room, it offers a sofa including a coffee table, multi plants, and a television shelf. In addition, decoration items are dispersed throughout the living area to provide a more residential atmosphere. At last, the office area contains either one or two desktops, including computer monitors, keyboards, and a reasonable number of office chairs. Further commodities such as pens, scissors, and other amenities that frequently change their place are also included to recreate a realistic everyday life situation. To make the dataset even more challenging, furniture and the appearance of other objects change over time to simulate human presence.

Table 5.1: Overview of each scenario's (S) specific properties and number of runs (R). Scenarios 0-4 have been captured in created environments in our lab, the last one is recorded in an actual apartment

S	#R	Environment	Device	Sync	GT
0	19	kitchen, office, living-room	HCD	✓	✓
1	28	kitchen, office, living-room	HCD		✓
2	20	2 rooms: kitchen, living-room	HCD	✓	✓
3	15	2 office desktops	HCD	✓	✓
4	15	kitchen, office, living-room	Marvin		✓
5	10	actual apartment	HCD	✓	

Figure 5.1: Overview of the environmental settings of the SMiLE Laboratory, in which the IndoorMCD was recorded. The illustrated configuration resembles the environmental conditions and scenery setup of the scenarios 0, 1, and 4 which contains a typical urban housing setting including a kitchen, living room, and office assembly.

Overall, the benchmark consists of 105 individual sequences arranged in five different scenarios captured using multiple commercial off-the-shelf (COTS) sensor systems providing RGB-D information and IMU measurements. Within each scenario, the complexity of the trajectory and challenges confronting the algorithms being evaluated increase with each data sequence. Basically, the data sequences can be divided into three different categories. While the first runs only contain a small quantity of rotation and translations in the elementary sequences, the trajectories increase in length and amount of movement in the advanced lapses. They ultimately include loops and revisits of previously explored areas. The final sequences in form of the long-runs add further environmental changes that can be observed when places are visited multiple times. In order to simplify the recording process, the data sequences are not directly captured on Rollin' Justin since the robotic system is quite cumbersome, especially in narrow situations. Instead, the hardware setup for the data acquisition process consists of three Intel RealSense D435i sensor systems, denoted as *left*, *front*, and *right*, in two different configurations. At first, the majority of the dataset was recorded using the hand-held camera device (HCD), which offers 6 DoF and integrates all sensors in a compact configuration, as depicted in Figure 5.2c. The small form factor allows simple and uncomplicated use by the operator and enables mobile manipulation. Apart from the hand device, data sequences were also recorded using the Mock-up Platform for Audio Research and Vision on Rollin' Justin (MARVIn) to simulate the characteristics of wheel-based systems. The mockup platform is illustrated in Figure 5.2b. In particular, this design is intended to mimic the FoV of sensors equipped on real assistant systems such as Rollin' Justin. With this setup, the motion variability is effectively reduced to only 3 DoF consisting of two translations x and y, as well as the rotation θ around their orthogonal axis.

The camera settings are directly derived from Rollin' Justin's current configuration. A summary of the most important properties is provided in Table 5.2. Following the manufacturer's recommendation for optimal performance [22], the image resolution is set to 640 × 480 pixels at a frame rate of 15 Hz.

For the experimental examination of MMVO, we decided to focus on a subset of the IndoorMCD dataset. In particular, the elementary and advanced sequences from scenario 1 and 4 are selected as the database for the evaluation. At this point, the long-runs are intentionally omitted since they are rather targeting benchmarking frameworks with mapping and long-term localization capabilities to preserve global consistency. For this reason, it is preferable to rely on individual shorter sequences with a certain number of rotations and translations for the evaluation of MMVO. Within the latter scenario, the data sequences were recorded using MARVIn that perfectly resembles the characteristics of our expected hardware configuration and, more importantly, the associated motion variability in terms of the prevailing degrees of

Table 5.2: Hardware properties and settings within the IndoorMCD dataset for the experimental evaluation.

Parameter	**Specification**
Camera model	Intel RealSense D435i
Color image sensor	OmniVision OV2740
Image resolution	640 × 480 pixels (VGA)
Frame rate	15 FPS
Exposure	Automated exposure time
White balance	Automated white balance settings
Accelerometer sample rate	400 Hz
Gyroscope sample rate	250 Hz

(a) (b) (c)

Figure 5.2: Overview of the utilized hardware platforms and recording devices. The IndoorMCD dataset was recorded using (c) HCD and (b) MARVIn to simulate the sensor characteristics of the humanoid assistance robot (a) Rollin' Justin.

freedom. In addition, we decide to head a step further and test the limits of our framework by evaluating data sequences with 6 DoF and a higher level of distortions recorded by the HCD in scenario 1. Since the experimental evaluation focuses on the performance characteristics of MMVO in a standalone application, a multi-camera setup is not required. Therefore, we only consider the data sequences from the *front* sensor, which is mounted at an elevated position and thus provides the most comprehensive view. The utilized data sequences are summarized in Table 5.3.

5.2.2 Camera Calibration

The intrinsic parameters responsible for the perspective projection and the distortion figures of the visual sensors are estimated using the DLR Calibration Laboratory [90]. Within the calibration process, the pinhole camera model is utilized, and the desired parameters are obtained using different views of a 2-D checkerboard target, including a distinctive origin for each sensor. These parameters consist of the focal-lengths f_x and f_y, the principal point (c_x, c_y), the skew k_{skew}, and the distortion contributions $\boldsymbol{K}$ and $\boldsymbol{P}$. The depth map is aligned to the color image on the hardware side of the RealSense devices resulting in a pixel-to-pixel correspondence in the images. In addition, the Brown-Conrady [91] model can be applied to remove distortion from the color image.

Each device is calibrated using the color sensor in terms of extrinsic relations at the system level. Since the experimental evaluation focuses on the performance characteristics of MMVO in a standalone application and does not require a multi-sensor setup, the exact procedure will not be further elaborated.

5.2.3 Ground Truth

In order to provide an adequate valuation basis for the performance metrics in Section 5.1, time-synchronized ground truth trajectories relating to the change of scene between individual frames in the evaluated data sequences have to be established. Within IndoorMCD, highly accurate ground truth estimations are obtained using a Vicon MX T40 motion capture tracking system. Therefore, the recording devices are equipped with several reflective markers, which are monitored by up to six infrared cameras mounted overhead on the ceiling, as illustrated in Figure 5.1. Overall, the system operates at $100\,\mathrm{Hz}$ and generates the trajectory of the tracked recording devices with an accuracy of less than $1\,\mathrm{mm}$.

For calibrating the motion capture system to the origin of the data acquisition platform, several reflective markers are placed on the checkerboard and registered manually to its origin. Multiple images of the calibration target are captured by the *front* camera, which is at the same time the origin of the respective recording device. In the end, the relation between the Vicon tracking system and the system's origin is estimated by summarizing the transformation of the *front* camera to the checkerboard and individual marker positions in the motion capture system.

5.3 Evaluation Results

After all previously specified prerequisites are fulfilled, the developed multi-modal feature collaboration methods are examined in this section. Therefore, different system configurations are created, and the set of conclusive evaluation metrics introduced in Section 5.1 are applied to the resulting trajectories of the selected data sequences. At first, the feature combination and collaboration methods are analyzed in a standalone setup. In the next step, MMVO is benchmarked against conventional applications and approaches. With this, our focus is directed towards the RPE and completeness ratio of each data sequence to relate the accuracy and the tracking stability of the multi-modal system in the state-of-the-art context. At last, a run-time analysis is conducted to assess the real-time capability of the proposed VO application. In all

Table 5.3: Overview of the data sequences included in the experimental evaluation.

Scenario	Run	Sequence Designation	n_{frames}	$t_{sequence}$	s_{path}
1	0	mcd5_hcd_nosync_s1r0	235	15.7 s	4.92 m
	1	mcd5_hcd_nosync_s1r1	248	16.5 s	4.22 m
	2	mcd5_hcd_nosync_s1r2	331	22.1 s	5.89 m
	3	mcd5_hcd_nosync_s1r3	347	23.1 s	6.42 m
	4	mcd5_hcd_nosync_s1r4	301	20.1 s	5.86 m
	5	mcd5_hcd_nosync_s1r5	350	23.3 s	4.91 m
	6	mcd5_hcd_nosync_s1r6	428	28.5 s	7.23 m
	7	mcd5_hcd_nosync_s1r7	342	22.8 s	5.72 m
	8	mcd5_hcd_nosync_s1r8	436	29.1 s	7.75 m
4	0	mcd5_marvin_s4r0	286	19.1 s	2.77 m
	1	mcd5_marvin_s4r1	439	29.3 s	4.10 m
	2	mcd5_marvin_s4r2	588	39.2 s	4.44 m
	3	mcd5_marvin_s4r3	601	40.1 s	5.78 m
	4	mcd5_marvin_s4r4	489	32.6 s	4.92 m
	5	mcd5_marvin_s4r5	818	54.5 s	7.47 m
	6	mcd5_marvin_s4r6	643	42.9 s	4.71 m
	7	mcd5_marvin_s4r7	848	56.5 s	5.93 m
	8	mcd5_marvin_s4r8	931	62.1 s	5.79 m

evaluation processes, the maximum number of landmarks to be extracted is set to 500 per frame for each utilized extraction algorithm. Therefore, the initial landmark collection contains maximum 2 000 feature entities in MMVO's intended multi-feature configuration with four different types of detectors. The time interval Δ is set to 15, indicating the relative drift per second.

5.3.1 Standalone Analysis

Before benchmarking MMVO against state-of-the-art algorithms, we first want to investigate the influence of the proposed multi-modal feature collaboration methods in a standalone setup. For this reason, trajectories from three different system configurations are generated and compared. Since our focus is directed towards identifying the potential of individual approaches, all system configurations are provided with the same feature collection to maintain equal conditions. Therefore, the multi-feature setup resembles the default setting of MMVO, which consists of landmarks extracted by the ORB, GFTT, CenSurE, and LSD detection algorithms. The first setup provides the performance baseline, in which no additional prioritization, filtering, and outlier rejection steps are executed apart from the standard processes within the tracking pipeline. At this point, individual landmarks from different feature types are treated equally and forwarded to the motion estimation module. In this particular case, line segments detected by the LSD algorithm are not included since this type of feature geometry cannot be utilized in the reference setup. The second and third configurations build on top of the first one. They are further equipped with respective methods for multi-feature collaboration regarding FEF^2 in MMVO (MFC) and the collaborated motion estimation approach in MMVO (CME). Within these setups, feature information from all four detectors can be combined, and possible synergies are elaborated.

Based on the different system configurations, trajectories are generated for individual data sequences and examined in combination with the provided ground truth reference. The results, including the RPE and completeness ratio of each evaluated data sequence, are summarized in Table 5.4.

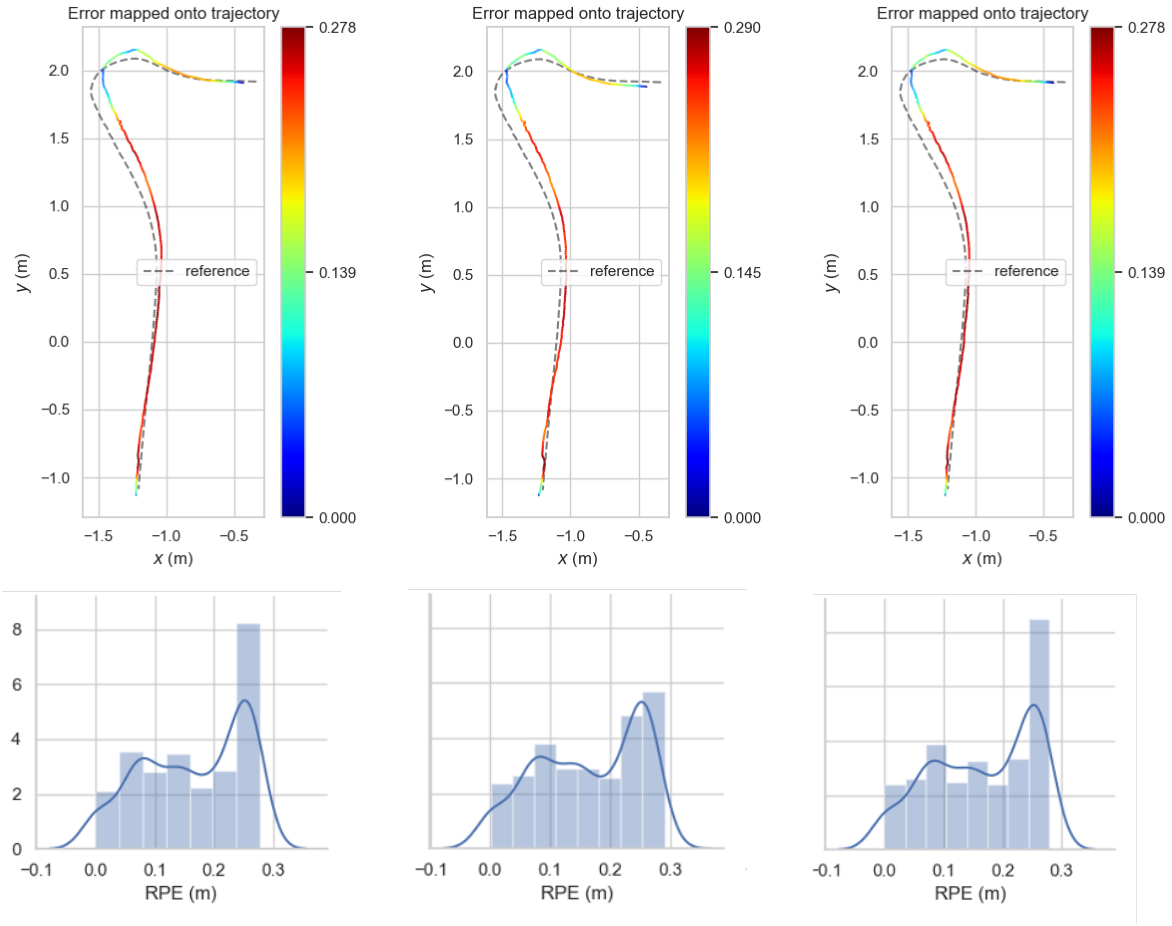
At first, the analysis starts with scenario 4, in which the data sequences were recorded by MARVIn that perfectly resemble the characteristics of our expected hardware configuration on Rollin' Justin. Apart from the fact that all setups were able to complete the dataset without LoT, it is noticeable that the achieved values of the quality measure are very similar among the different methods. More precisely, the RPE scores only vary within a magnitude of a few millimeters regardless of the arithmetic mean or maximum value. For this reason, the trajectories are reviewed individually as the next step to further investigate the background of this behavior. As an example, Figure 5.3 illustrates the trajectories and the associated RPE distributions of the second run within this scenario. A closer look at the ground tracks reveals that the estimated course in the baseline case in Figure 5.3a is already reasonably accurate in terms of the global frame. Since the multi-feature collaboration approach is primarily conceptualized for improving the robustness and reliability of the feature collections to the motion estimation process, it indicates that the overall quality of the features is sufficiently high. Also in the collaboration motion estimation in Figure 5.3c, the behavior is consistent with our initial assumption. Within this multi-modal method, information from the line features are utilized in an additional optimization step after an initial pose is estimated using point features. In this case, the initial estimation is already reasonably accurate, and the camera pose can only be marginally enhanced, or in some cases, not at all. A closer examination of the respective histograms confirms this statement. While the proportion with the highest RPE values is slightly reduced in the case of the multi-feature collaboration setup in Figure 5.3b, the magnitude of the RPE values remains

very similar. Therefore, it can be concluded that the scope for improvement within these data sequences is already saturated, and no mentionable enhancements can be achieved with the proposed multi-modal approaches. Overall, the influences of these methods are insignificant in this particular scenario and can be neglected with good conscience.

In the next step, the influences of the multi-modal feature collaboration methods are investigated using data sequences recorded in more challenging conditions with 6 DoF and an overall higher level of photometric distortions. In contrast to the results of the performance indicators in the previous scenario, the impact of these methods is considerably more notable, as indicated in Table 5.4. Here, the mean RPE is reduced by approximately $5\,\mathrm{cm}$ on average in the multi-feature approach and up to $100\,\mathrm{cm}$ in sequence 6 and $15\,\mathrm{cm}$ in sequence 0. At the same time, a similar effect is also visible in the case of the collaborated motion estimation. Although the overall impact of this particular method is not as significant as that of the previous approach, the positive effects are nevertheless evident. Figure 5.4 illustrates the trajectories and the associated RPE distributions of the second run within this scenario. Starting with the MMVO setup, including the multi-feature collaboration component in Figure 5.4b, the mean RPE of the generated trajectory is reduced by approximately $3\,\mathrm{cm}$ compared to the baseline case in Figure 5.4a. The improvement in tracking accuracy is explained by taking a closer look at both configurations' statistical RPE distribution and ground tracks. While the first indication of the deteriorating precision of the estimated camera poses is implicitly provided by comparing the respective ground tracks, the statistical distribution of the achieved RPE values is more significant in terms of ensuring local consistency. As the sequences in this scenario were recorded by a hand-held device without any support, the overall quality of the obtained features is expected to be much lower than in the previous scenario, with a higher fraction of matching outliers. In the case of the multi-feature collaboration setup, the landmarks are evaluated and prioritized to identify a set of highly distinctive landmarks. Therefore, the initial collection is filtered, and the most promising feature correspondences are forwarded to subsequent processes in the tracking

Table 5.4: RPE in m and the completeness ratio (CR) of the standalone evaluation. MMVO with multi-feature (MF) setup including ORB, GFTT and CenSurE landmarks provides the baseline for the systems with multi-feature collaboration (MFC) and collaborated motion estimation (CME) modules.

Scen.	Run	MMVO (MF)			MMVO (MFC)			MMVO (CME)		
		RPE_{mean}	RPE_{max}	CR	RPE_{mean}	RPE_{max}	CR	RPE_{mean}	RPE_{max}	CR
1	0	0.61052	1.24225	100 %	0.45837	1.52418	100 %	0.61053	1.24225	100 %
	1	0.31789	1.04061	100 %	0.28201	0.63108	100 %	0.29707	0.59583	100 %
	2	0.44076	1.02512	100 %	0.41204	0.86710	100 %	0.42907	1.06638	100 %
	3	0.43022	0.90421	100 %	0.39608	1.36455	100 %	0.35085	0.80264	100 %
	4	0.45977	1.39624	100 %	0.40734	0.68540	100 %	0.45976	1.39624	100 %
	5	0.40428	1.63362	100 %	0.32440	0.93329	100 %	0.39777	1.56193	100 %
	6	0.52279	1.71953	100 %	0.42524	1.02885	100 %	0.52279	1.71953	100 %
	7	0.34460	0.75738	100 %	0.31396	0.62094	100 %	0.34460	0.75739	100 %
	8	0.37344	1.02520	100 %	0.36543	1.18372	100 %	0.37344	1.02519	100 %
4	0	0.22149	0.38122	100 %	0.22680	0.37908	100 %	0.21283	0.38060	100 %
	1	0.20643	0.37462	100 %	0.20601	0.37550	100 %	0.20643	0.37461	100 %
	2	0.16324	0.27868	100 %	0.16264	0.29057	100 %	0.16366	0.27840	100 %
	3	0.21227	0.35841	100 %	0.21345	0.35963	100 %	0.21226	0.35841	100 %
	4	0.21777	0.37761	100 %	0.21495	0.37623	100 %	0.21786	0.37643	100 %
	5	0.19207	0.33091	100 %	0.19157	0.33097	100 %	0.19206	0.33091	100 %
	6	0.15967	0.27812	100 %	0.15962	0.27875	100 %	0.15928	0.27685	100 %
	7	0.14875	0.31521	100 %	0.14841	0.31528	100 %	0.14824	0.31606	100 %
	8	0.13385	0.29871	100 %	0.13320	0.29649	100 %	0.13384	0.29871	100 %



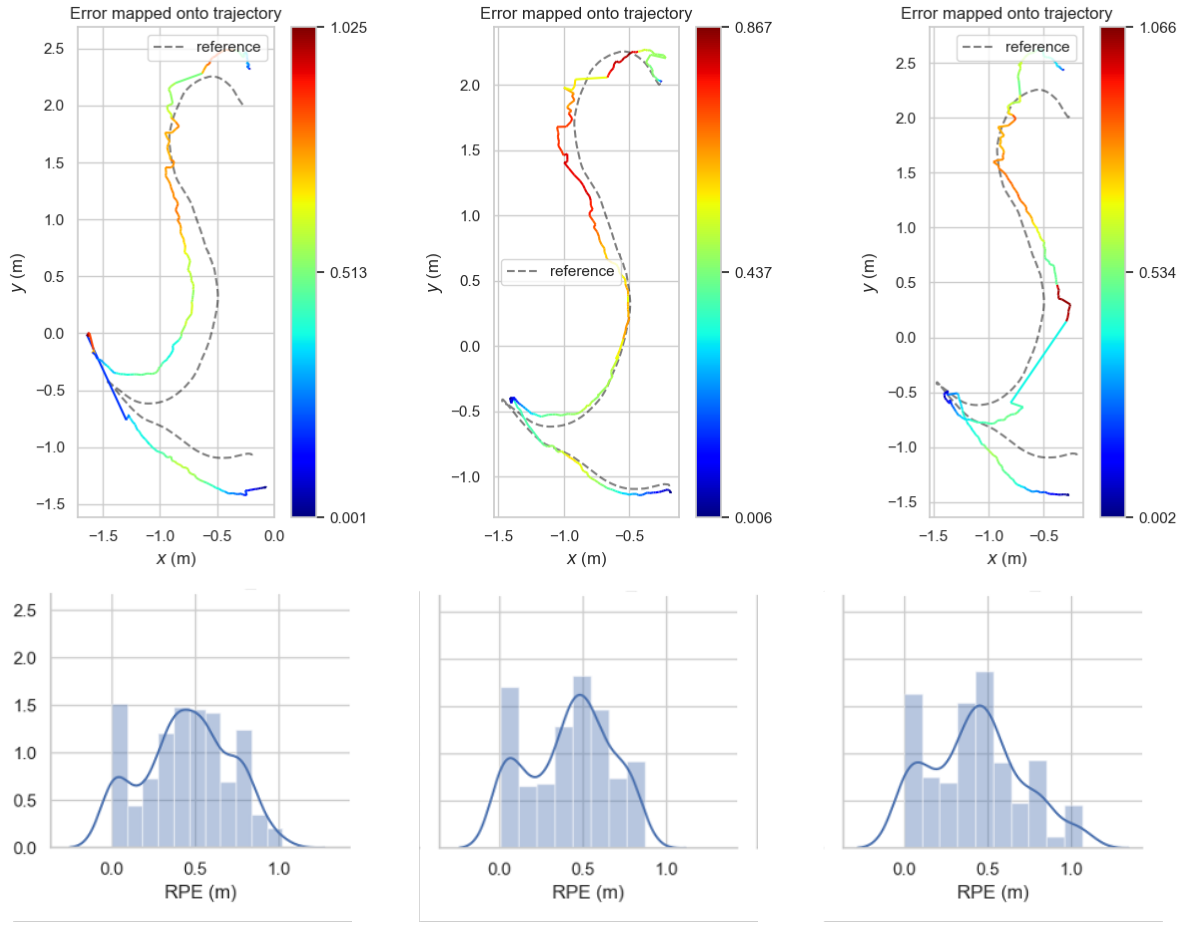
(a) MMVO (MF) (b) MMVO (MFC) (c) MMVO (CME)

Figure 5.3: Ground track trajectories and RPE distribution in the standalone analysis of scenario 4 run 2. Illustration of the results of (a) MMVO with multi-feature (MF) setup, (b) MMVO with multi-feature collaboration (MFC), and (c) MMVO in connection with the collaborated motion estimation (CME) module.

pipeline. Although the motion estimation module is equipped with an outlier rejection routine, the accuracy of the camera pose estimation can be improved by the additional prioritization and filtering process. On the other side, the collaborated motion estimation also achieved reasonable improvements in the accuracy of the estimated trajectory indicated by the successful reduction of the mean RPE value. However, the evaluation result also indicates that this combination is more sensitive to estimation outliers since the maximum error is increased compared to the baseline.

5.3.2 System Performance Benchmark

Following the standalone assessment of the different multi-modal feature collaboration methods, we want to relate the accuracy and the tracking stability of the multi-modal system in the context of what is considered state-of-the-art. Therefore, the performance of our multi-modal system is benchmarked against the tracking results of ORB-SLAM 2. Due to the fact that the overall system performance is highly dependent on the framework structure and implementation details on the software side, the comparison between these two applications does not always reflect the true potential of the utilized methods. For this reason, an additional setup is included in the system performance benchmark, in which the characteristics of ORB features are integrated into the MMVO implementation in a single-feature setup. Therefore, landmarks detected by this



(a) MMVO (MF) (b) MMVO (MFC) (c) MMVO (CME)

Figure 5.4: Ground track trajectories and RPE distribution in the standalone analysis of scenario 1 run 2. Illustration of the results of (a) MMVO with multi-feature (MF) setup, (b) MMVO with multi-feature collaboration (MFC), and (c) MMVO in connection with the collaborated motion estimation (CME) module.

feature extraction algorithm provide the database for the subsequent processing steps within the tracking pipeline.

Based on the different system configurations, trajectories are generated for individual data sequences and examined in combination with the provided ground truth reference. The results, including the RPE and completeness ratio of each evaluated data sequence, are summarized in Table 5.5.

Starting with the results from scenario 4, the error values of all three algorithms are in a similar order of magnitude at first glance. As already stated in the standalone analysis, the similarity is caused by the characteristics of this particular scenario, in which photometric and disturbances are sufficiently low. Therefore, all three approaches are able to establish an adequate number of reliable correspondences between the individual frames. On closer examination, a general trend emerges, in which MMVO achieves a slightly better tracking accuracy with both feature collaboration methods. Compared to the state-of-the-art algorithm, the RPE values are improved in most sequences, and the difference between the respective error values is under $1\,\mathrm{mm}$.

In the more challenging scenario 1, in which the algorithms are confronted with higher degrees of photometric distortions and more rapid motion sequences, the result of individual tracking algorithms are further apart and more distributed. In contrast to the previous scenario, in

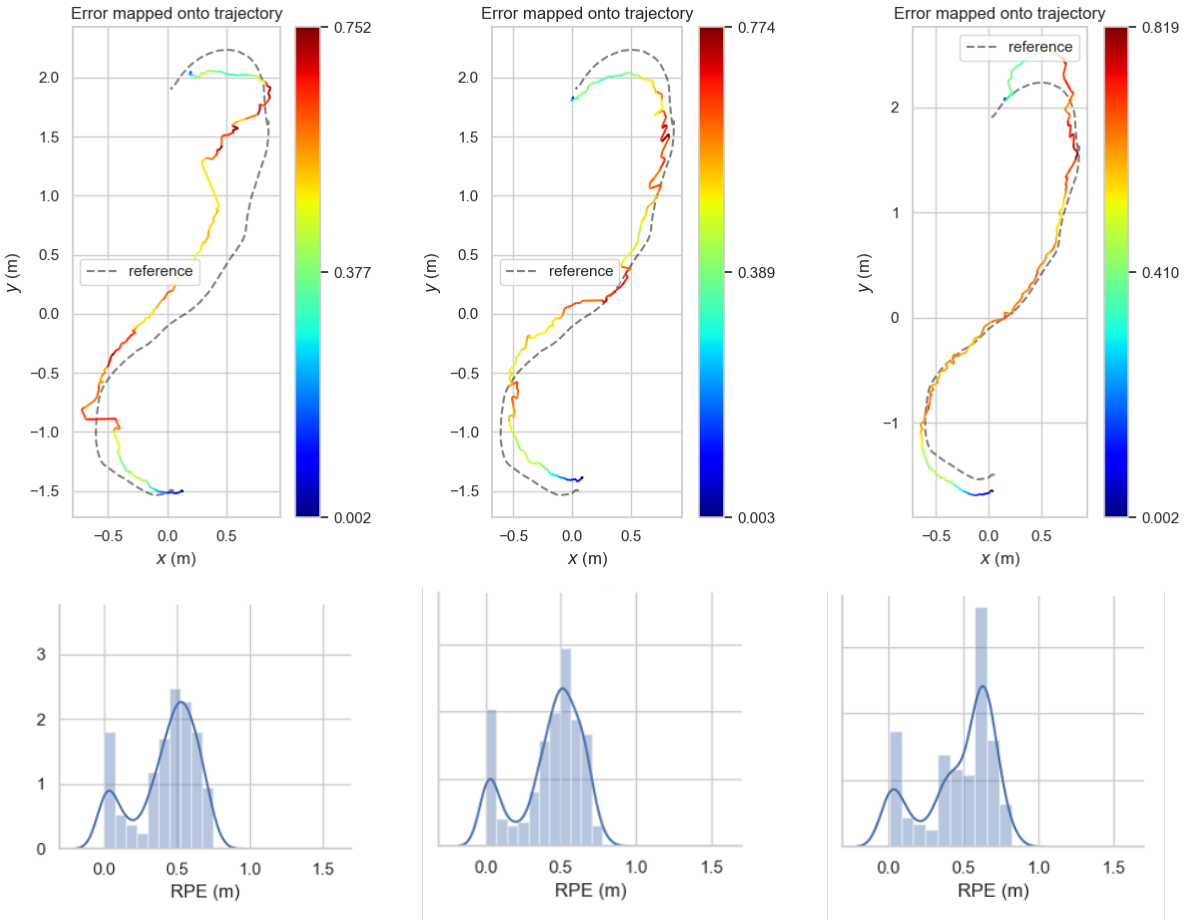
which all three approaches were able to fully complete the data sequences without LoT, the completeness ratios are more diverse in these sequences. While all three approaches were able to fully complete the previous sequences without LoT, the completeness ratio differs from algorithm to algorithm as well as from sequence to sequence. At this point, one of the multi-modal application's main advantages emerges in its enhanced tracking stability. While MMVO, in its intended configuration, completed most of the data sequences without LoT, it is the exception, with the methods solely based on the features from the ORB detector. When examining the achieved RPE values, no definitive trend can be established at first glance since the results fluctuate depending on the contemplated sequence. However, the results are more explainable once the completeness ratio is considered. Since the depicted error values refer to the fraction of the data processed by the respective method, the relations between the remaining frames after theLoT event occurred are, therefore, not included. A closer look into evaluations with similar tracking progress confirms this.

Figure 5.5 illustrates the trajectories and the associated RPE distributions of the fourth run within this scenario. A peculiarity of this presented data sequence is that all three tracking methods provided a complete trajectory. For this reason, it provides the best basis for analyzing the achieved performance parameters. Based on the obtained mean and maximum RPE values in Table 5.5, the racking results of both MMVO implementations are more precise than the one achieved by ORB-SLAM 2 in terms of local consistency. The performance evaluation of theMMVO-based methods is more complicated and cannot be easily assessed based on the mean values since they are very similar in the magnitude of a few millimeters. On closer inspection of the respective histograms in Figure 5.5, it can be observed that the distribution curve has been shifted downwards in Figure 5.5b in relation to Figure 5.5a, resulting in a lower magnitude of the obtained RPE values in the case of the multi-modal approach.

In summary, it can be stated that the previously established trend also continues in this scenario. The performance parameters of the multi-modal approach are within the region of the bench-

Table 5.5: RPE and completeness ratio (CR) of the overall system evaluation. Performance benchmark between MMVO with only ORB features, the complete MMVO setup with multi-feature collaboration (MFC) and collaborated motion estimation (CME) modules, and ORB-SLAM 2 with default parameters.

Scen.	Run	MMVO (ORB Features)			MMVO (MFC + CME)			ORB-SLAM 2		
		RPE_{mean}	RPE_{max}	CR	RPE_{mean}	RPE_{max}	CR	RPE_{mean}	RPE_{max}	CR
1	0	0.47554	1.15943	67.2 %	0.45833	1.52418	100 %	0.43038	0.91567	100 %
	1	0.38950	0.56039	81.0 %	0.28201	0.63108	100 %	0.26759	0.52420	79.9 %
	2	0.44926	0.89855	82.5 %	0.41204	0.80510	100 %	0.41192	0.81492	100 %
	3	0.35477	1.45178	38.3 %	0.39608	1.36455	100 %	0.16148	0.41365	37.5 %
	4	0.42355	0.75204	100 %	0.41801	0.77437	100 %	0.46211	0.81867	100 %
	5	0.23996	0.75093	37.7 %	0.29204	0.63021	100 %	0.23047	0.58242	37.7 %
	6	0.62739	3.50029	30.6 %	0.42256	1.30662	100 %	0.25468	0.57355	23.9 %
	7	0.26097	0.54457	29.5 %	0.31396	0.62094	100 %	0.26649	0.63780	28.3 %
	8	0.31902	0.89269	30.3 %	0.36199	1.18389	100 %	0.18355	0.41359	34.6 %
4	0	0.25054	0.69947	100 %	0.22461	0.38072	100 %	0.22296	0.37851	100 %
	1	0.20704	0.37220	100 %	0.20600	0.37550	100 %	0.20122	0.37921	100 %
	2	0.16511	0.27500	100 %	0.16179	0.26936	100 %	0.16713	0.27875	100 %
	3	0.21286	0.35266	100 %	0.21244	0.36155	100 %	0.21226	0.35153	100 %
	4	0.23821	1.09597	100 %	0.21495	0.37623	100 %	0.22013	0.38059	100 %
	5	0.19298	0.33402	100 %	0.19157	0.33096	100 %	0.19521	0.32212	100 %
	6	0.15838	0.27871	100 %	0.15962	0.27874	100 %	0.16293	0.28064	100 %
	7	0.15008	0.31294	100 %	0.14841	0.31528	100 %	0.15065	0.31889	100 %
	8	0.16363	1.26917	100 %	0.13254	0.29738	100 %	0.13419	0.29897	100 %



(a) MMVO (ORB) (b) MMVO (MFC + CME) (c) ORB-SLAM 2

Figure 5.5: Ground track trajectories and RPE distribution in the system performance benchmark of scenario 1 run 4. Illustration of the results of (a) MMVO with single-feature setup including ORB features, (b) MMVO with the multi-feature collaboration (MFC) and the collaborated motion estimation (CME) module, and ORB-SLAM 2.

marked state-of-the-art application. On an occasional basis, our approach is able to outperform the tracking capabilities of ORB-SLAM 2, as depicted in Figure 5.5 and a few other cases in Table 5.5. Compared to the single-feature setup in MMVO (ORB), the accuracy and overall robustness of the tracking pipeline are also enhanced.

5.3.3 Computation Time

Apart from the qualitative examination of the resulting trajectories, our focus is on the computation time in this section. Therefore, we iterated through all 18 data sequences to provide an adequate base for the run-time analysis. In addition, the execution time of the employed multi-modal feature collaboration methods should also be compared to the run-time of tracking systems with state-of-the-art approaches. However, the required computation time is highly dependent on the software-side implementation. For this reason, the evaluation baseline is also generated based on the MMVO implementation. With this, it is essential to mention that the recorded benchmark values only account for the pose tracking process of one relative displacement of the keyframe and the currently provided image. Other associated tasks surrounding the tracking pipeline are, therefore, not included.

Table 5.6 displays the mean execution time of the tracking pipeline for estimating the relative displacement between two frames. Further information concerning the statistical distribution is given in Figure 5.6. The combination of these two evaluation methods guarantees a reliable assessment of the computation time, especially in time-critical tasks, which is implicitly given with the target of developing a perceptual system. Unsurprisingly, the required mean processing time in the multi-feature setup is higher than in the single-feature approach by a factor of five. This is mainly caused by significantly more data being processed in the proposed approach, resulting in a mean run-time of nearly one-hundredth of a second.

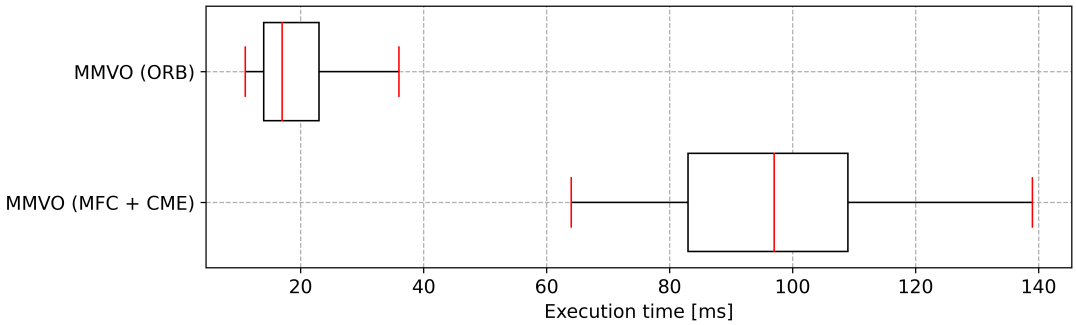


Figure 5.6: Statistical distribution of execution time of the tracking pipeline for estimating the relative displacement between two frames.

Table 5.6: Mean execution time of the tracking pipeline for estimating the relative displacement between two frames.

Module	Mean Execution Time [ms]
MMVO (ORB)	20.780
Feature Extraction	5.342
Feature Matching	3.589
Motion Estimation	4.534
MMVO (MFC + CME)	99.867
Feature Extraction	39.242
FEF^2	10.358
Feature Matching	22.515
Motion Estimation	21.427

6 Discussion

Based on the results from the experimental evaluation in the previous chapter, the findings and realizations are summarized and critically assessed. The discussion chapter is structured according to the order within the experimental evaluation.

At first, the elaborated multi-modal feature collaboration methods are analyzed in a standalone setup to identify the individual influence on the tracking system. Starting with the multi-feature collaboration, the results of the investigation have demonstrated that the accuracy of the tracking pipeline can be drastically enhanced by the prioritization and filtering algorithm. Within MMVO, this module is responsible for the valuation of each feature entity within the initial landmark collection to identify distinctive and more robust features. Consequently, it also provides a further filtering step for rejecting matching outliers in addition to the conventional method of applying Lowe's ratio test [80]. Although the motion estimation module is equipped with a separate outlier rejection scheme, the experimental evaluation revealed that this routine is insufficient. Especially in terms of deteriorating image quality and a higher level of photometric distortions induced by, e.g. rapid motion sequences, the additional prioritization, and filtering algorithm would be a valuable aid to the tracking pipeline. Apart from the mean RPE, which has been reduced by the feature valuation system in all data sequences without any exception, the maximum error value follows the general trend even though it reached a new global peak in three of the 18 sequences. Based on the available characteristic figures, no definitive explanation can be given for this phenomenon. For this reason, additional analysis has to be conducted to investigate the occurrence of this event further since it is most likely caused by the feature valuation and filtering system.

Apart from the first multi-modal method, the impact of the collaborated motion estimation pipeline is investigated in the second step. In this case, the results are not as straightforward as in the first approach. While in some cases, the effect of the additional optimization step is clearly visible and contributes positively to the overall tracking accuracy, the respective error measure remains the same as in the baseline configuration in more than 50 % of the cases. In particular, the characteristics of the mean RPE value have not been influenced either on the constructive or the destructive side. Although the effectiveness of this particular type of multi-modal feature collaboration method and its potential is confirmed by the experimental evaluation, it also revealed the shortcomings of the current implementation regarding the algorithm's robustness and reliability. In the current implementation stage, the collaborated motion estimation module consists of a sequential arrangement of two separate motion estimation processes. Therefore, the refinement stage depends on the initial estimation provided by the collection of point features. With this, it is advisable to compare the distribution of the achieved RPE values between these approaches. In case a more accurate estimation is achieved by the feature valuation and filtering system, as the mean RPE parameters suggest, it is most likely that the collaborated motion estimation process would rather have a negative effect on the overall precision of the trajectory. As an alternative approach, a combined motion estimation pipeline can be developed, in which

the line orientation information is explicitly included in the initial process, thus establishing a parallel configuration. This way, the dependency can be decoupled, resulting in a potential improvement in estimation accuracy.

After determining the influence of each multi-modal contribution in the standalone analysis, these methods are combined in the MMVO framework and placed in the context of state-of-the-art approaches as part of the evaluation of overall system performance. Within this, our algorithm is benchmarked against the camera tracking capabilities of ORB-SLAM 2 and MMVO in a single-feature setup, including landmarks provided by the ORB detector. As usual, the first step contains the qualitative assessment of the tracking accuracy based on the estimated trajectories and the corresponding performance metrics. Surprisingly, our expectations were exceeded by far since the performance indicators of the multi-modal system are approximately within the region of the state-of-the-art application. In some sequences, our approach is even able to outperform the tracking accuracy of ORB-SLAM 2. However, the most outstanding characteristic of the multi-feature MMVO is its significantly enhanced tracking capability and stability, even under the most unfavorable conditions. While MMVO, in its intended configuration, is able to complete all data sequences without LoT, it is rather an exception with the remaining methods that are solely based on ORB features. Combined with the previously analyzed qualitative result of the estimated trajectories, one might state that our novel development is considered the most comprehensive tracking algorithm within the scope of this examination. In addition, the implementation efficiency of the MMVO framework is implicitly outlined by comparing the RPE scores obtained by the single feature configuration to the achieved characteristics of ORB-SLAM 2. Since they are based on a similar initial feature collection, it is to be expected that their relative error indicators are within the same magnitude in the case of two equivalent implementations. However, the performance in the single-feature setup is less accurate than the one achieved by the state-of-the-art approach. This implies that the methods utilized in the ORB-SLAM 2's tracking pipeline are more advanced and can provide estimations with a smaller error margin. For this reason, promising methods such as generating a local map for the feature tracking process can be adapted to MMVO to improve the algorithm's performance as a possible next step.

In contrast to the encouraging qualitative results, the bottleneck of our proposed system is revealed by the evaluation of the necessary computation time. While the mean execution time in the single-feature setup is approximately $21\,\mathrm{ms}$ with outliers up to $38\,\mathrm{ms}$, the necessary computation time increases by a factor of five in the multi-feature approach. Therefore, the targeted real-time capability of a minimum of $15\,\mathrm{Hz}$ cannot be fulfilled with the current configuration. A closer inspection of the individual contributions shows that the feature extraction module registers the highest increase in run-time of all components, which is mainly caused by the line extraction algorithm. However, the additional information provided by the line features is deeply rooted in the concept of both multi-modal feature collaboration approaches. While the stability of the present collaborated motion estimation has to be significantly improved, the contribution of the geometric clustering method is not explicitly demonstrated in the experimental evaluation within the thesis. For this reason, the actual impact of the line clustering and filtering process has to be investigated in greater detail as the next step. Nevertheless, a general cost-benefit analysis regarding the practicability of employing line features would be recommended. Apart from the feature extraction module, another potential step in streamlining the overall footprint of MMVO is the development of a dynamic distribution and control routine. Depending on the actual situation and encountered photometric conditions, this system is targeted to adjust the initial composition of the feature collection dynamically. In the case of the overall conditions within scenario 4,

the experimental analysis revealed that, in most cases, all evaluated configurations are able to achieve a more or less equivalent result in Table 5.4 and Table 5.5. Therefore, the workload in all modules within the tracking pipeline can be reduced by utilizing ORB features in the majority while retaining a smaller subset of, e.g. GFTT landmarks. Conversely, the system is booted to its full capacity in more difficult circumstances in order to maintain the tracking process. Overall, it would be recommended to reassess the general design concept within MMVO to combine many different feature types regarding real-time capability. While one of the key design aspects within the ORB-SLAM family is to rely on one particular feature type for all processes within the scope of SLAM, the design choice in our approach is quite the opposite by combining the characteristics of four different detection algorithms. Thus, it would inevitably result in a heavier footprint for the entire application.

7 Conclusion

7.1 Summary

In this work, a robust and efficient front-end module responsible for tracking and short-term localization tasks is developed as part of a novel perception framework. Following the initial analysis of state-of-the-art approaches and the hardware properties, a feature-based VO algorithm, designated as MMVO, was proposed based on sensory data from an RGB-D setup and additional IMU measurements. Special attention was dedicated to establishing multiple modalities at different system levels during the conceptualization process. While a multi-modal setup was achieved in the hardware domain by consolidating information obtained by the visual perception system with IMU measurements, this concept could also be applied on the software-related side. Therefore, three different feature collaboration methods were proposed in two major arrangements.

The first approach centers around the concept that detections from different feature extraction algorithms of a specific geometry are able to cooperate. By utilizing several point feature extractors with different detection principles, algorithm-specific landmark collections were generated, and potential synergies between them could be formed to improve the overall quality of the detections. Within the intra-class feature collaboration, the robustness of individual landmark entities was analyzed based on their surroundings and prioritized using a custom-designed valuation system. As a second step, landmarks with different geometric properties could also collaborate in the inter-class feature collaboration to enhance the robustness of the tracking pipeline. In this case, unique characteristics of line segments were effectively combined with point features, resulting in the generation of line clusters. Alongside the information from the landmark matching process, these geometric clusters were used as an additional step in the feature valuation system, providing a valuable contribution to the rejection of matching outliers.

Apart from the previous approaches, which were focused on the data preparation toolchain for the subsequent processes in the tracking pipeline, the following method directly targeted the module responsible for calculating the relative displacement between two images. At this point, a sequentially arranged motion estimation process was proposed, in which the relative displacement was initially estimated using the point feature collection. In the second stage, the initial calculation is further refined by implicitly including the orientation information of line features.

After the feature collaboration methods had been successfully integrated into MMVO, an experimental evaluation was conducted to investigate the performance characteristics of our approach. While the standalone analysis of the multi-feature collaboration demonstrated that the accuracy of the tracking pipeline was significantly enhanced by the prioritization and filtering algorithm, the collaborated motion estimation results were not as straightforward as in the first approach. Although the additional optimization step positively contributed to the overall tracking accuracy in some cases, the reliability of this approach could not be clarified with absolute certainty. In the

next step, our proposed VO framework was placed in the context of state-of-the-art approaches and benchmarked against ORB-SLAM 2. As a result, it was identified that the tracking performance of our system regarding local consistency was comparable to the accuracy achieved by the state-of-the-art reference and even more accurate in some sequences. However, the most outstanding characteristic of MMVO is that it was able to complete all data sequences without loss of tracking, whereas it is an exception in ORB-SLAM 2. The subsequent run-time analysis revealed the bottleneck of our system in terms of the necessary computation time, which by far exceeds our defined requirement of a minimum frame rate of $15\,\mathrm{Hz}$. Therefore, MMVO cannot be classified as real-time capable in its present form.

7.2 Outlook

For further work, the real-time capability issue should be addressed first. Therefore, the system must be streamlined in various regions, e.g. by deploying a situation-based dynamic distribution and control routine. Taking it a step further, creating a deep-learning-based landmark detector is also conceivable, which summarizes the individual characteristics of the utilized feature extraction algorithms. By doing so, it could potentially contribute to the optimization of the necessary computation time within MMVO. Furthermore, a suitable motion model for handling IMU measurements has to be integrated into the system. Although MMVO is conceptualized as a multi-modal RGB-D-IMU framework, only the interfaces were provided on the system side.

List of Figures

List of Tables

Bibliography

[1] NASA, „NASA's Lunar Exploration Program Overview," 2020, p. 15.

[2] D. Heather et al., „The ESA PROSPECT payload for Luna 27: development status," 2021, pp. 1–2.

[3] E. Goldstein and J. Brockmole, *Sensation and Perception*, Cengage Learning, pp. 5–11, 2016.

[4] D. Schacter, D. T. Gilbert and D. M. Wegner, *Psychology (2nd Edition)*, New York, Worth, p. 123, 2011.

[5] A. Viberg. „"The verbs of perception: A typological study"," in: 2014, pp. 123–162.

[6] L. S. Roque et al., „Vision verbs dominate in conversation across cultures, but the ranking of non-visual verbs varies," *Cognitive Linguistics*, vol. 26, pp. 31 –60, 2014.

[7] M. O. A. Aqel et al., „Review of Visual Odometry: Types, Approaches, Challenges, and Applications," *SpringerPlus, 2016*.

[8] EUSPA, „Galileo Open Service - Service Definition Document," 2019, p. 64.

[9] H. G. Okuno and K. Nakadai, „Robot audition: Its rise and perspectives," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5610–5614.

[10] S. Shigemi. „"ASIMO and Humanoid Robot Research at Honda"," in: *Humanoid Robotics: A Reference*. Ed. by A. Goswami and P. Vadakkepat. Dordrecht: Springer Netherlands, 2019, pp. 55–90.

[11] S. Yamamoto et al., „Real-Time Robot Audition System That Recognizes Simultaneous Speech in The Real World," 2006, pp. 5333–5338.

[12] K. Nakadai et al., „Development, Deployment and Applications of Robot Audition Open Source Software HARK," *Journal of Robotics and Mechatronics*, vol. 29, no. 1, pp. 16–25, 2017.

[13] R. W. Moncrieff, „An instrument for measuring and classifying odors," *Journal of Applied Physiology*, vol. 16, no. 4, pp. 742–749, 1961.

[14] J. W. Gardner and P. N. Bartlett, „A brief history of electronic noses," *Sensors and Actuators B: Chemical*, vol. 18, no. 1, pp. 211 –220, 1994.

[15] R. Gutierrez-Osuna, „Pattern analysis for machine olfaction: a review," *IEEE Sensors Journal*, vol. 2, no. 3, pp. 189–202, 2002.

[16] T. Wen et al., „The Odor Characterizations and Reproductions in Machine Olfactions: A Review," *Sensors*, vol. 18, no. 7, 2018.

[17] M. Peris and L. Escuder-Gilabert, „A 21st century technique for food control: Electronic noses," *Analytica Chimica Acta*, vol. 638, no. 1, pp. 1–15, 2009.

[18] M. García et al., „Electronic nose for wine discrimination," *Sensors and Actuators B: Chemical*, vol. 113, no. 2, pp. 911–916, 2006.

[19] L. Zhang et al., „Classification of multiple indoor air contaminants by an electronic nose and a hybrid support vector machine," *Sensors and Actuators B: Chemical*, vol. 174, pp. 114–125, 2012.

[20] M. Fuchs et al., „Rollin' Justin - Design considerations and realization of a mobile platform for a humanoid upper body," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 4131–4137.

[21] I. Corporation. „*Intel RealSense D400 Series Product Family - Datasheet*," *337029-010*. Revision 010. 2021.

[22] A. Grunnet-Jepsen, J. N. Sweetser and J. Woodfill. „*Best-Known-Methods for Tuning Intel® RealSense™ D400 Depth Cameras for Best Performance*," Revision 2.0, BST-BMI055-DS000-10. 2020. Available: https://www.intelrealsense.com/download/9921/?_ga=2.81016023.1935368793.1637531522-517282294.1635858675.

[23] Intel. „*Intel Realsense Depth Camera D435i*," Accessed: 2021-11-10. Available: https://www.intelrealsense.com/depth-camera-d435i/.

[24] M. Sewtz, T. Bodenmüller and R. Triebel, „Design of a Microphone Array for Rollin Justin," in *Sound Source Localization and its Application for Robots*, 2019.

[25] Bosch Sensortec. „*BMI055 - Small, Versatile 6DoF Sensor Module - Datasheet*," *BST-BMI055-DS000-10*. Revision 1.4. 2021. Available: https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bmi055-ds000.pdf.

[26] B. Dynamics. „*Inside the lab: How does Altas work?*," visited on 2021-11-01. 2021. Available: https://www.youtube.com/watch?v=EezdinoG4mk.

[27] K. Yousif, A. Bab-Hadiashar and R. Hoseinnezhad, „An Overview to Visual Odometry and Visual SLAM: Applications to Mobile Robotics," *Intelligent Industrial Systems, 2015*.

[28] D. Scaramuzza and F. Fraundorfer, „Visual Odometry Tutorial - Part 1," *IEEE Robot. Automat. Mag., 2011*.

[29] J. Campbell et al., „A Robust Visual Odometry and Precipice Detection System Using Consumer-grade Monocular Vision," in *Proceedings of the IEEE International Conference on Robotics and Automation, 2005*.

[30] C. Cadena et al., „Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age," *IEEE Transactions on Robotics, 2016*.

[31] H. P. Moravec, „Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover," PhD thesis, Standford University, 1980.

[32] A. Comport, E. Malis and P. Rives, „Accurate Quadrifocal Tracking for Robust 3D Visual Odometry," in *Proceedings IEEE International Conference on Robotics and Automation, 2007*.

[33] C. Olson et al., „Robust stereo ego-motion for long distance navigation," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2000*.

[34] C. F. Olson et al., „Rover navigation using stereo ego-motion," *Robotics and Autonomous Systems, 2003*.

[35] S. Lacroix et al., „Rover Self Localization in Planetary-Like Environments," 1999.

[36] Y. Cheng, M. Maimone and L. Matthies, „Visual Odometry on the Mars Exploration Rovers," in *IEEE International Conference on Systems, Man and Cybernetics, 2005*.

[37] D. Nister, O. Naroditsky and J. Bergen, „Visual Odometry," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2004*.

[38] F. Fraundorfer and D. Scaramuzza, „Visual Odometry : Part II: Matching, Robustness, Optimization, and Applications," *IEEE Robotics & Automation Magazine, 2012*.

[39] M. Servières et al., „Visual and Visual-Inertial SLAM: State of the Art, Classification, and Experimental Benchmarking," *Journal of Sensors, 2021*.

[40] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2004.

[41] M. A. Fischler and R. C. Bolles, „Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM, 1981*.

[42] R. Smith, M. Self and P. Cheeseman, „A stochastic map for uncertain spatial relationships," in *Proceedings of the 4th International Symposium on Robotics Research, 1988*.

[43] A. J. Davison et al., „MonoSLAM: Real-Time Single Camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007*.

[44] S. Thrun et al., „FastSLAM: An Efficient Solution to the Simultaneous Localization And Mapping Problem with Unknown Data," *Journal of Machine Learning Research, 2004*.

[45] D. Chekhlov et al., „Real-Time and Robust Monocular SLAM Using Predictive Multi-resolution Descriptors," in *Proceedings of the 2nd International Symposium on Visual Computing (ISVC), 2006*.

[46] A. I. Mourikis and S. I. Roumeliotis, „A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation," in *Proceedings IEEE International Conference on Robotics and Automation, 2007*.

[47] M. Bloesch et al., „Robust visual inertial odometry using a direct EKF-based approach," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015*.

[48] G. Klein and D. Murray, „Parallel Tracking and Mapping for Small AR Workspaces," in *6th IEEE and ACM International Symposium on Mixed and Augmented Reality, 2007*.

[49] H. Strasdat, J. Montiel and A. Davison, „Scale drift-aware large scale monocular SLAM,"

[50] H. Strasdat et al., „Double window optimisation for constant time visual SLAM," in *International Conference on Computer Vision, 2011*.

[51] H. Lim, J. Lim and H. J. Kim, „Real-time 6-DOF monocular visual SLAM in a large-scale environment," in *IEEE International Conference on Robotics and Automation (ICRA), 2014*.

[52] R. Mur-Artal, J. M. M. Montiel and J. D. Tardós, „ORB-SLAM: A Versatile and Accurate Monocular SLAM System," *IEEE Transactions on Robotics, 2015*.

[53] E. Rublee et al., „ORB: An Efficient Alternative to SIFT or SURF," in *International Conference on Computer Vision, 2011*.

[54] R. A. Newcombe, S. J. Lovegrove and A. J. Davison, „DTAM: Dense tracking and mapping in real-time," in *International Conference on Computer Vision, 2011*.

[55] C. Forster et al., „SVO: Semidirect Visual Odometry for Monocular and Multicamera Systems," *IEEE Transactions on Robotics, 2017*.

[56] J. Engel, V. Koltun and D. Cremers, „Direct Sparse Odometry," *IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018*.

[57] R. A. Newcombe et al., „KinectFusion: Real-time dense surface mapping and tracking," in *10th IEEE International Symposium on Mixed and Augmented Reality, 2011*.

[58] T. Whelan et al., „Real-time large-scale dense RGB-D SLAM with volumetric fusion," *The International Journal of Robotics Research, 2015*.

[59] F. Endres et al., „3-D Mapping With an RGB-D Camera," *IEEE Transactions on Robotics, 2014*.

[60] C. Kerl, J. Sturm and D. Cremers, „Dense visual SLAM for RGB-D cameras,"

[61] T. Whelan et al., „ElasticFusion: Real-time dense SLAM and light source estimation," *The International Journal of Robotics Research, 2016*.

[62] R. Mur-Artal and J. D. Tardos, „ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," *IEEE Transactions on Robotics, 2017*.

[63] M. Sewtz et al., „Robust Approaches for Localization on Multi-camera Systems in Dynamic Environments," in *7th International Conference on Automation, Robotics and Applications (ICARA), 2021*.

[64] M. Kaess et al., „iSAM2: Incremental Smoothing and Mapping with Fluid Relinearization and Incremental Variable Reordering,"

[65] S. Leutenegger et al., „Keyframe-Based Visual-Inertial Odometry Using Nonlinear Optimization," *The International Journal of Robotics Research, 2014*.

[66] T. Qin, P. Li and S. Shen, „VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," *IEEE Transactions on Robotics, 2018*.

[67] C. Campos et al., „ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM," *IEEE Transactions on Robotics, 2021*.

[68] C. Evers and P. A. Naylor, „Acoustic SLAM," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 9, pp. 1484–1498, 2018.

[69] J. Coughlan and A. L. Yuille, „The Manhattan World Assumption: Regularities in Scene Statistics which Enable Bayesian Inference," in *Advances in Neural Information Processing Systems*, 2001.

[70] X. Luo and M. Sewtz, „Requirement Analysis for Perception on Assistant Robots in Multi-Modal Environment Conditions," in *ESA 16th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA), 2022*.

[71] C. G. Harris and M. J. Stephens, „A Combined Corner and Edge Detector," in *Alvey Vision Conference, 1988*.

[72] H. P. Moravec, „Obstacle avoidance and navigation in the real world by a seeing robot rover," 1980.

[73] S. Jianbo and Tomasi, „Good Features To Track," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1994*.

[74] M. Agrawal, K. Konolige and M. R. Blas, „CenSurE: Center Surround Extremas for Realtime Feature Detection and Matching," in *Computer Vision – ECCV, 2008*.

[75] N. Nain et al., „Fast Feature Point Detector," in *IEEE International Conference on Signal Image Technology and Internet Based Systems, 2008*.

[76] R. Gioi et al., „LSD: A Line Segment Detector," *Image Processing On Line, 2012*.

[77] M. Calonder et al., „BRIEF: Binary Robust Independent Elementary Features," in *Computer Vision – ECCV, 2010*.

[78] L. Zhang and R. Koch, „An Efficient and Robust Line Segment Matching Approach Based on LBD Descriptor and Pairwise Geometric Consistency," *Journal of Visual Communication and Image Representation, 2013*, vol. 24.

[79] M. Muja and D. G. Lowe, „Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration," in *VISAPP*, 2009.

[80] D. G. Lowe, „Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision, 2004*, vol. 60, no. 2.

[81] M. Norouzi, A. Punjani and D. J. Fleet, „Fast search in Hamming space with multi-index hashing," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3108–3115.

[82] E. Marchand, H. Uchiyama and F. Spindler, „Pose Estimation for Augmented Reality: A Hands-On Survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 12, pp. 2633 –2651, 2016. Available: https://hal.inria.fr/hal-01246370.

[83] X. Lin et al., „An Automatic Key-Frame Selection Method for Monocular Visual Odometry of Ground Vehicle," *IEEE Access*, vol. PP, pp. 1–1, 2019.

[84] J. J. Engel, T. Schöps and D. Cremers, „LSD-SLAM: Large-Scale Direct Monocular SLAM," in *ECCV, 2014*.

[85] R. Mur-Artal and J. D. Tardós, „Visual-Inertial Monocular SLAM With Map Reuse," *IEEE Robotics and Automation Letters, 2017*.

[86] J. Sturm et al., „A benchmark for the evaluation of RGB-D SLAM systems," in *International Conference on Intelligent Robots and Systems*, 2012.

[87] S. Umeyama, „Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 376–380, 1991.

[88] M. Grupp. „*evo: Python package for the evaluation of odometry and SLAM.*" https://github.com/MichaelGrupp/evo. 2017.

[89] M. Sewtz et al., „IndoorMCD: A Benchmark for Low-Cost Multi-Camera SLAM in Indoor Environments," *IEEE Robotics and Automation Letters*, 2023.

[90] K. H. Strobl and G. Hirzinger, „More accurate pinhole camera calibration with imperfect planar target," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011.

[91] A. E. Conrady, „Decentred Lens-Systems," *Monthly Notices of the Royal Astronomical Society*, vol. 79, no. 5, 1919. eprint: https://academic.oup.com/mnras/article-pdf/79/5/384/18250798/mnras79-0384.pdf. Available: https://no_no_doi.org/10.1093/mnras/79.5.384.

Own Publications

[63] M. Sewtz et al., „Robust Approaches for Localization on Multi-camera Systems in Dynamic Environments,“ in *7th International Conference on Automation, Robotics and Applications (ICARA), 2021*.

[70] X. Luo and M. Sewtz, „Requirement Analysis for Perception on Assistant Robots in Multi-Modal Environment Conditions,“ in *ESA 16th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA), 2022*.

[89] M. Sewtz et al., „IndoorMCD: A Benchmark for Low-Cost Multi-Camera SLAM in Indoor Environments,“ *IEEE Robotics and Automation Letters*, 2023.