Novel Camera Architectures for Localization and Mapping on Intelligent Mobile Platforms

Yifu Wang

A thesis submitted for the degree of
Doctor of Philosophy at
The Australian National University

March 2021

Except where otherwise indicated, this thesis is my own original work.

Yifu Wang
1 March 2021

To my beloved families.

Acknowledgments

Frankly speaking, I am usually not expressing my feelings, but I thought it would be worth to write it down in my PhD dissertation. I feel deeply honoured and humbled that my efforts during the last four years will be rewarded by the earning of my PhD degree.

I would like to express my deepest gratitude to my primary supervisor Assoc. Prof. Laurent Kneip, and Hongdong Li, the chair of my supervisory panel, for giving me the opportunity to study in one of the world-leading computer vision and robotic groups. During my PhD studies, they gave me patient and valuable supervision, which guided me to grow up from a newcomer into an individual researcher. In my four and half years of research, I basically focus on the topic of visual odometry and visual SLAM, which is always proclaimed as a quite matured computer vision technology from the perspective of theory. It is a tough and challenging road to come up with novel ideas or provide revolutionary improvements for such a well-studied topic, I would like to express my sincere appreciation to my primary supervisor Laurent, who provides me numerous helps both on my research and life. As one of the first students of Laurent, he spent a lot of his time on teaching me, not only from the perspective of theoretical knowledge and engineering skills, but also the way to develop critical thinking and strict academic attitude. I am also very grateful to Prof. Hongdong for his scientific advice and many insightful discussions and suggestions during my studies. For special reasons, I spent two years of my PhD career overseas, in order to closely collaborate with my primary supervisor Laurent, Hongdong has been supportive and has given me the freedom to pursue various projects without objection. I also have to thank my associate supervisor Dr. Viorela Ila, and my senior Yi Zhou, for their helpful advice and suggestions in general.

In the last two year of my PhD, I left Australia and was invited to visit the Mobile Perception Lab (MPL) at ShanghaiTech University, to jointly collaborate with Laurent. It was not until I came to Shanghai that I realized how important a good team is in the research field. I was lucky to be a part of MPL and I would like to show my appreciation to my friends there, Kun Huang, Xin Peng, Lan Hu, Yuchen Cao, Peng Wu, Zhanpeng Ouyang, Wanting Xu, Li Cui, Ling Gao, for their sharing and giving.

I am not alone when I pursuing my PhD degree, I also would like to show my appreciation to all my friends and colleagues in both Australia and China. It is really my pleasure to meet you guys for nearly 10 years, Zhirui Wang, Yizhou Yang, and Yan Han, the time we spent together in Canberra will never be forgotten.

Finally, I am deeply grateful to my families for their endless love, understanding and infinite support throughout my PhD studies.

Abstract

Self-localization and environment mapping play a very important role in many robotics application such as autonomous driving and mixed reality consumer products. Although the most powerful solutions rely on a multitude of sensors including lidars and camera, the community maintains a high interest in developing cost-effective, purely vision-based localization and mapping approaches. The core problem of standard vision-only solutions is accuracy and robustness, especially in challenging visual conditions. The thesis aims to introduce new solutions to localization and mapping problems on intelligent mobile devices by taking advantages of novel camera architectures. The thesis investigates on using surround-view multi-camera systems, which combine the benefits of omni-directional measurements with a sufficient baseline for producing measurements in metric scale, and event cameras, that perform well under challenging illumination conditions and have high temporal resolutions.

The thesis starts by looking into the motion estimation framework with multi-perspective camera systems. The framework could be divided into two sub-parts, a front-end module that initializes motion and estimates absolute pose after bootstrapping, and a back-end module that refines the estimate over a larger-scale sequence. First, the thesis proposes a complete real-time pipeline for visual odometry with non-overlapping, multi-perspective camera systems, and in particular presents a solution to the scale initialization problem, in order to solve the unobservability of metric scale under degenerate cases with such systems. Second, the thesis focuses on the further improvement of front-end relative pose estimation for vehicle-mounted surround-view multi-camera systems. It presents a new, reliable solution able to handle all kinds of relative displacements in the plane despite the possibly non-holonomic characteristics, and furthermore introduces a novel two-view optimization scheme which minimizes a geometrically relevant error without relying on 3D points related optimization variables. Third, the thesis explores the continues-time parametrization for exact modelling of non-holonomic ground vehicle trajectories in the back-end optimization of visual SLAM pipeline. It demonstrates the use of B-splines for an exact imposition of smooth, non-holonomic trajectories inside the 6 DoF bundle adjustment, and show that a significant improvement in robustness and accuracy in degrading visual conditions can be achieved.

In order to deal with challenges in scenarios with high dynamics, low texture distinctiveness, or challenging illumination conditions, the thesis focuses on the solution to localization and mapping problem on Autonomous Ground Vehicle (AGV) using event cameras. Inspired by the time-continuous parametrizations of image warping functions introduced by previous works, the thesis proposes two new algorithms to tackle several motion estimation problems by performing contrast maximization approach. It firstly looks at the fronto-parallel motion estimation of an event camera, in stark contrast to the prior art, a globally optimal solution to this motion estimation problem is derived by using a branch-and-bound optimization scheme. Then, the thesis introduces a new solution to handle the localization and mapping problem of single event camera by continuous ray warping and volumetric contrast maximization, which can perform joint optimization over motion and structure for cameras exerting both translational and rotational displacements in an arbitrarily structured environment. The present thesis thus makes important contributions on both front-end and back-end of SLAM pipelines based on novel, promising camera architectures.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Contribution

Sensation is one of the most significant functions of almost all creatures, preceding all other aspects such as cognition, behaviour and thought. For most of the vertebrates, including humans, they have more than one sensory system. Undoubtedly, creatures that have keen sensory systems have a tremendous advantage in terms of searching for food, finding preys, escaping their predator and locating mates. The evolutionary development of sensory systems, especially the visual system, is closely related to the external environment that creatures live in. For instance, eagles evolved a visual system that can find the preys at great distances, and bats evolved a auditory system that can navigate by echolocation in a low light scene. Being inspired by creatures, a lot of novel sensors are invented aiming at various challenging conditions. Meanwhile, in recent years, intelligent mobile platforms emerged as the next disrupting technology with a potential to transform our societies similar to the way this has been done by the invention of the automobile or the introduction of the internet. Important example applications are given by intelligent vehicles, intelligence augmentation devices such as smart phones or the Microsoft Hololens, and factory automation or service robots. The introduction of these solutions will lead to significant advantages such as more efficient and cost-effective transportation, partial or complete automatization of hard, tedious, or dangerous labor, as well as urgently needed assistance in the health care sector. Such intelligent mobile platforms are characterized by the fact that, in contrast to pure AI, we are talking about a physical device that is moving either actively or passively in the real world. The device is executing a certain task such as navigation in this environment, augmentation of the environment by virtual elements, or even manipulation of parts of the environment. Similar to the development of sensory systems in nature, the choice of sensors for tomorrow's intelligent mobile platforms needs to be chosen in order to satisfy the various boundary conditions.

Over the years, the ability of a machine to perceive and localize within its immediate surroundings has been recognized as a fundamental problem of several game-changing future technologies, and it has been investigated intensively across both industry and academia. Suppose an intelligent robot is placed into an unknown environment, the first task of the robot is to know where it is and what its surrounding environment is. Whether we are talking about a passively moving head-mounted display in an augmented reality application or an actively navigating self-driving vehicle, the 3D perception task remains similar: Process the continuous input data stream from all the available sensors to solve the inter-dependent problem of Simultaneous Localization And Mapping (SLAM). This is typically addressed by employing one of a few possible exteroceptive sensors, such as regular cameras, RGBD cameras, or 2D or 3D laser range scanners (i.e. lidars). Due to their high accuracy and robustness, the lidar-based multi-sensor alternatives represents the standard in many applications. However, a high price and often complicated design obstruct a large-scale deployment on close-to-market products, such as passenger vehicles. Hence the choice of sensor is often the result of combined requirements such as weight, size, or power restrictions, market restrictions, price restrictions, and sensing capabilities. In this particular regard, cameras have since ever represented a highly attractive alternative. They are small, lightweight, cheap and abundant, consume little energy, and return rich information about the environment at a potentially high frame-rate.

In this thesis, we looking at visual SLAM, which is a long-standing problem within the computer vision and robotics communities. Current pure vision-based solutions lack the level of robustness found in laser-based solutions, and are thus often complemented by additional sensors such as inertial measurement unit (IMU) on unmanned aerial vehicles (UAVs), and wheel encoders that measuring the rotational velocity of the wheels on ground vehicles. However, such sensors are not always the standard sensing equipment of intelligent mobile platforms, or accessing the signals of existing sensors such as wheel encoders may be prevented by the manufacturer. As a result, the development of robust, purely vision-based SLAM solutions remains a relevant topic in the development of intelligent mobile platforms. The thesis aims to introduce new reliable solutions to visual SLAM problem on both front-end and back-end module for intelligent mobile platforms by taking advantages of novel, promising camera architectures.

1.1 Literature Review

For a complete visual SLAM algorithm, the goal is to accurately estimate the trajectory of mobile platforms and reconstruct the 3D structure of the surrounding environment simultaneously. The pipeline of visual SLAM can be divided into two parallel modules: tracking and mapping. This thesis mainly investigates on tracking module of visual SLAM algorithms, thus we need to introduce an alternative, vision-based ego-motion estimation, also coined as Visual Odometry (VO) [Nistér et al., 2004]. The history of solving ego-motion estimation problem for mobile platforms could be traced back to early 1980s. [Moravec, 1980] firstly solved the ego-motion estimation problem testing on a planetary rover equipped with a single camera sliding on a rail. This project was motivated by the NASA Mars exploration program, and the goal is to provide an alternative to wheel odometry to measure the 6-degree-of-freedom (DoF) motion of all-terrain rovers, in the presence of wheel slippage in uneven and rough terrains. Although [Matthies and Shafer, 1986; Lacroix et al., 1999; Olson et al., 2000] investigated on motion estimation problem in the following two decades, these are mainly offline implementations and real-time working systems flourish only after the 2000s. In 2004, the landmark paper *Visual Odometry* written by [Nistér et al., 2004] was published and the term VO has become popular in the field of computer vision since then.

Regardless of VO or visual SLAM, a number of successful works have already been presented. The conclusion is that that no method is all-powerful and able to handle the motion estimation problem under arbitrary scenarios. While single camera solutions [Nistér et al., 2004; Strasdat and Davison, 2010; Klein and Murray, 2007; Mur-Artal et al., 2015; Newcombe et al., 2011b; Engel et al., 2014] are certainly the most interesting from a more scientific point of view, they are also challenged by many potential bottlenecks such as a limited field of view, moderate sampling rates, and a low ability to deal with texture-poor environments or agile motion. In addition to exteroceptive sensors such as inertial measurement units, the engineering standpoint therefore envisages the use of stereo [Konolige et al., 2007], depth [Newcombe et al., 2011a; Whelan et al., 2013; Steinbrücker et al., 2011; Tykkälä et al., 2011], or even light-field cameras that simplify or robustify the solution of the structure-from-motion problem by providing direct 3D depth-of-scene measurements. However, for stereo vision systems, using multiple cameras would potentially require excessive computational resources and also lead to substantial energy consumption. Besides, when the distance from the cameras to the observations becomes much larger than the distance between the stereo cameras, the stereo system would degenerate to the monocular case and becomes ineffective. We therefore investigate on the essence of motion estimation problems, starting from the novel camera architectures to provide reliable solutions.

In order to better understand our contributions in this thesis, we review the recent survey [Cadena et al., 2016] and some representative works are mentioned in the remainder of this section. They are either the landmark works in their field or represent the state-of-art approaches.

1.1.1 Feature-based Method

There are two predominant methods in visual SLAM: the feature-based method and direct methods. The feature-based method can also be divided into filtering methods and non-filtering methods.

- The most representative work for filtering method is Mono-SLAM [Davison et al., 2007], which also represents the first work that can recover the 6-DoF motion of a single camera in real-time. Mono-SLAM uses the extended-kalman-filter (EKF) as the core estimation approach, and build a map of landmarks with sparse image features, shown as Figure 1.1. However, there are several limitations to this work. For instance, the filter is prone to divergence and the method is sensitive to false landmark associations. This work inspires researchers to do repeatable localization by using filter-based visual SLAM framework, and a lot of successful approaches such as [Civera et al., 2009] and [Handa et al., 2010] are proposed based on this framework.

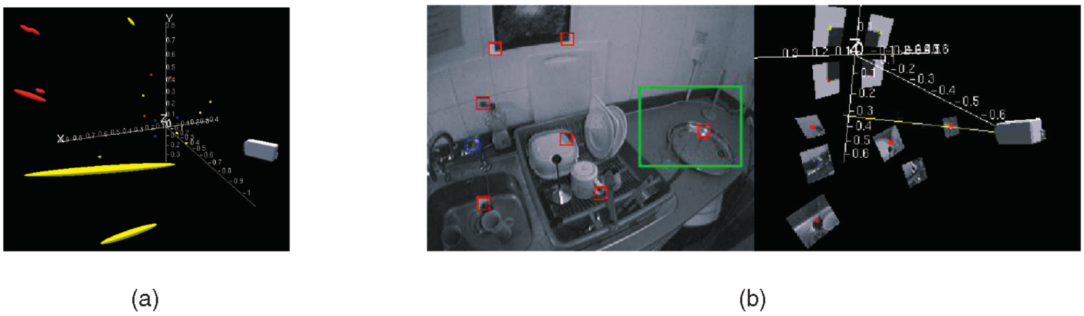


Figure 1.1: Illustration of (a) a snapshot of the probabilistic 3D map and (b) detected visually salient feature patches used in Mono-SLAM (source: [Davison et al., 2007])

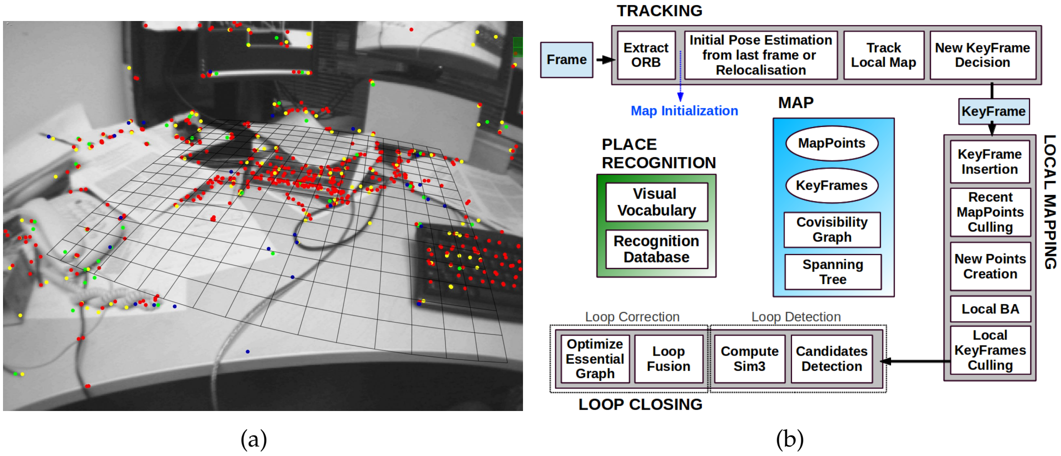


Figure 1.2: Illustration of existing keyframe-based methods. Left: A typical operation of PTAM. (source: [Klein and Murray, 2007]) Right: A system overview of ORB-SLAM (source: [Rublee et al., 2011])

- Non-filter methods have also been described as *keyframe methods*. Instead of using Bayesian filtering, non-filter methods commonly optimize the trajectory with global bundle adjustment by minimizing the reprojection errors over camera poses and 3D position of landmarks. Among non-filter methods, Parallel Tracking and Mapping (PTAM) is regarded as the standard for modern feature-based SLAM system [Klein and Murray, 2007]. PTAM is a pure vision-based SLAM pipeline designed for augmented reality (AR), and it requires no markers, pre-made maps or known template. An example of PTAM is shown in Figure 1.2(a). This approach firstly proposes to split tracking and mapping tasks into two separate threads, concentrate on processing keyframes to reduce redundant computation, and finally uses non-linear optimization methods as back-end full-map optimization. PTAM outperforms traditional EKF based methods in both efficiency and accuracy, by relying on efficient keyframe-based bundle adjustment for pose and map refinement. Although this method only works well in the small-scale indoor environment, its implementation style still highly inspires some state-of-art visual SLAM or VO pipelines like [Mur-Artal et al., 2015] and [Forster et al., 2014].

- It is worth mentioning that in the open-source communities for visual SLAM, ORB-SLAM [Mur-Artal et al., 2015] successfully builds a complete and practical SLAM system by combining many former research results. ORB-SLAM provides an impressive improvement with PTAM by applying some techniques in both implementation and theoretical level. An overview of the system is shown in Figure 1.2(b). Comparing with PTAM, an additional thread for loop closing is implemented in ORB-SLAM, in order to achieve consistent localization and mapping. It automatically selects the thread of calculating homography or fundamental matrix on map initialization stage, while manual operations are required to finish initialization in PTAM. Besides, just as its name implies, ORB-SLAM utilizes the ORB feature detector and descriptor [Rublee et al., 2011] rather than image patches used in PTAM, which highly improve the accuracy and robustness under scale and orientation change. Although ORB-SLAM is a traditional feature-based SLAM and similar to PTAM, it still receives much appreciation from both researchers and developers because of its impressive performance in practice.

1.1.2 Direct Method

A major problem of feature-based methods is that the performance of standard methods are highly depend on the results of feature extraction, description and matching, and it always fails in textureless environments. In the contrary of feature-based methods, direct methods formulate the motion estimation problem by directly optimizing the image intensity functions between sequenced frames, without sparse feature extraction and matching. Hence they can obtain ideal performance in accuracy and robustness even though not enough textures appear. Among the direct methods, there are several successful works are worth to mention.

- In 2011, [Newcombe et al., 2011b] firstly propose a system for real-time camera tracking and reconstruction by estimating the dense depth map at each keyframe, namely Dense Tracking and Mapping (DTAM). DTAM utilizes the dense geometric and photometric predictions from the dense surface model in tracking module, and it proved that feature-based methods can be replaced with a direct tracking approach. It also exploits hundreds of images available in a video stream from a single moving camera in order to compute the dense surface models more efficiently. Although DTAM represents a significant advance in the field of real-time visual SLAM, it has prior assumptions that brightness is constant in all stages of reconstruction and tracking, which would be sensitive to challenging illumination conditions. Moreover, DTAM requires excessive computational resources and it can only achieve real-time performance on GPU hardware.

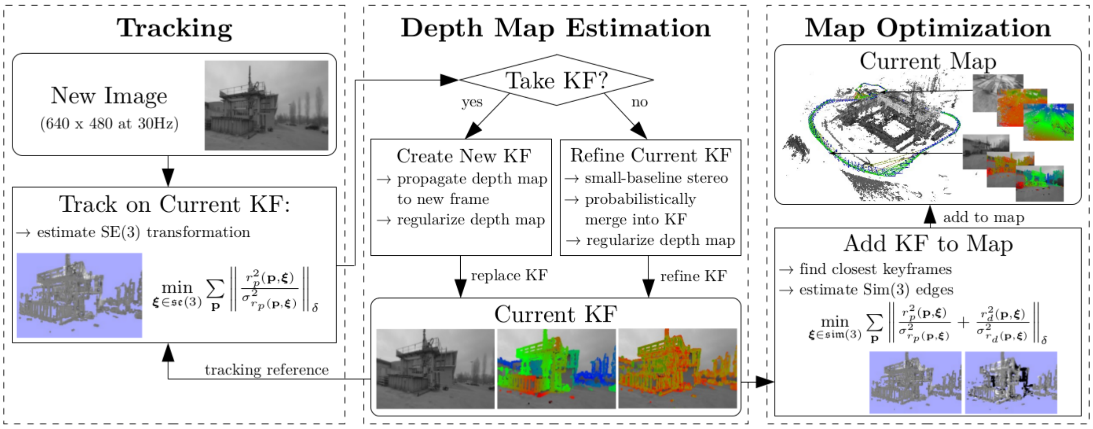

Tracking
New Image
(640 x 480 at 30Hz)
Track on Current KF:
→ estimate SE(3) transformation
$\min_{\xi \in \mathfrak{se}(3)} \sum_{\mathbf{p}} \left\| \frac{r_p^2(\mathbf{p},\xi)}{\sigma^2_{r_p(\mathbf{p},\xi)}} \right\|_\delta$
tracking reference
Depth Map Estimation
Take KF?
yes
no
Create New KF
→ propagate depth map to new frame
→ regularize depth map
Refine Current KF
→ small-baseline stereo
→ probabilistically merge into KF
→ regularize depth map
replace KF
refine KF
Current KF
Map Optimization
Current Map
add to map
Add KF to Map
→ find closest keyframes
→ estimate Sim(3) edges
$\min_{\xi \in \mathfrak{sim}(3)} \sum_{\mathbf{p}} \left\| \frac{r_p^2(\mathbf{p},\xi)}{\sigma^2_{r_p(\mathbf{p},\xi)}} + \frac{r_d^2(\mathbf{p},\xi)}{\sigma^2_{r_d(\mathbf{p},\xi)}} \right\|_\delta$

Figure 1.3: An overview of the complete LSD-SLAM algorithm. (source: [Engel et al., 2014])

- Different from DTAM, [Engel et al., 2014] propose a method that can achieve real-time performance on a CPU, namely Large-Scale Direct Monocular SLAM (LSD-SLAM). Comparing to DTAM, LSD-SLAM only utilizes pixels that on the boundaries of structures rather than every pixel over the image sequences. The major components of the LSD-SLAM algorithm are visualized in Figure 1.3; There are mainly two novelties of LSD-SLAM, which are explicitly incorporating and detecting scale-drift by aligning two keyframes on $\mathfrak{sim}(3)$, and taking the effect of noisy depth values into tracking with a probabilistic approach. Similar to DTAM, LSD-SLAM is also proved to be more robust and accurate than feature-based methods in textureless environments.

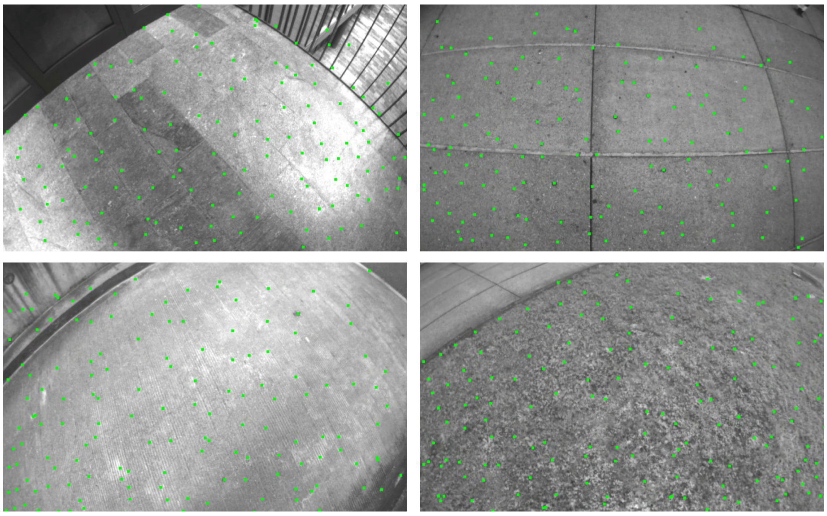


Figure 1.4: Illustration of a successful tracking in scenes of high-frequency texture using SVO. (source: [Forster et al., 2014])

- Semi-Direct Monocular Visual Odometry (SVO) [Forster et al., 2014] combines the robustness of direct methods and efficiency of feature-based methods. Without explicitly feature extraction and matching, SVO utilizes the pixel intensities around sparse features, and estimate the 6-DOF motion of a camera with a direct method, shown as Figure 1.4. Comparing with other methods, SVO is extremely efficient and it can run at 400 frames per second on a standard desktop.

1.1.3 SLAM with Deep Learning

In the past few years, the impact of deep learning in the field of computer vision is transformational. It has also promoted breakthroughs on visual SLAM problem. Generally, the state-of-art learning-based visual SLAM approaches can be divided into two directions:

- The straightforward direction is to use deep learning approaches to replace one or more modules in standard visual SLAM pipeline, such as feature detection, depth and pose estimation or loop closure detection [Tateno et al., 2017; Kendall et al., 2015; Yi et al., 2016]. These works take the advantages of deep learning methods and highly improve the performance of standard visual SLAM methods especially in challenging environments.

- Another direction is to learn and employ richer, class-specific models for the 3D geometry of complex, higher-level objects and structural components. Such features can be readily detected in images using the powerful semantic recognition capabilities given by deep neural networks [Salas-Moreno et al., 2013; Sünderhauf et al., 2017]. The motivation here is utilizing object-level representations to replace low-dimensional representations of complex objects, which not only can improve the accuracy of localization and mapping in complex environments, but also is of invaluable use for satisfying the higher-level perception needs on intelligent mobile platforms.

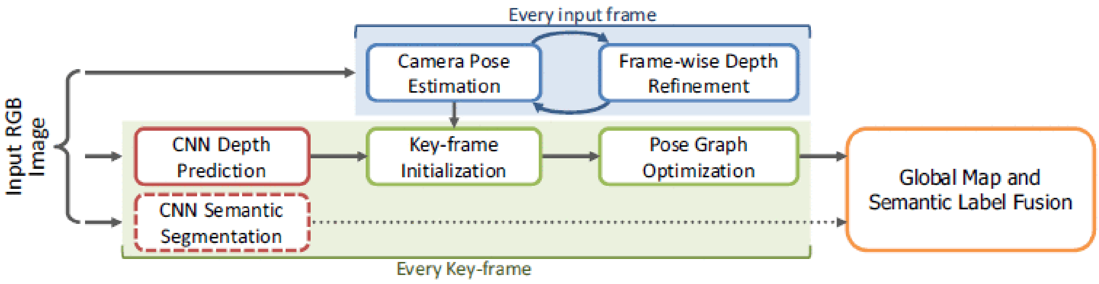


Figure 1.5: The flow diagram of CNN-SLAM. (source: [Tateno et al., 2017])

1.1.4 SLAM with Novel Sensors

Apart from the investigation of new algorithms, the robotics community also investigate on alternative sensors that can be leveraged for visual SLAM. Despite the advantageous properties of regular cameras given by low cost, weight and energy consumption, they also come at a certain cost such as low latency, motion blur and low dynamic range. A great number of research has been devoted to novel vision sensors that do not have the above-listed disadvantages. There are a few noteworthy new sensors and their application for visual SLAM are mentioned below:

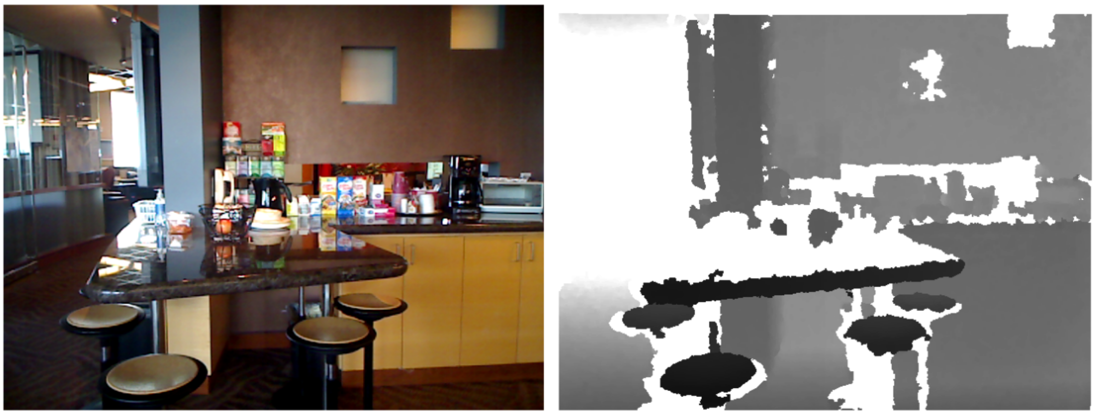
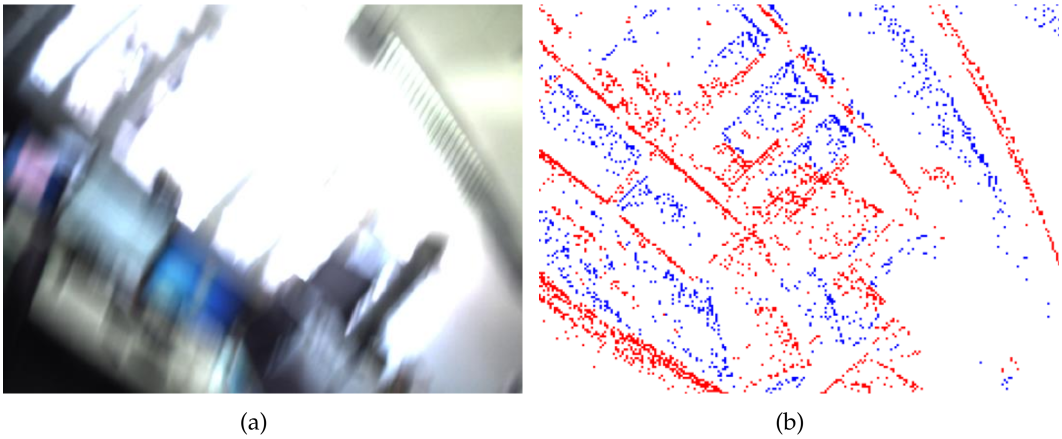


Figure 1.6: Illustration of RGB image (left) and depth (right) information captured by an RGB-D camera. (source: [Henry et al., 2014])

- Light-emitting depth cameras, are also known as RGB-D cameras, which have been widely used in mobile robots since the advent of the Microsoft Kinect game console in 2010. An example of an observed frame with an RGB-D camera is shown in Figure 1.6. RGB-D cameras provide high-frequency, dense depth images by taking the advantages of structure light or time of flight (ToF) principles, and the prior knowledge about the depth of the scene can significantly improve the accuracy of monocular SLAM methods. While a number of successful works have already been presented [Endres et al., 2013; Henry et al., 2014; Steinbrücker et al., 2011; Kerl et al., 2013], all of these systems can only operate on indoor environments since the RGB-D sensors are sensitive to illumination and unapplicable under direct sunlight.

- Event cameras are bio-inspired visual sensors which, unlike regular cameras, do no longer measure the raw intensity of an incoming light beam, but a quantized version of the relative intensity with respect to a reference time. The quantization happens at every pixel independently, and an event is triggered as soon as the absolute value of the logarithm of the relative intensity exceeds a certain threshold. The asynchronous nature and high temporal resolution of event cameras mean that the fired event patterns do not suffer from artifacts such as motion blur. The result is high performance in highly dynamic scenarios, and very low latency. Furthermore, event cameras have very good high dynamic range, which contributes to their strong potential to handle challenging illumination scenarios. Regarding full SLAM frameworks, one of the notable works is given by the EVO pipeline presented by [Rebecq et al., 2016], which relies on their earlier proposed EMVS plane sweeping based mapping approach [Rebecq et al., 2018]. Another notable work is given by [Kim et al., 2016], who present a real-time filter-based tracking and mapping framework. [Zihao Zhu et al., 2017] introduced a event-based visual-inertial odometry pipeline. The proposed method tracks a set of features in the event stream and then using an Extended Kalman Filter to fuse these tracks. More successful works can be found in a recent overview of research on event-based vision given by [Gallego et al., 2020].


(a)

(b)

Figure 1.7: Illustration of the performance of regular cameras and event cameras under high-speed motion. Left: A blurry image captured by regular camera Right: A visualization of accumulated events over a specific time interval (source: [Scaramuzza, 2020])

1.2 Motivation and Contributions

While a lot of efforts have been made on purely vision-based localization and mapping problem, the conclusion is that only a few of them focus on handling the motion estimation problem for intelligent mobile platforms. In order to improve the performance of tracking on such platforms, we aim to develop efficient, reliable solutions by taking advantages of novel camera architectures.

1.2.1 On scale initialization in non-overlapping multi-perspective visual odometry

In Chapter 2, we start by looking into the motion estimation framework with multi-perspective camera systems (MPCs). Multi-perspective camera systems pointing into all directions are cost-effective sensor systems and have already become part of the standard sensing equipment on modern intelligent platforms. Comparing with monocular or stereo camera systems, MPCs combine the benefits from different directions. If pointing the cameras into different, opposite directions, as shown in Figure 1.8, the flow fields caused by translational and rotational motion become very distinctive [Kneip and Lynen, 2013], meaning that MPC solutions are strong at avoiding motion degeneracies. Furthermore, omni-directional observation of the environment makes failures due to texture-poor situations much more unlikely. In contrast to regular omni-directional cameras, MPCs maintain the advantage of not introducing any significant lens distortions in the perceived visual information. Just like plain monocular cameras, MPCs also remain kinetic depth sensors. This means that they have no inherent limitations like stereo or depth cameras, which have limited range, or—in the latter case—cannot be used outdoors. As a final benefit, MPC systems are able to produce measurements in metric scale even if there is no internal overlap in the cameras' field of view.

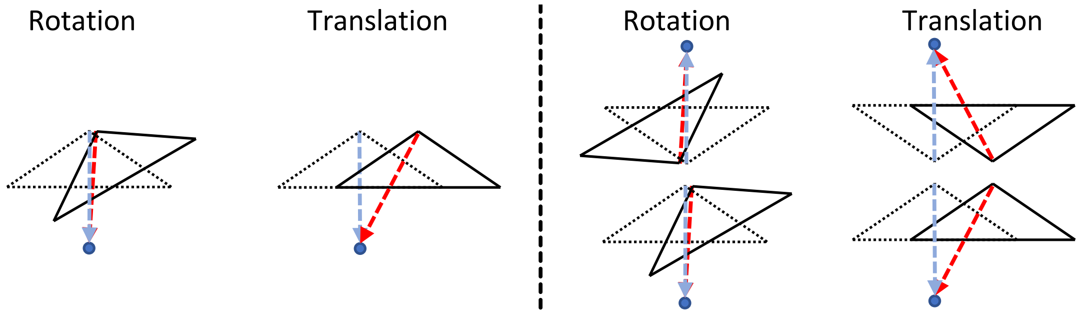


Figure 1.8: Similarity (left) and dissimilartiy (right) of disparities caused by different motion and different camera types. Left: single perspective camera. Right: non-overlapping multi-perspective camera systems

From the perspective of cost, MPCs are becoming increasingly important in most recent designs from the automotive or the consumer electronics industry. A large number of affordable visual onboard sensors looking into various directions to provide complete capturing of the surrounding environment. An example of the fields of view of an intelligent vehicle's visual sensors is shown in Figure 1.9. The drawback with many such arrangements, however, is that most the cameras do not share any significant overlap in their field of view. We call those camera arrays *non-overlapping MPCs*.

The proper handling of non-overlapping MPCs requires the solution of two fundamental problems. As discussed in [Clipp et al., 2008], non-overlapping MPCs

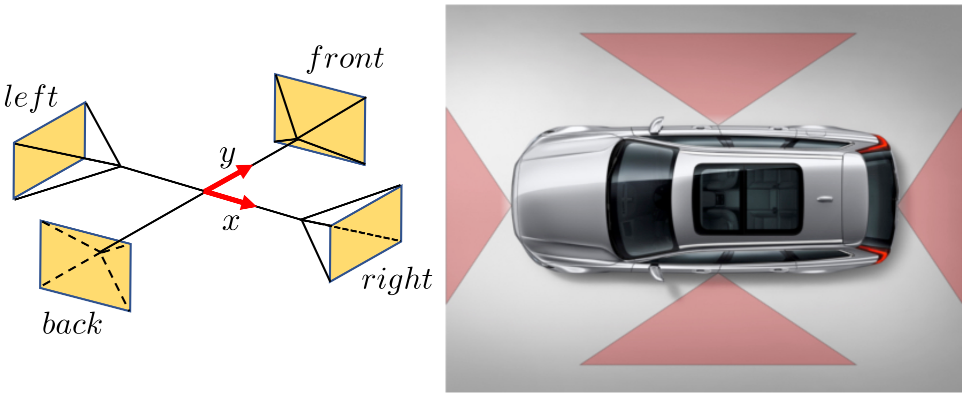


Figure 1.9: Example fields of view of a multi-perspective camera mounted on an intelligent vehicle.

are easily affected by motion degeneracies that cause scale unobservability, such as straight or Ackermann motion. This is a severe problem especially in automotive applications or in general during the bootstrapping phase, where no scale information can be propagated from prior processing. In conclusion, in order to truly benefit from the omni-directional measurements of MPCs, the measurements need to be processed jointly in each step of the computation. This is challenging as classical formulations of space resectioning and bundle adjustment all rely on a simple perspective camera model.

1.2.2 Reliable relative pose estimation for vehicle-mounted surround-view camera systems

While we will present an unprecedented integration of the paradigm of *Using many cameras as one* into a full end-to-end real-time visual odometry pipeline in the Chapter 2, there still remains space for further improvements. One remaining problem is that the success of our joint linear bootstrapping approach still depends on sufficiently good relative rotations estimated from each camera individually at the very beginning. Therefore further research efforts are required to push generalized relative pose methods towards a robust recovery of relative rotations even in the case of motion degeneracies. In Chapter 3, from a more specific perspective, we focus on a vehicle-mounted surround-view camera system, and notably aims at a solution to the problem of finding the planar motion of the vehicle. Such non-overlapping surround-view camera systems often include four fish-eye cameras pointing into the forward, backward, and side-ways directions, and they are also described by the generalized camera model.

Among the massive body of existing work on solving relative pose estimation problem for ground vehicles, no method is all-powerful and able to handle the possibly non-holonomic planar vehicle motion. Generalized relative pose solvers solving for all 6 degrees-of-freedom have substantial computational complexity and solution multiplicity in the minimal case [Stewénius and Nistér, 2005], or require too many samples and linearizations in the non-minimal case [Li et al., 2008], thus leading to unstable results under noise. Some generalized solvers even degenerate in the case of non-holonomic motion [Kneip and Furgale, 2014].

Several relative pose solvers [Scaramuzza et al., 2009; Huang et al., 2019] specialise in the non-holonomic planar vehicle motion estimation, including solvers designed for multi-camera systems [Lee et al., 2013]. While the aforementioned methods are very robust, they rely on the ideal assumption of a fixed steering angle; the centre of rotation as introduced by the Ackermann motion model is however a dynamic point for the majority of time during which a vehicle is taking a turn. Although a single-view solver relying on the more correct planar motion model has been presented [Booij and Zivkovic, 2009], a generalized planar motion solver that simultaneously exploits the information from multiple cameras appears as a gap in the literature.

1.2.3 B-splines for visual SLAM on non-holonomic ground vehicles

While the two previous points will introduce a complete real-time pipeline for visual odometry with non-overlapping multi-perspective camera systems, and also focus on how to reliably solve relative pose estimation problem with such systems in the front-end module. There is a crucial gap in our proposed pipeline which is to introduce a robust and accurate back-end optimization approach.

Bundle adjustment plays an essential role in most state-of-the-art feature-based visual SLAM, and it is typically used for the final refinement stage to approximate initial scene estimates and remove the scale drift in incremental reconstructions. It jointly optimizes the 6-DoF camera poses and 3D position of landmarks by minimizing the reprojection error corresponding to the image distance between the projected and measured image points. Such objective functions are generally optimized using non-linear least-squares algorithms. The non-linear problem can be efficiently solved by implementing iterative methods such as Levenberg-Marquardt, which uses an appropriate initial guess to generate a sequence of improving approximate solutions and finally converge to a local minimum of the objective function.

From the above, we can see that bundle adjustment is highly depend on a sufficiently good initialization obtained from front-end solvers. However, in contrary to most public datasets for autonomous driving, such as KITTI benchmark datasets [Geiger et al., 2013], there are many complicated and challenging scenarios in practical applications, which makes the failure of front-end relative pose solver very likely. That is exactly one of the reasons why current pure vision-based solutions always lack the level of robustness found in laser-based solutions, and are thus often complemented by additional sensors. We notice that the trajectory in visual SLAM frameworks is commonly represented by a discrete set of camera poses each associated with one of the captured images. It is clear to observe that the discrete representation is too general and does in fact not respect the kinematic motion constraints of ground vehicles. As a result, in order to develop a robust, purely vision-based (or inertial-supported) SLAM solutions for self-driving vehicles, in Chapter 4, we take a specific kinematic constraint on the vehicle motion into account.

For unconstrained motion, a camera has 6-DoF and it represents that the camera is free to change the orientation/position in 3D space. While for drift-less ground vehicles shown in Figure 1.10, certain displacements become impossible or only executable as a complex combination of displacements. It could be approximated to a more restrictive motion model, namely the Ackermann model. This restrictive motion model essentially reflects the fact that the infinitesimal motion is a rotation about an Instantaneous Centre of Rotation (ICR) which lies on the extended non-steering two-wheel axis. It forces the local trajectory to be a circular arc in the plane, and the heading of the vehicle to remain tangential to the arc. The model depends on only two parameters, which are the radius of the circle and the inscribed angle of the arc. This kinematic constraint employs a more restrictive but exact geometric representation for the kind of smooth trajectory that we have on drift-less, non-holonomic ground vehicles. The present thesis will introduce a new back-end optimization framework that relies on continuous-time parametrizations to continuously enforce the vehicle kinematic constraints.

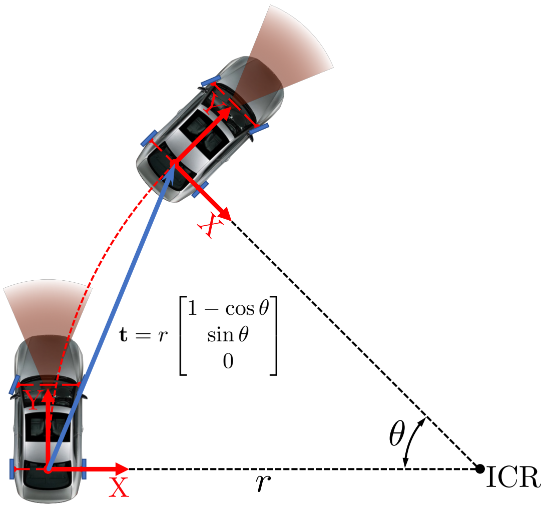

Y
X
$\mathbf{t} = r \begin{bmatrix} 1 - \cos\theta \\ \sin\theta \\ 0 \end{bmatrix}$
Y
X
θ
r
ICR

Figure 1.10: Kinematic constraints of the Ackermann steering model.

1.2.4 Globally-optimal event camera motion estimation for planar ground vehicles

In all the aforementioned points, we will investigate how to use multiple regular cameras as one system to develop a powerful SLAM system. Regular cameras have been widely regarded as a highly attractive alternative in most SLAM system, however, despite their advantageous properties from an economic point of view, they also come at a certain cost (cf. Figure 1.11):

- Regular cameras only measure perspective projections of the environment, and the reconstruction of especially the dense geometry of the environment therefore proposes a solvable but challenging problem.

- Regular cameras measure at a certain fixed frame-rate, and therefore return the information with a certain minimum latency.
- Regular cameras suffer from motion blur in situations of high dynamics, a circumstance that easily occurs for example if mounted on an intelligence augmentation device.
- Regular cameras only measure photometric reflections with limited dynamic range, thus leading to poor performance in challenging (especially changing) illumination conditions in one and the same view.

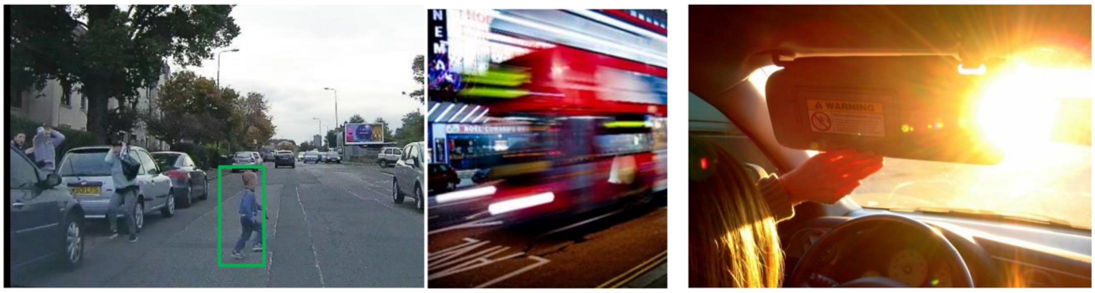


Figure 1.11: Disadvantages of regular cameras: low latency, motion blur, and low dynamic range. (source: [Scaramuzza, 2020])

Each of the issues above can make failures of existing visual SLAM system developed for regular cameras very likely. Hence we look into recently emerging sensors that are capable to deal with the challenging issues above.

In Chapter 5, we look at a new type of camera sensor, namely a dynamic vision sensor or event camera, which improves the camera performance in the aforementioned challenging scenario. In contrary to regular cameras, event cameras are neuromorphic sensors that do not directly measure the raw intensity of an incoming light beam, but a quantized version of the relative intensity with respect to a reference time. The quantization happens at every pixel independently, and an event is triggered as soon as the absolute value of the logarithm of the difference in brightness exceeds a preset threshold. Figure 1.12 illustrates the basic principle of operation of event cameras. If neither camera nor environment moves, no events will be triggered (except noise). As soon as there are relative motion between the camera and the environment, it can be clearly observed that events being triggered particularly along the high-gradient regions in the image. In contrary to regular cameras, the pixels operate asynchronously and independently of one another. There are no frames, and the virtual frame rate can be orders of magnitude larger. Every event is associated with a timestamp read from a central clock, which has a temporal resolution in the order of 1μs. The asynchronous nature and high temporal resolution of event cameras mean that the triggered event patterns do not suffer from artifacts such as motion blur. The result is high performance in highly dynamic scenarios, and very low latency. Furthermore, event cameras have very good high dynamic range, which contributes to their strong potential to handle challenging illumination scenarios.

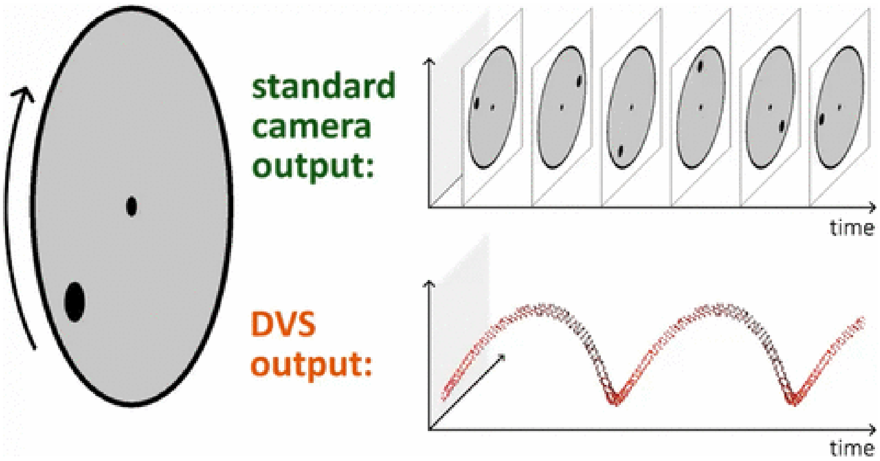


Figure 1.12: Basic principle of operation of event cameras. (source: [Scaramuzza, 2020])

Although event cameras are commercially available since about a decade now, there are still significant gaps in the literature thus leaving their potential somewhat unexplored. The computer vision and robotics community maintain a high interest in developing fully convincing large-scale visual SLAM frameworks with single or multiple event cameras, in order to push the limits of conventional visual SLAM frameworks and effectively increase the perception abilities of mobile intelligent platforms in challenging real-world conditions.

Regarding localization and mapping with event cameras, some preliminary methods such as [Mueggler et al., 2015], [Reverter Valeiras et al., 2016], [Gallego et al., 2017] and [Bryner et al., 2019] are focusing on camera tracking with known priors. [Gallego et al., 2018] and [Gallego et al., 2019] recently introduced a unifying framework called contrast maximization that allowing the solution of several important problems for event cameras, in particular motion estimation problems in which the effect of camera motion may be described by a homography (e.g. motion in front of a plane, pure rotation). However, all methods above proceed by local optimizations rather than globally optimal solutions, and contrast maximization for motion estimation is in general a non-convex problem, which means that local optimization may be sensitive to the initial parameters and may not find the global optimum. The present thesis is introducing a new globally optimal motion estimation framework for event cameras.

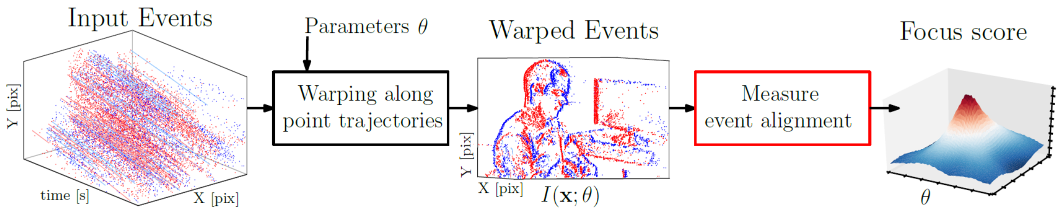


Figure 1.13: Contrast Maximization framework. Optimal parameters will maximize the sharpness of the image of warped events (IWE) (source: [Gallego et al., 2019])

1.2.5 Event camera tracking and mapping by volumetric contrast maximization

As mentioned in Section 1.2.4, some existing works have employed time-continuous parametrizations of image warping functions. Based on the assumption that events are pre-dominantly triggered by high-gradients edges in the image, the optimal image warping parameters will cause the events to warp onto a sharp edge-map in a reference view called the Image of Warped Events (IWE). The optimal warping parameters are hence found by maximizing contrast in the IWE. Figure 1.13 illustrate the principle of the contrast maximization framework. Various reward functions to evaluate contrast have been presented and analysed in the recent works of Gallego et al. [Gallego et al., 2018, 2019] and Stoffregen and Kleeman [Stoffregen and Kleeman, 2019], and successfully used for solving a variety of problems with event cameras such as optical flow [Zhu et al., 2017; Gallego et al., 2018; Stoffregen and Kleeman, 2017; Ye et al., 2018; Zhu et al., 2019a, 2018b], segmentation [Stoffregen and Kleeman, 2017; Stoffregen et al., 2019; Mitrokhin et al., 2018], 3D reconstruction [Rebecq et al., 2018; Zhu et al., 2018a, 2019a; Ye et al., 2018], and motion estimation [Gallego and Scaramuzza, 2017; Gallego et al., 2018]. The main problem with the construction of the IWE is that it relies on a low-dimensional image-to-image warping function, which—in the case of both translational and rotational displacements—is only possible if the model is homographic or if knowledge about the depth of the scene is prior available. The present thesis is introducing a new method that extending the idea of contrast maximization into 3D, a technique that will enable us to handle situations in which we perceive non-planar environments under arbitrary motion and with no priors on the depth of events.

1.3 Thesis Outline

In Chapter 2, 3 and 4, we look into the motion estimation problem with a novel type of sensor system, namely multi-perspective camera systems. Such systems are widely equipped on modern vehicles and other intelligent mobile platforms. More specifically, the chapter is organized as follows:

- In Chapter 2, we propose a complete real-time pipeline for visual odometry with non-overlapping, multi-perspective camera systems, and in particular present a solution to the scale initialization problem. We firstly introduce a novel joint linear bootstrapping method in the initialization stage, in order to solve the unobservability of metric scale under degenerate cases. Furthermore, the present approach uses the measurements from all cameras simultaneously at all processing stages including bootstrapping, pose tracking, and mapping, which can truly benefit from the omni-directional measurements of multi-perspective camera systems.

- In Chapter 3, we focus on the further improvement of front-end relative pose estimation for vehicle-mounted surround-view multi-camera systems. We present the first generalized relative pose solver for general planar vehicle motion and non-overlapping multi-camera systems. We present a new univariate objective function that relies on a parallel evaluation of the epipolar geometry for each individual camera, rather than simply replacing the motion parametrization in existing formulations that rely on the generalized essential matrix (which degenerates in the case of non-overlapping multi-camera arrays). We introduce a modified, iteratively reweighted optimization of the planar motion that minimizes the geometrically relevant object space error, and we carefully analyze the behavior of the objective function and its ability to overcome rotation-translation ambiguities by exploiting omni-directional measurement distributions and geometrically meaningful residuals. Most notably, the objective function remains a uni-variate expression, and therefore outperforms two-view bundle adjustment in terms of computational efficiency.

- In Chapter 4, we explore the continues-time parametrization for an exact modelling of non-holonomic ground vehicle trajectories in the back-end optimization of visual SLAM pipeline. We use the B-spline based, smooth, continuous-time trajectory representation introduced by [Furgale et al., 2015] to represent the motion of ground vehicles. We introduce different constraints and compare the resulting frameworks against conventional solutions. We demonstrate a significant advantage in robustness and accuracy on both simulated and real data.

In Chapter 5 and 6, in order to deal with challenges in scenarios with high dynamics, low texture distinctiveness, or challenging illumination conditions, we focus on the solution to localization and mapping problem on Autonomous Ground Vehicle (AGV) using event cameras. We propose two new algorithms to tackle several motion estimation problems by performing the contrast maximization approach:

- In Chapter 5, we introduce the first globally optimal solution to contrast maximization for unwarped event streams, and apply the idea of homography estimation via contrast maximization to the real-world case of non-holonomic motion estimation with a downward-facing camera mounted on an AGV. We solve the global maximization of contrast functions via Branch and Bound. We derive bounds for contrast estimation functions. The bounds are furthermore calculated recursively, which enables efficient processing. We successfully apply this strategy to Autonomous Ground Vehicle (AGV) planar motion estimation with a downward-facing event camera, a problem that is complicated by motion blur, challenging illumination conditions, and indistinctive, noisy textures. We prove that using an event camera can solve these challenges, hence outperforming alternatives given by regular cameras.

- In Chapter 6, we present a new solution to tracking and mapping with an event camera by continuous ray warping and volumetric contrast maximization, and validate our approach by applying it to AGV motion estimation and 3D reconstruction with a single vehicle-mounted event camera. We extend the idea of contrast maximization into 3D. Using a time-continuous trajectory model, the 3D location of the landmarks corresponding to events is modelled by time-continuous ray warping in space, and the optimal motion parameters are found by maximizing contrast within a volumetric ray density field. Our method is the first to perform joint optimization over motion and structure for cameras exerting both translational and rotational displacements in an arbitrarily structured environment. We successfully apply our framework to Autonomous Ground Vehicle (AGV) motion estimation with a forward-facing event camera. We prove that using only an event camera, we can provide good quality, continuous visual localization and mapping results able to compete with regular camera alternatives, especially as visual conditions degrade.

1.4 List Publications

1.4.1 Published

- [Wang and Kneip, 2017] **Y Wang** and L Kneip. On scale initialization in non-overlapping multi-perspective visual odometry. In Proceedings of the International Conference on Computer Vision Systems(ICVS), Shenzhen, July 2017. **Best Student Paper Award**

- [Huang et al., 2019] K Huang, **Y Wang** and L Kneip. Motion estimation of non-holonomic ground vehicles from a single feature correspondence measured over n views. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR), June. 2019.

- [Wang et al., 2020] **Y Wang**, K Huang, X Peng, H Li and L Kneip. Reliable frame-to-frame motion estimation for vehicle-mounted surround-view camera systems. In Proceedings of the 2020 IEEE International conference on robotics and automation (ICRA), June. 2020.

- [Peng et al., 2020] X Peng*, **Y Wang***, L Gao* and L Kneip. Globally-Optimal Event Camera Motion Estimation. In Proceedings of the European Conference on Computer Vision (ECCV), Aug. 2020.

- [Peng et al., 2021] X. Peng, L. Gao, **Y. Wang** and L. Kneip, "Globally-Optimal Contrast Maximisation for Event Cameras," in IEEE Transactions on Pattern Analysis and Machine Intelligence, Jan. 2021.

1.4.2 Under Review

- **Y Wang**, W Peng, X Peng, L Gao, H Li and L Kneip. ETAM: Event Camera Tracking and Mapping by Continuous Ray Warping and Volumetric Contrast Maximization. Submitted to the 2021 IEEE International conference on robotics and automation (ICRA).

- K Huang, **Y Wang** and L Kneip. B-splines for purely vision-based localization and mapping on non-holonomic ground vehicles. Submitted to the 2021 IEEE International conference on robotics and automation (ICRA).

Chapter 2

On scale initialization in non-overlapping multi-perspective visual odometry

In this chapter, we introduce a complete pipeline for motion estimation with non-overlapping multi-perspective camera systems, and in particular presents a solution to the scale initialization problem. Multi-perspective camera systems (MPCs) pointing into all directions represent an increasingly interesting solution for visual localization and mapping. They combine the benefits of omni-directional measurements with a sufficient baseline for producing measurements in metric scale. However, the observability of metric scales suffers from degenerate cases if the cameras do not share any overlap in their field of view. This problem is of particular importance in many relevant practical applications, and it impacts most heavily on the difficulty of bootstrapping the structure-from-motion process. We evaluate our method on both simulated and real data, thus proving robust initialization capacity as well as best-in-class performance regarding the overall motion estimation accuracy.

2.1 Related work

The motion estimation problem with MPCs can be approached in two fundamentally different ways. The first one consists of a loosely-coupled scheme where the information in each camera is used to solve individual monocular structure-from-motion problems, and the results from every camera are then fused in a subsequent pose averaging module. Kazik et al. [Kazik et al., 2012] apply this solution strategy to a stereo camera rig with two cameras pointing into opposite directions. The inherent difficulty of this approach results from the scale invariance of the individual monocular structure-from-motion results. Individual visual scales first have to be resolved through an application of the hand-eye calibration constraint [Horaud and Dornaika, 1995] before the individual pose results can be fused. Furthermore, the fact that the measurements of each camera are processed independently means that the benefit of having omni-directional measurements remains effectively unexploited during the geometric computations.

The second solution strategy assumes that the frames captured by each camera are synchronized, and hence can be bundled in a multi-frame measurement that contains one image of each camera from the same instant in time. Relying on the idea of *Using many cameras as one* [Pless, 2003], the fundamental problems of structure from motion can now be solved jointly for the entire MPC system, rather than for each camera individually. The measurements captured by the entire MPC can notably be described using a generalized camera, a model that envisages the description of measured image points via spatial rays that intersect with the corresponding camera's center, all expressed in a common frame for the entire MPC. By relying on the generalized camera model, the problems of joint absolute and relative camera pose estimation for the entire MPC rig have been successfully solved [Nistér and Stewénius, 2006; Kneip et al., 2013, 2014; Pless, 2003; Stewénius and Nistér, 2005; Li et al., 2008; Kneip and Li, 2014]. An excellent summary of the state-of-the-art in generalized camera pose computation is provided by the OpenGV library[Kneip and Furgale, 2014], a relatively complete collection of algorithms for solving related problems.

Despite the fact that closed-form solutions for the underlying algebraic geometry problems of generalized absolute and relative camera pose computation have already been presented, a full end-to-end pipeline for visual odometry with a non-overlapping MPC system that relies exclusively on the generalized camera paradigm remains an open problem. The problem mostly lies in the bootstrapping phase. As explained in [Clipp et al., 2008], the relative pose for a multi-camera system can only be computed if the motion does not suffer from the degenerate case of Ackermann-like motion (which includes the case of purely straight motion). Unfortunately, in a visual odometry scenario, the images often originate from a smooth trajectory with only moderate dynamics, hence causing the motion between two sufficiently close frames to be almost always very close to the degenerate case. Kneip and Li [Kneip and Li, 2014] claim that the rotation can still be found, but we confirmed through our experiments that even the quality of the relative rotation is not sufficiently good to reliably bootstrap MPC visual odometry. A robust initialization procedure, as well as a complete, real-time end-to-end pipeline, notably, are the main contributions of this work.

The chapter is organized as follows. Section 2.2 provides an overview of our complete non-overlapping MPC motion estimation pipeline as well as the joint bootstrapping and global optimization modules. Section 2.3 finally presents the promising results we have obtained on both simulated and real data.

2.2 Joint motion estimation with non-overlapping MPCs

This section outlines our complete MPCs motion estimation pipeline. We start with an overview of the entire framework, explaining the state machine and the resulting sequence of operations especially during the initialization procedure. We then look at two important sub-problems of the initialization, namely the robust retrieval of absolute orientations for the first frames of a sequence, as well as a joint linear recovery of the corresponding relative translations and 3D points. We conclude with an insight into the final bundle adjustment back-end that is entered once the initialization is completed.

2.2.1 Notations and prior assumptions

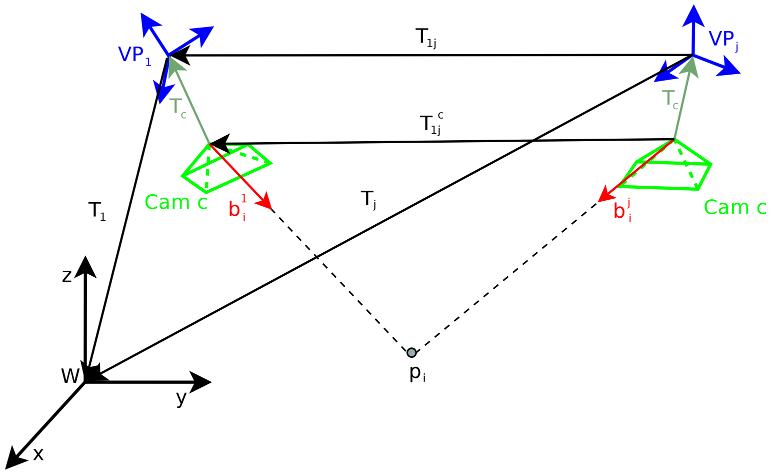


Figure 2.1: Notations used throughout this chapter (best viewed in color). Please see text for detailed explanations.

We start by introducing the geometry of the generalized camera system. The MPC frames of a video sequence are denoted by VP_j, where $j = \{1, \cdots, m\}$. Their poses are expressed by transformation matrices $\mathbf{T}_j = \begin{bmatrix} \mathbf{R}_j & \mathbf{t}_j \\ \mathbf{0} & 1 \end{bmatrix}$ such that $\mathbf{T}_j\mathbf{x}$ transforms $\mathbf{x}$ from the MPC to the world frame (denoted W). Let us now assume that our MPC has k cameras. This leads to the definition of transformation matrices $\mathbf{T}_c = \begin{bmatrix} \mathbf{R}_c & \mathbf{t}_c \\ \mathbf{0} & 1 \end{bmatrix}$, where $c \in \{1, \cdots, k\}$. They permit the transformation of points from the respective camera frame c to the MPC frame. Assuming that the MPC rig is static, these transformations are constant and determined through a prior extrinsic calibration process. We also define the relative transformation $\mathbf{T}_{1j} = \begin{bmatrix} \mathbf{R}_{1j} & \mathbf{t}_{1j} \\ \mathbf{0} & 1 \end{bmatrix}$ that allows us to transform points from VP_j back to VP_1. We furthermore assume that—given that we are in a visual odometry scenario and that the cameras have no overlap in their fields of view—the cameras do not share any point observations. We therefore can associate each one of our points $\mathbf{p}_i, i \in \{1, \cdots, n\}$ to one specific camera within the rig, denoted by the index c_i. To conclude, we also assume that the intrinsic camera parameters are known, which is why we can always transform 2D points into spatial unit vectors pointing from the individual camera centers to the respective world points. We denote these measurements $\mathbf{b}_i^j$, meaning the measurement of point

$\mathbf{p}_i$ (with camera c_i) in the MPC frame VP_j. Our derivations furthermore utilize the transformation $\mathbf{T}^c_{1j}$, which permits the direct transformation of points from the camera frame c in VP_j to the camera frame c in VP_1. All variables are indicated in Figure 2.1.

2.2.2 Framework overview

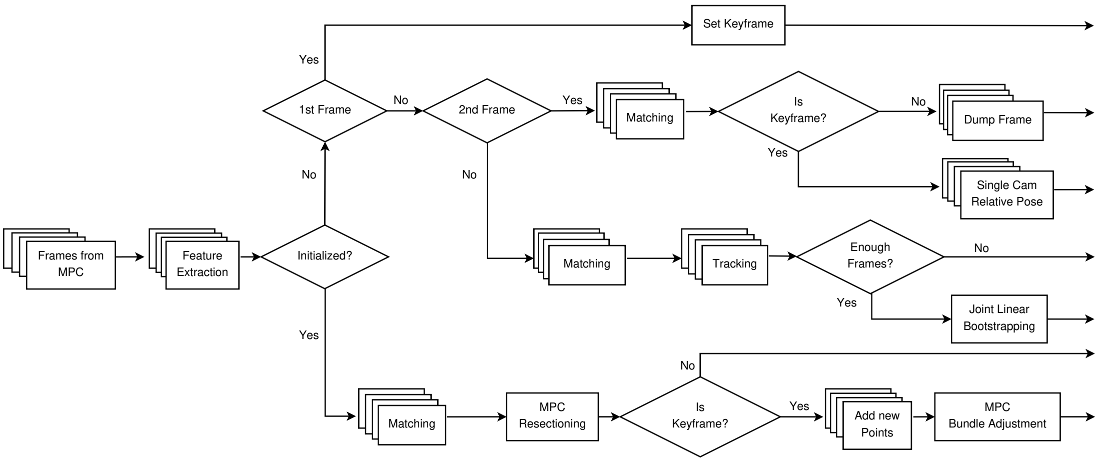


Figure 2.2: Overview of the proposed visual localization and mapping pipeline for MPC systems. The flowchart in particular outlines the detailed idea behind the initialization procedure.

A flowchart of our proposed method detailing all steps including the initialization procedure is illustrated in Figure 2.2. After the definition of a first (multi-perspective) keyframe[1], the algorithm keeps matching inter-camera correspondences between the first and subsequent MPC frames until the average of the median frame-to-frame disparity for each camera surpasses a predefined threshold (verified in the decision nodes "Is Keyframe?"). Once this happens, we add a second keyframe and compute all $\mathbf{T}^c_{12}$ using classical single camera calibrated relative pose computation [Stewénius et al., 2006]. We furthermore triangulate an individual point cloud for every camera in the MPC array. Subsequent frames from the individual cameras are then aligned with respect to these maps using classical single camera calibrated absolute pose computation [Kneip et al., 2011]. Once enough frames are collected, the initialization is completed by the joint, linear MPC pose initialization module outlined in Sections 2.2.3 and 2.2.4. Note that individual single camera tracking is only performed in order to eliminate outlier measurements and obtain prior knowledge about relative rotations. It bypasses the weakness of methods such as [Kneip and Li, 2014] of not being able to deliver robust generalized relative pose results in most practically

[1] *Keyframes* are simply frames that are retained in a buffer of frames due to sufficient local distinctiveness [Klein and Murray, 2007].

relevant cases. The actual final initialization step and all subsequent modules then perform joint MPC measurement processing.

After the initialization is completed, the frames of each new MPC pose are matched individually to the frames of the most recent MPC keyframe, but the alignment is solved jointly using generalized camera absolute pose computation [Kneip et al., 2013]. We keep checking the local distinctiveness of every MPC frame by evaluating the frame-to-frame disparities in the above outlined manner, and add new keyframes everytime the threshold is surpassed. To conclude, we add new 3D points everytime a new keyframe is added, and perform generalized windowed bundle adjustment to jointly optimize over several recent MPC poses and the 3D landmark positions. This back-end optimization procedure is outlined in Section 2.2.5.

2.2.3 Initial estimation of relative rotations

The very first part of our computation executes visual odometry in each camera individually. In order to make use of the relative orientations, we propose to first eliminate the redundancy in the information. This is done by first combining the computed orientations with the camera-to-MPC transformations $\mathbf{T}_c$ in order to obtain relative orientation estimates for the entire MPC rig. We now have k samples for the MPC frame-to-frame orientations in the frame buffer. We apply L_1 rotation averaging based on the Weiszfeld algorithm as outlined in [Hartley et al., 2013] in order to obtain an accurate, unique representation.

2.2.4 Joint linear bootstrapping

The computation steps until here provide sets of inlier inter-camera correspondences and reasonable relative rotations between subsequent MPC frames. The missing variables towards a successful bootstrapping of the computation are given by MPC positions and point depths. Translations and point depths can also be taken from the prior individual visual odometry computations [Kazik et al., 2012], but they may be unreliable, and—more importantly—have different unknown visual scale factors that would first have to be resolved.

We propose a new solution to this problem which solves for all scaled variables (i.e. positions and point depths) through one joint, closed-form, linear initialization procedure. What we are exploiting here is the known fact that structure from motion can be formulated as a linear problem once the relative rotations are subtracted from the computation (although results will not minimize a geometrically meaningful error anymore).

Let us assume that we have two MPC view-points VP_1 and VP_j. We start by formulating the hand-eye calibration constraint for a camera c inside the MPC

$$\begin{cases} \mathbf{t}_c = \mathbf{t}_{1j} + \mathbf{R}_{1j} \cdot \mathbf{t}_c + \mathbf{R}_{1j} \cdot \mathbf{R}_c \cdot \mathbf{t}^c_{j1} \\ \mathbf{R}_c = \mathbf{R}_{1j} \cdot \mathbf{R}_c \cdot \mathbf{R}^c_{j1} \end{cases} \tag{2.1}$$

Let us now assume that there is one observed world point $\mathbf{p}_i$ giving rise to the measurements $\mathbf{b}_i^1$ and $\mathbf{b}_i^j$ inside the camera. The latter now has the index c_i. The point inside the first camera is simply given as $\lambda_i \cdot \mathbf{b}_i^1$, where λ_i denotes the depth of $\mathbf{p}_i$ seen from camera c_i in VP_1. We now apply $\mathbf{T}_{j1}^{c_i}$ and transform this point into camera c_i of VP_j. In here, the point obviously needs to align with the direction $\mathbf{b}_i^j$, which leads us to the constraint

$$(\mathbf{R}_{j1}^{c_i} \cdot \lambda_i \cdot \mathbf{b}_i^1 + \mathbf{t}_{j1}^{c_i}) \times \mathbf{b}_i^j = 0. \tag{2.2}$$

By replacing (2.1) in (2.2), we finally arrive at

$$(\mathbf{R}_{c_i}^T \cdot \mathbf{R}_{1j}^T \cdot \mathbf{R}_{c_i} \cdot \lambda_i \cdot \mathbf{b}_i^1) \times \mathbf{b}_i^j - (\mathbf{R}_{c_i}^T \cdot \mathbf{R}_{1j}^T \cdot \mathbf{t}_{1j}) \times \mathbf{b}_i^j = -\mathbf{R}_{c_i}^T \cdot \mathbf{R}_{1j}^T (\mathbf{t}_{c_i} - \mathbf{R}_{1j} \cdot \mathbf{t}_{c_i}) \times \mathbf{b}_i^j. \tag{2.3}$$

Let us now assume that we have n points and m MPC frames. The unknowns are hence given by λ_i, where $i \in \{1, \cdots, n\}$, and $\mathbf{t}_{1j}$, where $j \in \{2, \cdots, m\}$. We only use fully observed points, meaning that each point $\mathbf{p}_i$ is observed by camera c_i in each MPC frame, thus generating the measurement sequence $\left\{\mathbf{b}_i^1, \cdots, \mathbf{b}_i^m\right\}$. All pair-wise constraints in the form of (2.3) can now be grouped in one large linear problem $\mathbf{A}\mathbf{x} = \mathbf{b}$, where

$$\mathbf{A} = \begin{bmatrix} (\mathbf{R}_{c_1}^T \mathbf{R}_{12}^T \mathbf{R}_{c_1} \mathbf{b}_1^1) \times \mathbf{b}_1^2 & & & [\mathbf{b}_1^2]_\times \mathbf{R}_{c_1}^T \mathbf{R}_{12}^T & & \\ & \cdots & & \cdots & & \\ & & (\mathbf{R}_{c_n}^T \mathbf{R}_{12}^T \mathbf{R}_{c_n} \mathbf{b}_n^1) \times \mathbf{b}_n^2 & [\mathbf{b}_n^2]_\times \mathbf{R}_{c_n}^T \mathbf{R}_{12}^T & & \\ \cdots & \cdots & \cdots & & \cdots & \\ (\mathbf{R}_{c_1}^T \mathbf{R}_{1m}^T \mathbf{R}_{c_1} \mathbf{b}_1^1) \times \mathbf{b}_1^m & & & & & [\mathbf{b}_1^m]_\times \mathbf{R}_{c_1}^T \mathbf{R}_{1m}^T \\ & \cdots & & & & \cdots \\ & & (\mathbf{R}_{c_n}^T \mathbf{R}_{1m}^T \mathbf{R}_{c_n} \mathbf{b}_n^1) \times \mathbf{b}_n^m & & & [\mathbf{b}_n^m]_\times \mathbf{R}_{c_n}^T \mathbf{R}_{1m}^T \end{bmatrix} \tag{2.4}$$

$$\mathbf{x} = \begin{bmatrix} \lambda_1 \\ \cdots \\ \lambda_n \\ \mathbf{t}_{12} \\ \cdots \\ \mathbf{t}_{1m} \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} -\mathbf{R}_{c_1}^T \mathbf{R}_{12}^T (\mathbf{t}_{c_1} - \mathbf{R}_{12} \mathbf{t}_{c_1}) \times \mathbf{b}_1^2 \\ \cdots \\ -\mathbf{R}_{c_n}^T \mathbf{R}_{12}^T (\mathbf{t}_{c_n} - \mathbf{R}_{12} \mathbf{t}_{c_n}) \times \mathbf{b}_n^2 \\ \cdots \\ -\mathbf{R}_{c_1}^T \mathbf{R}_{1m}^T (\mathbf{t}_{c_1} - \mathbf{R}_{1m} \mathbf{t}_{c_1}) \times \mathbf{b}_1^m \\ \cdots \\ -\mathbf{R}_{c_n}^T \mathbf{R}_{1m}^T (\mathbf{t}_{c_n} - \mathbf{R}_{1m} \mathbf{t}_{c_n}) \times \mathbf{b}_n^m \end{bmatrix} \tag{2.5}$$

$\mathbf{A}$ and $\mathbf{b}$ can be computed from the known extrinsics, inlier measurements, and relative rotations, whereas $\mathbf{x}$ contains all unknowns.

The non-homogeneous linear problem $\mathbf{A}\mathbf{x} = \mathbf{b}$ could be solved by a standard technique such as QR decomposition, thus resulting in $\mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$. However, in order to improve efficiency, we utilize the Schur-complement trick and exploit the sparsity pattern of the matrix. Matrix $\mathbf{A}^\mathbf{T}\mathbf{A}$ is divided into four smaller sub-blocks $\mathbf{A}^T\mathbf{A} = \begin{bmatrix} \mathbf{P} & \mathbf{Q} \\ \mathbf{R} & \mathbf{S} \end{bmatrix}$, and our two vectors $\mathbf{x}$ and $\mathbf{b}$ are decomposed accordingly thus resulting in $\mathbf{x} = [\mathbf{x}_1^T, \mathbf{x}_2^T]^T$ and $\mathbf{A}^\mathbf{T}\mathbf{b} = [(\mathbf{A}^\mathbf{T}\mathbf{b}_1)^T, (\mathbf{A}^\mathbf{T}\mathbf{b}_2)^T]^T$. Substituted into the original equation $\mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{A}^T\mathbf{b}$, and after variable elimination, we obtain

$$\begin{cases} \mathbf{P}\mathbf{x}_1 = \mathbf{A}^\mathbf{T}\mathbf{b}_1 - \mathbf{Q}\mathbf{x}_2 \\ (\mathbf{S} - \mathbf{R}\mathbf{P}^{-1}\mathbf{Q})\mathbf{x}_2 = \mathbf{A}^\mathbf{T}\mathbf{b}_2 - \mathbf{R}\mathbf{P}^{-1}\mathbf{A}^\mathbf{T}\mathbf{b}_1 \end{cases} \tag{2.6}$$

This form permits us to first solve for $\mathbf{x}_2$ individually, a much smaller problem due to the relatively small number of MPC frames. $\mathbf{x}_1$ is subsequently retrieved by simple variable back-substitution.

2.2.5 Multi-perspective windowed bundle adjustment

After bootstrapping, we can continuously use multi-perspective absolute camera pose computation [Nistér and Stewénius, 2006] in order to align subsequent MPC frames with respect to the local point cloud. Furthermore, we keep buffering keyframes each time the average frame-to-frame disparity exceeds a given threshold. This in fact already constitutes a complete procedure for MPC visual odometry. In order to improve the accuracy of the solution, we add a windowed bundle adjustment back-end to our pipeline [Hartley and Zisserman, 2004]. The goal of windowed bundle adjustment(BA) is to optimize 3D point positions and estimated MPC poses over all correspondences observed in a certain number of most recent keyframes. The key idea here is that points are generally observed in more than just two keyframes. By minimizing the reprojection error of every point into every observation frame, we implicitly take multi-view constraints into account, thus improving the final accuracy of both structure and camera poses. The computation is restricted to a bounded window of keyframes not to compromise computational efficiency. This form of non-linear optimization is also known as *sliding window bundle adjustment*.

Let us define the set $\mathcal{J}_i = \{j_1, \cdots, j_k\}$ as the set of MPC keyframe indices for which camera c_i observes the point $\mathbf{p}_i$. Let us furthermore assume that the size of the optimization window is s, and the set of points is already limited to points that have at least two observations within the s most recent keyframes. The objective of windowed bundle adjustment can now be formulated as

$$\begin{aligned} &\left\{\hat{\mathbf{T}}_{m-s+1}, \cdots, \hat{\mathbf{T}}_m, \hat{\mathbf{p}}_1, \cdots, \hat{\mathbf{p}}_n\right\} = \\ &\underset{\mathbf{T}_{m-s+1},\cdots,\mathbf{T}_m,\mathbf{p}_1,\cdots,\mathbf{p}_n}{\operatorname{argmin}} \sum_{i=1}^{n} \sum_{j\in\mathcal{J}_i} \|\pi_{c_i}(\tilde{\mathbf{b}}_i^j) - \pi_{c_i}(\mathbf{T}_{c_i}^{-1}\mathbf{T}_j^{-1}\tilde{\mathbf{p}}_i)\|^2. \end{aligned} \tag{2.7}$$

where

- $\mathbf{T}_j$ is parametrized minimally as a function of 6 variables.
- $j \in \{m - s + 1, \cdots, m\}$.
- π_{c_i} is the known (precalibrated) camera-to-world function of camera c_i. It transforms 3D points in homogeneous form into 2D Euclidean points.
- $\tilde{\mathbf{x}} =$ takes the homogeneous form of $\mathbf{x}$ by appending a 1.

- $\pi_{c_i}(\tilde{\mathbf{b}}_i^j)$ is the original, measured image location of the spatial direction $\mathbf{b}_i^j$.

2.3 Experimental results

We test our algorithm on both simulated and real data. The simulation experiments analyze the noise resilience of our linear bootstrapping algorithm. The real data experiment then evaluates the performance of the complete pipeline by comparing the obtained results against ground truth data collected with an external motion tracking system, as well as a loosely-coupled alternative.

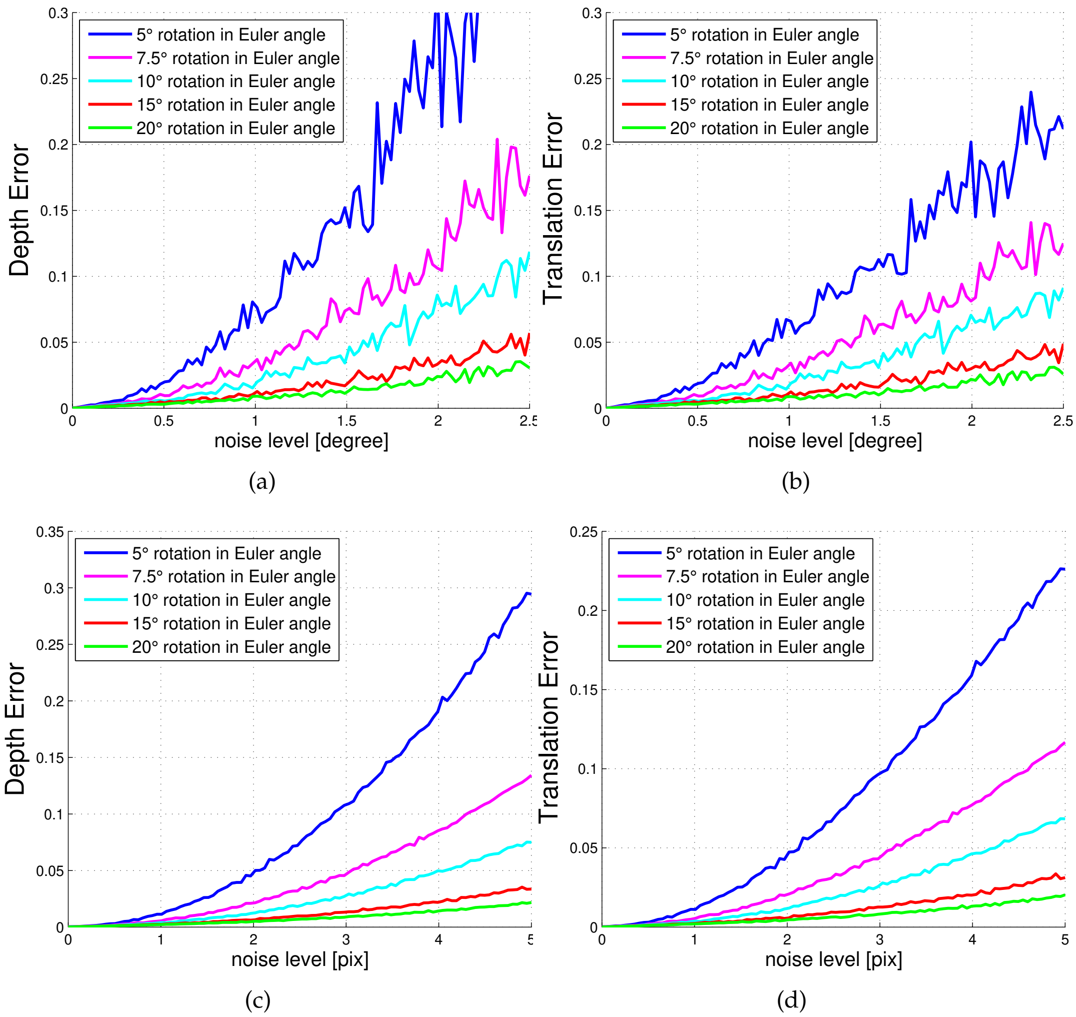
2.3.1 Results on simulated data

We perform experiments on synthetic data to analyze the performance of our linear MPC pose initialization module in the presence of varying levels of noise. In all our simulation experiments, we simply use 2 cameras pointing into opposite directions, and generate 10 random points in front of each camera. We furthermore generate 10 homogeneously distributed camera poses generating near fronto-parallel motion for both cameras. To conclude, we add an oscillating rotation about the main direction of motion. The maximum amplitude of this rotation is set to either $5°, 7.5°, 10°, 15°$ or $20°$, which creates an increasing distance to the degenerate case of Ackerman motion. We perform two separate experiments in which we add noise to either the relative rotations or the 2D bearing vector measurements pointing from the camera centers to the landmarks. The error for each noise level is averaged over 1000 random experiments.

In our first experiment, we add noise to the relative rotations by multiplying them with another random rotation matrix that is derived from uniformly sampled Euler angles with a maximum value reaching from zero to $2.5°$ degrees. The reported errors are the relative depth error of the 3D points $\frac{\|\lambda_{est}\|}{\|\lambda_{true}\|}$, and the relative translation magnitude error $\frac{\|\mathbf{t}_{est}\|}{\|\mathbf{t}_{true}\|}$. The errors are indicated in Figures 2.3(a) and 2.3(b), respectively.

In our second experiment, we simulate noise on the bearing vectors by adding a random angular deviation θ_{rand} such that $\tan\theta_{\text{rand}} < \frac{\sigma}{f}$, where f is a virtual focal length of 500 pixels, and σ is a virtual maximum pixel noise level reaching from 0 to 5 pixels. We analyze the same errors and the results are reported in Figures 2.3(c) and 2.3(d), respectively.

As can be concluded from the results, a reasonable amount of noise in both the point observations as well as the relative rotations can be tolerated. However, the correct functionality of the linear solver depends critically on the observability of the metric scale. Limiting the maximum amplitude of the out-of-plane rotation to a low angle (e.g. below $5°$) can quickly compromise the stability of the solver and cause very large errors. In practice, this means that accurate results can only be expected if we add sufficiently many frames with sufficiently rich dynamics to our solver.

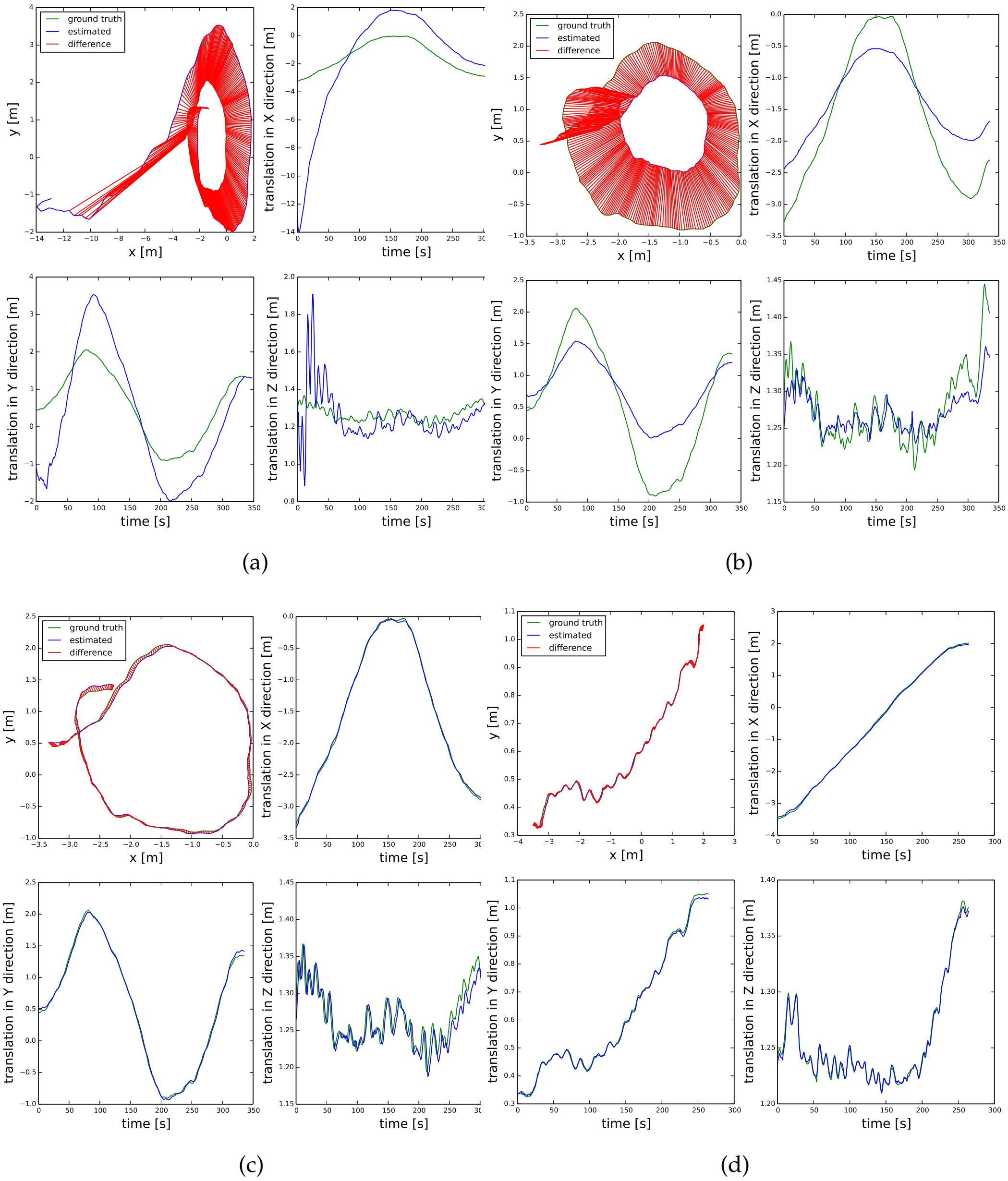


(a) (b) (c) (d)

Figure 2.3: Benchmark of our linear bootstrapping algorithm showing relative translation and 3D point depths error for different levels of noise in the relative rotation and 2D landmark observations. The experiment is repeated for different "out-of-plane" dynamics, which causes significant differences in the scale observability of the problem. Note that each value in the figures is averaged over 1000 random experiments.

2.3.2 Results on real data

We have been given access to the data already used in [Kazik et al., 2012], which allows us to compare our method against accurate ground truth measurements obtained by an external tracking device, a loosely-coupled alternative [Kazik et al., 2012], and a more traditional approach from the literature [Clipp et al., 2008]. The data consists of two different sequences captured with a synchronized, non-overlapping stereo rig that contains two cameras facing opposite directions. For further details about the hardware including intrinsic parameter values as well as the extrinsic calibration procedure, the reader is kindly referred to [Kazik et al., 2012]. The sequences are henceforth referred to as the *circular* and *straight* motion sequences. In the circular motion sequence, the rig moves with significant out-of-plane rotation along a large loop. In the straight motion sequence, the rig simply moves forward with significantly reduced out-of-plane rotation. Both *circular* and *straight* datasets run



(a)

(b)

(c)

(d)

Figure 2.4: (c) and (d): Results of our method on *circular* and *straight* sequences collected by [Kazik et al., 2012]. (a): Results of our method without proposed initialization module on the *circular* sequence. (b): Results of our method without windowed bundle adjustment module on the *circular* sequence.

at 10FPS. All experiments are conducted on a regular desktop computer with 8GB RAM and an Intel Core i7 2.8 GHz CPU. Our C++ implementation runs in real-time, and uses OpenCV [Bradski, 2000], Eigen [Guennebaud et al., 2010], OpenGV [Kneip and Furgale, 2014] and the Ceres Solver library [Agarwal et al., 2010].

In order to assess the impact of our proposed linear bootstrapping and generalized sliding window bundle adjustment modules, we analyze three different algorithm configurations on the circular motion sequence. In our first test—indicated in Figure 2.4(a)—we do not use our proposed initialization procedure, but simply rely on the method presented in [Kneip and Li, 2014] to bootstrap the algorithm from a pair of sufficiently separated frames in the beginning of the sequence. We tested numerous entry points, but the algorithm consistently fails to produce a good initial relative translation, thus resulting in severely distorted trajectories. In our second test—indicated in Figure 2.4(b)—we rely on our linear bootstrapping algorithm to initialize the structure-from-motion process, but still do not activate windowed bundle adjustment. The obtained results are already much better, but still relatively far away from ground truth. It seems that our linear solver is able to produce meaningful initial values, but—due to the ill-posed nature of the problem—still has some error and further error is accumulated throughout the sequence. In our final test—indicated in Figure 2.4(c)—we then also activate the sliding window bundle adjustment, thus leading to high-quality results with very little drift away from ground truth. Once a sufficiently close initialization point is given, the non-linear optimization module is consistently able to compensate remaining scale and orientation errors. Finally, Figure 2.4(d) shows that the algorithm is also able to successfully process the more challenging straight motion sequence.

Method	Ratio of Norms
Approach by [Clipp et al., 2008]	1.005 ± 0.071
Approach by [Kazik et al., 2012]	0.90 ± 0.28
Our Method	0.996 ± 0.038
Method	**Vector Error**
Approach by [Clipp et al., 2008]	0.079 ± 0.061
Approach by [Kazik et al., 2012]	0.23 ± 0.19
Our Method	0.092 ± 0.049

Table 2.1: Performance comparison against [Kazik et al., 2012] and [Clipp et al., 2008]

Similar to [Kazik et al., 2012] we also calculated the ratio of the norms of the estimated and the ground truth translations as well as the relative translation vector error. The results are indicated in Figure 2.5. Table 2.1 furthermore compares our result against the results obtained in [Kazik et al., 2012] and [Clipp et al., 2008]. It can be observed that our method operates closest to the ideal ratio of 1 with the smallest standard deviation with respect to the ratio of norms of the estimated and ground truth translations. Looking at the relative translation vector error ratio, our

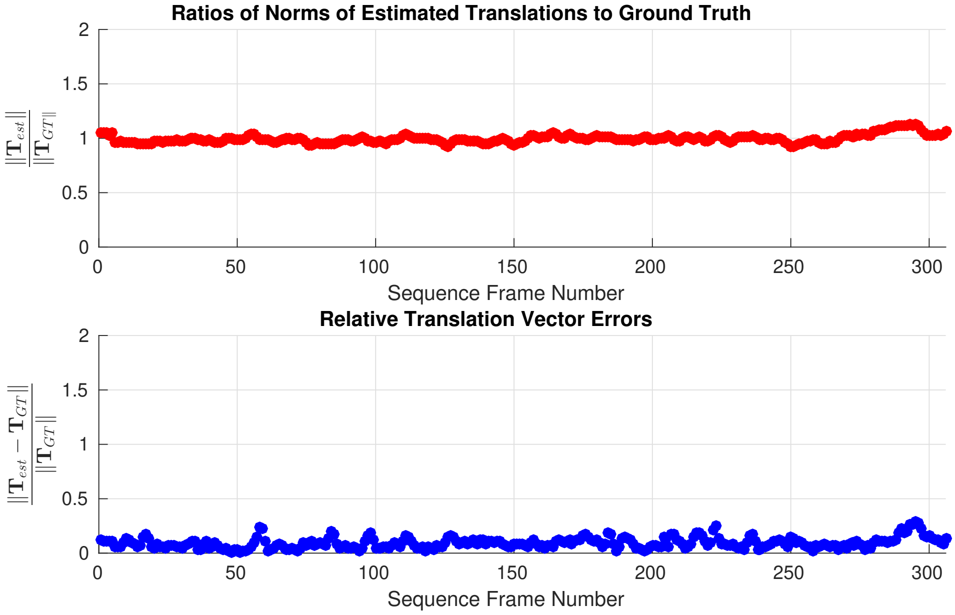


Figure 2.5: Ratios of Norms of Estimated Translations to Ground Truth and Relative Translation Vector Errors.

result is very close to the one obtained in [Kazik et al., 2012], and again achieves smaller standard deviation. The better standard deviation makes us believe that part of the reason for the slightly worse mean may be biases originating from an imperfect alignment with ground truth.

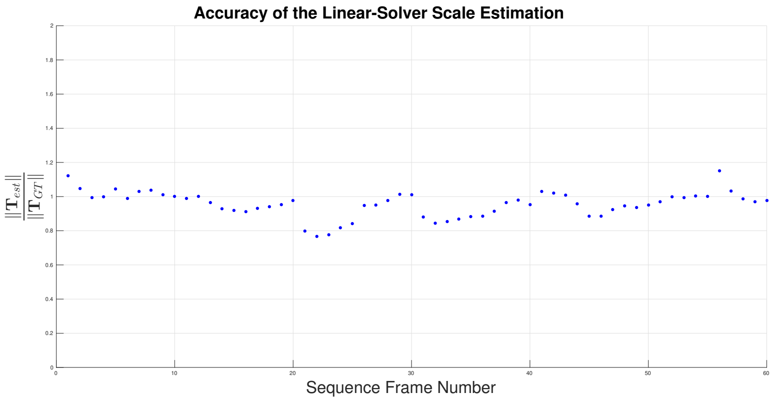


Figure 2.6: Accuracy of the linear bootstrapping technique for various starting points across the entire circular motion sequence.

As a final test, we consider it important to verify the performance of our linear bootstrapping algorithm on real data. Rather than applying it just in the very beginning of the dataset, we therefore test if the initialization method can work for arbitrary starting positions across the entire circular motion sequence. The test result of the ratio of norms of estimated and ground truth translations is indicated in Figure 2.6. The mean value of the ratio equals to 0.956 and the standard deviation is 0.075. We can conclude that, at least on this sequence, the linear initialization module performs consistently well.

2.4 Discussion

This chapter introduces a complete pipeline for motion estimation with non-overlapping multi-perspective camera systems. The main novelty lies in the fact that nearly all processing stages including bootstrapping, pose tracking, and mapping use the measurements from all cameras simultaneously. The approach is compared against a loosely coupled alternative, thus proving that the joint exploitation of the omnidirectional measurements leads to superior motion estimation accuracy.

While our result represents an unprecedented integration of the paradigm of *Using many cameras as one* into a full end-to-end real-time visual odometry pipeline, there still remains space for further improvements. For example, one remaining problem is that the success of our approach still depends on sufficiently good relative rotations estimated from each camera individually at the very beginning. Future research therefore consists of pushing generalized relative pose methods towards a robust recovery of relative rotations even in the case of motion degeneracies. A further point consists of parameterizing poses with similarity transformations, which would simplify drift compensation in the case of extended periods of scale unobservability.

Chapter 3

Reliable relative pose estimation for vehicle-mounted surround-view camera systems

In the previous chapter, we have seen how to construct a complete real-time pipeline for visual odometry with non-overlapping, multi-perspective camera systems, as well as a robust scale initialization procedure. Although our proposed method works impressively well, it is necessary to depend on sufficiently good relative rotations that estimated from each camera individually at the very beginning. In this chapter, we look at an efficient and reliable method specifically designed for estimating relative pose with non-overlapping multi-camera systems, which the motion can be approximated to remain in a plane. Modern vehicles are often equipped with a surround-view multi-camera system. The current interest in autonomous driving invites the investigation of how to use such systems for a reliable estimation of relative vehicle displacement. Existing camera pose algorithms either work for a single camera, make overly simplified assumptions, are computationally expensive, or simply become degenerate under non-holonomic vehicle motion. In this chapter, we introduce a new, reliable solution able to handle all kinds of relative displacements in the plane despite the possibly non-holonomic characteristics. We furthermore introduce a novel two-view optimization scheme which minimizes a geometrically relevant error without relying on 3D points related optimization variables. Our method leads to highly reliable and accurate frame-to-frame visual odometry with a full-size, vehicle-mounted surround-view camera system.

3.1 Related work

The most important related solvers that either exploit the geometry of generalized cameras [Stewénius and Nistér, 2005; Li et al., 2008; Kneip and Furgale, 2014] or non-holonomic motion constraints [Scaramuzza et al., 2009; Huang et al., 2019; Lee et al., 2013] have already been introduced in Section 1.2.2. The 6-point solver presented in [Stewénius and Nistér, 2005] proposes the solution to the generalized relative pose problem based on the Gröbner basis theory, and uses 6 ray-correspondences in order to come up with 64 solutions. Li et al. [Li et al., 2008] provide a linear algorithm that requires 17 correspondences in general, and 16 or 14 correspondences in certain special situations to solve the generalized relative pose problem. A solution that factorizes the generalized relative pose problem as an iterative optimization over relative rotation is presented in [Kneip and Furgale, 2014]. The first work that exploits the non-holonomic constraints of planar vehicles to parameterize the motion with only 1 feature correspondence is given by [Scaramuzza et al., 2009] and extended to *n* views in [Huang et al., 2019]. [Lee et al., 2013] furthermore applies the model to multi-perspective camera systems.

Further related work is given by [Pless, 2003], who is the first to introduce the paradigm of *using many cameras as one*, and Lee et al. [Lee et al., 2014], who look at the generalized relative pose problem with a known reference direction. This problem is highly related to ours in that it only solves for a one-dimensional degree of freedom rotation. However, as shown in their chapter, the algorithm again potentially degenerates for planar vehicle motion, most notably if the relative rotation becomes identity. Our work is also naturally related to the standard relative pose problem. A good overview of epipolar geometry is given in [Hartley and Zisserman, 2004]. The most popular solvers for the relative pose problem are given by [Hartley, 1997] and [Stewénius et al., 2006]. Another foundational work for ours is presented by Kneip et al. [Kneip et al., 2013], who are the first to formulate epipolar geometry as an eigenvalue minimization problem in the space of rotations.

This chapter is organized as follows. Section 3.2 provides a brief review of epipolar geometry and its formulation as an eigenvalue problem. Section 3.3 introduces our objective functions and solution strategy. Section 3.4 carefully analyze the behavior of the objective function and its ability to overcome rotation-translation ambiguities by exploiting omni-directional measurement distributions and geometrically meaningful residuals. Section 3.5 finally demonstrates our results on both simulated and real data. Our simulations compare robustness, accuracy, and computational efficiency of multiple algorithms, and demonstrate that our proposed method is able to outperform in all aspects. The potential of our method is finally confirmed on multiple real-world datasets, where we demonstrate highly reliable and accurate visual odometry performance.

3.2 Foundations

This section reviews the basic idea of direct optimization of frame-to-frame rotation. We start by introducing the geometry of generalized cameras. We furthermore summarize the prior centralized and generalized methods, and conclude with a brief motivation of our new solver.

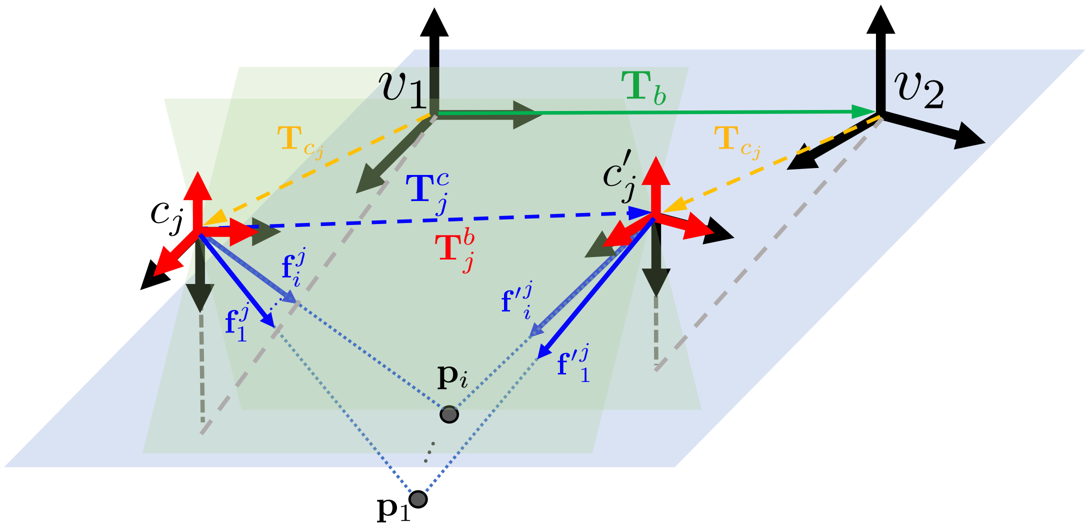


Figure 3.1: Geometry of the generalized camera system.

3.2.1 Notations and prior assumptions

We assume that we have an intrinsically and extrinsically calibrated multi-camera system. Without loss of generality, each 2D image point can therefore easily be expressed as a normalized 3D bearing vector. Considering two consecutive frames, let $\mathbf{f}^j_i$ and $\mathbf{f}'^j_i$ be the unit bearing vectors pointing at the same 3D world point $\mathbf{p}_i$ from the jth camera at the first and second frame, respectively. Let $\mathbf{R}_{c_j}$ furthermore be the rotation from the camera to the common body frame b, and $\mathbf{t}_{c_j}$ the position of camera j inside the body frame. Let $\mathbf{t}_b$ and $\mathbf{R}_b$ furthermore denote the relative pose of the body frame between two subsequent view-points, such that $\mathbf{p}_i = \mathbf{R}_b \mathbf{p}'_i + \mathbf{t}_b$ transforms points from the second frame b' back to the first frame b. To conclude, let $\mathbf{t}^c_j$ and $\mathbf{R}^c_j$ be the equivalent transformation parameters seen from camera j, i.e. transforming points from camera frame c'_j back to camera frame c_j.

3.2.2 Brief review of epipolar geometry

The epipolar incidence relationship is given by $\mathbf{f}^{jT}_i \lfloor \mathbf{t}^c_j \rfloor_\times \mathbf{R}^c_j \mathbf{f}'^j_i = 0$, and most algebraic solvers therefore minimize the sum of squared errors

$$\underset{\mathbf{t}^c_j, \mathbf{R}^c_j}{\operatorname{argmin}} \sum_i (\mathbf{f}^{jT}_i \lfloor \mathbf{t}^c_j \rfloor_\times \mathbf{R}^c_j \mathbf{f}'^j_i)^2 \tag{3.1}$$

As illustrated in [Kneip et al., 2013], we can apply the scalar triple product rule to the algebraic incidence relationship, and—by defining the epipolar plane normal vector as

$$\mathbf{n}^j_i = \mathbf{f}^j_i \times \mathbf{R}^c_j \mathbf{f}'^j_i \tag{3.2}$$

—we easily arrive at the following modified objective for the algebraic energy minimization

$$\underset{\mathbf{t}_j^c, \mathbf{R}_j^c}{\operatorname{argmin}} \; \mathbf{t}_j^{cT} \left(\sum_i \mathbf{n}_i^j \mathbf{n}_i^{jT} \right) \mathbf{t}_j^c. \tag{3.3}$$

This objective is simple to solve by an eigenvalue minimization of the matrix $\sum_i \mathbf{n}_i^j \mathbf{n}_i^{jT}$, which only depends on $\mathbf{R}_j^c$. Furthermore, as illustrated in [Kneip and Furgale, 2014], the matrix can be augmented to a 4×4 matrix to solve for the generalized case. However, as further explained in [Kneip and Furgale, 2014], this objective is not well suited for multi-camera arrays, as the latter case leads to a zero energy for identity rotation in the eigenvalue minimization objective. As further illustrated in [Li et al., 2008] and [Lee et al., 2014], critical motions with non-overlapping multi-camera systems even affect linear formulations using the generalized essential matrix. The following section introduces a solution able to handle all kinds of planar motion.

3.3 Theory

We start by seeing a new uni-variate objective function, which enables the parallel evaluation of the epipolar geometry for each individual camera as a multi-eigenvalue problem. We then proceed to see both algebraic and geometric variants of the energy minimized in our approach, which is optimized over the space of rotations only, and introduce an iteratively reweighted optimization of the planar motion that minimizes the geometrically relevant object space error. The section concludes with the derivation of the relative translation.

3.3.1 Formulation as a multi-eigenvalue problem

We now proceed to the core of our contribution, which is a novel algorithm for estimating general planar motion for non-overlapping multi-camera systems. Let $\mathbf{n}_i^j$ still be an epipolar plane normal vector, given by (3.2). As illustrated in Figure 3.1, in the case of a calibrated multi-camera system, we can use the known extrinsic rotation $\mathbf{R}_{c_j}$ to rotate all the observed unit bearing vectors of each camera into a frame that is still centered at camera c_j but has similar orientation than the local body frame. We can thus obtain an alternative multi-camera system in which all cameras simply have the same orientation than the local body frame. It can be easily observed that all the new normal vectors expressed in the body frame and given by

$$\mathbf{n}_i^{b_j} = (\mathbf{R}_{c_j} \mathbf{f}_i^j) \times \mathbf{R}_b (\mathbf{R}_{c_j} \mathbf{f}'^j_i) \tag{3.4}$$

still span a plane that is orthogonal to the translation $\mathbf{t}_j$. Similar to [Kneip et al., 2013; Kneip and Furgale, 2014], our target remains a solution to the relative displacement that depends only on $\mathbf{R}_b$.

In order to enforce all normal vectors of each camera to obey the coplanarity condition, the basic approach consists of stacking the normal vectors from corresponding cameras into the matrix $\mathbf{N}_j = [\mathbf{n}_1^j \ \ ... \ \ \mathbf{n}_i^j]^T$ such that $\mathbf{N}_j\mathbf{t}_j^c = 0$. Thus, the relative rotation $\mathbf{R}_b$ can be derived by jointly minimizing the smallest eigenvalue of the matrices $\mathbf{M}_j = \mathbf{N}_j\mathbf{N}_j^T$ from each camera. If $\lambda_{\mathbf{M}_j,min}$ denotes the smallest eigenvalue of $\mathbf{M}_j$, our final objective becomes

$$\mathbf{R}_b = \text{argmin}_{\mathbf{R}_b} \sum_j (\lambda_{\mathbf{M}_j,min})^2, \text{ where} \tag{3.5}$$

$$\mathbf{M}_j = \sum_{i=1}^{n_j}((\mathbf{R}_{c_j}\mathbf{f}_i^j) \times \mathbf{R}_b(\mathbf{R}_{c_j}\mathbf{f'}_i^j))((\mathbf{R}_{c_j}\mathbf{f}_i^j) \times \mathbf{R}_b(\mathbf{R}_{c_j}\mathbf{f'}_i^j))^T. \tag{3.6}$$

It is important to realize that this objective is different from (3.3) in [Kneip et al., 2013]. Unless proceeding to a generalization as presented in [Kneip and Furgale, 2014], it is not possible to add all normal vectors to one co-planarity condition as each camera has a potentially different translation vector, and therefore defines a different plane for its epipolar plane normal vectors. However, the rotation is the same for each camera, and—owing to the fact that the eigenvalue formulation only depends on the relative rotation—we may still jointly minimize all objectives. In the following, we concentrate on the case of planar motion, for which the relative rotation has only a single degree of freedom. Note however that the formulation makes no assumptions about the translation, and may therefore be equally applied to any relative displacements for which at least 2 of the rotational degrees of freedom are known (e.g. zero, or measured by an alternative sensor).

We choose the Cayley [Cayley, 1846] parameters $\mathbf{v} = [0 \ 0 \ z]^T$ to represent the rotation $\mathbf{R}_b$, the latter being given as

$$\mathbf{R}_b = 2(\mathbf{v}\mathbf{v}^T - \lfloor\mathbf{v}\rfloor_\times) + (1 - \mathbf{v}^T\mathbf{v})\mathbf{I}. \tag{3.7}$$

Note that we omit the scale factor as it equally affects all terms in all energies. The result is a very efficient non-linear optimization over a single parameter only. Note that the direction of each camera's relative translation can be recovered by looking at the eigenvector that corresponds to the smallest eigenvalue. As shown in Section 3.3.3, they can be further used to compute the scaled relative translation between the two viewpoints once the relative rotation $\mathbf{R}_b$ has been found.

Similar to [Kneip et al., 2013], the non-linear problem can be efficiently solved by implementing a Levenberg-Marquardt scheme. In our constraint (3.5), the only unknown parameter with respect to the sum of squares of the smallest eigenvalues is the Cayley parameter z of rotation $\mathbf{R}_b$. Owing to the fact that the angular velocities of the vehicle's motion are similar from frame to frame, a starting point for the optimization is easily given by propagating the relative rotation between the previous pair of viewpoints. An exhaustive search can be used to initialize the rotation if no prior is available.

3.3.2 Object-space error based refinement

For a perspective camera, a purely translational displacement that is parallel to the image plane can cause a very similar disparity than a pure rotation around an orthogonal axis in the image plane, and vice-versa. The hereby described *rotation-translation ambiguity* is furthermore amplified by sideways looking, fronto-parallel cameras, especially if they have a very limited field of view (FoV). The separation into 4 eigenvalue problems that are solved in parallel naturally raises the question of how this affects the algorithm's ability to deal with such ambiguities. Though the algebraic solution is not geometrically or statistically meaningful, it generally leads to satisfying results at a low computational cost. However, the rotation-translation ambiguity can easily lead to local minima in the algebraic objective error in the above described case. In an aim to solve this problem, we introduce an object-space error based objective as an iterative refinement step. It is to be understood as an efficient replacement of 2-view bundle adjustment using the more traditional reprojection error. We define the object-space error as the distance between the rays defined by $\mathbf{f}_i^j$ and $\mathbf{f}'^j_i$. Starting from the definition of the distance between two skew lines [Gellert et al., 1989], we derive the geometric object-space error for a single correspondence to be

$$d = \frac{((\mathbf{R}_{c_j}\mathbf{f}_i^j) \times \mathbf{R}_b(\mathbf{R}_{c_j}\mathbf{f}'^j_i)) \cdot \vec{\mathbf{t}}_j^c}{\|(\mathbf{R}_{c_j}\mathbf{f}_i^j) \times \mathbf{R}_b(\mathbf{R}_{c_j}\mathbf{f}'^j_i)\|}. \tag{3.8}$$

$\vec{\mathbf{t}}_j^c$ represents the direction of the relative translation. The optimization problem is finally given as

$$\{\mathbf{R}_b, \vec{\mathbf{t}}_j^c\} = \operatorname{argmin}_{\mathbf{R}_b, \vec{\mathbf{t}}_j^c} \sum_{i=0}^{n_j} \left(\frac{((\mathbf{R}_{c_j}\mathbf{f}_i^j) \times \mathbf{R}_b(\mathbf{R}_{c_j}\mathbf{f}'^j_i)) \cdot \vec{\mathbf{t}}_j^c}{\|(\mathbf{R}_{c_j}\mathbf{f}_i^j) \times \mathbf{R}_b(\mathbf{R}_{c_j}\mathbf{f}'^j_i)\|}\right)^2. \tag{3.9}$$

It is easy to see that the same objective can again be minimized by solving the iteratively reweighted eigenvalue-minimization problem

$$\mathbf{R}_b = \operatorname{argmin}_{\mathbf{R}_b} \sum_j (\lambda_{\tilde{\mathbf{M}}_j, min})^2, \text{ where} \tag{3.10}$$

$$\tilde{\mathbf{M}}_j = \sum_{i=1}^{n_j} \frac{(\mathbf{n}_i^{b_j})(\mathbf{n}_i^{b_j})^T}{\|\mathbf{n}_i^{b_j}\|_2^2}. \tag{3.11}$$

The proposed object-space error minimization strategy still depends only on the relative rotation $\mathbf{R}_b$, meaning a one-dimensional optimization space in the case of planar motion. We confirmed through a series of simulation experiments that the minimization of the object-space error is much more stable for different FoVs than the algebraic objective. It can effectively avoid wrong minima caused by rotation-translation ambiguity. The detailed analysis of the rotation-translation ambiguity for both forward and side-ways moving cameras can be found in Section 3.4. The computational complexity of the presented object-space error minimization objective is significantly lower than the one of standard two-view bundle adjustment. The latter not only optimizes over both rotation and translation parameters, but—if using the classical reprojection error—also over the 3D coordinates of each landmark.

3.3.3 Recovery of relative translation

In order to recover the translation in absolute scale, we start by formulating the hand-eye calibration constraint for camera c_j inside the multi-camera system [Hartley and Zisserman, 2004]:

$$\begin{cases} \mathbf{t}_b = \mathbf{t}_{c_j} + \mathbf{R}_{c_j}\mathbf{t}_j^c - \mathbf{R}_{c_j}\mathbf{R}_j^c\mathbf{R}_{c_j}^T\mathbf{t}_{c_j} \\ \mathbf{R}_b = \mathbf{R}_{c_j} \cdot \mathbf{R}_j^c \cdot \mathbf{R}_{c_j}^T \end{cases} \tag{3.12}$$

As mentioned in Section 3.3.1, we can compensate for each camera's extrinsic rotation $\mathbf{R}_{c_j}$, and thus obtain the simpler constraint

$$\mathbf{t}_b = \mathbf{t}_{c_j} + \mathbf{t}_j^c - \mathbf{R}_b\mathbf{t}_{c_j}. \tag{3.13}$$

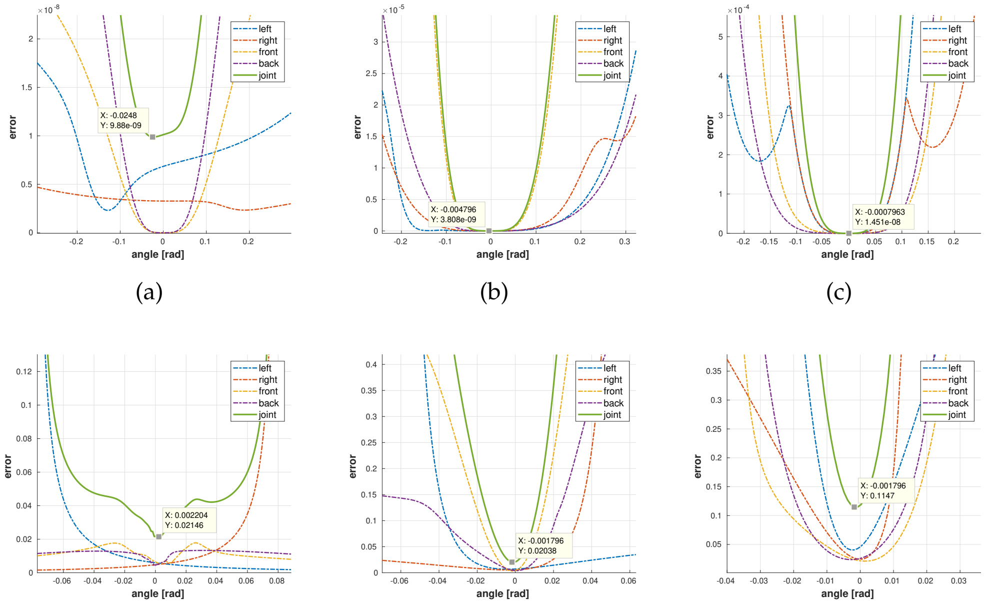
For each relative translation $\mathbf{t}_j^c = \lambda_j \cdot \vec{\mathbf{t}}_j^c$, directions $\vec{\mathbf{t}}_j^c$ can be easily computed by composing $\mathbf{M}_j$ and deriving the eigenvector corresponding to the optimized $\mathbf{R}_b$. All pair-wise constraints in the form of (3.13) can now be grouped into a linear problem $\mathbf{Ax} = \mathbf{b}$, where

$$\mathbf{A} = \begin{bmatrix} \vec{\mathbf{t}}_1^c & & -\mathbf{I} \\ & \dots & \dots \\ & \vec{\mathbf{t}}_4^c & -\mathbf{I} \end{bmatrix}, \ \mathbf{b} = \begin{bmatrix} (\mathbf{R}_b - \mathbf{I})\mathbf{t}_{c_1} \\ \dots \\ (\mathbf{R}_b - \mathbf{I})\mathbf{t}_{c_4} \end{bmatrix}, \tag{3.14}$$

and $\mathbf{x} = \begin{bmatrix}\lambda_1 \ \dots \lambda_4 \ \mathbf{t}_b^T\end{bmatrix}^T$. $\mathbf{A}$ and $\mathbf{b}$ can be computed from the known extrinsics and the relative rotation $\mathbf{R}_b$, whereas $\mathbf{x}$ contains all unknowns. The non-homogeneous linear problem $\mathbf{Ax} = \mathbf{b}$ can be solved by a standard technique such as singular value decomposition (SVD). Note that the system shows an obvious characteristic of non-overlapping multi-camera arrays, namely that metric scale remains unobservable if $\mathbf{R}_b = \mathbf{I}$. The rotation however remains computable.

3.4 Analysis of rotation-translation ambiguity

In this section, we analyse the energy plots of both the algebraic and the geometric object-space error in different scenarios. We vary the cameras' field-of-view and show results for 10°, 60°, and 135° to evaluate the behavior of each objective function's performance. We fix the relative rotation between the two viewpoints to 0 degrees, which is sufficiently general to illustrate the behavior. We furthermore plot the energy for each individual camera as well as the jointly estimated energy. Our analysis consists of single random cases for which the energies are depicted in Figure 3.2(a). They are however sufficiently representative to explain the general behavior of the problem.



(a) (b) (c)

(d) (e) (f)

Figure 3.2: Behavior of the object functions under different field-of-views. Top row: energy plot obtained by algebraic error. Bottom row: energy plot obtained by object-space error. The field-of-view increases from left to right by 10°, 60°, and 135° respectively.

In the small field-of-view case, it can be easily observed that sideways cameras are much more likely to suffer from rotation-translation ambiguities, and as a result may easily converge to wrong minima. Although the front and back cameras help to reshape the overall energy plot, the ground truth rotation may still present larger energy than the global minimum in the joint energy, thus leading to a solution with a substantial error. By contrast, the global minimum in the object-space energy (cf. Figure 3.2(d)) is much closer to the real solution. It shows that the geometric error—though more affected by local minima—takes advantage over the algebraic error in small field-of-view cases[1]. In a normal field-of-view scenario, Figure 3.2(b) indicates that the deviation of the minimum for sideways pointing cameras is diminishing with respect to the small field-of-view case, and the performance has an overall improvement. Minimizing the object-space error continues to show stable convergence to the correct solution (cf. Figure 3.2(e)). Finally, as can be observed from Figure 3.2(c), the influence of local minima on the plots of the sideways facing cameras is gradually reduced as the cameras' field-of-view is increasing. Note that alternative algebraic solvers are similarly affected, which also explains the reason why the generalized eigenvalue solver presented in [Kneip and Furgale, 2014] is only suited for omni-directional measurements in each camera.

[1] Note that what matters is not the absolute field of view of the camera, but the effective field of view given by the observations.

The conclusion of our analysis is that the minimization of the object-space error is much more stable for different FoVs than the algebraic objective. However, the object-space error is more affected by local minima, and therefore depends on a good starting point. Our choice therefore is to execute algebraic and object-space error minimization in sequence, and—in practice—make sure that the cameras satisfy a minimum requirement on the field-of-view. The overall accuracy and computational efficiency of our method, as well as a comparison against standard classical two-view reprojection error minimization for different fields-of-view, are further investigated in Section 3.5.4.

3.5 Experimental evaluation

We test our algorithm on both synthetic and real data. Our solver depends on the planar motion and is designed for non-overlapping multi-camera systems. Our experiments therefore focus on a comparison against previous relative pose solvers for generalized cameras, which are a 2-point RANSAC algorithm relying on a non-holonomic motion assumption, the linearized 17-point algorithm, and a generalized eigenvalue minimization algorithm. In order to demonstrate the benefit of using multiple cameras pointing into different directions, we also include a comparison against centralized, single-camera algorithms such as the traditional 8-point algorithm and 1-point RANSAC, the latter again relying on a non-holonomic motion assumption. We execute different comparative simulation experiments to evaluate accuracy and noise resilience, the performance of the proposed object-space error based non-linear refinement, and the performance when embedded into a RANSAC scheme [Fischler and Bolles, 1981]. We conclude with continuous frame-to-frame motion estimation demonstrations on both small-scale indoor and large-scale outdoor datasets captured by a 4-camera system mounted on either a turtlebot or a full-size car (cf. live demo on Youtube[2]). Ground truth for the indoor sequence is delivered by a highly accurate external motion capture system.

3.5.1 Outline of the simulation experiments

The surround-view camera system we investigate in simulation highly resembles the multi-camera system on real experiments, and has four cameras pointing into all directions (cf. Figure 3.3). The cameras all lie in the same horizontal plane and have a distance between 0.6 and 1m away from the body origin. In each experiment iteration, we fix the frame of the first viewpoint to coincide with the world frame. We then add 6 further views by adopting a linearly changing rotational velocity, therefore generating realistic, non-circular motion trajectories. We finally calculate the relative displacement between the first and the last viewpoint. The final relative rotation angle lies between 3 and 6 degrees, and the displacement between frames is set to 0.6m. We generate random correspondences for each camera by defining random 3D land-

[2] https://youtu.be/mtqFAzmh9E4

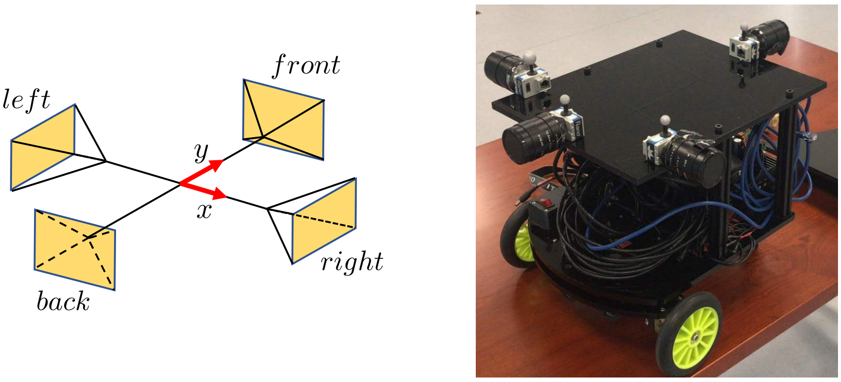
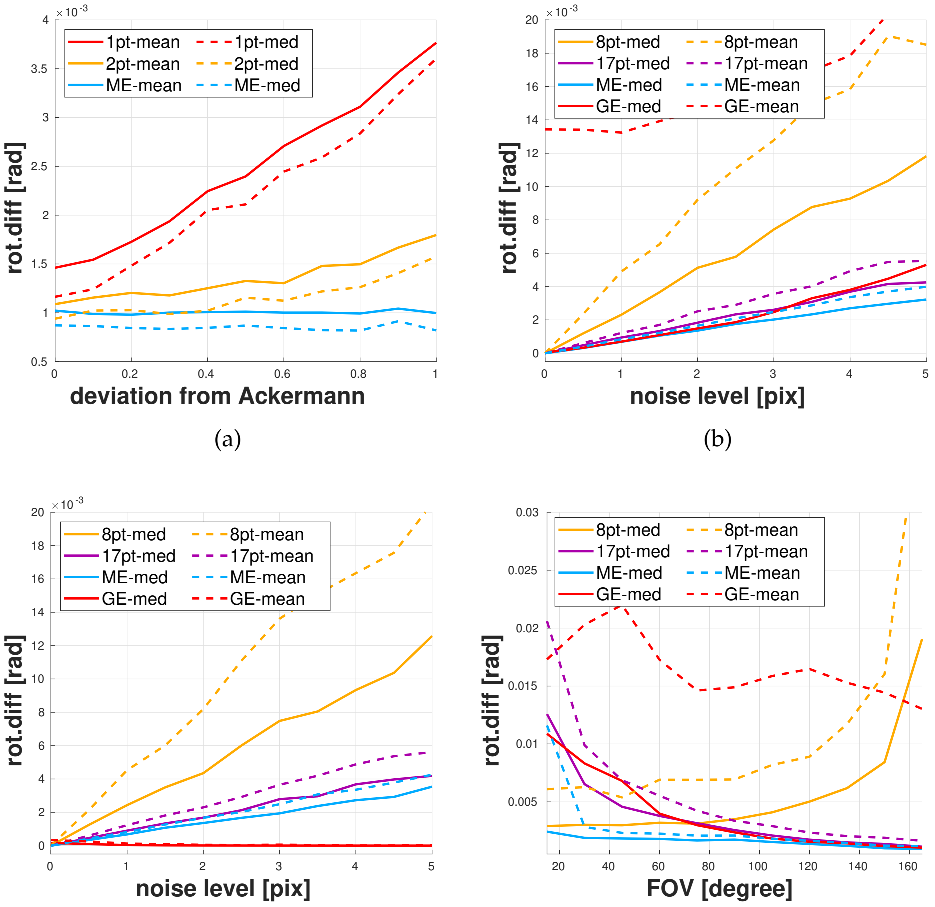


Figure 3.3: Non-overlapping, surround-view four-camera system as analyzed in this chapter.

marks located within the field of view of each camera in the first viewpoint, assign random depths between 1 and 8m, and finally reproject the obtained 3D landmarks into each camera of the second viewpoint. Noise is added to the measurements by extracting the orthogonal plane of each bearing vector, and adding noise based on a virtual spherical camera with focal length 800 pixels. Outliers are added by replacing bearing vectors such that they point towards new, randomly generated landmarks. We analyze the performance of our method under different conditions, which are a dynamic rotational velocity, purely translational motion, a varying field-of-view, and a changing outlier fraction. We execute 1000 random constellations for each experiment, and—due to scale observability issues—report primarily on the accuracy of the relative rotation estimation.

3.5.2 Comparison against minimal solvers

We compare our method (**ME**) against state-of-the-art minimal solvers for planar motion, which is the 2-point RANSAC algorithm by [Lee et al., 2013] (**2-pt**) and the 1-point RANSAC algorithm proposed by [Scaramuzza et al., 2009] (**1-pt**). These algorithms adopt the Ackermann motion model, and thus make the assumption that the motion is non-holonomic with a fixed centre of rotation (ICR) for the entire relative displacement. As a result, they have a reduced number of required correspondences. We generate 20 points in each camera for all algorithms, and use the minimal number of points for each method to solve the problem hypotheses (1 point for **1-pt**, 2 points for **2-pt** and 3 points per camera in our algorithm). No outliers are added to the data. We repeat the experiment for changing deviations from pure Ackermann motion defined by the linearly changing per-frame rotation change $\omega = 0.04k \cdot i + \omega_0$, where $i = 1, \ldots, 6$, $\omega_0 = 0.2^\circ$, and k is varied from 0 to 10. The ICR is extrapolated by assuming the constant forward velocity $v = 0.1$m per second. The results are indicated in Figure 3.4(a). As expected, our model outperforms as the deviation from non-holonomic motion is increasing.



(a)

(b)

(c)

(d)

Figure 3.4: Comparison between our proposed method **ME** and the **1pt**, **2pt**, **8pt**, **17pt**, **GE** method for different perturbation factors. Each value is averaged over 1000 random experiments. Details are provided in the text.

3.5.3 Comparison against non-minimal solvers

We compare our method against alternative non-minimal, generalized solvers (**17-pt** [Li et al., 2008] and **GE** [Kneip and Furgale, 2014]) as well as a central method (**8-pt** [Hartley, 1997]) commonly applied in vehicle motion estimation with a forward-facing camera. We use 5 points in each camera for all generalized algorithms and 8 points in the forward camera for **8-pt**. We run only the solvers and do not add nonlinear refinement. We conduct three types of experiments:

- *Significant rotation*: The field-of-view of each camera is fixed to 120°, and the noise level is varied between 0 and 5 pixels. The results are indicated in Figure 3.4(b). As can be observed, all generalized solvers out-perform the centralized method, and **ME** performs better than **17-pt** in terms of both the mean and median error. As stated in [Kneip and Furgale, 2014], **GE** has been designed for omnidirectional cameras and occasionally converges into wrong local minima, thus leading to an increased mean error with repect to **ME**.

- *Pure translations*: We repeat the same experiment but simply force the motion to be purely translational. The result is illustrated in Figure 3.4(c). As expected, **17-pt** and **ME** maintain a higher level of accuracy than **8-pt**. As furthermore explained in [Kneip and Furgale, 2014], **GE** is affected by a constant zero energy for identity rotation. While this leads to perfect performance in this experiment, it is to be interpreted as a weakness. **GE** is unable to distinguish small from zero rotation angles.

- *Variation of the field of view*: We vary the field-of-view from 15° to 165°. As shown in Figure 3.4(d), the centralized method **8-pt** applied in the forward facing camera performs better for very small fields-of-view. As explained in Section 3.3, side-ways looking cameras are affected by the rotation-translation ambiguity. The effect worsens for a decreasing field-of-view, and potentially affects all generalized camera solvers. However, as soon as the field-of-view is sufficiently large (75° for our settings), generalized solvers start to outperform. **ME** furthermore clearly outperforms other methods and beats **8-pt** for any FoV larger than 30°.

3.5.4 Behavior of object-space error based refinement

Figure 3.5 shows the comparison between our proposed object-space error minimizer and standard two-view bundle adjustment. Both depend on a sufficiently good initialization. However, as stated in Section 3.3.2, standard two-view bundle adjustment reduces reprojection errors over rotation, translation, and structure parameters, while the proposed joint eigenvalue minimization based object-space error reduction involves only the rotational degrees of freedom (which—in the case of planar motion—is only a single degree of freedom). As indicated in Figure 3.5, object-space error minimization shows comparable performance than 2-view bundle adjustment for varying fields-of-view. However, owing to its uni-variate nature, the proposed objective is minimized 8 times faster.

3.5.5 Overall performance within RANSAC

Before moving on to real data experiments, we add a final experiment with outliers to also compare the behavior with respect to full generalized relative pose solvers if embedded into a RANSAC scheme. We add up to 30% outliers, and use the same outlier threshold and inlier verification criterium for each algorithm. We test three elements including the execution time of each algorithm, the required number of RANSAC iterations, and the percentage of all true inliers found by each method. Results are depicted in Figure 3.6. Our method's processing time is 0.23ms, **17-pt** uses 0.11ms, **GE** uses 0.3ms, and **8-pt** is the fastest at 0.05ms. As already indicated in [Li et al., 2008], the linear **17-pt** method requires too many samples and even 1000 iterations may be insufficient to perform successful inlier identification. Although

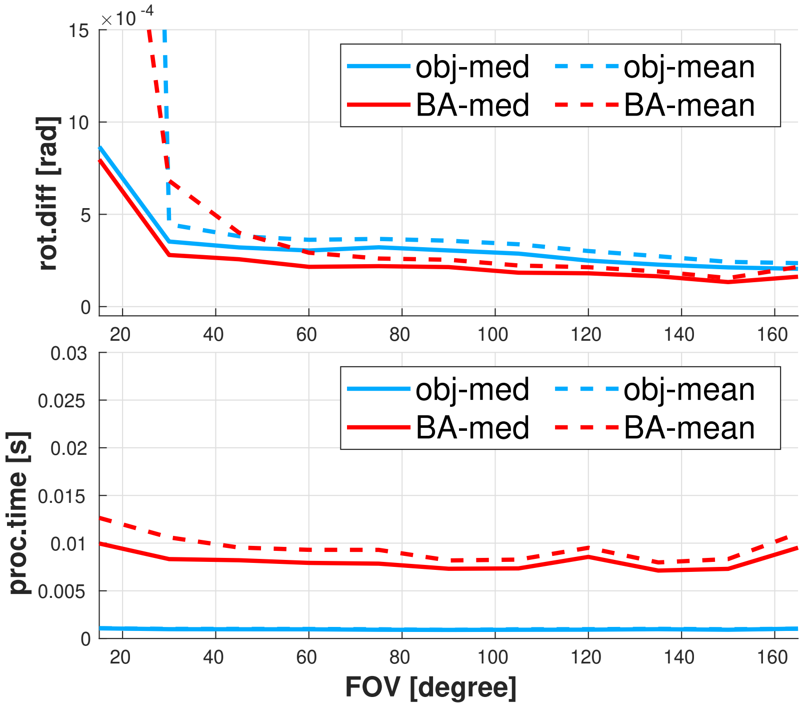


Figure 3.5: Accuracy of the different geometric optimization method and average execution time.

our method requires more samples (3 points per camera) than **GE** (8 points) and **8-pt** (8 points) and resulting in slightly more iterations, our method also shows high computational efficiency. It consumes 2 times the time of the linear solver **17-pt** and is 1.5 times faster than **GE**. To conclude, our solution finds the largest percentage of all true inliers, which demonstrates that—given a sufficiently large number of iterations—our method has a low probability to miss the global minimum of the objective function.

3.5.6 Results on a real multi-camera system

In order to demonstrate the performance of our algorithm on real images, we apply it to two sequences representing both an indoor and an outdoor example. All datasets are captured by a fully calibrated and synchronized surround-view multi-camera system mounted on either a turtlebot (cf. illustrated in Figure 3.3) or a full-size car. The indoor dataset allows us to compare our method against highly accurate ground truth captured by a motion tracking system. It provides a mix of characteristics with both straight forward motion and significant rotation parts. All methods are embedded into a RANSAC scheme and applied on a frame-to-frame basis. We use object-space error minimization for **ME** and standard 2-view bundle adjustment for all remaining algorithms as a two-view refinement procedure. We do not add a multi-frame back-end optimization module (i.e. sliding-window bundle-adjustment)

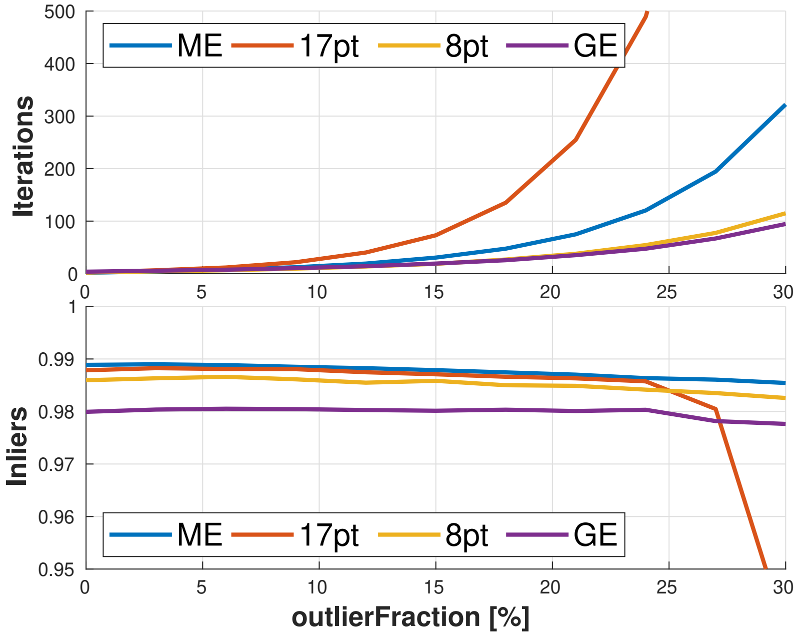


Figure 3.6: Average number of iterations and found inlier rates.

as this permits the observation of the original performance of each method. Implementations are made in C++, and use OpenCV [Bradski, 2000] and OpenGV [Kneip et al., 2014] for image processing and geometry problems, respectively. All experiments are conducted on an Intel Core i7 2.4 GHz CPU with 8GB RAM. Figure 3.7 shows our results obtained on the indoor dataset and compares them against all alternative algorithms. The following is worth noting:

- The trajectories all suffer from slow error accumulation, which means that all algorithms successfully process the entire 2000 frames without any gross errors. Our algorithm **ME** clearly outperforms both **17-pt** and **2pt**. Note that **8-pt** and **GE** are not included in the results, as they are both unable to provide competitive results.

- The observations concerning rotation-translation ambiguity are consistent with our prior analysis. We therefore implement a firewall strategy to prevent occasional convergence to wrong local minima. We check the solution obtained from only the front and back cameras, and compare it against the solution obtained from the entire system. If the two solutions have obvious differences, we down-weight the energy contribution of the sideways facing cameras. As shown in Figure 3.7, this strategy leads to the best result.

- Note furthermore that **2pt** is confined to the front and back cameras, as we observed that it performs much better than a full 4-camera alternative. Nonetheless, it suffers from the strict Ackermann-motion assumption, and leads to a similar error accumulation like **17-pt**.

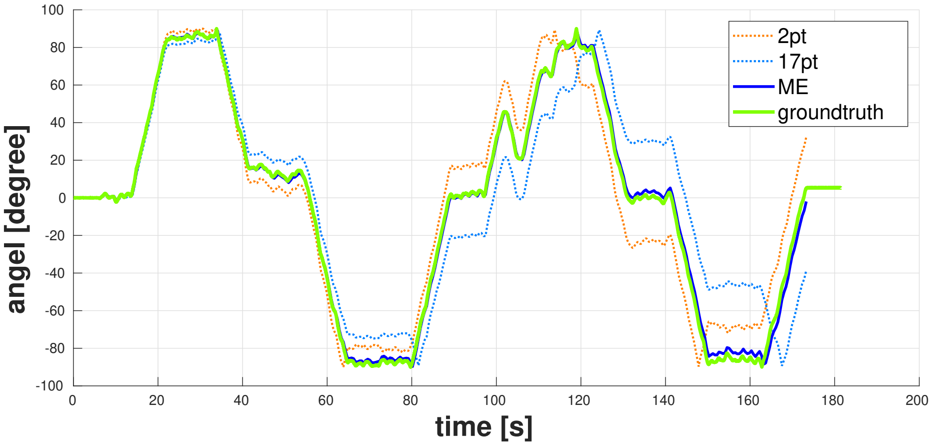


Figure 3.7: Evaluation of the dead-reckoned absolute orientation of a real multi-camera rig moving in an indoor environment. Ground-truth is provided by an Optitrack motion tracking system.

The real-time execution of the algorithm and further qualitative results on a full-size vehicle moving outdoors can be found on Youtube[3].

3.6 Discussion

Our work stands in contrast with many prior closed-form solutions presented in the literature, as it relies on an iterative optimization scheme. However, by exploiting simple linear algebra relationships and the planarity of the motion, the dimensionality of the energy minimization problem is reduced to one, and can hence be solved very effectively. Furthermore, the fact that the minimized energy depends only on the rotation parameters and the insight that these parameters are shared between all cameras permits us to minimize multiple single camera objectives in parallel rather than a single generalized objective. As demonstrated through our results, the formulation is free of singularities and amenable to highly accurate and reliable, continuous motion estimation for surround-view camera systems. It is our belief that this contribution must be of interest to the intelligent vehicles community, and direct our future work towards an extension of the approach over multiple temporal frames, and further integrated into our previously proposed complete real-time pipeline for visual odometry with MPC systems.

[3] https://youtu.be/mtqFAzmh9E4

Chapter 4

B-splines for visual SLAM on non-holonomic ground vehicles

In previous chapters, we have seen how to efficiently and reliably estimate the trajectory of generalized camera systems undergo an arbitrary planar motion. The proposed methods provide an attractive solution to localization and mapping on smart ground vehicles. However, existing methods including the methods that we proposed in the previous chapters are sensitive to the quality of visual inputs, they might come across challenges under degraded visual conditions, such as poor texture environments, insufficient field-of-view of cameras, and challenging illumination conditions. Hence, the accuracy and especially robustness of vision-only solutions still remain rivalled by more expensive, lidar-based multi-sensor alternatives. In this chapter, we show that a significant increase in robustness can be achieved by taking non-holonomic kinematic constraints on the vehicle motion into account. Rather than using approximate planar motion models or simple, pair-wise regularization terms, we demonstrate the use of B-splines for an exact imposition of smooth, non-holonomic trajectories inside the 6-DoF bundle adjustment. We introduce different formulations and compare them in terms of computational efficiency and accuracy against traditional solutions. We evaluate our method on both simulated and real data, and thus demonstrate a significant improvement in both robustness and accuracy in degraded visual conditions.

4.1 Related work

Most modern solutions to localization and mapping for self-driving ground vehicles rely on powerful 3D lidars and high-definition 3D maps of the environment [Levinson et al., 2007; Wan et al., 2018]. A complete review of lidar-based solutions would however go beyond the scope of this chapter. We limit our discussion to purely vision-based solutions. Existing methods using only a single camera may easily lead to drift accumulation and robustness issues, the frameworks tested specifically on ground vehicle applications therefore often employ a stereo [Nistér et al., 2006; Konolige et al., 2007; Howard, 2008; Kitt et al., 2010] or a surround-view multi-camera array [Furgale et al., 2013; Heng et al., 2018]. The latter is particularly interesting as it is already commonly installed in modern cars.

We look into the improvement of a purely vision-based solution by taking vehicle kinematics related constraints into the optimization framework. Such techniques have been already commonly applied to vision-based multi-sensor solutions using additional odometers to measure the rotational velocity of each wheel. The state-of-the-art EKF filter [Wu et al., 2017], particle filter [Yap et al., 2011], and optimization-based [Quan et al., 2018; Kang et al., 2019] solutions rely on a drift-less planar motion model that derived from a dual-drive or Ackermann steering platform. They perform relatively high-frequent integration of wheel odometry in order to provide an adequate initial guess on the relative displacement between subsequent views. Censi et al. [Censi et al., 2013] furthermore consider simultaneous extrinsic calibration between cameras and odometers. For skid-steering mobile platforms, several works [Yi et al., 2009; Martinez et al., 2017; Lv et al., 2017] also applied a closely related vehicle motion model in filtering and optimization-based frameworks. Although the slippage exists, non-holonomic models relying on the Instantaneous Centre of Rotation (ICR) still explain the motion of skid-steering platforms relatively well [Martinez et al., 2005], which explain the reason why our methods could potentially be applied to such platforms. Further related work is given by Zhang et al. [Zhang et al., 2019], who still relies on a drift-less non-holonomic motion model, but extends the estimation to full 6-DoF motion by introducing the motion-manifold and manifold-based integration of wheel odometry signals.

When we focus on pure vision-based solutions, the non-holonomic constraints need to be enforced purely by the motion model, which is much more difficult. Scaramuzza [Scaramuzza et al., 2009; Scaramuzza, 2011] presents highly robust solutions to estimate the relative motion with a single vehicle-mounted camera by introducing the Ackermann motion model. Huang et al. [Huang et al., 2019] recently extended the method to an n-frame solver, while Lee et al. [Lee et al., 2013] successfully applied it to a multi-camera array. Long et al. [Zong et al., 2017] and Li et al. [Li et al., 2018] have included similar constraints into windowed optimization frameworks, which essentially penalize trajectory deviations from an approximate piece-wise circular arc model.

A more correct model-based imposition of non-holonomic constraints on ground vehicle motion has been proposed in robotics and control societies. The related works mostly aim at planning feasible trajectories for drift-less ground vehicles, a challenging problem if both spatial and temporal constraints need to be taken into account. An in-depth introduction to the topic is given in [Souéres and Boissonnat, 1998]. The general idea consists of using continuous-time models for smooth collision-free trajectory planning on non-holonomic curvature-bounded vehicles. For example, Lundberg [Lundberg, 2017] introduces clothoid-based smoothing in A* path planning.

From a purely geometric point of view, a drift-less, non-holonomic ground vehicle moves along smooth trajectories in space, and—more importantly—heads toward the vehicle motion direction. This motivates our use of the continuous-time trajectory model as proposed by Furgale et al. [Furgale et al., 2015]. While parametrizing a smooth vehicle trajectory, the representation and in particular its first-order differential is easily used to additionally enforce the vehicle heading to remain tangential to the trajectory.

The chapter is organized as follows. Section 4.2 provides a brief review of B-splines and kinematic motion constraints on drift-less non-holonomic platforms. Section 4.3 introduces different realizations of the objective, and Section 4.4 finally concludes with our results on both simulated and real data.

4.2 Preliminaries

This section reviews the basic idea of continuous-time parametrizations, which play an important role in motion estimation when dealing with smooth trajectories or temporally dense sampling sensors. There are various alternatives for the basis functions, such as FFTs, discrete cosine transforms, polynomial kernels, or Bézier splines. In this chapter, we will use the efficient and smooth B-spline parametrization [Piegl and Tiller, 2012] as already illustrated by Furgale et al. [Furgale et al., 2015]. We start by reciting the basic form of B-splines and their initialization, and conclude with a brief review of preliminaries on the non-holonomic motion.

4.2.1 B-splines

According to the definition, a B-spline curve is defined as a linear combination of control points and B-spline basis functions. The basic form of a pth-degree B-spline curve is given as follows:

$$\mathbf{c}(u) = \sum_{i=0}^{n} N_{i,p}(u)\mathbf{p}_i, \qquad a \leq u \leq b, \tag{4.1}$$

where u is the continuous-time parameter, $\{\mathbf{p}_i\}$ are the $n+1$ control points, and $\{N_{i,p}(u)\}$ are the $n+1$ pth-degree B-spline basis functions defined on the monotonically increasing and non-uniform knot vector $\mathbf{u} = \{\underbrace{a,\ldots,a}_{p+1}, u_{p+1},\ldots,u_n,\underbrace{b,\ldots,b}_{p+1}\}$. The procedure to compute a point on a B-spline curve for a fixed u value consists of three main steps:

- We firstly find u's knot span k in the knot vector $\mathbf{u}$. k is defined such that $u \in [u_k, u_{k+1})$, and an exception is given when $u = u_{n+1}$ in which case $k = n$;
- Then, we evaluate the basis functions $\{N_{i,p}(u)\}$, which are non-zero only if $i = \{k-p, k-p+1, \ldots, k\}$;
- Finally, the values of the non-zero basis functions are multiplied with their respective control points, and sum up the terms.

It should be noticed that the form of a B-spline and the basis functions are generally fixed, the shape of the curve is influenced by the control points only.

In this chapter, we focus on exploring B-spline as a representation for the kind of smooth trajectory that we have on drift-less, non-holonomic ground vehicles. We therefore need to discuss B-splines initialization based on discrete sets of camera poses. The splines are initialized from a set of samples along the curve, and there is a spline curve approximation algorithm proposed by Piegl and Tiller [Piegl and Tiller, 2012] for the initialization. We assign a parameter value $\bar{u}_k$ to each sample, and define an appropriate knot vector $\mathbf{u}$. The control points can then be solved from a simple linear system. The detailed steps are as follows:

- **Data registration:** Suppose we have a set of discrete vehicle poses obtained from corresponding images by front-end solver, and the timestamp of an individual image is used as a parameter value $\bar{u}_k$.

- **Knot spacing:** In order to define the knot vector $\mathbf{u}$, an automatic knot spacing algorithm (9.69) of [Piegl and Tiller, 2012] is applied in our method. This algorithm guarantees that every knot span contains at least one $\bar{u}_k$. From a temporal perspective, this algorithm will give a nearly uniformly distributed knot vector.

- **Least squares curve approximation:** Assume that points $\{\mathbf{d}_0, \ldots, \mathbf{d}_m\}$ are given, a pth-degree nonrational curve could be approximated which satisfying that $\mathbf{d}_0 = \mathbf{c}(a)$ and $\mathbf{d}_m = \mathbf{c}(b)$, and the remaining $\mathbf{d}_k$ are then used to optimize the control points in the sense of the least-squares objective

$$\underset{\{\mathbf{p}_i\}}{\arg\min} \sum_{k=1}^{m-1} \|\mathbf{d}_k - \mathbf{c}(\bar{u}_k)\|^2. \tag{4.2}$$

- **B-spline initialization for rotations:** We use the method of Kang and Park [Kang and Park, 1999], which uses B-splines to approximate unit quaternions. The basic idea behind this method is to use 4-dimensional B-splines, and then project the resulting curve onto the unit three-sphere.

4.2.2 Non-holonomic motion

As mentioned in previous Section 1.2.3, the ground vehicle with a non-steering two-wheel axis commonly undergo non-holonomic planar motion. Under the assumption that the wheels do not undergo any drift, certain displacements become impossible or only executable as a complex combination of displacements. This kinematic constraint is reflected in the Ackermann steering model and the instantaneous heading of the vehicle is parallel to its velocity. The constraint has been already exploited in purely vision-based algorithms, however only based on the approximation of a piece-wise constant steering angle:

- As proposed by Scaramuzza et al. [Scaramuzza et al., 2009], the motion of a ground vehicle could be approximated to lies on a plane, and the vehicle have a locally constant steering angle. The trajectory between subsequent views is therefore approximated by an arc of a circle, and the heading remains tangential to this arc. Hence the motion could be parameterized by the inscribed arc-angle θ as well as the radius of this circle r, and both the relative rotation and translation are expressed as functions of these parameters. The matter is illustrated in Figure 1.10.

- Inspired by [Scaramuzza et al., 2009], Peng et al. [Peng et al., 2019] introduce a constraint when relative rotations $\mathbf{R}$ and translations $\mathbf{t}$ under the approximation of a piece-wise constant steering angle or circular arc model. The constraint is formulated as follows:

$$\left((\mathbf{I}+\mathbf{R})\begin{bmatrix}0 & 1 & 0\end{bmatrix}^T\right) \times \mathbf{t} = 0, \tag{4.3}$$

  It can be added as a regularization term in a common bundle adjustment framework. We call this the *R-t* constraint.

The models mentioned above are only an approximation of the original infinitesimal constraints on the velocity and the position of the ICR. In the following sections, we will look into the use of continuous-time parametrizations to continuously enforce identity between the body's velocity direction $\frac{\mathbf{v}_b}{\|\mathbf{v}_b\|}$ and the vehicle's forward axis y_b, which is the original infinitesimal constraint. We call this the *R-v* constraint.

4.3 Optimization of non-holonomic trajectories

In order to enforce *R-v* and *R-t* constraint, we use a spline to represent the non-holonomic vehicle trajectory in continuous time. The *R-v* constraint can be formulated by the first-order differential of the spline, which represents the instantaneous velocity of the vehicle. We introduce and compare multiple formulations starting from conventional bundle adjustment.

4.3.1 Conventional Bundle Adjustment (CBA)

Conventional Bundle Adjustment (CBA) boils down to minimizing reprojection errors with respect to directly parametrized camera poses and 3D landmarks. Specifically, the non-linear objective is given by

$$\min_{\substack{\{\mathbf{t}_{b_j}\}\\ \{\mathbf{q}_{b_j}\}\\ \{\mathbf{x}_i\}}} \underbrace{\sum_{i,j} \rho\left(\|f_p\left(\mathbf{T}_{sb}\begin{bmatrix}\mathbf{R}(\mathbf{q}_{b_j}) & \mathbf{t}_{b_j}\\ 0 & 1\end{bmatrix}^{-1}\mathbf{x}_i\right) - \mathbf{m}_{ij}\|^2\right)}_{\text{conventional bundle adjustment (CBA)}}, \tag{4.4}$$

where

- $\mathbf{R}(\mathbf{q})$ is the rotation matrix constructed from quaternion representation $\mathbf{q}$.
- $\mathbf{t}_{b_j}$ and $\mathbf{q}_{b_j}$ are optimized pose parameters.
- $\{\mathbf{x}_i\}$ are landmarks in homogeneous representation.

- $\mathbf{m}_{ij}$ is the observation of landmark i in frame j.

- $\rho(\cdot)$ represents a loss function (e.g. Huber loss) to mitigate the influence of outliers.

- $f_p(\cdot)$ is the projection of $\{\mathbf{x}_i\}$ onto the image plane with calibrated cameras.

- $\mathbf{T}_{sb}$ represents the extrinsic parameters that transform observed points from the vehicle to the camera frame.

As shown in Figure 4.1(a), blue nodes in the graphical model are residual blocks of re-projection error.

4.3.2 CBA with *R-t* constraints (CBARt)

We now proceed to the aforementioned the *R-t* constraint (4.3) in Section 4.2.2, the constraint can be added as a pairwise, soft regularization constraint into the formulation of conventional bundle adjustment similar to [Zong et al., 2017] and [Li et al., 2018]. We therefore obtain

$$\min_{\substack{\{\mathbf{t}_{b_j}\}\\ \{\mathbf{q}_{b_j}\}\\ \{\mathbf{x}_i\}}} \underbrace{\sum_{i,j} \rho\left(\| f_p\left(\mathbf{T}_{sb} \begin{bmatrix} \mathbf{R}(\mathbf{q}_{b_j}) & \mathbf{t}_{b_j} \\ 0 & 1 \end{bmatrix}^{-1} \mathbf{x}_i \right) - \mathbf{m}_{ij} \|^2 \right)}_{\text{conventional bundle adjustment (CBA)}} \\ \underbrace{+ \sum_j w_r \| \left(\left(\mathbf{I} + \mathbf{R}_{b_{j-1}b_j} \right) \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right) \times \mathbf{t}_{b_{j-1}b_j} \|^2}_{\textit{R-t constraint}}, \tag{4.5}$$

where

- w_r is a scalar weight for the R-t constraints.

- $\mathbf{R}_{b_{j-1}b_j}$ and $\mathbf{t}_{b_{j-1}b_j}$ denotes the relative rotation and translation between subsequent views.

However, such objectives still optimize a discrete set of poses, and regularizes it against a piece-wise circular arc model. The graphical model of this formulation is shown in Figure 4.1(b), and the yellow nodes indicate *R-t* constraints.

4.3.3 CBA with spline regression (CBASpRv)

We now proceed to introduce a continuous-time model into bundle adjustments. Although we still use CBA as the basis formulation for optimizing a discrete set of poses, we replace the previous *R-t* constraint with a soft, regularising *R-v* constraint that using a 3D spline approximated from the optimized positions $\mathbf{t}_b$. The objective can now be rewritten as

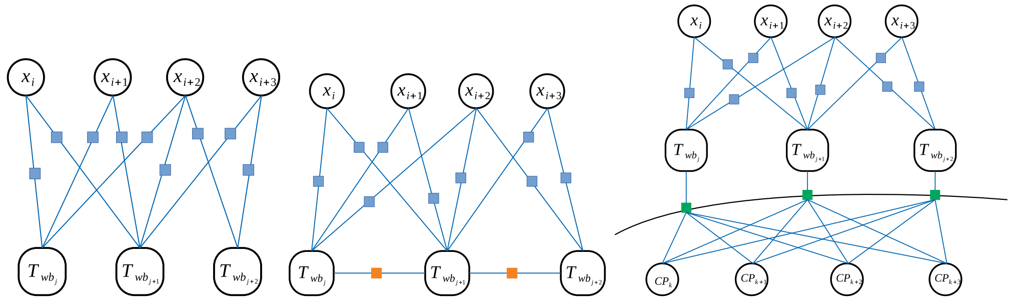
$$\min_{\substack{\{\mathbf{t}_{b_j}\}\{\mathbf{q}_{b_j}\}\\ \{\mathbf{x}_i\},\mathcal{P}}} \underbrace{\sum_{i,j} \rho\left(\left\|f_p\left(\mathbf{T}_{sb}\begin{bmatrix}\mathbf{R}(\mathbf{q}_{b_j}) & \mathbf{t}_{b_j}\\ 0 & 1\end{bmatrix}^{-1}\mathbf{x}_i\right) - \mathbf{m}_{ij}\right\|^2\right)}_{\text{conventional bundle adjustment (CBA)}} \tag{4.6}$$

$$+ \underbrace{\sum_j w_s\|\mathbf{t}_{b_j} - \mathbf{c}_1(t_j)\|^2}_{\text{smoothness constraint}} + \underbrace{\sum_j w_c \left\|\mathbf{R}(\mathbf{q}_{b_j})\begin{bmatrix}0\\1\\0\end{bmatrix} - \eta(\mathbf{c}'_1(t_j))\right\|^2}_{\text{R-v constraint}},$$

where

- $\mathcal{P}$ is the set of control points of $\mathbf{c}_1(t)$.
- $\mathbf{c}_1(t)$ denotes the alternatingly updated spline.
- t_j is the timestamp at jth frame.
- $\eta(\mathbf{a}) = \frac{\mathbf{a}}{\|\mathbf{a}\|}$.
- $\mathbf{c}'(t)$ denotes the first-order derivative of $\mathbf{c}(t)$.
- w_s, w_c are scalar weights.

Note that the spline used in objective above is computed by summing over products between control points and the fixed, first-order derivatives of the basis functions. The objective function can also be easily adjusted to the 3-Dof planar case, by changing the degree of control points from 6 to 3, including a 2-Dof translation on the plane and a 1-Dof rotation along normal axis. This is indeed the version that is used in Section 6.4.3 of Chapter 6. The green nodes are combined smoothness and R-v constraints, shown as in Figure 4.1(c), and *CP* represent control points.



(a) (b) (c)

Figure 4.1: Graphical models of the different methods: (a) CBA; (b) CBARt; (c) CBASpRv.

4.4 Experimental results

In this section, we test our algorithm on both synthetic and real data. The main purpose of our experiments is to demonstrate the ability of our methods to handle degraded visual conditions. We therefore analyse the influence of the connectivity of the graph by varying number of landmarks and number of observations, in addition to the commonly analysed influence of noise on the image measurements $\mathbf{m}_{ij}$. In order to evaluate both accuracy and computational efficiency of our methods, we take a few well-chosen, synthetic sequences, and execute different experiments focus on a comparison against all aforementioned objective functions. We conclude with tests on popular, real-data benchmark sequences and compare our methods against an established alternative from the open-source community: ORB-SLAM [Mur-Artal and Tardós, 2017]. Ground truth for the sequences is delivered by an accurate Inertial Navigation System. All our experiments are conducted on a laptop with 8GB RAM and an Intel Core i7 2.4 GHz CPU, and the C++ implementations use OpenCV [Bradski, 2000], Eigen [Guennebaud et al., 2010], and the Ceres Solver [Agarwal et al., 2010] optimization toolbox with automatic differentiation.

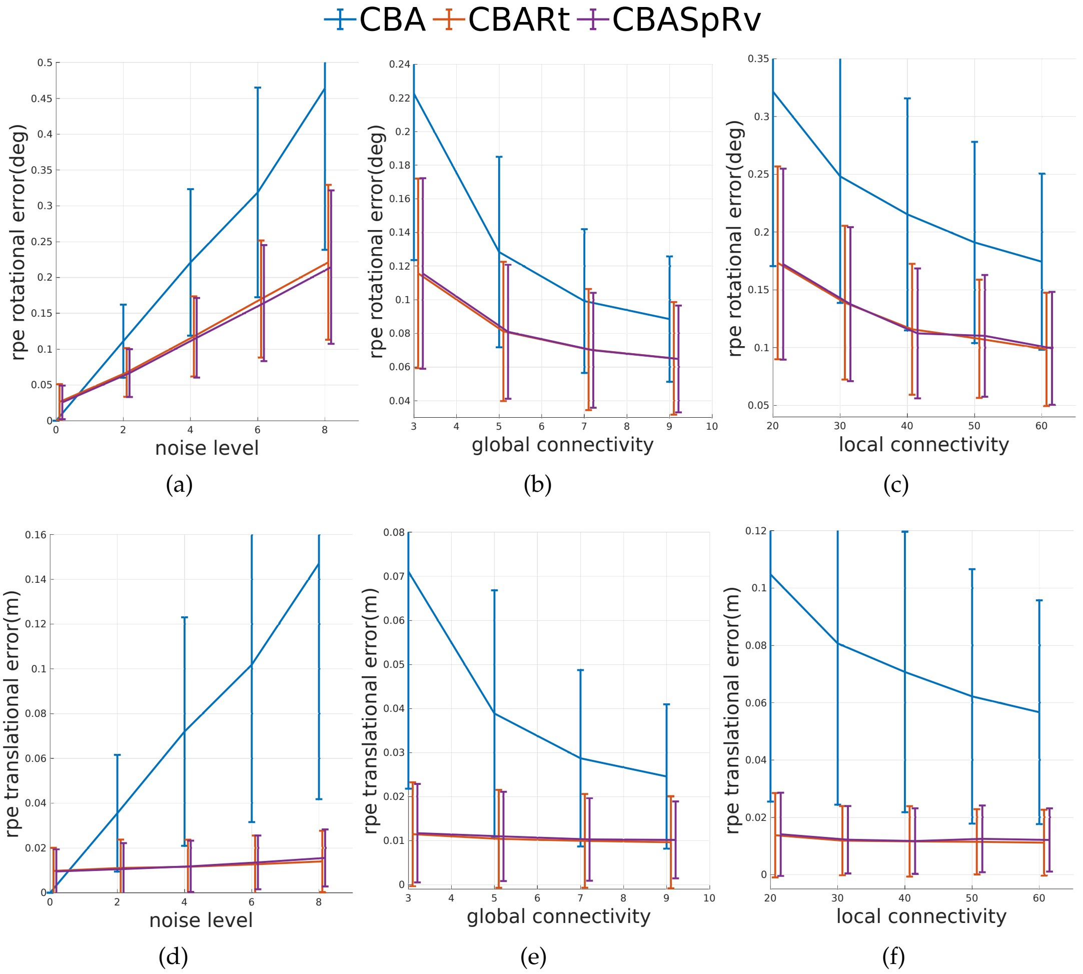
4.4.1 Results on synthetic data

We start by using realistic trajectories that directly taken from the ground truth trajectories from KITTI sequences [Geiger et al., 2012]. Both the intrinsic and extrinsic parameters $f_p(\cdot)$ and $\mathbf{T}_{sb}$ are taken from the KITTI sequences as well. However, instead of using original images, we generate synthetic correspondences by defining uniformly distributed random image points in each view. The number of points denotes the local connectivity. We define random depths for these points by sampling from a uniform distribution between 6 and 30 meters. The corresponding world points (landmarks) are finally projected into all nearby views to generate all possible correspondences in the graph. Note that the number of observations per landmark is however capped by the global connectivity setting. We also perform a boundary check to make sure that all reprojected points are visible in the virtual views. Finally, we add zero-mean normally distributed noise to the observations. For each analysis and noise or connectivity setting, we calculate the mean and standard deviation of the sliding pair-wise Relative Pose Error (RPE) with respect to ground truth, which individually analyses rotation and translation errors. The rotation error is calculated using (2.15) in [Ma et al., 2012]. For the translation error, our evaluation differs from the one in [Sturm et al., 2012] in that we ignore the scale of the relative translations which are unobservable in a monocular setting. Results are indicated in the Figure 4.2. We conduct three types of experiments:

- **Noise level**: The noise level is varied between 0 to 5 pixels. The result is indicated in Figures 4.2(a) and 4.2(d), adding kinematic constraints leads to a large reduction of errors; the proposed methods using continuous-time parametrizations outperforms CBA in most cases, especially in terms of the translational error. Both CBASpRv and CBARt present high robustness against increasing noise levels.

- **Global connectivity**: We repeat the same experiments but vary the global connectivity from 3 to 10 frames. As shown in Figures 4.2(b) and 4.2(e), the proposed methods maintain a high level of accuracy as the graph's global connectivity degrades. CBASpRv and CBARt again outperform CBA for both low and high connectivity.

- **Local connectivity**: We vary the local connectivity from 20 to 60 landmarks.



(a) (b) (c)

(d) (e) (f)

Figure 4.2: Mean and standard deviation of RPE for different methods on synthetic data using KITTI-VO-05 ground truth trajectories. The default noise level used in the experiments is set to 4 pixels, the default global connectivity is set to 3 frames, and the default local connectivity is set to 40 landmarks. Column 1 analyse rotational(first row) and translational(second row) errors for varying noise levels, column 2 for varying maximum number of observations for each landmark, and column 3 for varying number of landmarks per frame.

The result is illustrated in Figures 4.2(c) and 4.2(f). As expected, our proposed kinematic methods still perform significantly better as the number of observations per frame decreases.

4.4.2 Results on artificial trajectories

In the previous sections, our experiments illustrate that the CBARt method has provided competitive results when using realistic trajectories. However, taking sparsely sampled views of course diminishes the validity of the Ackermann motion model. A lower frame density implies that the piece-wise circular arc-based regularization of CBARt is less valid and its performance degrades. Here we add additional experiments in which the trajectory is formed by sinusoidal curves in the plane for better understanding. On average, we place less than 10 keyframes over the span of a single period, which simulates high dynamics scenario. All other experimental settings are similar to the previous section. As shown in Figure 4.3, the performance of CBARt decreases and shows similar performance with CBA. CBASpRv still clearly outperforms.

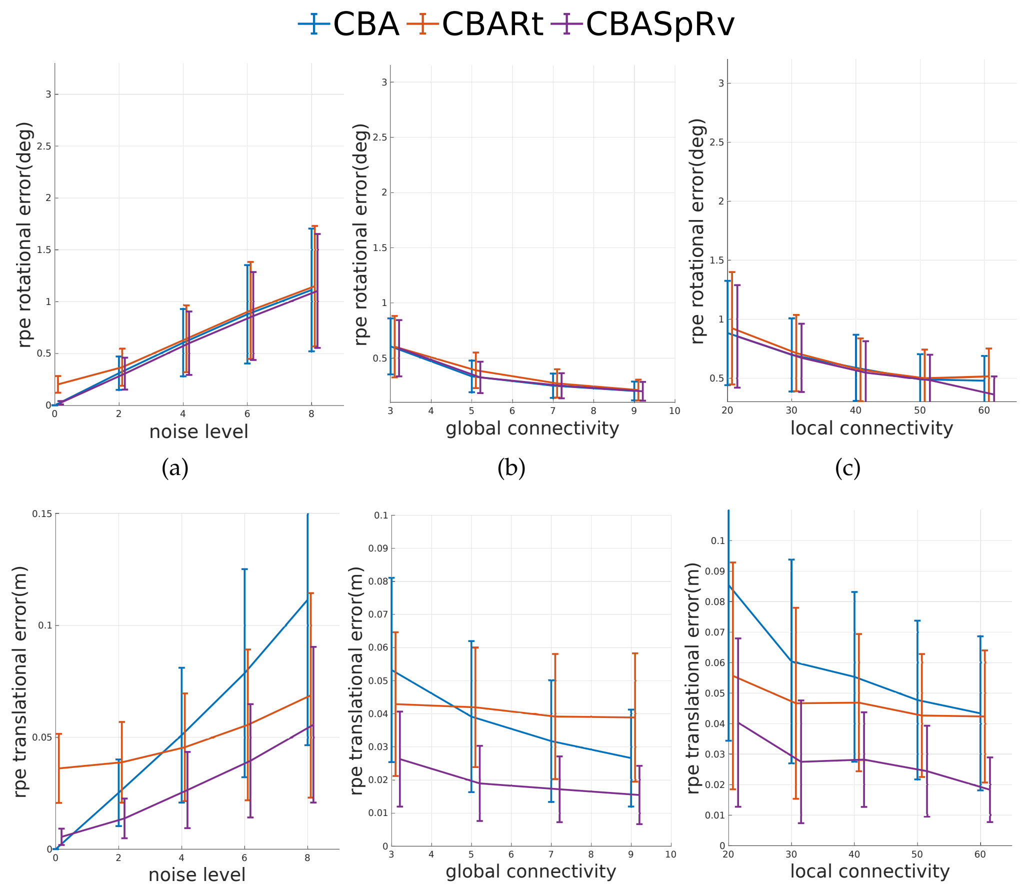
4.4.3 Experiments on real data

In order to demonstrate the performance of our methods on real images, we use the KITTI benchmark datasets [Geiger et al., 2012], which are fully calibrated and contain images captured by a forward-looking camera mounted on a passenger vehicle driving through different environments. The datasets contain signals from high-end GPS/IMU sensors, which allow us to compare our results against ground truth. Several different sequences are used in order to provide a mix of motion characteristics reaching from significant turns and height variations to simple forward motion. We extract ORB [Rublee et al., 2011] features and use the *flann* matcher from OpenCV. We furthermore use the 1-point RANSAC method by Scaramuzza et al. [Scaramuzza, 2011] to initialize the motion and identify inlier correspondences for triangulation. The results for all methods are shown in Figure 4.4. We present individual results for six different KITTI sequences, which contain a mix of motion characteristics. KITTI-VO-01 and KITTI-VO-04 are an empty highway and a short straight road segment resulting in poor or simple graphical models, respectively.

We again evaluate all results as a function of artificially added noise, in order to measure the influence of degraded visual conditions. The blue dotted line indicates the error of the initialization. As illustrated in results, CBA generally has the worst performance, followed by CBARt. CBASpRv provides the best performance, it generally improves the accuracy, especially in terms of the translational error. Note that CBARt shows relatively large errors on KITTI-VO-00, which we trace back to a single difficult, badly modelled subpart of the trajectory between frames 250 and 300. From a qualitative point of view, several example results are indicated in Figures 4.5(a), 4.5(b) and 4.5(c), which show the comparison between our proposed methods and CBA with ground-truth, and our proposed methods performs closer to ground-truth than CBA.

4.4.4 Comparison against ORB-SLAM

As a final test, we let our kinematically constrained optimization compete against an established alternative from the open-source community: ORB-SLAM [Mur-Artal and Tardós, 2017]. RPE results are again indicated in Table 4.1. It is true that our comparison against the heavily engineered ORB-SLAM framework does not show any general advantages, and the results confirm that simple CBA is not able to compete with ORB-SLAM. While our proposed methods with kinematic constraints perform on par with and occasionally even outperform ORB-SLAM. It basically depends on the dataset. Under challenging illumination conditions, the difference can be poten-



(a) (b) (c)

(d) (e) (f)

Figure 4.3: Mean and standard deviation of RPE for different methods on synthetic data using more sparsely sampled artificial trajectories. All experimental settings except the number of keyframes are similar to Figure 4.2. Varying noise levels, maximum number of observations for each landmark and number of landmarks per frame are on columns 1 2 and 3 respectively.

Dataset	method	mean(**t**)	stddev(**t**)	mean(**R**)	stddev(**R**)
VO-01	ORB-SLAM	0.1293	0.1676	**0.3149**	0.4548
	CBA	0.0170	0.0413	0.3580	0.5445
	CBASpRv	0.0082	0.0046	0.3929	0.4717
VO-04	ORB-SLAM	0.0073	0.0034	**0.0451**	**0.0312**
	CBA	0.0079	0.0039	0.0775	0.0392
	CBASpRv	**0.0050**	0.0032	0.0784	0.0392
VO-05	ORB-SLAM	**0.0098**	**0.0106**	**0.0455**	**0.0264**
	CBA	0.0233	0.0795	0.1434	0.4881
	CBASpRv	0.0111	0.0135	0.1292	0.1898
VO-06	ORB-SLAM	0.0076	0.0074	**0.0432**	**0.0277**
	CBA	0.0145	0.0411	0.1039	0.2494
	CBASpRv	**0.0057**	**0.0065**	0.0951	0.0821

Table 4.1: Comparison against ORB-SLAM. Error in **t**: [m], errors in **R**: [deg].

tially large. We would like to emphasise that—although ORB-SLAM also uses CBA in the back-end—it is a heavily engineered framework that performs additional tasks to reinforce the health and quality of the underlying graphical model. We therefore again conclude that the addition of kinematic constraints generally models the motion well, and increases the ability to handle degraded visual measurements.

4.4.5 Computational efficiency

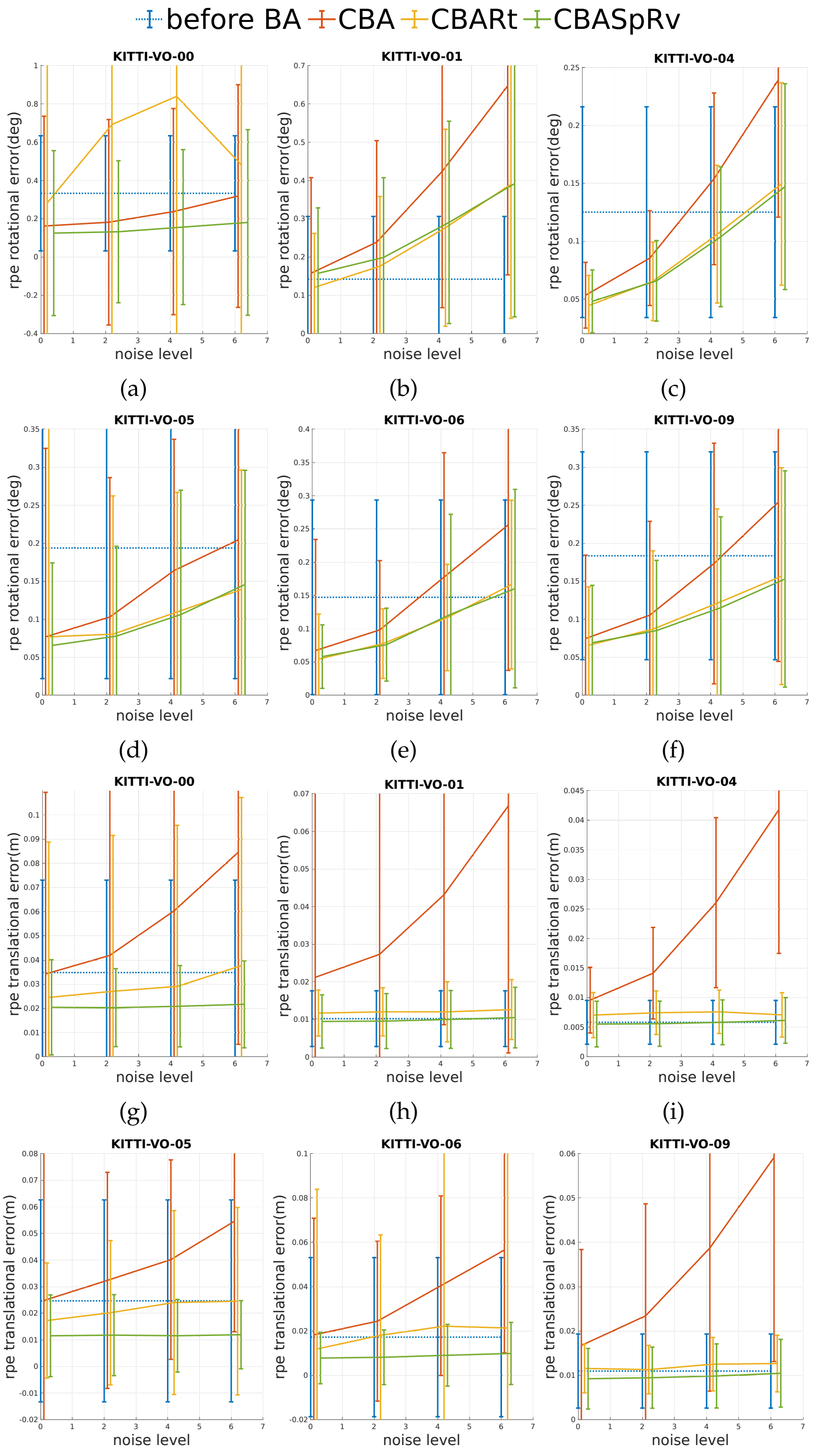
To conclude, we compare the computational efficiency of the different methods. As indicated in Table 4.2, with automatically differentiated B-spline-based implementations taking about double the time of conventional bundle adjustment, which is still acceptable for back-end optimization.

l.c.	CBA	CBARt	CBASpRv
20	9.294	8.963	22.225
30	13.887	15.661	30.790
40	18.162	19.775	57.784
50	22.194	22.555	62.120
60	25.977	25.263	68.009

Table 4.2: Average optimization time in seconds per 50 iterations for the proposed and the baseline methods. The optimization in simulation is over 1000 frames using the KITTI-VO-05 trajectory. *l.c.* denotes the local connectivity, and thus the number of generated landmarks per frame. The noise level is set to 4 pixels, and the global connectivity to 3 frames.

4.5 Discussion

In this chapter, we introduce continuous-time trajectory parametrizations for an exact modelling of non-holonomic ground vehicle trajectories in bundle adjustment. The proposed method shows significant improvements on both accuracy and robustness of monocular visual odometry, especially as the connectivity of the graph or the quality of the measurements degrades. This work fills the crucial gap in a robust, reliable motion estimation pipeline on intelligent mobile platforms. Although we only test our methods with a single camera, it is our belief that our general objective functions could be integrated into our aforementioned multi-perspective camera systems for further improvements.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Figure 4.4: Mean and standard deviation of RPE for different methods on real images from the KITTI benchmark. The first two rows show rotational errors, while the last two rows show translation errors. Each column presents results on a different dataset. *before BA* (blue dotted line) denotes the initial error before optimization.

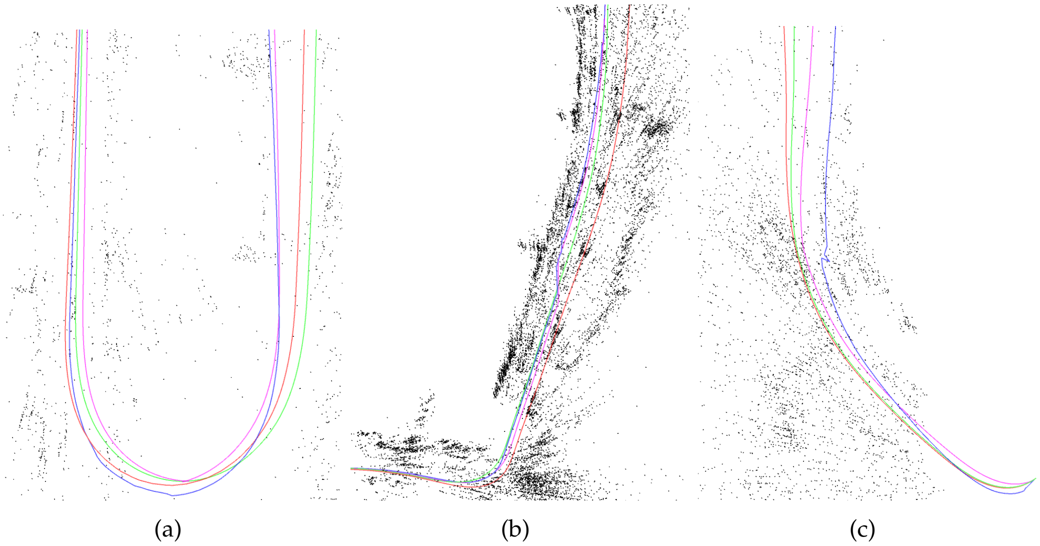
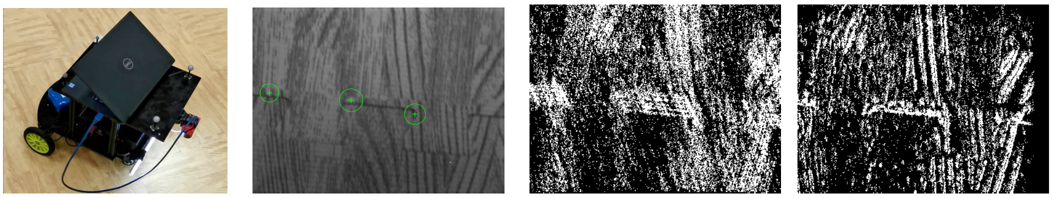


Figure 4.5: Example trajectory segments for ground truth (Red), CBA (Blue), CBARt (Magenta), CBASpRv (Green). Left: U-turn on KITTI VO-06. Center: Uneven surface on KITTI VO-00. Right: Sharp turn on KITTI VO-10.

Chapter 5

Globally-optimal event camera motion estimation for planar ground vehicles

In previous chapters, we have made a lot of efforts on purely vision-based localization and mapping problem for intelligent mobile platforms. Our proposed pipelines take advantages of kinematic constraints, and have developed efficient, reliable solutions on vehicle-mounted, multi-perspective camera systems. Although regular cameras have represented a highly attractive alternative of sensors for powerful SLAM systems, existing visual SLAM systems with regular cameras still come across challenges such as high dynamics scenes, changing illumination conditions, etc. In order to improve the camera performance in the above situations, we look at a new type of camera sensor, called a dynamic vision sensor, or event camera. Event cameras are bio-inspired sensors that perform well in HDR conditions and have high temporal resolutions. However, different from traditional frame-based cameras, event cameras measure asynchronous pixel-level brightness changes and return them in a highly discretized format, hence new algorithms are needed. The present chapter looks at fronto-parallel motion estimation of an event camera. The flow of the events is modeled by a general homographic warping in a space-time volume, and the objective is formulated as maximization of contrast within the image of unwarped events. However, in stark contrast to the prior art, we derive a globally optimal solution to this generally non-convex problem, and thus remove the dependency on a good initial guess. Our algorithm relies on branch-and-bound optimization for which we derive novel, recursive upper and lower bounds for contrast estimation functions. The practical validity of our approach is supported by a highly successful application to AGV motion estimation with a downward-facing event camera, a challenging scenario in which the sensor experiences fronto-parallel motion in front of noisy, fast moving textures(cf. Figure 5.1).



(a) AGV (b) wood grain foam (c) $\boldsymbol{\theta} = \mathbf{0}$ (d) $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$

Figure 5.1: (a): AGV equipped with a downward facing event camera for vehicle motion estimation. (b)-(d): collected image with detectable corners, image of warped events with $\boldsymbol{\theta} = \mathbf{0}$, and image of warped events with optimal parameters $\hat{\boldsymbol{\theta}}$.

5.1 Related work

A good overview of recent research on event-based vision is given by [Gallego et al., 2020]. In this section, our work aim at fronto-parallel motion estimation of an event camera. The flow of the events is hereby modelled by a general homographic warping in a space-time volume, and motion may be estimated by maximization of contrast in the image of unwarped events [Gallego et al., 2018]. Various reward functions that maximize contrast have been presented and analysed in the recent works of Gallego et al. [Gallego et al., 2019] and Stoffregen and Kleeman [Stoffregen and Kleeman, 2019], and successfully used for solving a variety of problems with event cameras such as optical flow [Zhu et al., 2017; Gallego et al., 2018; Stoffregen and Kleeman, 2017; Ye et al., 2018; Zhu et al., 2019a, 2018b], segmentation [Stoffregen and Kleeman, 2017; Stoffregen et al., 2019; Mitrokhin et al., 2018], 3D reconstruction [Rebecq et al., 2018; Zhu et al., 2018a, 2019a; Ye et al., 2018], and motion estimation [Gallego and Scaramuzza, 2017; Gallego et al., 2018]. Our work focuses on the latter problem of camera motion estimation. However—different from many of the aforementioned works—we propose the first globally optimal solution to the underlying contrast maximization problem, an important point given its generally non-convex nature.

The chapter is organized as follows. Section 5.2 provides a brief review of contrast maximization framework. Section 5.3 proceed to the core of our contribution, which is a globally optimal solution to contrast maximization using Branch and Bound (BnB) algorithm. Section 5.4 applies the globally-optimal contrast maximization framework to the real-world case of non-holonomic motion estimation with a downward-facing camera mounted on an AGV. Section 5.5 finally concludes with our results on both simulated and real data.

5.2 Contrast maximization framework

An event camera outputs a sequence of *events* denoting temporal logarithmic brightness changes above a certain threshold. An event $e = \{\mathbf{x}, t, s\}$ is described by its pixel position $\mathbf{x} = [x\ y]^T$, timestamp t, and polarity s (the latter indicates whether the brightness is increasing or decreasing, and is ignored in the present work). Our method is inspired by contrast maximization framework, which is recently introduced by Gallego et al. [Gallego et al., 2018] as a unifying framework allowing the solution of several important problems for dynamic vision sensors, in particular motion estimation problems in which the effect of camera motion may be described by a homography (e.g. motion in front of a plane, pure rotation). The key idea is that events are significantly more likely to be triggered by high gradient regions in the image (i.e. the edges). If the motion is estimated correctly, events which are triggered by the same point will be accumulated by the same pixel in the reference view, and the resulting Image of Warped Events (IWE) will therefore become a sharp edge map. The question is how the accumulation is done, and how the sharpness of the IWE is characterised. Gallego et al. propose to optimize the alignment of the events by maximizing the contrast in the IWE.

Suppose we are given a set of N events $\mathcal{E} = \{e_k\}_{k=1}^{N}$. We define a general warping function $\mathbf{x}'_k = W(\mathbf{x}_k, t_k; \boldsymbol{\theta})$ that returns the position $\mathbf{x}'_k$ of an event e_k in the reference view at time t_{ref}. $\boldsymbol{\theta}$ is a vector of warping parameters. The IWE is generated by accumulating warped events at each discrete pixel location:

$$I(\mathbf{p}_{ij}; \boldsymbol{\theta}) = \sum_{k=1}^{N} \mathbf{1}(\mathbf{p}_{ij} - \mathbf{x}'_k) = \sum_{k=1}^{N} \mathbf{1}(\mathbf{p}_{ij} - W(\mathbf{x}_k, t_k; \boldsymbol{\theta}))\,, \tag{5.1}$$

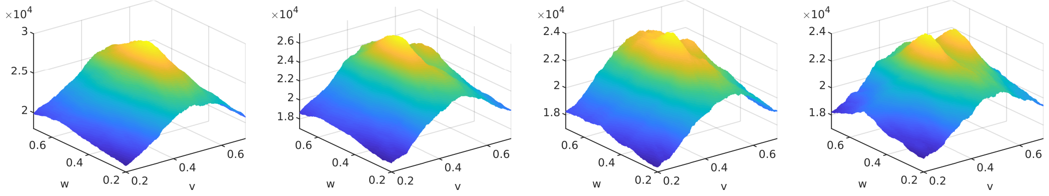
where $\mathbf{1}(\cdot)$ is an indicator function that counts 1 if the absolute value of $(\mathbf{p}_{ij} - \mathbf{x}'_k)$ is less than a threshold ϵ in each coordinate, and otherwise 0. $\mathbf{p}_{ij}$ is a pixel in the IWE with coordinates $[i\ j]^T$, and we refer to it as an *accumulator* location. We set $\epsilon = 0.5$ such that each warped event will increment one accumulator only.

Existing approaches replace the indicator function with a Gaussian kernel to make the IWE a smooth function of the warped events, and thus solve contrast maximization problems via local optimization methods (cf. [Gallego and Scaramuzza, 2017; Gallego et al., 2018, 2019]). In contrast, we show how our proposed method is able to find the global optimum of the above, discrete objective function.

As introduced in [Stoffregen and Kleeman, 2019; Gallego et al., 2019], reward functions for event un-warping all rely on the idea of maximizing the contrast or sharpness of the IWE (they have also been denoted as *focus loss functions*). They proceed by integration over the entire set of accumulators, which we denote $\mathcal{P}$. In this work, we use the Sum of Squares (SoS) as loss function, which is given by:

$$L_{\text{SoS}}(\boldsymbol{\theta}) = \sum_{\mathbf{p}_{ij} \in \mathcal{P}} I(\mathbf{p}_{ij}; \boldsymbol{\theta})^2 \tag{5.2}$$

There are also a lot of reward functions, which will not be introduced in the present dissertation.



(a) N/E = 0 (b) N/E = 0.02 (c) N/E = 0.10 (d) N/E = 0.18

Figure 5.2: Visualization of the Sum of Squares contrast function. The camera is moving in front of a plane, and the motion parameters are given by translational and rotational velocity (cf. Section 5.4). The sub-figures from left to right are functions with increasing Noise-to-Events (N/E) ratios. Note that contrast functions are non-convex.

5.3 Globally maximized contrast using Branch and Bound

Let us now proceed to the main contribution of our work, which is a derivation of bounds on the above objectives as required by Branch and Bound. Figure 5.2 illustrates how contrast maximization for motion estimation is in general a non-convex problem, meaning that local optimization may be sensitive to the initial parameters and not find the global optimum. We tackle this problem by introducing a globally optimal solution to contrast maximization using Branch and Bound (BnB) optimization. BnB is an algorithmic paradigm in which the solution space is subdivided into branches in which we then find upper and lower bounds for the maximal objective value. The globally optimal solution is isolated by an iterative search in which entire branches are discarded if their upper bound for the maximum objective value remains lower than the corresponding lower bound in another branch. The most important factor deciding the effectiveness of this approach is given by the tightness of the bounds.

Our core contribution is given by a recursive method to efficiently calculate upper and lower bounds for the maximum value of a contrast maximization function over a given branch. In short, the main idea is given by expressing a bound over $(N+1)$ events as a function of the bound over N events plus the contribution of one additional event. The strategy can be similarly applied to other contrast functions, and we limit the exposition to the derivation of bounds for L_{SoS} in the present thesis.

5.3.1 Objective Function

In the following, we assume that $L = L_{\text{SoS}}$. The maximum objective function value over all N events in a given time interval $[t_{\text{ref}}, t_{\text{ref}} + \Delta T]$ is given by

$$L_N = \max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \sum_{\mathbf{p}_{ij} \in \mathcal{P}} \left[\sum_{k=1}^{N} \mathbf{1}\left(\mathbf{p}_{ij} - W(\mathbf{x}_k, t_k; \boldsymbol{\theta}) \right) \right]^2 , \tag{5.3}$$

where $\boldsymbol{\theta}$ is the search space (i.e. branch or sub-branch) over which we want to maximize the objective. Most globally optimal methods for geometric computer vision problems find bounds by a spatial division of the problem into individual, simpler maximization sub-problems (cf. [Campbell et al., 2017]). However, the contrast maximization objective is related to the distribution over the entire IWE and not just individual accumulators, which complicates this strategy.

5.3.2 Upper Bound and Lower Bound

The bounds are calculated recursively by processing the events and one-by-one, each time updating the IWE. The events are notably processed in temporal order with increasing timestamps.

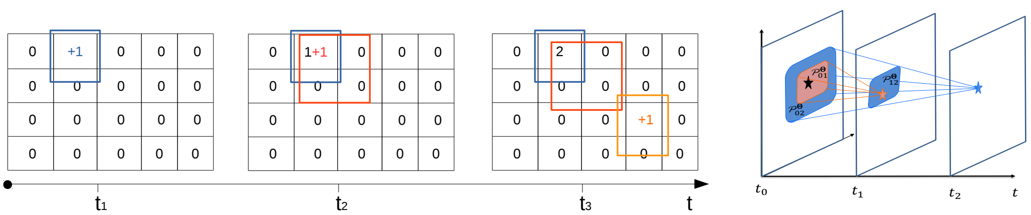
For the lower bound, it is readily given by evaluating the contrast function at an arbitrary point on the interval $\boldsymbol{\theta}$, which is commonly picked as the interval center $\boldsymbol{\theta}_0$. We present a recursive rule to efficiently evaluate the lower bound. For search space $\boldsymbol{\theta}$ centered at $\boldsymbol{\theta}_0$, according to the definition of the sum of the square focus loss function, the lower bound of SoS-based contrast maximization could be given by

$$\underline{L_{N+1}} = \underline{L_N} + 1 + 2I^N(\boldsymbol{\eta}^{\theta_0}_{N+1}; \boldsymbol{\theta}_0)\,, \tag{5.4}$$

where $I^N(\mathbf{p}_{ij}; \boldsymbol{\theta}_0)$ is the incrementally constructed IWE, its exponent N denotes the number of events that have already been taken into account, and

$$\boldsymbol{\eta}^{\theta_0}_{N+1} = \text{round}(W(\mathbf{x}_{N+1}, t_{N+1}; \boldsymbol{\theta}_0)) \tag{5.5}$$

returns the accumulator closest to the warped position of the $(N+1)$-th event. Note that the IWE is iteratively updated by incrementing the accumulator which locates closest to $\boldsymbol{\eta}^{\theta_0}_{N+1}$.


+1
1+1
2
+1
t₁
t₂
t₃
t
$\mathcal{P}^{\Theta}_{01}$
$\mathcal{P}^{\Theta}_{02}$
$\mathcal{P}^{\Theta}_{12}$
t_0
t_1
t_2
t

(a) (b)

Figure 5.3: (a) Incremental update of the IWE. For each new event e, we choose and increment the currently maximal accumulator in the bounding box $\mathcal{P}^{\boldsymbol{\Theta}}$ around all possible locations $W(\mathbf{x}, t; \boldsymbol{\theta} \in \boldsymbol{\Theta})$. We simply increment the center of the bounding box if no other accumulator exists. (b) Bounding boxes of two temporally distinct events generated by the same point in 3D.

We now proceed to our main contribution, a recursive upper bound for the contrast maximization problem. Let us define $\mathcal{P}^{\boldsymbol{\Theta}}_i$ as the bounding box around all possible locations $W(\mathbf{x}_i, t_i; \boldsymbol{\theta} \in \boldsymbol{\Theta})$ of the unwarped event. Given a search space $\boldsymbol{\theta} \in \boldsymbol{\Theta}$, for a small enough time interval, if $W(\mathbf{x}_i, t_i; \boldsymbol{\theta}) = W(\mathbf{x}_j, t_j; \boldsymbol{\theta})$ and $0 < i < j \leq N$, we have $\mathcal{P}_i^{\boldsymbol{\Theta}} \subseteq \mathcal{P}_j^{\boldsymbol{\Theta}}$. An intuitive explanation is given in Figure 5.3(b). Hence the upper bound of the objective function L_N for SoS-based contrast maximization satisfies

$$\begin{aligned} L_{N+1} &= L_N + 1 + 2I^N(\boldsymbol{\eta}_{N+1}^{\hat{\theta}}; \hat{\boldsymbol{\theta}}) && (5.6)\\ &\leq \overline{L_N} + 1 + 2Q^N = \overline{L_{N+1}}, && (5.7)\\ \text{where } Q^N &= \max_{\mathbf{p}_{ij} \in \mathcal{P}_{N+1}^{\boldsymbol{\Theta}}} \bar{I}^N(\mathbf{p}_{ij}) \geq I^N(\boldsymbol{\eta}_{N+1}^{\hat{\theta}}; \hat{\boldsymbol{\theta}}) \end{aligned}$$

$\mathcal{P}_{N+1}^{\boldsymbol{\Theta}}$ is a bounding box for the $(N+1)$-th event. $\hat{\boldsymbol{\theta}}$ is the optimal parameter set that maximizes L_{N+1} over the interval $\boldsymbol{\Theta}$. $\bar{I}^N(\mathbf{p}_{ij})$ is the value of pixel $\mathbf{p}_{ij}$ in the upper bound IWE, a recursively constructed image in which we always increment the maximum accumulator within the bounding box $\mathcal{P}_{N+1}^{\boldsymbol{\Theta}}$ (i.e. the one that we used to define the value of Q^N. The incremental construction of $\bar{I}^N(\mathbf{p}_{ij})$ is illustrated in Figure 5.3(a). Note that for $N = 0$, it is obvious that $L_0 = \overline{L_0} = 0$. It varies for different loss function and will be further used for initialization.

Our globally-optimal contrast maximization framework (GOCMF) is outlined in Algorithm 1 and Algorithm 2. We propose a nested strategy for calculating upper bounds, in which the outer layer *RB* evaluates the objective function, while the inner layer *BB* estimates the bounding box $\mathcal{P}_N^{\boldsymbol{\Theta}}$ and depends on the specific motion parametrization.

Algorithm 1 GOCMF: globally optimal contrast maximization framework

Input: event set $\mathcal{E}$, initial search space $\boldsymbol{\Theta}$, branching limit N_b

Output: optimal warping parameters $\hat{\boldsymbol{\theta}}$

```
Initialize $\hat{\boldsymbol{\theta}}$ with the center of $\boldsymbol{\Theta}$,
$L^* \leftarrow 0$ , $S \leftarrow$ {RB($\mathcal{E}$, $\boldsymbol{\Theta}$), $\boldsymbol{\Theta}$}
Push $S$ into queue $Q$, $S^* \leftarrow S$
while $i < N_b$ do
    if $S^*.\underline{L}, == S^*.\overline{L}$ then
        $\hat{\boldsymbol{\theta}} \leftarrow$ Center of $S^*.\boldsymbol{\Theta}$, break
    for each node $S \in Q$ do
        Pop $S$, split into subspaces $S_j$
        for all subspaces $S_j$ do
            $\{S_j.\underline{L}, S_j.\overline{L}\} \leftarrow$ RB($\mathcal{E}$, $\boldsymbol{\Theta}_j$)
            if $S_j.\underline{L} > L^*$ then
                $L^* \leftarrow S_j.\underline{L}$ , $S^* \leftarrow S_j$
            Push $S_j$ into $Q$
    Prune branches in $Q$
    $i \leftarrow i + 1$
return $\hat{\boldsymbol{\theta}}$
```

Algorithm 2 RB: recursive bounds calculation

```
Input: event set $\mathcal{E}$, search space $\boldsymbol{\Theta}$
Output: lower bound $\underline{L}$, upper bound $\overline{L}$
Initialize accumulator images $\overline{I}$ and $I$ with zeros
Initialize $\underline{L}$, $\overline{L}$
$\boldsymbol{\theta}_0 \leftarrow$ center of $\boldsymbol{\Theta}$
for each event $e_k \in \mathcal{E}$ do
    $\mathcal{P}_k^{\boldsymbol{\Theta}} \leftarrow BB(W(\cdot), \boldsymbol{\Theta}, e_k)$
    $Q \leftarrow \max_{\mathbf{p}_{ij} \in \mathcal{P}_k^{\boldsymbol{\Theta}}} \overline{I}(\mathbf{p}_{ij})$
    $\boldsymbol{\eta}_k^{\theta_0} \leftarrow \mathrm{round}(W(\mathbf{x}_k, t_k; \boldsymbol{\theta}_0))$
    Update $\underline{L}$, $\overline{L}$
    $\boldsymbol{\nu}_k \leftarrow \mathrm{argmax}_{\mathbf{p}_{ij} \in \mathcal{P}_k^{\boldsymbol{\Theta}}} \overline{I}(\mathbf{p}_{ij})$
    $\overline{I}(\boldsymbol{\nu}_k) \leftarrow \overline{I}(\boldsymbol{\nu}_k) + 1$
    $I(\boldsymbol{\eta}_k^{\theta_0}) \leftarrow I(\boldsymbol{\eta}_k^{\theta_0}) + 1$
return $\underline{L}$, $\overline{L}$
```

5.4 Application to visual odometry with a downward-facing event camera

Motion estimation for planar Autonomous Ground Vehicles (AGVs) is an important problem in intelligent transportation [Wang et al., 2020; Peng et al., 2019; Huang et al., 2019]. An interesting alternative is given by employing a downward instead of a forward-facing camera, thus permitting direct observation of the ground plane with known depth. This largely simplifies the geometry of the problem and notably turns the image-to-image warping into a homographic mapping that is linear in homogeneous space. The strategy is widely used in relevant applications such as sweeping robots and factory AGVs, and a good review is presented in [Aqel et al., 2016]. However, the method is affected by potentially severe challenges given by the image appearance: a) reliable feature matching or even extraction may be difficult for certain noisy ground textures, b) fast motion may easily lead to motion blur, and c) stable appearance may require artificial illumination. Many existing methods therefore do not employ feature correspondences but aim at a correspondence-less alignment or even a full photometric image alignment. Besides more classical RANSAC-based hypothesise-and-test schemes [Chen et al., 2018], the community therefore has also developed appearance-based template matching approaches [Dille et al., 2010; Nourani-Vatani et al., 2009; Yu et al., 2011; Nourani-Vatani and Borges, 2011; Gonzalez et al., 2012], solvers based on efficient second-order minimization [Lovegrove et al., 2011; Zienkiewicz and Davison, 2015; Jordan and Zell, 2016], and methods exploiting the Fast Fourier Transform [Piyathilaka and Munasinghe, 2010; Birem et al., 2018], the Fourier-Mellin Transform [Guo et al., 2005; Kazik and Göktoğan, 2011], or the Improved Fourier Mellin Invariant [Xu et al., 2019; Bülow and Birk, 2009]. In an attempt to tackle highly self-similar ground textures, Dille et al. [Dille et al., 2010]

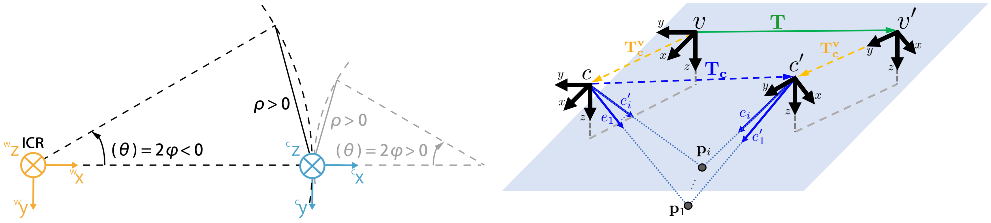


Figure 5.4: Left: *The Ackermann steering model* with the ICR [Gao et al., 2020]. Both left and a right turn are illustrated. Right: Connections between vehicle displacement, extrinsic transformation, and relative camera pose.

propose to use an optical flow sensor instead of a regular CMOS camera.

A critical question is given by the position of the camera. The camera may hang in the front or rear of the vehicle, which gives increased distance to the ground plane and in turn reduces motion blur. However, it also causes moving shadows in the image, and generally complicates the stabilisation of the image appearance and thus repeatable feature detection or region-based matching. A common alternative therefore is given by installing the camera underneath the vehicle paired with an artificial light source (e.g. [Dille et al., 2010; Birem et al., 2018]). However, the short distance to the ground plane may easily lead to unwanted motion blur. We therefore consider an event camera as a highly interesting and much more dynamic alternative visual sensor for this particular scenario.

5.4.1 Homographic Mapping and Bounding Box Extraction

We use the globally-optimal BnB solver for correspondence-less AGV motion presented in [Gao et al., 2020], which also employs a normal, downward-facing camera. We employ the two-dimensional Ackermann steering model describing the commonly non-holonomic motion of an AGV. Employing this 2-DoF model leads to benefits in BnB, the complexity of which strongly depends on the dimensionality of the solution space. As illustrated in Figure 5.4, the Ackermann model constrains the motion of the vehicle to follow a circular-arc trajectory about an Instantaneous Centre of Rotation (ICR). The motion between successive frames can be conveniently described at the hand of two parameters: the half-angle of the relative rotation angle ϕ, and the baseline between the two views ρ. However, the alignment of the events requires a temporal parametrization of the relative pose, which is why we employ the angular velocity $\omega = \frac{\theta}{t} = \frac{2\phi}{t}$ as well as the translational velocity $v = \omega r = \omega \rho \frac{1}{2\sin(\phi)}$ in our model. Note that the time interval t is a parameter that needs to be carefully adjusted depending on the movement speed of the camera. A shorter time window does not convey enough information under slow motion, while a longer time window will diminish the validity of the locally constant velocity model. As an example, for the experiments presented in Section 5.5, we use 0.04s as the time parameter in all sequences. The relative transformation from vehicle frame v' back to v is therefore given by

$$\mathbf{R}_v = \begin{bmatrix} \cos(\omega t) & -\sin(\omega t) & 0 \\ \sin(\omega t) & \cos(\omega t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \mathbf{t}_v = \frac{v}{\omega} \begin{bmatrix} 1 - \cos(\omega t) \\ \sin(\omega t) \\ 0 \end{bmatrix} . \tag{5.8}$$

In practice, the vehicle frame hardly coincides with the camera frame. The orientation and the height of the origin can be chosen to be identical, and the camera may be laterally mounted in the centre of the vehicle. However, there is likely to be a displacement along the forward direction, which we denote by the signed variable s. In other words, $\mathbf{R}_v^c = \mathbf{I}_{3\times 3}$ and $\mathbf{t}_v^c = \begin{bmatrix} 0 & s & 0 \end{bmatrix}^T$. As illustrated in Figure 5.4, the transformation from camera pose c' (at an arbitrary future timestamp) to c (at the initial timestamp t_{ref}) is therefore given by

$$\begin{aligned} \mathbf{R}_c &= \mathbf{R}_v^{cT} \mathbf{R}_v \mathbf{R}_v^c , \\ \mathbf{t}_c &= -\mathbf{R}_v^{cT} \mathbf{t}_v^c + \mathbf{R}_v^{cT} \mathbf{t}_v + \mathbf{R}_v^{cT} \mathbf{R}_v \mathbf{t}_v^c . \end{aligned} \tag{5.9}$$

Using the known plane normal vector $\mathbf{n} = \begin{bmatrix} 0 & 0 & -1 \end{bmatrix}^T$ and depth-of-plane d, the image warping function $W(\mathbf{x}_k, t_k; [\omega\ v]^T)$ that permits the transfer of an event $e_k = \{\mathbf{x}_k, t_k, s_k\}$ into the reference view at t_{ref} is finally given by the planar homography equation

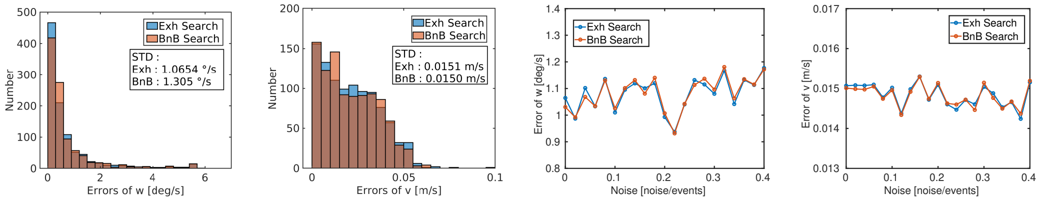
$$\mathbf{H} \begin{bmatrix} \mathbf{x}_k^T & 1 \end{bmatrix}^T = \mathbf{K}(\mathbf{R}_c - \frac{\mathbf{t}_c \mathbf{n}^\mathbf{T}}{d}) \mathbf{K}^{\mathbf{-1}} \begin{bmatrix} \mathbf{x}_k^T & 1 \end{bmatrix}^T . \tag{5.10}$$

Note that $\mathbf{K}$ here denotes a regular perspective camera calibration matrix with homogeneous focal length f, zero skew, and a principal point at $\begin{bmatrix} u_0 & v_0 \end{bmatrix}^T$. Note further that the substituted time parameter needs to be equal to $t = t_k - t_{\text{ref}}$, and that the result needs to be dehomogenised. After expansion, we easily obtain

$$\begin{aligned} \mathbf{x}_k' &= W(\mathbf{x}_k, t_k; [\omega\ v]^T) = \begin{bmatrix} x_k' & y_k' \end{bmatrix}^T \qquad (5.11) \\ &= \begin{bmatrix} -[y_k - v_0 + s\frac{f}{d}] \sin(\omega t) + [x_k - u_0 - \frac{f}{d}(\frac{v}{w})] \cos(\omega t) + \frac{f}{d}(\frac{v}{w}) + u_0 \\ [x_k - u_0 - \frac{f}{d}(\frac{v}{w})] \sin(\omega t) + [y_k - v_0 + s\frac{f}{d}] \cos(\omega t) - s\frac{f}{d} + v_0 \end{bmatrix} . \end{aligned}$$

Finally, the bounding box $\mathcal{P}_k^{\mathbf{\Theta}}$ is found by bounding the values of x_k' and y_k' over the intervals $\omega \in \mathcal{W} = [\omega_{\min}; \omega_{\max}]$ and $v \in \mathcal{V} = [v_{\min}; v_{\max}]$. The bounding is easily achieved if simply considering monotonicity of functions over given sub-branches. For example, if $\omega_{\min} \geq 0$, $v_{\min} \geq 0$, $x_k \geq u_0$, and $y_k \geq v_0 - s\frac{f}{d}$, we obtain

$$\begin{aligned} \underline{x_k'} &= -[y_k - v_0 + s\frac{f}{d}] \sin(\omega_{\max} t) + [x_k - u_0 - \frac{f}{d}(\frac{v_{\min}}{w_{\max}})] \cos(\omega_{\max} t) + \frac{f}{d}(\frac{v_{\min}}{w_{\max}}) + u_0 , \\ \overline{x_k'} &= -[y_k - v_0 + s\frac{f}{d}] \sin(\omega_{\min} t) + [x_k - u_0 - \frac{f}{d}(\frac{v_{\max}}{w_{\min}})] \cos(\omega_{\min} t) + \frac{f}{d}(\frac{v_{\max}}{w_{\min}}) + u_0 , \\ \underline{y_k'} &= [x_k - u_0 - \frac{f}{d}(\frac{v_{\max}}{w_{\min}})] \sin(\omega_{\min} t) + [y_k - v_0 + s\frac{f}{d}] \cos(\omega_{\max} t) - s\frac{f}{d} + v_0 , \text{ and} \\ \overline{y_k'} &= [x_k - u_0 - \frac{f}{d}(\frac{v_{\min}}{w_{\max}})] \sin(\omega_{\max} t) + [y_k - v_0 + s\frac{f}{d}] \cos(\omega_{\min} t) - s\frac{f}{d} + v_0 . \end{aligned} \tag{5.12}$$



(a) (b) (c) (d)

Figure 5.5: Simulation Results. (a) and (b) indicate the error distribution for ω and v over all experiments for both our proposed method as well as an exhaustive search. (c) and (d) visualise the average error of the estimated parameters caused by adding salt and pepper noise on the event stream. Results are averaged over 1000 random experiments. Note that our proposed method has excellent robustness even for N/E ratios up to 40%.

5.5 Experimental evaluation

We present two suites of experiments. The first one validates the global optimality, accuracy and robustness of our solver on simulated data. The second one then applies it to the real-world scenario of AGV motion estimation.

5.5.1 Accuracy and Robustness of Globally Optimal Motion Estimation

We start by evaluating the accuracy of the motion estimation with contrast maximisation function L_{SoS} over synthetic data. As already implied in [Gallego et al., 2019], L_{SoS} can be considered as a solid starting point for the evaluation. Our synthetic data consists of randomly generated horizontal and vertical line segments on a plane at a depth of 2.0m. We consider Ackermann motion with an angular velocity $\omega = 28.6479°/\text{s}$ (0.5rad/s) and a linear velocity $v = 0.5\text{m/s}$. Events are generated by randomly choosing a 3D point on a line, and reprojecting it into a random camera pose sampled by a random timestamp within the interval $[0, 0.1s]$. The result of our method is finally evaluated by running BnB over the search space $\mathcal{W} = [0.4, 0.6]$ and $\mathcal{V} = [0.4, 0.6]$, and comparing the retrieved solution against the result of an exhaustive search with sampling points every $\delta\omega = 0.001\text{rad/s}$ and $\delta v = 0.001\text{m/s}$. BnB is furthermore configured to terminate the search if $|\omega_{max} - \omega_{min}| \leq 0.00078\text{rad/s}$ or $|v_{max} - v_{min}| \leq 0.00078\text{m/s}$. The experiment is repeated 1000 times.

Figures 5.5(a) and 5.5(b) illustrate the distribution of the errors for both methods in the noise-free case. The standard deviation of the exhaustive search and BnB are $\sigma_\omega = 1.0645°/\text{s}$, $\sigma_v = 0.0151\text{m/s}$ and $\sigma_\omega = 1.305°/\text{s}$, $\sigma_v = 0.0150\text{m/s}$, respectively. While this result suggests that BnB works well and sustainably returns a result very close to the optimum found by exhaustive search, we still note that the optimum identified by both methods has a bias with respect to ground truth, even in the noise-free case. Note however that this is related to the nature of the contrast maximisation function, and not our globally optimal solution strategy.

In order to analyse robustness, we randomly add salt and pepper noise to the event stream with noise-to-event (N/E) ratios between 0 and 0.4 (Example objective functions for different N/E ratios have already been illustrated in Figure 5.2). Figure 5.5(c) and 5.5(d) show the error for each noise level again averaged over 1000 experiments. As can be observed, the errors are very similar and behave more or less independently of the amount of added noise. The latter result underlines the high robustness of our approach.

5.5.2 Application to real data and comparison against alternatives

We apply our method to real data collected by a DAVIS346 event camera, which outputs events streams with a maximum time resolution of 1μs as well as regular frames at a frame rate of 30Hz. Images have a resolution of 346×260. We mount the camera on the front of a XQ-4 Pro robot and let it face downward. The displacement from the non-steering axis to the camera is $s = -0.45$m, and the height difference between camera and ground is $d = 0.23$m. We recorded several motion sequences on a wood grain foam which has highly self-similar texture and poses a challenge to reliably extract and match features. Ground truth is obtained via an Opti-track optical motion tracking system. Our algorithm is working in undistorted coordinates, which is why normalisation and undistortion are computed in advance.

Method	**Line w[°/s]**	**Line v[m/s]**	**Circle w[°/s]**	**Circle v[m/s]**	**Curve w[°/s]**	**Curve v[m/s]**
SoS	**0.5127**	**0.0086**	**1.0884**	**0.0083**	**3.0091**	0.0208
IFMI	145.3741	1.0594	8.1092	0.0243	12.8047	**0.0192**
GOVO	6.9705	0.2409	4.5506	0.0642	9.8652	0.0590

Table 5.1: RMS errors for different datasets and methods.

We test the algorithm over various types of motions, including a straight line, a circle, and an arbitrarily curved trajectory. We compare our method with two state-of-the-art approaches for regular images, namely the correspondence-less globally optimal feature-based approach (GOVO) from [Gao et al., 2020], as well as the Improved Fourier Mellin Invariant transform (IFMI) in [Xu et al., 2019; Bülow and Birk, 2009]. These two alternatives are frame-based algorithms specifically designed for planar AGV motion estimation under featureless conditions. Figure 5.1 shows an example frame of the wood grain foam texture, and Figure 5.6 the results obtained for all methods. As can be observed, GOVO finds as little as three corner features for some of the images, thus making it difficult to accurately recover the vehicle displacement despite the globally-optimal correspondence-less nature of the algorithm. Both IFMI and GOVO occasionally lose tracking (especially for linear motion), which leaves our proposed globally-optimal event-based method outperforms others. The qualitative results are shown in Table 5.1, which provides the RMS errors of the estimated dynamic parameters. It proves that our event-based motion estimation method outperforms the intensity-camera-based alternatives.

BnB vs Gradient Ascent: We apply both gradient descent as well as BnB to the

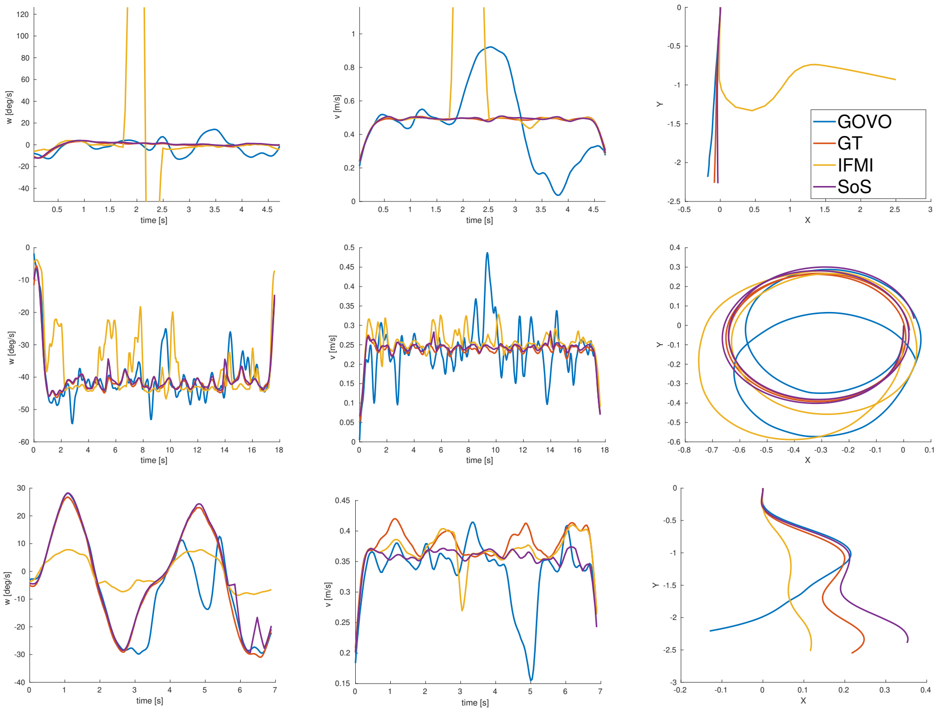
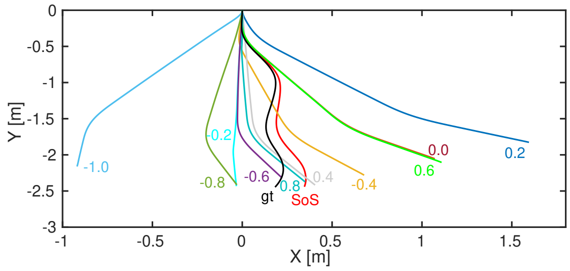


Figure 5.6: Results for all methods over different datasets. The first two columns are errors over time for ω and v, and the third column illustrates a bird's eye view onto the integrated trajectories.

Foam dataset with curved motion. For the first temporal interval and the local search method, we vary the initial angular velocity ω and linear velocity v between -1 and 0.8 with steps of 0.2 (rad/s or m/s, respectively). For later intervals, we use the previous local optimum. Figure 5.7 illustrates the estimated trajectories for all initial values, compared against ground truth and our BnB search method. RMS errors are also indicated. As clearly shown, even the best initial guess eventually diverges under a local search strategy, thus leading to clearly inferior results compared to our globally optimal search.



Method	w[°/s]	v[m/s]
SoS	3.0091	0.0208
GA	11.5023	0.0379

Figure 5.7: Estimated trajectories by our method (SoS), gradient ascent with various initializations, and ground truth (gt). The table indicates the RMS errors for the best performing gradient ascent run and SoS.

5.6 Discussion

In this chapter, we introduce the first globally optimal solution to contrast maximization for un-warped event streams. To the best of our knowledge, we are also the first to apply the idea of homography estimation via contrast maximization to the real-world case of non-holonomic motion estimation with a downward facing camera mounted on an AGV. The challenging conditions in this scenario favorise dynamic vision sensors over regular frame-based cameras, a claim that is supported by our experimental results. The latter furthermore prove that global solutions are important and significantly outperform incremental local refinement. The recursive formulation of our bound lets us find the global optimum over event streams of 0.04s within less than one minute, a respectable achievement given the typically low computational efficiency of BnB solvers.

Chapter 6

Event camera tracking and mapping by volumetric contrast maximization

In the previous chapter, we have introduced the first globally optimal solution to the problem of non-holonomic motion estimation with a downward-facing camera mounted on an AGV by performing contrast maximization for unwarped event streams. However, as outlined in the aforementioned chapter, one of the main disadvantages of the image-warping based IWE contrast maximization is that it does not involve any unknown depth parameters. As a result, it is not possible to use the objective in situations in which the flow at each location in the image is also a function of the depth. For this reason, we present a new solution to tracking and mapping with an event camera. The motion of the camera contains both rotation and translation, and the displacements happen in an arbitrarily structured environment, the image warping therefore may no longer be represented by a low-dimensional homographic warping. A new solution to this problem is introduced by performing contrast maximization in 3D. The 3D location of the rays cast for each event is smoothly varied as a function of a continuous-time motion parametrization, and the optimal parameters are found by maximizing the contrast in a volumetric ray density field. Our method thus performs joint optimization over motion and structure. The practical validity of our approach is supported by an application to AGV motion estimation and 3D reconstruction with a single vehicle-mounted event camera. The method performs at least on par with regular cameras, and eventually outperforms in challenging visual conditions.

6.1 Related work

Past solutions to vision-based localization and mapping therefore looked at alternative solution attempts. Note that there are lots of works on the localization and mapping problems individually, a listing of which would go beyond the scope of this introduction. Here we only focus on combined solutions to both problems that use only a single event camera, and that are able to handle combined rotational and translational displacements in unknown, arbitrarily structured environments. There are surprisingly few works that solve this problem, which is proof of its difficulty. The first solution to this problem is given by Kim et al. [Kim et al., 2016], who propose a complex framework of three individual filters. Results are limited to small scale environments and small, dedicated motions. A geometric attempt is given by Rebecq et al. [Rebecq et al., 2016], who present a combination of a tracker and their ray-density based structure extraction method EMVS [Rebecq et al., 2018]. However, the framework alternates between the tracking and mapping solutions, which leaves open questions as to how to safely bootstrap the system. Zhu et al. [Zhu et al., 2019b] finally present a promising learning-based approach. It does however depend on vast amounts of training data, and provides no guarantees of optimality or generality.

The chapter is organized as follows. Section 6.2 provides a brief review of contrast maximization framework. Section 6.3 introduces how we extend the idea of contrast maximization into 3D and represent the motion with a continuous-time camera trajectory model. Section 6.4 applies the idea of volumetric contrast maximization to the real-world case of non-holonomic motion estimation with a forward-facing camera mounted on an AGV. Section 6.5 finally demonstrates our results on multiple, real datasets, both the accuracy and quality of the estimated trajectories, as well as the quality of the implicitly modelled 3D structure are assessed.

6.2 Contrast maximization

We are given a set of N events $\mathcal{E} = \{e_k\}_{k=1}^{N}$ happening over a certain time interval, where each event $e_k = \{\mathbf{x}_k, t_k, b_k\}$ is defined by its image location $\mathbf{x}_k = [x_k \quad y_k]^T$, timestamp t_k, and polarity b_k. Note that the set is ordered, meaning that if $\mathcal{E} = \{\ldots, e_i, \ldots, e_j, \ldots\}$, then $t_i \leq t_j$. We furthermore assume that image warping during the entire time interval can be parametrized as a continuous-time function of a certain parameter vector $\boldsymbol{\theta}$, and define the warping function $\mathbf{x'}_k = \mathbf{W}(\mathbf{x}_k, t_k|\boldsymbol{\theta})$ that warps an event with location $\mathbf{x}_k$ and timestamp t_k into a reference view at t_r. For example, if the camera undergoes pure rotation, and we use a constant rotational velocity model, the warping of an event into the reference view can be achieved by extrapolating the relative rotation and the corresponding homography at infinity.

As mentioned in the previous section 5.2, the contrast maximization framework proposed by [Gallego et al., 2018] aims at optimizing the alignment of the events by maximizing the contrast in the Image of Warped Events (IWE). The IWE at point $\mathbf{x}$ is defined by

$$I(\mathbf{x}|\boldsymbol{\theta}) = \sum_{k=1}^{N} e^{-\frac{\|\mathbf{x}-\mathbf{x'}_k\|^2}{2\sigma}}, \tag{6.1}$$

and it is evaluated discretely for each pixel centre location. While the application of a Gaussian kernel makes sure that events which are closer to a certain pixel contribute more than events that are further away, it also makes sure that the IWE and its contrast remain smooth functions of the motion parameters and thus optimizable through gradient-based methods. According to [Stoffregen and Kleeman, 2019] and

[Gallego et al., 2019], the contrast or sharpness of the IWE may finally be evaluated using one of several possible focus loss functions. Here we use the perhaps most common one, given by the IWE variance

$$f_{Var}(I) = \frac{1}{N_p} \sum_{i,j} (I(\mathbf{x}_{ij}|\boldsymbol{\theta}) - \mu_I)^2. \tag{6.2}$$

μ_I is the mean value of I, N_p the number of pixels in I, and i and j are indices that loop through all the rows and columns of the IWE. As neatly visualised by the heat maps in [Gallego et al., 2019], the highest variance of the IWE gives the highest contrast location, and thus the optimal motion parameters causing the best alignment of the warped events.

The framework allows us to tackle several important motion estimation problems for event-based vision, such as optical flow estimation, motion segmentation, or pure rotational motion estimation. However, note that for an arbitrary point to be warped into the reference view, the warping must either be homographic, or the parameter vector θ must contain the depth for each event at the time it was captured. Both are rather restrictive towards general camera motion estimation in arbitrary environments. Current state-of-art contrast maximization methods are therefore only able to handle a particular set of problems such as motion in front of a plane, or pure rotation.

6.3 Volumetric contrast maximization using ray warping

Let us now proceed to our main contribution, which consists of extending the idea of contrast maximization into 3D, a technique that will enable us to handle situations in which we perceive non-planar environments under arbitrary motion and with no priors on the depth of events. Our main idea is illustrated in Figure 6.1. We introduce a continuous-time camera trajectory model as done in Furgale et al. [Furgale et al., 2015], which parametrizes both the position and the orientation of the sensor as a smooth, continuous function of time. For a given event, we may then use its timestamp to extrapolate the position and orientation of the event camera at the time the event was captured. Combined with the normalised spatial direction of the event inside the camera frame, each event can be translated into a spatial ray for which the starting point and orientation depend on the continuous trajectory parameters. Rather than evaluating the density of points for pixels in the image, we then propose to evaluate the density of rays at discrete locations in a volume in front of a reference view. We denote this volumetric density field the *Volume of Warped Events (VWE)*. The intuition is analogous to the IWE: the assumption is that there is a limited number of spatial (appearance or geometric) edges that will cause sufficiently large gradients in the image. Under the optimal motion parameters, the rays of the events will therefore intersect along those spatial edges and cause maximum *ray density* in those regions. In other words, the optimal motion parameters may be found by maximizing the contrast in the VWE. The important question is again given by how to express the ray density in the VWE.

The structure of the VWE field is inspired by the space-sweeping approach of [Rebecq et al., 2018] et al., who propose to estimate 3D structure regardless of explicit data associations and photometric information by finding local maxima in a spatial ray density field. However, their method assumes known camera poses, and they use an alternative camera tracking scheme in their previous work [Rebecq et al., 2016]. To the best of our knowledge, we are the first to propose the maximization of the contrast in the volumetric ray density field, and thus implicitly perform joint optimization over the continuous camera trajectory parameters and the 3D structure.

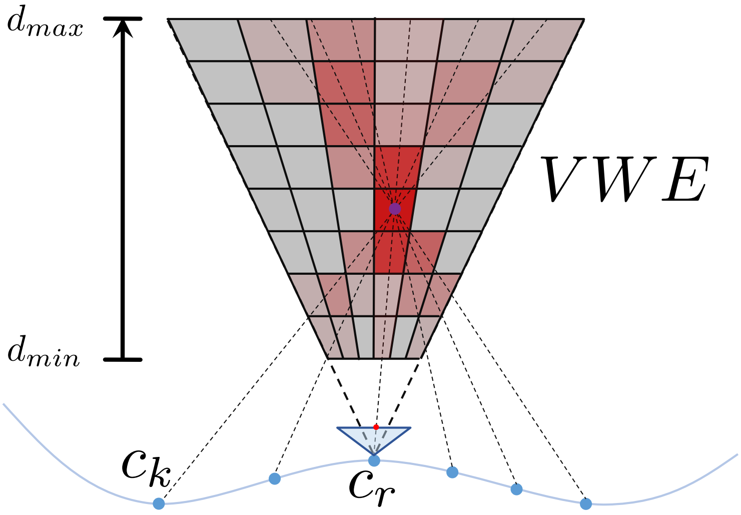


Figure 6.1: Volume of Warped Events: Events are transformed into rays that are warped in space based on a continuous-time trajectory model. We evaluate the ray density in a volume in front of a reference view, and maximize its contrast as a function of the continuous motion parameters.

6.3.1 Continuous Ray Warping

Suppose our event camera is pre-calibrated and we are given camera-to-image and image-to-camera transformation functions $\pi(\cdot)$ and $\pi^{-1}(\cdot)$, respectively. The latter transforms image locations into spatial directions in the camera frame, i.e.

$$\mathbf{f}_k = \pi^{-1}(\mathbf{x}_k). \tag{6.3}$$

In terms of the extrinsics, the trajectory of the camera is kept general for now and simply represented by a minimal, time-continuous, smoothly varying 6-vector

$$\mathbf{s}(t|\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{t}(t|\boldsymbol{\theta}) \\ \mathbf{q}(t|\boldsymbol{\theta}) \end{bmatrix}, \tag{6.4}$$

where $\boldsymbol{\theta}$ still represents a set of continuous motion parameters, $\mathbf{t}$ the position of the camera expressed in a world frame, and $\mathbf{q}$ its orientation as a Rodriguez vector. Note that the dimensionality of $\boldsymbol{\theta}$ is left unspecified for now. However, as will be shown in Section 6.4, it may indeed have only one or two parameters for certain special types of planar displacements. Besides their inherently smooth property, the continuous-time trajectory model has the obvious ability of being able to register information coming from temporally dense sampling sensors, such as event cameras. The transformation from camera to world at time t is given by

$$\mathbf{T}(t|\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{R}(\mathbf{q}(t|\boldsymbol{\theta})) & \mathbf{t}(t|\boldsymbol{\theta}) \\ \mathbf{0}^{\mathsf{T}} & 1 \end{bmatrix}. \tag{6.5}$$

With reference to Figure 6.1, c_k represents the camera frame at time t_k where a certain event e_k has been captured. The absolute pose of the frame at the time of capturing e_k is given by $\mathbf{T}_{wk} = \mathbf{T}(t_k|\boldsymbol{\theta})$. Now let c_r be the reference frame in which we define the projective sampling volume for the VWE. The absolute pose of c_r is given by $\mathbf{T}_{wr} = \mathbf{T}(t_r|\boldsymbol{\theta})$. The relative transformation is finally given as

$$\begin{aligned} \mathbf{T}_{rk} &= \begin{bmatrix} \mathbf{R}_r(t_k|\boldsymbol{\theta}) & \mathbf{t}_r(t_k|\boldsymbol{\theta}) \\ \mathbf{0}^{\mathsf{T}} & 1 \end{bmatrix} = \mathbf{T}_{wr}^{-1}\mathbf{T}_{wk} \qquad (6.6) \\ &= \begin{bmatrix} \mathbf{R}(\mathbf{q}(t_r|\boldsymbol{\theta})) & \mathbf{t}(t_r|\boldsymbol{\theta}) \\ \mathbf{0}^{\mathsf{T}} & 1 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{R}(\mathbf{q}(t_k|\boldsymbol{\theta})) & \mathbf{t}(t_k|\boldsymbol{\theta}) \\ \mathbf{0}^{\mathsf{T}} & 1 \end{bmatrix}. \end{aligned}$$

Finally, let λ represents the unknown depth along the ray. Any point on the ray seen from the reference view can be parametrized using the equation

$$\mathbf{p}_k(\lambda) = \lambda \mathbf{R}_r(t_k|\boldsymbol{\theta})\mathbf{f}_k + \mathbf{t}_r(t_k|\boldsymbol{\theta}). \tag{6.7}$$

6.3.2 VWE and spatial contrast maximization

We are now going back to our question of how to express the ray density in the VWE. The VWE is defined in a volumetric, projective sampling grid as illustrated in Figure 6.2. Let $\mathbf{v}$ be the centre of a voxel. The density of the rays in a voxel is now expressed as a function of the orthogonal distance between the voxel centre $\mathbf{v}$ (expressed in the reference view) and each individual ray. This spatial point-to-line distance is also called the *object space distance*, and it is given by

$$\epsilon_k^r(\mathbf{v}|\boldsymbol{\theta}) = \|(\mathbf{I} - \mathbf{V}_k)(\mathbf{R}_r^{\mathsf{T}}(t_k|\boldsymbol{\theta})(\mathbf{v} - \mathbf{t}_r(t_k|\boldsymbol{\theta})))\|, \tag{6.8}$$

where we have used the rotation $\mathbf{R}_r^{\mathsf{T}}(t_k|\boldsymbol{\theta})$ and translation $-\mathbf{R}_r^{\mathsf{T}}(t_k|\boldsymbol{\theta})\mathbf{t}_r(t_k|\boldsymbol{\theta})$ to transform the voxel centre $\mathbf{v}$ into the camera viewpoint at time t_k, and the householder matrix $(\mathbf{I} - \mathbf{V}_k) = (\mathbf{I} - \frac{\mathbf{f}_k\mathbf{f}_k^T}{\mathbf{f}_k^T\mathbf{f}_k})$ to project this point onto the normal plane of the observation direction $\mathbf{f}_k$. An example of object space distances for one voxel is indicated in Figure 6.2.

Supposing that we have N events, the final VWE is again given in smooth form by applying a Gaussian kernel and summing up the object space distances of every

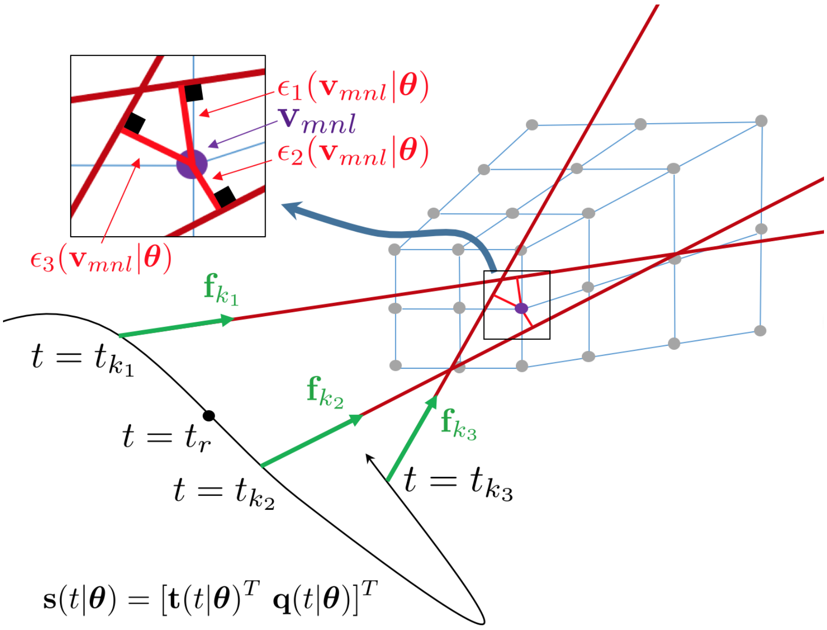

$\epsilon_1(\mathbf{v}_{mnl}|\boldsymbol{\theta})$
$\mathbf{v}_{mnl}$
$\epsilon_2(\mathbf{v}_{mnl}|\boldsymbol{\theta})$
$\epsilon_3(\mathbf{v}_{mnl}|\boldsymbol{\theta})$
$\mathbf{f}_{k_1}$
$t = t_{k_1}$
$t = t_r$
$\mathbf{f}_{k_2}$
$\mathbf{f}_{k_3}$
$t = t_{k_2}$
$t = t_{k_3}$
$\mathbf{s}(t|\boldsymbol{\theta}) = [\mathbf{t}(t|\boldsymbol{\theta})^T \ \mathbf{q}(t|\boldsymbol{\theta})]^T$

Figure 6.2: Warped rays with object space distances for one example voxel $\mathbf{v}_{mnl}$.

event with respect to the voxel centre $\mathbf{v}$

$$V^r(\mathbf{v}|\boldsymbol{\theta}) = \sum_{k=1}^{N} e^{-\frac{\epsilon_k^r(\mathbf{v}|\boldsymbol{\theta})^2}{2\sigma}}. \tag{6.9}$$

The standard deviation σ of the Gaussian kernels is actually not constant, but chosen as a function of the depth of the corresponding voxel from the centre of the reference view.

The final optimization objective is again given by maximizing the variance of the VWE, which expresses the sharpness of the edges reflected in the volumetric density field

$$f_{Var}(V^r) = \frac{1}{N_v} \sum_{m,n,l} (V^r(\mathbf{v}_{mnl}|\boldsymbol{\theta}) - \mu_{V^r})^2. \tag{6.10}$$

μ_{V^r} is the mean value of V^r, N_v the number of voxels in the entire volume, and m, n, and l now iterate through the voxels in the volume. Figure 6.3 visualizes an example VWE for wrong and correct motion parameters. For correct motion parameters (cf. 6.3(a) and 6.3(b)), the density field presents higher values and more contrast than for wrong motion parameters (cf. 6.3(c) and 6.3(d)).

6.3.3 Global optimization over longer trajectories

We perform global optimization by simultaneously maximizing the contrast in multiple VWEs cast from neighbouring reference views. Let $\{t_{r_1}, \ldots, t_{r_M}\}$ be the time

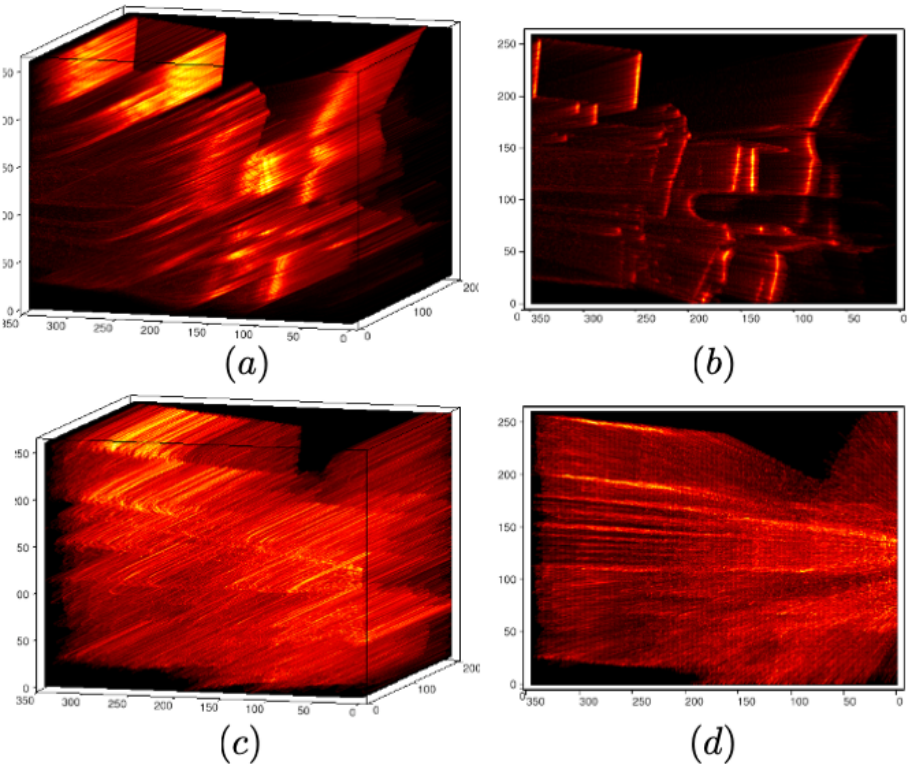


Figure 6.3: Volumetric ray density fields for correct ((a) and (b)) and wrong ((c) and (d)) motion parameters.

instants at which individual VWEs are placed. For simplicity, the time instants are regularly spaced such that $t_{r_{i+1}} - t_{r_i} = \tau_1$. We furthermore define time intervals $[t_{r_i} - \frac{\tau_2}{2} : t_{r_i} + \frac{\tau_2}{2}]$ for each corresponding field V^{r_i}, which define the subset of events that will be used for registration in that reference view. More specifically, event e_k is used in V^{r_i} if $t_k \in [t_{r_i} - \frac{\tau_2}{2} : t_{r_i} + \frac{\tau_2}{2}]$. The overall global optimization objective becomes

$$\max_{\boldsymbol{\theta}} \sum_{i=1}^{M} \frac{1}{N_v} \sum_{m,n,l} (V^{r_i}(\mathbf{v}_{mnl}|\boldsymbol{\theta}) - \mu_{V^{r_i}})^2, \tag{6.11}$$

$$\text{where } V^{r_i}(\mathbf{v}|\boldsymbol{\theta}) = \sum_{e_k \in \mathcal{E}_{t_{r_i}-\frac{1}{2}\tau_2}^{t_{r_i}+\frac{1}{2}\tau_2}} e^{-\frac{\epsilon_k^{r_i}(\mathbf{v}|\boldsymbol{\theta})^2}{2\sigma}}, \tag{6.12}$$

and $\mathcal{E}_{t_{r_i}-\frac{1}{2}\tau_2}^{t_{r_i}+\frac{1}{2}\tau_2}$ is defined as the subset of all the events e_k for which $t_k \in [t_{r_i} - \frac{\tau_2}{2} : t_{r_i} + \frac{\tau_2}{2}]$. The global optimization strategy is depicted in Figure 6.4. Note that τ_1 may be chosen such that neighbouring volumes are overlapping, and τ_2 may be chosen such that events are considered in more than just a single volumetric density field (i.e. $\tau_2 > \tau_1$). These choices make sure that the implicit graph behind this optimization problem is well connected and effects such as scale propagation take place.

6.4 Application to AGV with a forward-facing event camera

We evaluate our method on a planar Autonomous Ground Vehicle (AGV), on which we mount a single forward-facing event camera. Many solutions for the case of a regular, monocular camera exist, such simple relative pose solvers [Nistér et al., 2004] or full visual SLAM frameworks [Mur-Artal and Tardós, 2017]. The application of an event camera promises strong advantages in situations of high motion dynamics or—as shown in this work—challenging illumination conditions. Our motion estimation framework is divided into two sub-parts, a front-end module that initializes motion over shorter segments, and a back-end module that refines the estimate over larger-scale sequences. Both will be introduced after a short overview of the framework.

6.4.1 Framework overview

A flowchart of our proposed framework detailing all steps is illustrated in Figure 6.5. The front-end initialization module groups the events into sufficiently small subsets such that the motion on these subsets can be locally approximated using a simplified first-order constant velocity model. Furthermore, the front-end performs contrast

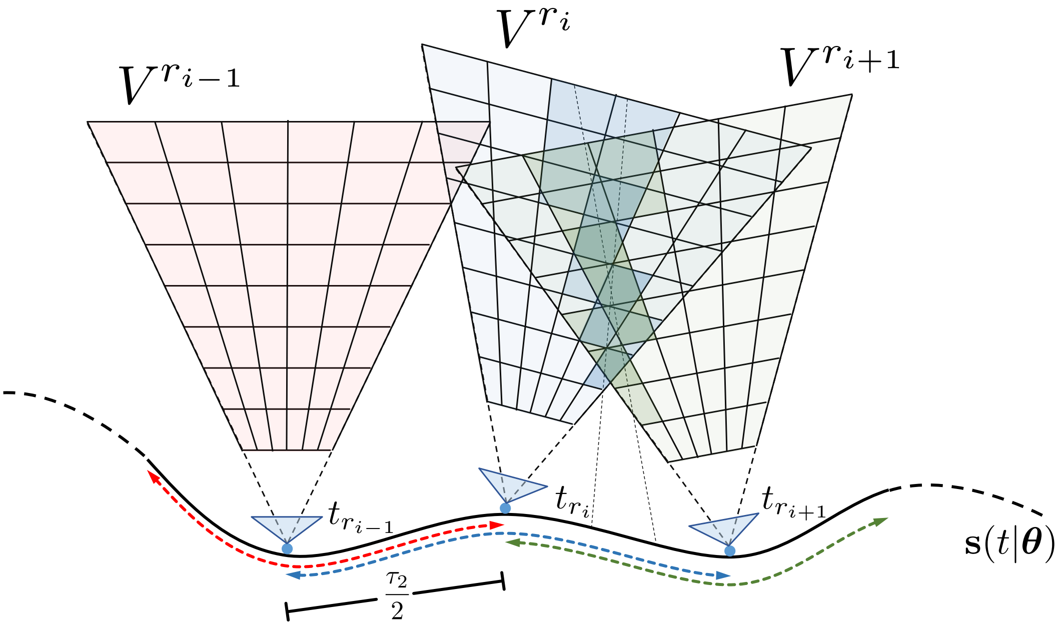


Figure 6.4: Global optimization over multiple reference volumes. The volumes may have spatial overlap. There is an individual time span $[t_{r_i} - \frac{1}{2}\tau_2 : t_{r_i} + \frac{1}{2}\tau_2]$ associated with each reference volume V^{r_i} from which events will be considered (marked by the red, blue and green arrows). The time-spans may have temporal overlap. As illustrated, two events may hence both appear in two distinct density fields, which reinforces scale propagation in the optimization.

maximization using a single VWE only. After a sufficient number of events and initial relative displacements have been accumulated, our method then proceeds to the back-end optimization part. The latter initializes a larger-scale, smooth, continuous-time trajectory model and executes the multi-volume optimization outlined in (6.11).

6.4.2 Front-end single-frame optimization

For the local approximation of the motion, we use a parametrization that is inspired by [Scaramuzza et al., 2009] and [Huang et al., 2019]. Based on the assumptions of a driftless, non-holonomic platform, and locally constant velocities, the continuous motion of the planar vehicle may be approximated to lie on an arc of a circle to which the heading of the vehicle remains tangential. This motion model is also known as the Ackermann motion model, and the centre of the circle is commonly referred to as the Instantaneous Centre of Rotation (ICR). The matter is illustrated in Figure 1.10 of Section 1.2.3. The model has only two degrees of freedom, which largely simplifies the geometry of the problem. It is given by the forward velocity v and the rotational velocity ω.

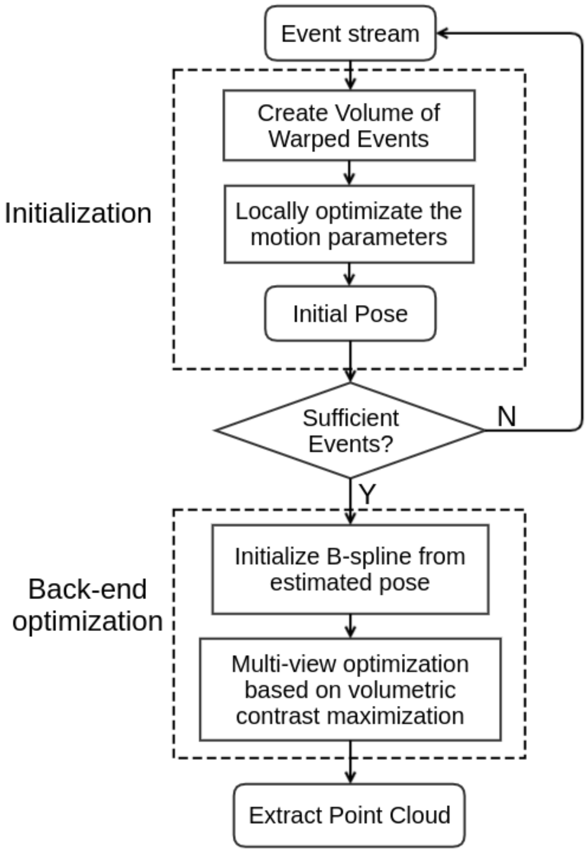

Event stream
Initialization
Create Volume of Warped Events
Locally optimizate the motion parameters
Initial Pose
Sufficient Events?
N
Y
Back-end optimization
Initialize B-spline from estimated pose
Multi-view optimization based on volumetric contrast maximization
Extract Point Cloud

Figure 6.5: Block diagram of our method containing both the front-end initialization and the back-end optimization part.

Using the convention and equations from [Huang et al., 2019] (y axis points forward, x axis to the right), it is straightforward to show that the relative transformation from a frame at time t_k to a nearby reference frame at time t_r is given by

$$\begin{aligned} \mathbf{R}_r^v(t_k|\boldsymbol{\theta}) &= \begin{bmatrix} \cos\omega(t_k - t_r) & -\sin\omega(t_k - t_r) & 0 \\ \sin\omega(t_k - t_r) & \cos\omega(t_k - t_r) & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ \mathbf{t}_r^v(t_k|\boldsymbol{\theta}) &= \frac{v}{\omega} \begin{bmatrix} 1 - \cos(\omega(t_k - t_r)) \\ \sin(\omega(t_k - t_r)) \\ 0 \end{bmatrix}. \end{aligned} \tag{6.13}$$

Given that scale is unobservable, we fix the forward velocity v to the configured speed of the vehicle (correct scale propagation is taken into account in the later global optimization scheme). As a result, the local motion initialization scheme over a single volume has only one degree of freedom, and the parameter vector becomes $\boldsymbol{\theta} = \omega$. Note furthermore that the original Ackermann model requires the camera to be mounted in the centre of the non-steering axis, which—in practice—hardly ever is the case. We therefore add the extrinsic calibration parameters $\mathbf{R}_{vc}$ and $\mathbf{t}_{vc}$ which transform points back-and-forth between the camera and the vehicle reference frames. Finally, the continuous parametrizations for the relative translation $\mathbf{t}_r(t_k|\boldsymbol{\theta})$ and rotation $\mathbf{R}_r(t_k|\boldsymbol{\theta})$ are given by

$$\begin{cases} \mathbf{R}_r(t_k|\boldsymbol{\theta}) = \mathbf{R}_{vc}^T \mathbf{R}_r^v(t_k|\boldsymbol{\theta}) \mathbf{R}_{vc} \\ \mathbf{t}_r(t_k|\boldsymbol{\theta}) = -\mathbf{R}_{vc}^T \mathbf{t}_{vc} + \mathbf{R}_{vc}^T \mathbf{t}_r^v(t_k|\boldsymbol{\theta}) + \mathbf{R}_{vc}^T \mathbf{R}_r^v(t_k|\boldsymbol{\theta}) \mathbf{t}_{vc}. \end{cases} \tag{6.14}$$

Figure 6.6 shows an example of local, volumetric contrast maximization on a local subset of the events captured by an event camera mounted on a non-holonomic platform. It illustrates the variance of the VWE as a function of our unique degree of freedom ω. As can be observed, the contrast of the VWE peaks at the vertical dashed line (red), which indicates the groundtruth angular velocity. This behavior is typical, and the motion parameters can be efficiently solved by local gradient-based optimization methods once a rough initial guess is given.

6.4.3 Back-end multi-frame optimization

The front-end obviously estimates the motion over short time periods, only, and furthermore relies on the approximation of locally constant velocities and a circular arc trajectory. We add a global back-end optimization over the entire trajectory which relies on a more general model for representing smooth planar motion. We use a two-dimensional, p-th degree B-spline curve

$$\mathbf{c}_{2\times 1}(t) = \sum_{i=0}^{n} N_{i,p}(t)\mathbf{p}_i, \qquad a \leq t \leq b, \tag{6.15}$$

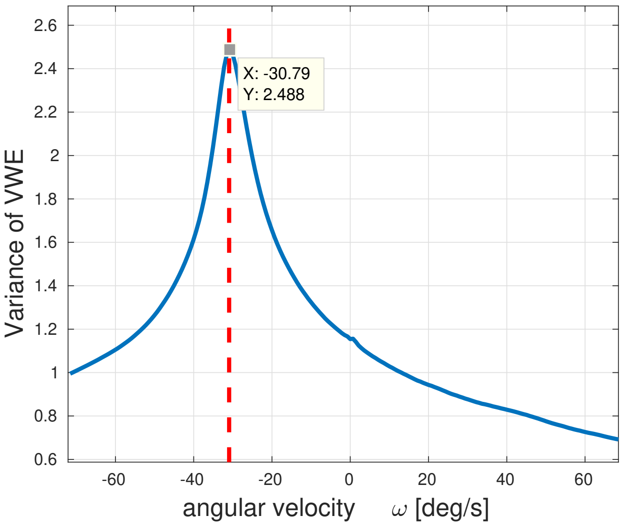


Figure 6.6: Contrast of VWE over angular velocity for events captured on a non-holonomic platform.

where the $\{\mathbf{p}_i\}$ stand for the $n+1$ two-dimensional control points that control the shape of the smooth, planar trajectory, and the $\{N_{i,p}(t)\}$ are the known pth-degree B-spline basis functions. The basic form of B-splines and their initialization are reviewed in Section 4.2.1 of Chapter 4, and readers are also invited to read up [Piegl and Tiller, 2012] and [Furgale et al., 2015] to see the more foundations of B-splines and an example application. Here we only focus on establishing the link to our smooth camera pose functions used in the optimization objective (6.11).

The parameter vector $\boldsymbol{\theta}$ may be defined as the stacked control points of the spline, i.e.

$$\boldsymbol{\theta} = [\mathbf{p}_0^T \ \dots \ \mathbf{p}_n^T]^T. \tag{6.16}$$

The spline directly models the position in the plane, so we easily obtain

$$\mathbf{t}^v(t|\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{c}(t) \\ 0 \end{bmatrix}. \tag{6.17}$$

For planar motion, the orientation is given by a pure rotation about the vertical axis, and we furthermore exploit the fact that for drift-less non-holonomic vehicles, the heading of the vehicle remains tangential to the trajectory. If the heading of the vehicle is still defined as the y axis, and the z axis points vertically upwards, the orientation of the vehicle is finally given as

$$\mathbf{R}^v(t|\boldsymbol{\theta}) = \begin{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \dot{\mathbf{c}}(t) & \dot{\mathbf{c}}(t) & \mathbf{0} \\ 0 & 0 & 1 \end{bmatrix}. \tag{6.18}$$

Note that only the temporal basis functions depend on time, and that $\dot{\mathbf{c}}(t)$ therefore also is a spline-based function of the same control points. The control point vector is initialized from the approximate trajectory given by the front end using the spline curve approximation given by the automatic knots spacing algorithms (9.68) and (9.69) of [Piegl and Tiller, 2012].

6.5 Implementation and validation

In this section, we briefly introduce implementation details of our method and then test our algorithm on multiple both synthetic and real datasets. We assess both accuracy and quality of the estimated trajectories, as well as the quality of the implicitly modelled 3D structure.

6.5.1 Implementation details

With respect to (6.9), we don't consider every event for every kernel, but ignore associations when the distance goes above a certain threshold. We utilize the event back-projection approach proposed in [Rebecq et al., 2018] to efficiently find the neighbouring voxels of a spatial ray. The details of this algorithm can be found in Section 7.1 of [Rebecq et al., 2018]. We furthermore use a simple gradient-ascent scheme to solve our volumetric contrast maximization problems. Especially in (6.13), the fixation of the forward velocity v leaves the angular velocity ω as the only unknown parameter, thus making the front-end constraint a uni-variate problem. Owing to the fact that the angular velocity of the vehicle's motion is relatively smooth, the starting point for the optimization can simply be obtained by propagation from a previous value. If no prior value is available, we simply find one through a Fibonacci search. Note that $\tau_1 = 0.08s$, and $\tau_2 = 2 \cdot \tau_1$.

In order to recover the implicitly modelled 3D structure of the environment, we simply reuse the Event-based Multi-View Stereo (EMVS) method from [Rebecq et al., 2018]. We extract the point cloud by first collapsing our VWE into a depth map with an associated 2D confidence map. The semi-dense depth map is then generated by adaptive thresholding of the confidence map, followed by conversion into a point cloud. Several noise filters are applied during the procedure.

6.5.2 Experiment setup

In order to demonstrate the performance of our algorithm, we apply it to both synthetic and real datasets. In the synthetic case, we use large-scale outdoor sequences from the KITTI benchmark [Geiger et al., 2013] and convert the image sequences into event data by using the method of Gehrig et al. [Gehrig et al., 2020]. The datasets are fully calibrated and contain images captured by a forward-looking camera mounted on a vehicle driving through a city. Experiments on real data are conducted by collecting several small-scale indoor sequences with a DAVIS346 event camera. The camera is mounted forward-facing on a turtlebot XQ-4 Pro. The camera has a resolution of 346×260. The output event stream has a maximum temporal resolution of 1μs. The camera is fully calibrated, and also captures regular frames at a frame rate of 30Hz. Implementations are made in C++, and use Eigen [Guennebaud et al., 2010] and Ceres [Agarwal et al., 2010] to solve the local and global optimization problems, respectively.

We compare our approach against traditional camera alternatives. Our current implementation focuses on non-holonomic planar motion, which is why we use the 1-point RANSAC algorithm for Ackermann motion [Scaramuzza et al., 2009] as a solid baseline algorithm for the regular camera alternative. We also let our method compete against an established alternative from the open-source community: ORB-SLAM [Mur-Artal and Tardós, 2017]. Note that we rescale all monocular, scale-invariant results to align as well as possible with groundtruth, which we obtain from the original KITTI datasets or an Opti-track system.

It should be noted that a direct comparison against alternative event-based VO/SLAM projects is difficult for several reasons. Til date, there are no open-source implementations and we are the first to even evaluate a monocular, event-based pipeline on a popular, established benchmark sequence. Furthermore, as stated in Section III. D of [Rebecq et al., 2016] and Section 3.5 of [Kim et al., 2016], the few existing alternatives either depend strongly on the quality of an initial 3D map (cf. [Rebecq et al., 2016]), or suffer from slowly converging depth estimates (cf. [Kim et al., 2016]). As shown in their experiments, they therefore require hovering motion in front of the same scene to provide sufficient time for the mapping back-end to converge. In contrast, our method performs joint optimization of trajectory and structure in near real-time, and thus successfully handles the continuous forward-exploration scenario.

6.5.3 Experiment on synthetic data

To prove the effectiveness of our method—which denote **ETAM**—, we apply it to synthetic sequences generated from the KITTI benchmark datasets [Geiger et al., 2013]. These datasets represent a fairly normal usecase without high motion dynamics or challenging illumination. We use the publicly available tool proposed by [Gehrig et al., 2020] to convert the regular videos into events streams. We compare our method against two alternatives, which is the state-of-the-art **ORB-SLAM** algorithm [Mur-Artal and Tardós, 2017] and the classical 1-point Ransac algorithm—denoted **1pt**—for planar motion [Scaramuzza et al., 2009]. The evaluation is performed on sequences *VO-00* and *VO-07*.

The qualitative performance is illustrated in Figure 6.7. All algorithms successfully process the sequences without any gross errors, and our system is slightly less accurate than **ORB-SLAM** on these high quality datasets. We furthermore believe that the decrease in performance is mostly explained by the approximate motion model, which ignores the slight pitch angle variations that could result from uneveness of the ground surface. Furthermore, we perform on par with **1pt**, which also relies on a non-holonomic planar motion model. To the best of our knowledge, this result is the first to demonstrate a monocular event camera solution that returns comparable results to regular camera alternatives.

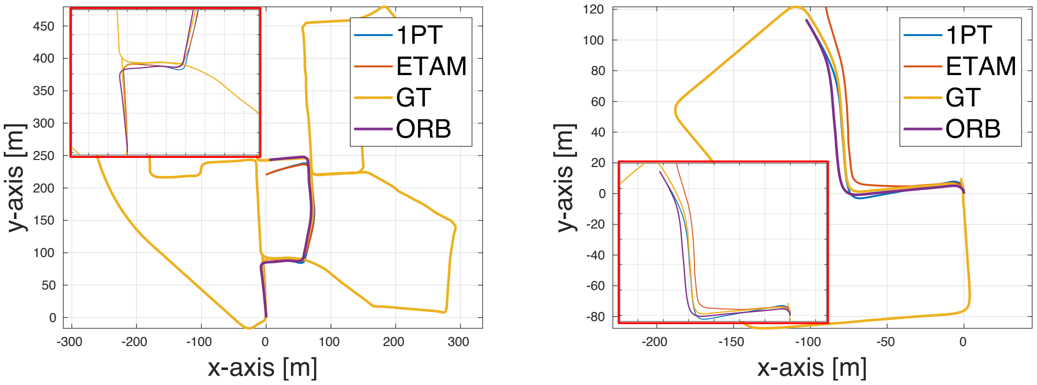


Figure 6.7: Results for both our method and regular camera alternatives on long outdoor trajectories

6.5.4 Experiment on real data

In order to demonstrate the performance of our algorithm on real data, we collect further sequences with a DAVIS346 event camera. The datasets are captured indoors to simulate different illumination conditions and capture groundtruth via Opti-track. We first apply them to two shorter sequences in which the camera follows an either circular (*Circle*) or purely translational trajectory (*Str*). Next, we perform a test on a much longer sequence with more complex motion (*Long2*). While the first three sequences are recorded under good illumination conditions, we conclude with another sequence with varying lighting conditions by toggling external illumination while the dataset is recorded (*HDR*).

ORB-SLAM proves to be fragile when applied to our indoor sequences. The images have low resolution and the proximity of the structure as well as fast vehicle rotations furthermore induce large frame-to-frame disparities, ultimately causing ORB-SLAM to break in such forward-exploration scenarios. We therefore assess the performance by a quantitative comparison of relative pose errors between **ETAM** and **1pt**. Results for all sequences are summarized in Table 6.1. It shows the root-mean-square or median of all deviations between estimated and groundtruth short-term relative rotation and translation displacements. Note that—in order to minimize the impact of unobservable scale—the error of the relative translation is evaluated by considering only the direction. Furthermore, errors are assessed per time, as it is clear that larger intervals may lead to more drift. We therefore employ the unit deg/s for both rotational and translational errors. The best performance is always highlighted in bold.

method	Circular motion			
	Rmse(**R**)	Median(**R**)	Rmse(**t**)	Median(**t**)
1pt	2.4526	2.3330	0.5427	**0.0296**
ETAM	**1.3275**	**0.6443**	**0.2826**	0.0322

method	Purely translational motion			
	Rmse(**R**)	Median(**R**)	Rmse(**t**)	Median(**t**)
1pt	**0.6997**	**0.5300**	0.5179	0.0369
ETAM	0.9769	0.6334	**0.4637**	**0.0235**

method	Long trajectory			
	Rmse(**R**)	Median(**R**)	Rmse(**t**)	Median(**t**)
1pt	1.8516	1.5829	0.1675	0.1718
ETAM	**1.6901**	**1.3417**	**0.1631**	**0.1703**

method	Challenging illumination conditions			
	Rmse(**R**)	Median(**R**)	Rmse(**t**)	Median(**t**)
1pt	-	-	-	-
ETAM	**1.6042**	**0.9093**	**0.0686**	**0.0084**

Table 6.1: Accuracy on different sequences. Unit: $[deg/s]$.

It can be easily observed that **ETAM** outperforms **1pt** on most datasets, and it is able to continuously track entire sequences with high accuracy even as illumination conditions become more challenging. In contrast, regular camera based visual odometry with 1-point RANSAC fails due to poor contrast or motion blur in dark or varying illumination settings (cf. Figure 6.8). Due to the forward-facing arrange-

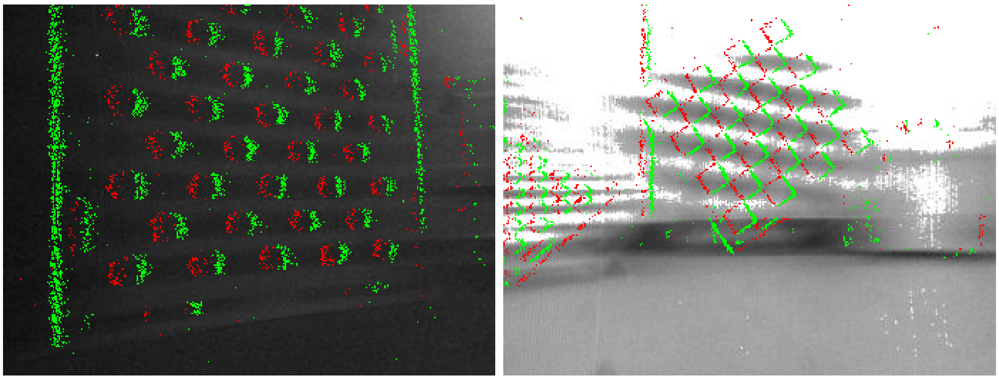


Figure 6.8: *Challenging illumination conditions.* Regular frames suffer from poor contrast or motion blur when lights are off (left) or on (right), respectively. Events in turn preserve the visual information of the structure.

ment, the purely translational displacement on the other hand triggers much fewer events than trajectories with rotational displacements, hence the slightly inferior performance for this type of motion.

The qualitative performance is illustrated in Figure 6.9, visualizing the angular velocity over time, dead-reckoned absolute orientations, and top views of complete trajectories for both algorithms and groundtruth (denoted **gt**). The figures in the left column are from the sequence *Long trajectory*, and the right ones are from the sequence *Challenging illumination conditions*. The absolute orientations and trajectories all suffer from slow error accumulation, and it can be observed that **ETAM** shows better performance than **1pt**. Our event-based method is able to work robustly in all challenging conditions.

6.5.5 Computational efficiency

All experiments are conducted on an Intel Core i7 2.4 GHz CPU. The total cumulative processing time for each sequence is summarized in Table 6.2. It remains below the actual length of each dataset, thus indicating real-time capability.

	Circle	*Str*	*Long1*	*Long2*	*HDR*
Dataset length	14.0s	10.4s	43.3s	40.4s	17.9s
Processing time	8.6s	8.5s	35.6s	24.9s	13.8s

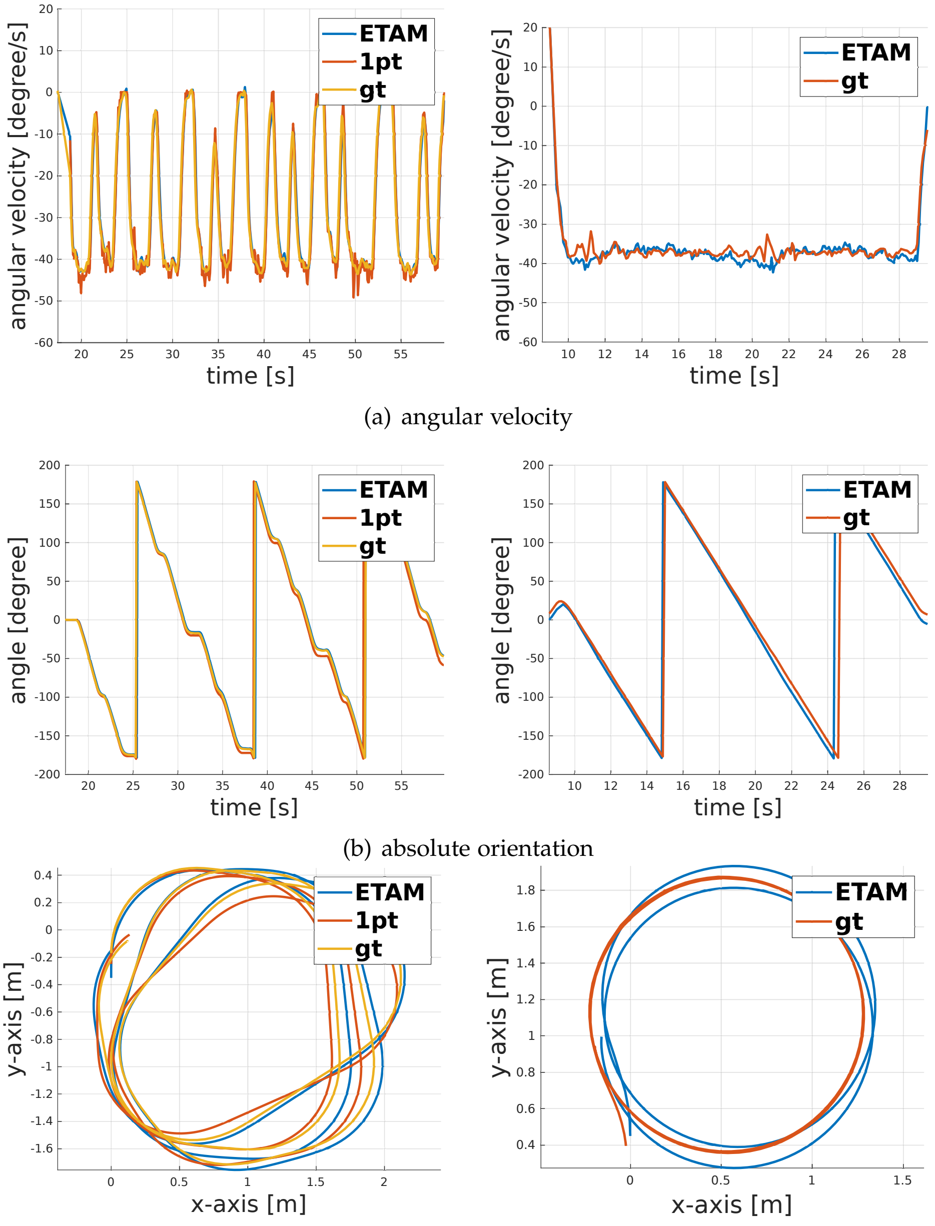
Table 6.2: Processing time in seconds for the proposed method.

6.5.6 Reconstruction result

Figures 6.10 and 6.11 finally visualize reconstruction results of the indoor scene. Figures 6.10(b) and 6.10(c) show a side perspective and a bird-eye view onto the final result. The coloured semi-dense points represent the reconstructed structure while the sparse white points in the centre denote the discretized trajectory. As can be clearly observed, our method produces a visually reasonable reconstruction similar to what one would obtain using a sparse or semi-dense method on regular images. Further visualisations of re-projected point clouds overlayed onto real images are visualized in Figures 6.11 (a) to (f). The depth of points is indicated by the colour, which reaches from red for closer points to blue for far-away points. Note that we clean up isolated noisy points by applying a radius filter. However, no additional depth fusion strategy is applied.

6.6 Discussion

Our main novelty consists of a single, joint objective that optimizes smooth motion directly from events, without the need of a prior derivation of 3D structure. This is achieved by constructing a volumetric ray density field, in which we then maximize contrast as a function of smooth motion parameters. As a result, the approach is able to bootstrap spatial motion in arbitrarily structured environments. The formulation is tested on the important application of ground vehicle motion estimation, and potential advantages in challenging illumination conditions are verified. While this is a highly promising result, our next step consists of extending the operation to more dynamic, full 3D motion, which we believe is possible if using the additional input of an IMU.

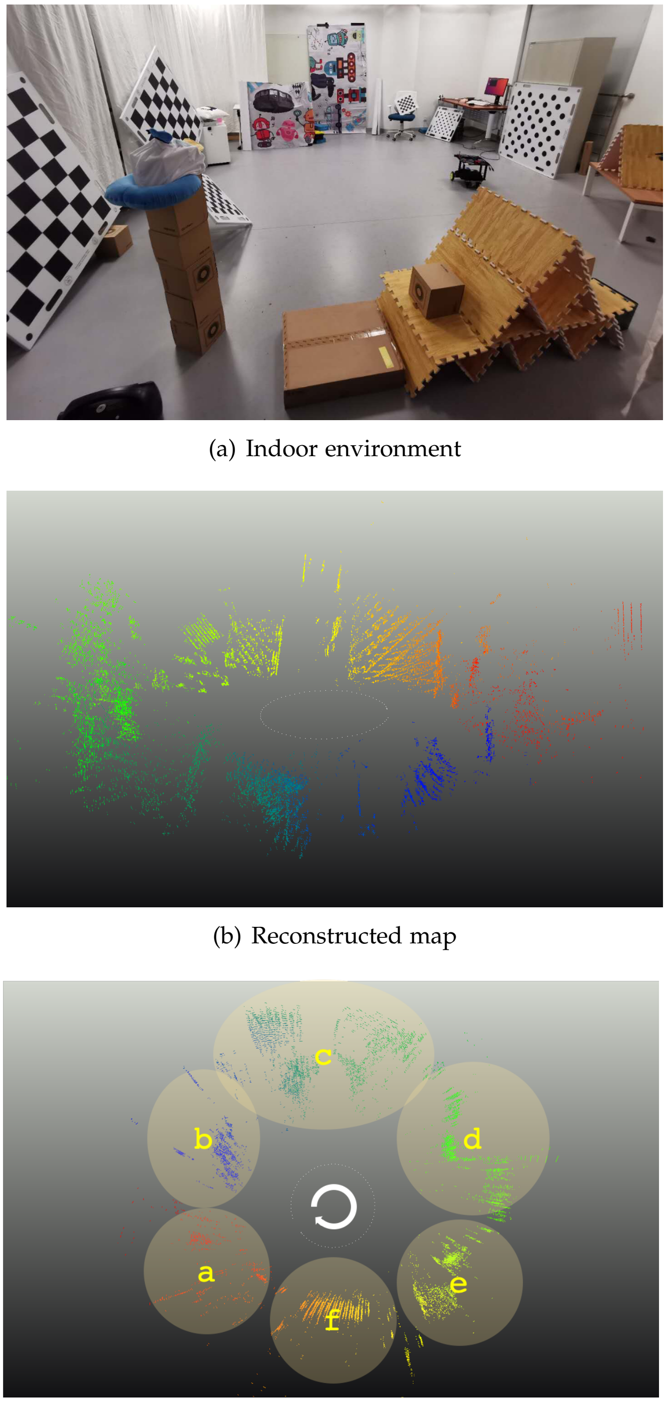


(a) angular velocity

(b) absolute orientation

(c) integrated trajectories

Figure 6.9: Results for both our method and 1pt-RANSAC on a long trajectory, indoor sequence (left) and under challenging illumination conditions (right).

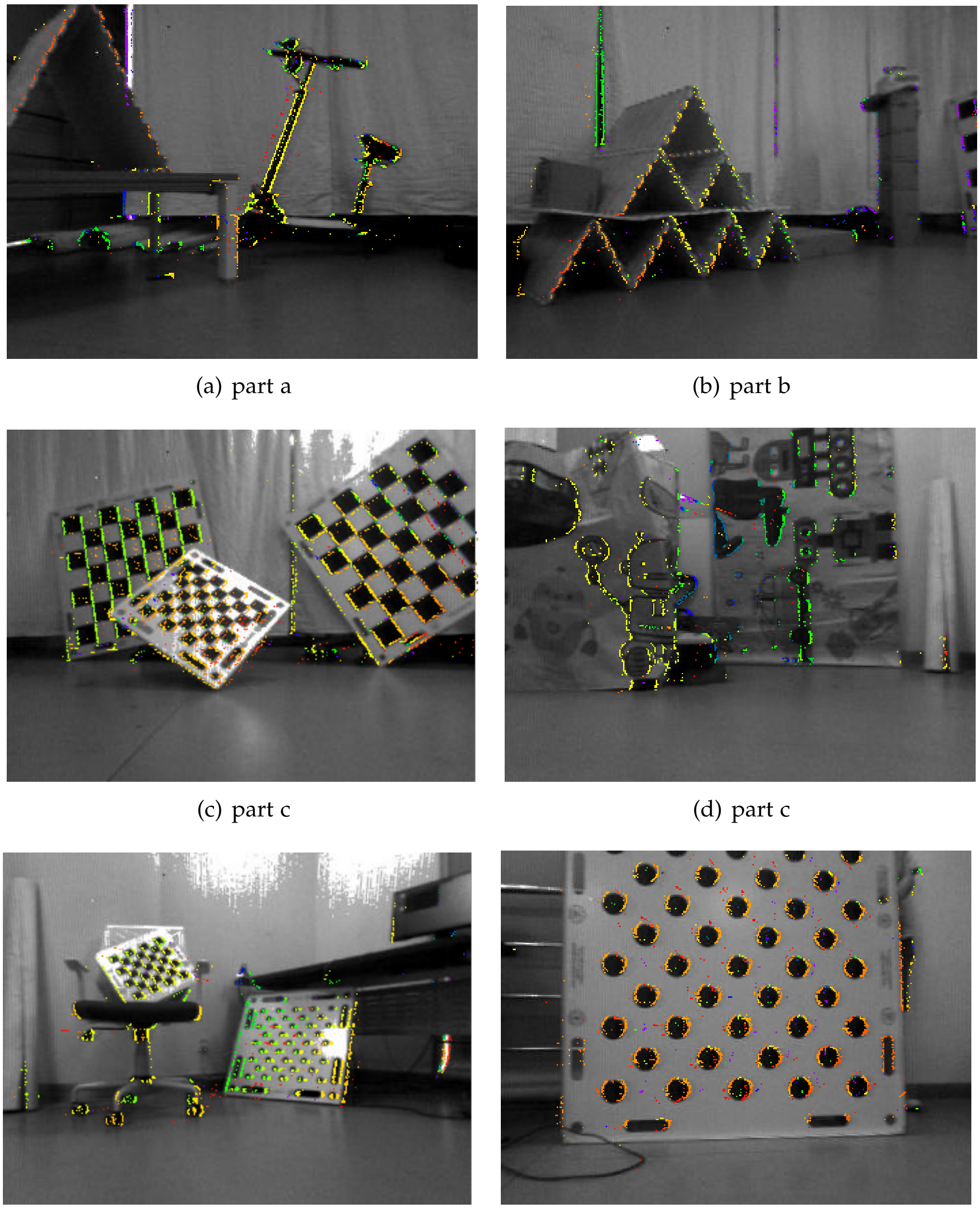


(a) Indoor environment

(b) Reconstructed map

(c) Topdown view

Figure 6.10: Reconstruction of an indoor scene. Figure (a) shows a real image of the environment. Figures (b) and (c) are different perspectives onto the reconstructed structure.



(a) part a

(b) part b

(c) part c

(d) part c

(e) part e

(f) part f

Figure 6.11: (a)-(f) are back-projections of the marked structure parts shown in Figure 6.10 overlaid onto the corresponding images captured under those poses. Warmer colors indicate closer points, while colder colors indicate larger depth.

Chapter 7

Conclusion

Over the past decade, automated real-time visual localization and mapping has often been proclaimed as a mature computer vision technology. However, it is only with the emergence of novel, billion-dollar industries such as autonomous driving, robotics, and mixed reality consumer products that this technology gets now put to a serious test. This prompts us to develop more reliable, efficient vision-based systems based on novel camera architectures.

7.1 Summary and contributions

The present thesis researches the development of several algorithms important for SLAM techniques based on novel camera architectures. The covered content reaches from an accurate, efficient visual odometry pipeline for multi-perspective systems to a robust back-end optimization method based on continuous-time trajectory parametrizations, as well as devising specialized algorithms for novel sensors namely event cameras, in order to deal with challenging scenarios. The present thesis is looking into discovering novel theoretical formulations and optimization algorithms that will gradually unlock the full potential of novel camera architectures. More specifically, this work addresses the following points.

7.1.1 Improving efficiency, accuracy and robustness

Visual odometry or visual SLAM encompasses a class of modular frameworks that crucially depend on a number of sub-solutions to be available. The present work has addressed several of these sub-problems:

- A complete real-time pipeline for visual odometry, as well as a robust scale initialization procedure for non-overlapping, multi-perspective camera systems are developed in Chapter 2. Our proposed pipeline differs from loosely-coupled alternatives, nearly all processing stages in our pipeline including bootstrapping, pose tracking, back-end optimization and mapping use the measurements from all cameras jointly, which is proven to have superior motion estimation accuracy with multi-perspective camera systems.

- Based on the idea of *Using many cameras as one*, in Chapter 3, this thesis has presented an efficient and reliable frame-to-frame motion estimation method specifically designed for surround-view, vehicle-mounted multi-camera systems, that the motion can be approximated to remain in a plane. It successfully solves one remaining problem in Chapter 2, which is how to robustly recover the relative rotations for generalized camera model at the very beginning, instead of estimating the relative rotations from each camera individually. The thesis presents a new univariate objective function by formulating the epipolar geometry as an eigenvalue problem, which can be effectively solved based on an iterative optimization scheme. The present work achieves not only competitive computational efficiency but also superior accuracy at the same time. The demo video can be found on Youtube[1].

- In Chapter 4, a robust back-end optimization scheme using B-splines for an exact imposition of smooth, non-holonomic trajectories inside the 6-DoF bundle adjustment is presented. While a number of successful VO/SLAM frameworks have already been presented, including the proposed ones in our previous chapters, they may come across challenges as the connectivity of the graph or the quality of the measurements degrades. Our method brings a significant improvement in robustness by taking non-holonomic kinematic constraints on the vehicle motion into account. The present work exploits B-spline as a representation for the kind of smooth trajectory, and evaluated a variety of formulations that imposing the kinematic constraints in the experiment. The potential of our method is finally confirmed on multiple publicly available datasets, which has demonstrated the improvements on both accuracy and robustness of monocular visual odometry.

7.1.2 Exploration of novel sensors

In order to deal with challenges in scenarios with high dynamics, low texture distinctiveness, or challenging illumination conditions that typically go beyond the limitation of regular cameras, a new camera architecture is investigated in this thesis. Event cameras are bio-inspired visual sensors which, unlike regular cameras, do not measure photometric intensities, but logarithmic, pixel-level, relative brightness changes. Data is furthermore returned asynchronously in quantized, pulse-like form; each pixel independently triggers an *event* whenever the above-mentioned, logarithmic quantity exceeds a certain threshold. Owing to their high temporal resolution of around 1μs, event cameras have very low latency and can capture very fast events. On the other hand, data in the form of events appears in an unconventional format, algorithms for event cameras are therefore needed to be specially devised.

- In Chapter 5, the present work looks at the relative pose estimation problem with a single event camera, which is used for the initialization or bootstrapping of a VO/SLAM framework if no prior knowledge about the environment is

[1] https://youtu.be/mtqFAzmh9E4

available. A globally optimal correspondence-less registration method based on contrast maximization framework is presented in our work, which primarily relies on the branch-and-bound optimization paradigm. The practical validity of our approach is supported by a highly successful application to AGV motion estimation with a downward-facing event camera. Our method outperforms regular cameras in such challenging scenarios where the sensor experiences fronto-parallel motion in front of noisy, fast moving textures.

- In Chapter 6, the present work concludes our thesis with a new solution to tracking and mapping with an event camera, by exploiting contrast maximization in 3D, which can jointly handle the localization and mapping problem of single event camera by continuous ray warping and volumetric contrast maximization in an arbitrarily structured environment. It overcomes one of the main weakness of image-warping based IWE contrast maximization, which is that IWE contrast maximization only able to handle a particular set of problems that can be represented by homographic warping or the depth parameters are known. The 3D location of the rays cast for each event is parametrized as a smooth, continuous function of time, and evaluate the ray density in a volume to find the optimal continuous motion parameters by maximizing its contrast. Our method is tested on ground vehicle motion estimation with a forward-facing event camera, it performs joint optimization of trajectory and structure in near real-time, and thus successfully handles the continuous forward-exploration scenario. Therefore our method could be considered as a very important innovation in event-based visual odometry.

7.2 Future Work

By the end of each chapter, we have concluded our contributions and discuss the potential improvements of our proposed algorithms. In addition, there are several indispensable extensions to our contributions will be discussed in the following.

7.2.1 Multi-perspective bundle adjustment using B-splines

As mentioned in Section 4.5, our B-spline based bundle adjustment is only tested on a monocular visual odometry framework, future research therefore consists of taking generalized camera system into account. Based on the multi-perspective window bundle adjustment scheme proposed in Section 2.2.5, we introduce the continuous-time model into multi-perspective bundle adjustments. The formulation (CBASpRv) which includes alternating spline regression and the *R-v* constraint has the best performance for graphs with stable connectivity.

We assume that we have a set of MPC keyframe $\mathcal{J}$ for which camera c_k observes the point $\mathbf{x}_i$. $\mathbf{t}_{b_j}$ and $\mathbf{q}_{b_j}$ are optimized pose parameters of MPC frame, and $\mathbf{R}(\mathbf{q})$ is the rotation matrix constructed from quaternion representation $\mathbf{q}$. The objective of novel bundle adjustment can now be reformulated as:

$$\min_{\substack{\{\mathbf{t}_{b_j}\}\{\mathbf{q}_{b_j}\} \\ \{\mathbf{x}_i\},\mathcal{P}}} \underbrace{\sum_{i,j}\sum_{k}\rho\left(\|f_p^{c_k}\left(\mathbf{T}_{c_k b}\begin{bmatrix}\mathbf{R}(\mathbf{q}_{b_j}) & \mathbf{t}_{b_j}\\ 0 & 1\end{bmatrix}^{-1}\mathbf{x}_i\right) - \mathbf{m}_{ij}^{c_k}\|^2\right)}_{\text{conventional bundle adjustment (CBA)}} \tag{7.1}$$

$$+\underbrace{\sum_j w_s\|\mathbf{t}_{b_j} - \mathbf{c}_1(t_j)\|^2}_{\text{smoothness constraint}} + \underbrace{\sum_j w_c\|\mathbf{R}(\mathbf{q}_{b_j})\begin{bmatrix}0\\1\\0\end{bmatrix} - \eta(\mathbf{c}_1'(t_j))\|^2}_{\text{R-v constraint}},$$

where

- $\{\mathbf{x}_i\}$ are landmarks in homogeneous representation.
- $\mathbf{m}_{ij}^{c_k}$ is the observation of landmark i from the respective camera frame c_k in MPC frame j.
- $\rho(\cdot)$ represents a loss function (e.g. Huber loss) to mitigate the influence of outliers.
- $f_p^{c_k}(\cdot)$ is the projection of $\{\mathbf{x}_i\}$ onto the image plane with calibrated camera c_k.
- $\mathbf{T}_{c_k b}$ represents the extrinsic parameters that transform observed points from the vehicle to the respective camera frame c_k.
- $\mathcal{P}$ is the set of control points of $\mathbf{c}_1(t)$.
- $\mathbf{c}_1(t)$ denotes the alternatingly updated spline.
- t_j is the timestamp at jth MPC frame.
- $\eta(\mathbf{a}) = \frac{\mathbf{a}}{\|\mathbf{a}\|}$.
- $\mathbf{c}'(t)$ denotes the first-order derivative of $\mathbf{c}(t)$.
- w_s, w_c are scalar weights.

The formulation of all aforementioned objective functions proposed in Chapter 4 will be derived in the future, and it would be interesting to test the developed back-end optimization module on aforementioned vehicle-mounted surround-view camera systems in practical cases.

7.2.2 Real-time 6-DoF Tracking and Mapping with an Event Camera

The work discussed in Chapter 6 is currently limited to planar AGV motion, and it is necessary to apply our concept to a 6-DoF scenario. Actually, the problem of full (potentially more agile) 6-DoF motion is significantly harder. Although we truly believe that our current work represents an important step in the right direction, this scenario requires further research which we currently conduct. To move one step beyond, we believe that it is important to adaptively sample space and thus enable the large scale optimization of longer trajectories. The intuition here lies in the fact that only appearance edges will trigger events, and that—as a result—the high-density regions in the final VWE are likely to align with the corresponding wire-frame structure. Our goal will be to identify this wire-frame structure to prune density estimation in uninteresting regions, and thus speed up the estimation by reducing the number of kernels. We will furthermore aim at adjusting the kernel centers and covariances for anisotropic weighting of the object space displacement vectors (i.e. displacements parallel to the 3D edge will be penalized less than displacements orthogonal to the 3D edge). Besides, we consider using the additional input of an IMU, which can provide an adequate initial guess on the relative displacement between subsequent views. To conclude, kernel covariances will be used to approximate gradient edges, information which can then be used in combination with event polarities to perform global localization in a given map. This estimation would naturally include the estimation of first-order dynamics.

7.2.3 Event-camera calibration

For the experiments conducted in Chapter 5 and 6, we notice that accuracy of intrinsic parameters is particularly important for VO/SLAM systems with event cameras. It directly affects whether our event-based methods can outperform regular cameras. In general, camera calibration is a crucial solution of which the quality will impact on the overall performance within the final application scenario, particularly if talking about accuracy in the context of a geometric problem.

Inspired by the continuous-time trajectory model used in Chapter 4 and 6, our principle involves the development of novel, practicable camera calibration methods that do not involve the need for specialized camera calibration targets with flashing LED lights used in existing approaches, but merely use a chess-board like pattern. While a static event camera will not be able to observe any passive calibration targets, it is clear that we will eventually start to perceive events along the high-gradient regions of the projected target as soon as there is relative motion between the target and the frame. Ignoring initialization questions, by knowing the exact calibration pattern, it therefore becomes possible to jointly optimize continuous motion parameters as well as intrinsic parameters by minimizing the distance between each event and its nearest reprojected edge. From a mathematical point of view, this problem may be formulated as follows.

We denote the event by $e_k = \{\mathbf{x}_k, t_k, b_k\}$, where $\mathbf{x}_k = [x_k \quad y_k]^T$ denotes the position of the event in the image, t_k the time-stamp of the event, and b_k its polarity (equals to 1 if the intensity at that pixel has increased, and to -1 if the intensity decreased).

We furthermore denote the camera pose by the transformation parameters from the target reference frame to the camera frame. It is given by the rotation matrix $\mathbf{R}(\boldsymbol{\theta}_{\mathbf{R}}(t))$ and the translation vector $\mathbf{t}(\boldsymbol{\theta}_{\mathbf{t}}(t))$. Note that both are dynamic and vary as a function of the minimal 6-DoF continuous-time trajectory parametrization

$$\boldsymbol{\theta}(t) = \begin{bmatrix} \mathbf{t}(\boldsymbol{\theta}_{\mathbf{t}}(t)) \\ \mathbf{R}(\boldsymbol{\theta}_{\mathbf{R}}(t)) \end{bmatrix}, \tag{7.2}$$

The warping function from the pattern to the image plane is therefore given by the planar homography

$$\mathbf{H}(t, \boldsymbol{\theta}(t), \mathbf{K}) = \mathbf{K} \begin{bmatrix} \mathbf{R}^1(\boldsymbol{\theta}_{\mathbf{R}}(t)) & \\ & + \mathbf{t}(\boldsymbol{\theta}_{\mathbf{t}}(t)) \\ \mathbf{R}^2(\boldsymbol{\theta}_{\mathbf{R}}(t)) & \end{bmatrix} \tag{7.3}$$

Note that $\mathbf{R}^1(\boldsymbol{\theta}_{\mathbf{R}}(t))$ and $\mathbf{R}^2(\boldsymbol{\theta}_{\mathbf{R}}(t))$ denote the first and second column of the rotation matrix $\mathbf{R}(\boldsymbol{\theta}_{\mathbf{R}}(t))$. To conclude, let us define the function $DF(\mathbf{x})$ (entered in homogeneous form) and the nearest edge in the pattern. The final cost function that permits the optimization of the intrinsic matrix $\mathbf{K}$ as well as the parameters $\boldsymbol{\theta}(t)$ of the motion is given by the sum of squared distances between the unwarped event locations (using the corresponding event time-stamp) and their nearest edge

$$E = \sum_{e_k} \{DF(\mathbf{H}^{-1}(t_k, \boldsymbol{\theta}(t_k), \mathbf{K})[x_k \quad y_k \quad 1]^T)\} \tag{7.4}$$

Note that rather than using a traditional chess-board pattern, it would be beneficial to use a pattern like the one presented in Figure 7.1, which would ensure that joint observation of events caused by orthogonal edges would continuously happen. As a result, the dynamic motion parameters would remain continuously observable.

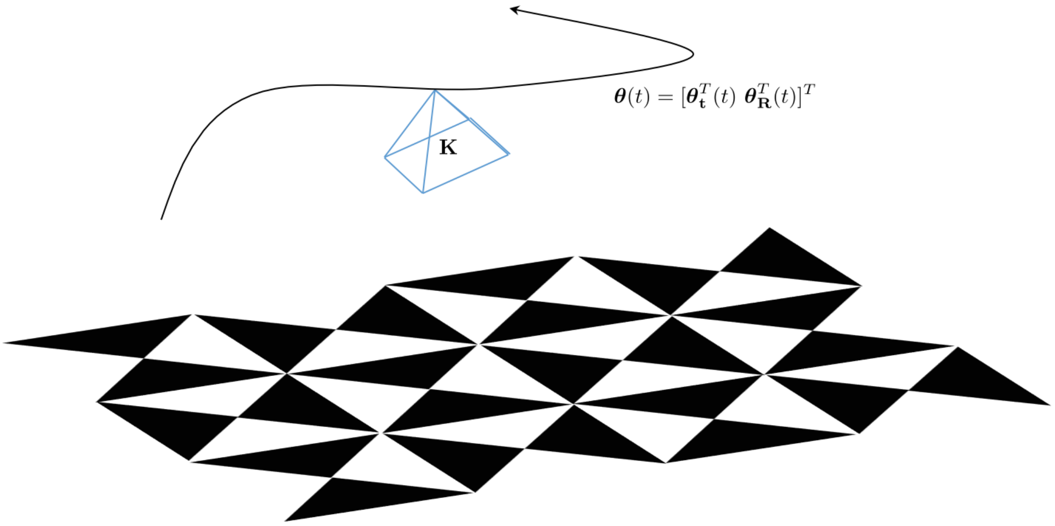

$\boldsymbol{\theta}(t) = [\boldsymbol{\theta}_{\mathbf{t}}^T(t)\ \boldsymbol{\theta}_{\mathbf{R}}^T(t)]^T$
K

Figure 7.1: Pattern, procedure and optimization variables for the proposed event camera calibration

Bibliography

AGARWAL, S.; MIERLE, K.; AND OTHERS, 2010. Ceres solver. http://ceres-solver.org. (cited on pages 31, 58, and 93)

AQEL, M. O.; MARHABAN, M. H.; SARIPAN, M. I.; AND ISMAIL, N. B., 2016. Review of visual odometry: types, approaches, challenges, and applications. *SpringerPlus*, 5, 1 (2016), 1897. (cited on page 73)

BIREM, M.; KLEIHORST, R.; AND EL-GHOUTI, N., 2018. Visual odometry based on the fourier transform using a monocular ground-facing camera. *Journal of Real-Time Image Processing*, 14, 3 (2018), 637–646. (cited on pages 73 and 74)

BOOIJ, O. AND ZIVKOVIC, Z., 2009. The planar two point algorithm. Technical Report IAS-UVA-09-05, University of Amsterdam. (cited on page 12)

BRADSKI, G., 2000. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, (2000). (cited on pages 31, 48, and 58)

BRYNER, S.; GALLEGO, G.; REBECQ, H.; AND SCARAMUZZA, D., 2019. Event-based, direct camera tracking from a photometric 3d map using nonlinear optimization. In *2019 International Conference on Robotics and Automation (ICRA)*, 325–331. IEEE. (cited on page 15)

BÜLOW, H. AND BIRK, A., 2009. Fast and robust photomapping with an unmanned aerial vehicle (uav). In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3368–3373. IEEE. (cited on pages 73 and 77)

CADENA, C.; CARLONE, L.; CARRILLO, H.; LATIF, Y.; SCARAMUZZA, D.; NEIRA, J.; REID, I.; AND LEONARD, J., 2016. Past, present, and future of simultaneous localization and mapping: Towards the robust-perception age. *IEEE Transactions on Robotics*, 32, 6 (2016), 1309–1332. (cited on page 3)

CAMPBELL, D.; PETERSSON, L.; KNEIP, L.; AND LI, H., 2017. Globally-optimal inlier set maximisation for simultaneous camera pose and feature correspondence. In *Proceedings of the IEEE International Conference on Computer Vision*, 1–10. (cited on page 71)

CAYLEY, A., 1846. About the algebraic structure of the orthogonal group and the other classical groups in a field of characteristic zero or a prime characteristic. *Reine Angewandte Mathematik*, 32 (1846). (cited on page 39)

Censi, A.; Franchi, A.; Marchionni, L.; and Oriolo, G., 2013. Simultaneous calibration of odometry and sensor parameters for mobile robots. *IEEE Transactions on Robotics (T-RO)*, 29, 2 (2013), 475–492. (cited on page 52)

Chen, X.; Vempati, A. S.; and Beardsley, P., 2018. Streetmap-mapping and localization on ground planes using a downward facing camera. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1672–1679. IEEE. (cited on page 73)

Civera, J.; Grasa, O.; Davison, A.; and Montiel, J., 2009. 1-point RANSAC for EKF-based structure from motion. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*. St. Louis, USA. (cited on page 3)

Clipp, B.; Kim, J.-H.; Frahm, J.-M.; Pollefeys, M.; and Hartley, R., 2008. Robust 6DOF Motion Estimation for Non-Overlapping, Multi-Camera Systems. In *Proceedings of the IEEE Workshop on Applications of Computer Vision*, 1–8. Washington, DC, USA. (cited on pages xix, 10, 22, 29, and 31)

Davison, A.; Reid, D.; Molton, D.; and Stasse, O., 2007. MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 26, 6 (2007), 1052–1067. (cited on pages xv, 3, and 4)

Dille, M.; Grocholsky, B.; and Singh, S., 2010. Outdoor downward-facing optical flow odometry with commodity sensors. In *Field and Service Robotics*, 183–193. Springer. (cited on pages 73 and 74)

Endres, F.; Hess, J.; Sturm, J.; Cremers, D.; and Burgard, W., 2013. 3-d mapping with an rgb-d camera. *IEEE transactions on robotics*, 30, 1 (2013), 177–187. (cited on page 8)

Engel, J.; Schöps, T.; and Cremers, D., 2014. LSD-SLAM: Large-scale direct monocular SLAM. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 834–849. (cited on pages xv, 3, and 6)

Fischler, M. and Bolles, R., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24, 6 (1981), 381–395. (cited on page 43)

Forster, C.; Pizzoli, M.; and Scaramuzza, D., 2014. Svo: Fast semi-direct monocular visual odometry. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*. (cited on pages xv, 5, 6, and 7)

Furgale, P.; Schwesinger, U.; Rufli, M.; Derendarz, W.; Grimmett, H.; Muhlfellner, P.; Wonneberger, S.; Li, B.; Schmidt, B.; Nguyen, T. N.; Cardarelli, E.; Cattani, S.; Brüning, S.; Horstmann, S.; Stellmacher, M.; Rottmann, S.; Mielenz, H.; Köser, K.; Timpner, J.; Beermann, M.; Häne, C.; Heng, L.; Lee, G. H.; Fraundorfer, F.; Iser, R.; Triebel, R.; Posner, I.; Newman, P.; Wolf, L.; Pollefeys, M.; Brosig, S.; Effertz, J.; Pradalier, C.; and Siegwart, R., 2013. Toward automated driving in cities using close-to-market sensors: an overview of the V-Charge project. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*. (cited on page 51)

Furgale, P.; Tong, C. H.; Barfoot, T. D.; and Sibley, G., 2015. Continuous-time batch trajectory estimation using temporal basis functions. *The International Journal of Robotics Research*, 34, 14 (2015), 1688–1710. (cited on pages 17, 52, 53, 83, and 91)

Gallego, G.; Delbruck, T.; Orchard, G. M.; Bartolozzi, C.; Taba, B.; Censi, A.; Leutenegger, S.; Davison, A.; Conradt, J.; Daniilidis, K.; and et al., 2020. Event-based vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2020), 1–1. doi:10.1109/tpami.2020.3008413. http://dx.doi.org/10.1109/TPAMI.2020.3008413. (cited on pages 9 and 68)

Gallego, G.; Gehrig, M.; and Scaramuzza, D., 2019. Focus is all you need: loss functions for event-based vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 12280–12289. (cited on pages xv, 15, 16, 68, 69, 76, and 83)

Gallego, G.; Lund, J. E.; Mueggler, E.; Rebecq, H.; Delbruck, T.; and Scaramuzza, D., 2017. Event-based, 6-dof camera tracking from photometric depth maps. *IEEE transactions on pattern analysis and machine intelligence*, 40, 10 (2017), 2402–2412. (cited on page 15)

Gallego, G.; Rebecq, H.; and Scaramuzza, D., 2018. A unifying contrast maximization framework for event cameras, with applications to motion, depth, and optical flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3867–3876. (cited on pages 15, 16, 68, 69, and 82)

Gallego, G. and Scaramuzza, D., 2017. Accurate angular velocity estimation with an event camera. *IEEE Robotics and Automation Letters*, 2, 2 (2017), 632–639. (cited on pages 16, 68, and 69)

Gao, L.; Su, J.; Cui, J.; Zeng, X.; Peng, X.; and Kneip, L., 2020. Efficient globally-optimal correspondence-less visual odometry for planar ground vehicles. In *2020 International Conference on Robotics and Automation (ICRA)*, 2696–2702. IEEE. (cited on pages xvii, 74, and 77)

Gehrig, D.; Gehrig, M.; Hidalgo-Carrió, J.; and Scaramuzza, D., 2020. Video to events: Recycling video datasets for event cameras. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*. (cited on pages 92 and 93)

Geiger, A.; Lenz, P.; Stiller, C.; and Urtasun, R., 2013. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, (2013). (cited on pages 12, 92, and 93)

Geiger, A.; Lenz, P.; and Urtasun, R., 2012. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (cited on pages 58 and 60)

Gellert, W.; Gottwald, S.; Hellwich, M.; Kästner, H.; and Künstner, H., 1989. *The VNR Concise Encyclopedia of Mathematics*. Van Nostrand Reinhold, New York, NY, USA, second edn. (cited on page 40)

Gonzalez, R.; Rodriguez, F.; Guzman, J. L.; Pradalier, C.; and Siegwart, R., 2012. Combined visual odometry and visual compass for off-road mobile robots localization. *Robotica*, 30, 6 (2012), 865–878. (cited on page 73)

Guennebaud, G.; Jacob, B.; et al., 2010. Eigen v3. http://eigen.tuxfamily.org. (cited on pages 31, 58, and 93)

Guo, X.; Xu, Z.; Lu, Y.; and Pang, Y., 2005. An application of fourier-mellin transform in image registration. In *The Fifth International Conference on Computer and Information Technology (CIT'05)*, 619–623. IEEE. (cited on page 73)

Handa, A.; Chli, M.; Strasdat, H.; and Davison, A., 2010. Scalable active matching. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. (cited on page 3)

Hartley, R., 1997. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 19 (1997), 580–?593. (cited on pages 36 and 45)

Hartley, R.; Trumpf, J.; Yuchao, D.; and Li, H., 2013. Rotation averaging. *International Journal of Computer Vision (IJCV)*, 103, 3 (2013), 267–305. (cited on page 25)

Hartley, R. and Zisserman, A., 2004. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, second edn. (cited on pages 27, 36, and 41)

Heng, L.; Choi, B.; Cui, Z.; Geppert, M.; Hu, S.; Kuan, B.; Liu, P.; Nguyen, R.; Yeo, Y. C.; Geiger, A.; Lee, G. H.; Pollefeys, M.; and Sattler, T., 2018. Project AutoVision: Localization and 3D scene perception for an autonomous vehicle with a multi-camera system. *arXiv*, 1809.05477 (2018). (cited on page 51)

Henry, P.; Krainin, M.; Herbst, E.; Ren, X.; and Fox, D., 2014. Rgb-d mapping: Using depth cameras for dense 3d modeling of indoor environments. In *Experimental robotics*, 477–491. Springer. (cited on pages xv and 8)

Horaud, R. and Dornaika, F., 1995. Hand-Eye Calibration. *International Journal of Robotics Research (IJRR)*, 14, 3 (1995), 195–210. (cited on page 21)

Howard, A., 2008. Real-time stereo visual odometry for autonomous ground vehicles. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*. Nice, France. (cited on page 51)

Huang, K.; Wang, Y.; and Kneip, L., 2019. Motion estimation of non-holonomic ground vehicles from a single feature correspondence measured over n views. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, USA. (cited on pages 12, 18, 35, 36, 52, 73, 89, and 90)

Jordan, J. and Zell, A., 2016. Ground plane based visual odometry for rgbd-cameras using orthogonal projection. *IFAC-PapersOnLine*, 49, 15 (2016), 108–113. (cited on page 73)

Kang, I. and Park, F., 1999. Cubic spline algorithms for orientation interpolation. *International Journal for Numerical Methods in Engineering*, 46, 1 (1999), 45–64. (cited on page 54)

Kang, R.; Xiong, L.; Xu, M.; Zhao, J.; and Zhang, P., 2019. Vins-vehicle: A tightly-coupled vehicle dynamics extension to visual-inertial state estimator. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 3593–3600. IEEE. (cited on page 52)

Kazik, T. and Göktoğan, A. H., 2011. Visual odometry based on the fourier-mellin transform for a rover using a monocular ground-facing camera. In *2011 IEEE International Conference on Mechatronics*, 469–474. IEEE. (cited on page 73)

Kazik, T.; Kneip, L.; Nikolic, J.; Pollefeys, M.; and Siegwart, R., 2012. Real-Time 6D Stereo Visual Odometry with Non-Overlapping Fields of View. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Providence, USA. (cited on pages xvi, xix, 21, 25, 29, 30, 31, and 32)

Kendall, A.; Grimes, M.; and Cipolla, R., 2015. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. (cited on page 7)

Kerl, C.; Sturm, J.; and Cremers, D., 2013. Robust odometry estimation for RGB-D cameras. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. (cited on page 8)

Kim, H.; Leutenegger, S.; and Davison, A. J., 2016. Real-time 3d reconstruction and 6-dof tracking with an event camera. In *European Conference on Computer Vision*, 349–364. Springer. (cited on pages 9, 82, and 93)

Kitt, B.; Geiger, A.; and Lategahn, H., 2010. Visual odometry based on stereo image sequences with RANSAC-based outlier rejection scheme. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*. (cited on page 51)

Klein, G. and Murray, D., 2007. Parallel tracking and mapping for small AR workspaces. In *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR)*. (cited on pages xv, 3, 4, and 24)

Kneip, L. and Furgale, P., 2014. OpenGV: A Unified and Generalized Approach to Real-Time Calibrated Geometric Vision. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. Hongkong. (cited on pages 12, 22, 31, 35, 36, 38, 39, 42, 45, and 46)

Kneip, L.; Furgale, P.; and Siegwart, R., 2013. Using Multi-Camera Systems in Robotics: Efficient Solutions to the NPnP Problem. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. Karlsruhe, Germany. (cited on pages 22, 25, 36, 37, 38, and 39)

Kneip, L. and Li, H., 2014. Efficient Computation of Relative Pose for Multi-Camera Systems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Columbus, USA. (cited on pages 22, 24, and 31)

Kneip, L.; Li, H.; and Seo, Y., 2014. UPnP: An Optimal O(n) Solution to the Absolute Pose Problem with Universal Applicability. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Zurich, Switzerland. (cited on pages 22 and 48)

Kneip, L. and Lynen, S., 2013. Direct Optimization of Frame-to-Frame Rotation. In *Proceedings of the International Conference on Computer Vision (ICCV)*. Sydney, Australia. (cited on page 10)

Kneip, L.; Scaramuzza, D.; and Siegwart, R., 2011. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Colorado Springs, USA. (cited on page 24)

Konolige, K.; Agrawal, M.; and Solà, J., 2007. Large scale visual odometry for rough terrain. In *Proceedings of the International Symposium on Robotics Research (ISRR)*. Hiroshima, Japan. (cited on pages 3 and 51)

Lacroix, S.; Mallet, A.; Chatila, R.; and Gallo, L., 1999. Rover self localization in planetary-like environments. In *Artificial Intelligence, Robotics and Automation in Space*. (cited on page 2)

Lee, G.; Pollefeys, M.; and Fraundorfer, F., 2014. Relative pose estimation for a multi-camera system with known vertical. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (cited on pages 36 and 38)

Lee, G. H.; Fraundorfer, F.; and Pollefeys, M., 2013. Motion estimation for a self-driving car with a generalized camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Portland, USA. (cited on pages 12, 35, 36, 44, and 52)

Levinson, J.; Montemerlo, M.; and Thrun, S., 2007. Map-based precision vehicle localization in urban environments. In *Proceedings of Robotics: Science and Systems (RSS)*, vol. 4, 1. (cited on page 51)

LI, H.; HARTLEY, R.; AND KIM, J.-H., 2008. A Linear Approach to Motion Estimation using Generalized Camera Models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–8. Anchorage, Alaska, USA. (cited on pages 12, 22, 35, 36, 38, 45, and 46)

LI, P.; QIN, T.; ET AL., 2018. Stereo vision-based semantic 3d object and ego-motion tracking for autonomous driving. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 646–661. (cited on pages 52 and 56)

LOVEGROVE, S.; DAVISON, A. J.; AND IBANEZ-GUZMÁN, J., 2011. Accurate visual odometry from a rear parking camera. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, 788–793. IEEE. (cited on page 73)

LUNDBERG, M., 2017. *Path planning for autonomous vehicles using clothoid based smoothing of A* generated paths and optimal control*. Ph.D. thesis, KTH Royal Institute of Technology, School of Engineering Sciences. (cited on page 52)

LV, W.; KANG, Y.; AND QIN, J., 2017. Indoor localization for skid-steering mobile robot by fusing encoder, gyroscope, and magnetometer. *IEEE Transactions on Systems, Man, and Cybernetics (SMC)*, 99 (2017), 1–13. (cited on page 52)

MA, Y.; SOATTO, S.; KOSECKA, J.; AND SASTRY, S. S., 2012. *An invitation to 3-d vision: from images to geometric models*, vol. 26. Springer Science & Business Media. (cited on page 58)

MARTINEZ, J. L.; MANDOW, A.; MORALES, J.; PEDRAZA, S.; AND GARCIA-CEREZO, A., 2005. Approximating kinematics for tracked mobile robots. *International Journal of Robotics Research (IJRR)*, 24, 10 (2005), 867–878. (cited on page 52)

MARTINEZ, J. L.; MORALES, J.; MANDOW, A.; PEDRAZA, S.; AND GARCIA-CEREZO, A., 2017. Inertia-based ICR kinematic model for tracked skid-steer robots. In *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 166–171. (cited on page 52)

MATTHIES, L. AND SHAFER, S. A., 1986. Error modelling in stereo navigation. In *Fall Joint Computer Conference*. (cited on page 2)

MITROKHIN, A.; FERMÜLLER, C.; PARAMESHWARA, C.; AND ALOIMONOS, Y., 2018. Event-based moving object detection and tracking. in 2018 ieee. In *RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1–9. (cited on pages 16 and 68)

MORAVEC, H., 1980. *Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover*. Ph.D. thesis, Stanford University. (cited on page 2)

MUEGGLER, E.; GALLEGO, G.; AND SCARAMUZZA, D., 2015. Continuous-time trajectory estimation for event-based vision sensors. In *Robotics: Science and Systems*. (cited on page 15)

Mur-Artal, R.; Montiel, J. M. M.; and Tardós, J. D., 2015. ORB-SLAM: a versatile and accurate monocular slam system. *IEEE Transactions on Robotics (T-RO)*, 31, 5 (2015), 1147–1163. (cited on pages 3 and 5)

Mur-Artal, R. and Tardós, J. D., 2017. ORB-SLAM2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33, 5 (2017), 1255–1262. (cited on pages 58, 61, 88, and 93)

Newcombe, R. A.; Izadi, S.; Hilliges, O.; Molyneaux, D.; Kim, D.; Davison, A. J.; Kohli, P.; Shotton, J.; Hodges, S.; and Fitzgibbon, A., 2011a. Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR)*. (cited on page 3)

Newcombe, R. A.; Lovegrove, S. J.; and Davison, A. J., 2011b. DTAM: Dense tracking and mapping in real-time. In *Proceedings of the International Conference on Computer Vision (ICCV)*. (cited on pages 3 and 5)

Nistér, D.; Naroditsky, O.; and Bergen, J., 2004. Visual odometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 652–659. Washington, DC, USA. (cited on pages 2, 3, and 88)

Nistér, D.; Naroditsky, O.; and Bergen, J., 2006. Visual odometry for ground vehicle applications. *Journal of Field Robotics (JFR)*, 23, 1 (2006), 3–20. Inaugural issue. (cited on page 51)

Nistér, D. and Stewénius, H., 2006. A minimal solution to the generalized 3-point pose problem. *Journal of Mathematical Imaging and Vision (JMIV)*, 27, 1 (2006), 67–79. (cited on pages 22 and 27)

Nourani-Vatani, N. and Borges, P. V. K., 2011. Correlation-based visual odometry for ground vehicles. *Journal of Field Robotics*, 28, 5 (2011), 742–768. (cited on page 73)

Nourani-Vatani, N.; Roberts, J.; and Srinivasan, M. V., 2009. Practical visual odometry for car-like vehicles. In *2009 IEEE International Conference on Robotics and Automation*, 3551–3557. IEEE. (cited on page 73)

Olson, C. F.; Matthies, L. H.; Schoppers, H.; and Maimone, M. W., 2000. Robust stereo ego-motion for long distance navigation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (cited on page 2)

Peng, X.; Cui, J.; and Kneip, L., 2019. Articulated multi-perspective cameras and their application to truck motion estimation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2052–2059. IEEE. (cited on pages 55 and 73)

Peng, X.; Gao, L.; Wang, Y.; and Kneip, L., 2021. Globally-optimal contrast maximisation for event cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2021), 1–1. doi:10.1109/TPAMI.2021.3053243. (cited on page 19)

Peng, X.; Wang, Y.; Gao, L.; and Kneip, L., 2020. Globally-optimal event camera motion estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. (cited on page 18)

Piegl, L. and Tiller, W., 2012. *The NURBS book*. Springer Science & Business Media. (cited on pages 53, 54, 91, and 92)

Piyathilaka, L. and Munasinghe, R., 2010. An experimental study on using visual odometry for short-run self localization of field robot. In *2010 Fifth International Conference on Information and Automation for Sustainability*, 150–155. IEEE. (cited on page 73)

Pless, R., 2003. Using many cameras as one. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 587–593. Madison, WI, USA. (cited on pages 22 and 36)

Quan, M.; Piao, S.; Tan, M.; and Huang, S.-S., 2018. Tightly-coupled Monocular Visual-odometric SLAM using Wheels and a MEMS Gyroscope. *arXiv*, 1804.04854 (2018). (cited on page 52)

Rebecq, H.; Gallego, G.; Mueggler, E.; and Scaramuzza, D., 2018. Emvs: Event-based multi-view stereo—3d reconstruction with an event camera in real-time. *International Journal of Computer Vision*, 126, 12 (2018), 1394–1414. (cited on pages 9, 16, 68, 82, 84, and 92)

Rebecq, H.; Horstschäfer, T.; Gallego, G.; and Scaramuzza, D., 2016. Evo: A geometric approach to event-based 6-dof parallel tracking and mapping in real time. *IEEE Robotics and Automation Letters*, 2, 2 (2016), 593–600. (cited on pages 9, 82, 84, and 93)

Reverter Valeiras, D.; Kime, S.; Ieng, S.-H.; and Benosman, R. B., 2016. An event-based solution to the perspective-n-point problem. *Frontiers in neuroscience*, 10 (2016), 208. (cited on page 15)

Rublee, E.; Rabaud, V.; Konolige, K.; and Bradski, G., 2011. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, 2564–2571. Ieee. (cited on pages xv, 4, 5, and 60)

Salas-Moreno, R. F.; Newcombe, R. A.; Strasdat, H.; Kelly, P. H. J.; and Davison, A. J., 2013. Slam++: Simultaneous localisation and mapping at the level of objects. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 1352–1359. (cited on page 7)

Scaramuzza, D., 2011. 1-point-ransac structure from motion for vehicle-mounted cameras by exploiting non-holonomic constraints. *International Journal of Computer Vision (IJCV)*, 95, 1 (2011), 74–85. (cited on pages 52 and 60)

SCARAMUZZA, D., 2020. Tutorial on event-based cameras. http://rpg.ifi.uzh.ch/docs/scaramuzza/Tutorial_on_Event_Cameras_Scaramuzza.pdf. (cited on pages xv, 9, 14, and 15)

SCARAMUZZA, D.; FRAUNDORFER, F.; AND SIEGWART, R., 2009. Real-time monocular visual odometry for on-road vehicles with 1-point ransac. *2009 IEEE International Conference on Robotics and Automation*, (2009), 4293–4299. (cited on pages 12, 35, 36, 44, 52, 54, 55, 89, and 93)

SOUÉRES, P. AND BOISSONNAT, J. D., 1998. Optimal trajectories for nonholonomic mobile robots. In *Robot Motion Planning and Control* (Ed. J.-P. LAUMOND), chap. 3, 93–166. Springer. (cited on page 52)

STEINBRÜCKER, F.; STURM, J.; AND CREMERS, D., 2011. Real-time visual odometry from dense RGB-D images. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*. (cited on pages 3 and 8)

STEWÉNIUS, H.; ENGELS, C.; AND NISTÉR, D., 2006. Recent developments on direct relative orientation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60, 4 (2006), 284–294. (cited on pages 24 and 36)

STEWÉNIUS, H. AND NISTÉR, D., 2005. Solutions to Minimal Generalized Relative Pose Problems. In *Workshop on Omnidirectional Vision (ICCV)*. Beijing, China. (cited on pages 12, 22, and 35)

STOFFREGEN, T.; GALLEGO, G.; DRUMMOND, T.; KLEEMAN, L.; AND SCARAMUZZA, D., 2019. Event-based motion segmentation by motion compensation. In *Proceedings of the IEEE International Conference on Computer Vision*, 7244–7253. (cited on pages 16 and 68)

STOFFREGEN, T. AND KLEEMAN, L., 2017. Simultaneous optical flow and segmentation (sofas) using dynamic vision sensor. *2017 Australasian Conference on Robotics and Automation (ACRA)*, (2017), 52–61. (cited on pages 16 and 68)

STOFFREGEN, T. AND KLEEMAN, L., 2019. Event cameras, contrast maximization and reward functions: an analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 12300–12308. (cited on pages 16, 68, 69, and 82)

STRASDAT, H. AND DAVISON, A., 2010. Scale Drift-Aware Large Scale Monocular SLAM. In *Proceedings of Robotics: Science and Systems (RSS)*. Zaragoza, Spain. (cited on page 3)

STURM, J.; ENGELHARD, N.; ENDRES, F.; BURGARD, W.; AND CREMERS, D., 2012. A benchmark for the evaluation of RGB-D SLAM systems. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*. Vilamoura-Algarve, Portugal. (cited on page 58)

Sünderhauf, N.; Pham, T. T.; Latif, Y.; Milford, M.; and Reid, I., 2017. Meaningful maps with object-oriented semantic mapping. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5079–5085. (cited on page 7)

Tateno, K.; Tombari, F.; Laina, I.; and Navab, N., 2017. Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. (cited on pages xv and 7)

Tykkälä, T.; Audras, C.; and Comport, A. I., 2011. Direct iterative closest point for real-time visual odometry. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*. (cited on page 3)

Wan, G.; Yang, X.; Cai, R.; Li, H.; Zhou, Y.; Wang, H.; and Song, S., 2018. Robust and precise vehicle localization based on multisensor fusion in diverse city scenes. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 4670–4677. (cited on page 51)

Wang, Y.; Huang, K.; Peng, X.; Li, H.; and Kneip, L., 2020. Reliable frame-to-frame motion estimation for vehicle-mounted surround-view camera systems. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 1660–1666. IEEE. (cited on pages 18 and 73)

Wang, Y. and Kneip, L., 2017. On scale initialization in non-overlapping multi-perspective visual odometry. In *International Conference on Computer Vision Systems*, 144–157. Springer. (cited on page 18)

Whelan, T.; Johannsson, H.; Kaess, M.; Leonard, J. J.; and McDonald, J., 2013. Robust real-time visual odometry for dense rgb-d mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. (cited on page 3)

Wu, K. J.; Guo, C. X.; Georgiou, G.; and Roumeliotis, S. I., 2017. VINS on wheels. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 5155–5162. (cited on page 52)

Xu, Q.; Chavez, A. G.; Bülow, H.; Birk, A.; and Schwertfeger, S., 2019. Improved fourier mellin invariant for robust rotation estimation with omni-cameras. In *2019 IEEE International Conference on Image Processing (ICIP)*, 320–324. IEEE. (cited on pages 73 and 77)

Yap, T.; Li, M.; Mourikis, A. I.; and Shelton, C. R., 2011. A particle filter for monocular vision aided odometry. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 5663–5669. (cited on page 52)

Ye, C.; Mitrokhin, A.; Parameshwara, C.; Fermüller, C.; Yorke, J. A.; and Aloimonos, Y., 2018. Unsupervised learning of dense optical flow and depth from sparse event data. *CoRR*, abs/1809.08625 (2018). http://arxiv.org/abs/1809.08625. (cited on pages 16 and 68)

Yi, J.; Wang, H.; Zhang, J.; Song, D.; Jayasuriya, S.; and Liu, J., 2009. Kinematic modelling and analysis of skid-steered mobile robots with applications to low-cost inertial-measurement unit-based motion estimation. *IEEE Transactions on Robotics (T-RO)*, 25, 5 (2009). (cited on page 52)

Yi, K.; Trulls, E.; Lepetit, V.; and Fua, P., 2016. Lift: Learned invariant feature transform. In *Proceedings of the European Conference on Computer Vision (ECCV)*, vol. 9910, 467–483. doi:10.1007/978-3-319-46466-4_28. (cited on page 7)

Yu, Y.; Pradalier, C.; and Zong, G., 2011. Appearance-based monocular visual odometry for ground vehicles. In *2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 862–867. IEEE. (cited on page 73)

Zhang, M.; Zuo, X.; Chen, Y.; and Li, M., 2019. Localization for ground robots: On manifold representation, integration, re-parameterization, and optimization. *arXiv*, 1909.03423 (2019). (cited on page 52)

Zhu, A. Z.; Atanasov, N.; and Daniilidis, K., 2017. Event-based feature tracking with probabilistic data association. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 4465–4470. IEEE. (cited on pages 16 and 68)

Zhu, A. Z.; Chen, Y.; and Daniilidis, K., 2018a. Realtime time synchronized event-based stereo. In *European Conference on Computer Vision*, 438–452. Springer. (cited on pages 16 and 68)

Zhu, A. Z.; Yuan, L.; Chaney, K.; and Daniilidis, K., 2018b. Ev-flownet: Self-supervised optical flow estimation for event-based cameras. *arXiv preprint arXiv:1802.06898*, (2018). (cited on pages 16 and 68)

Zhu, A. Z.; Yuan, L.; Chaney, K.; and Daniilidis, K., 2019a. Unsupervised event-based learning of optical flow, depth, and egomotion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 989–997. (cited on pages 16 and 68)

Zhu, D.; Xu, Z.; Dong, J.; Ye, C.; Hu, Y.; Su, H.; Liu, Z.; and Chen, G., 2019b. Neuromorphic visual odometry system for intelligent vehicle application with bio-inspired vision sensor. In *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2225–2232. IEEE. (cited on page 82)

Zienkiewicz, J. and Davison, A., 2015. Extrinsics autocalibration for dense planar visual odometry. *Journal of Field Robotics*, 32, 5 (2015), 803–825. (cited on page 73)

Zihao Zhu, A.; Atanasov, N.; and Daniilidis, K., 2017. Event-based visual inertial odometry. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5391–5399. (cited on page 9)

Zong, W.; Chen, L.; Zhang, C.; Wang, Z.; and Chen, Q., 2017. Vehicle model based visual-tag monocular ORB-SLAM. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 1441–1446. IEEE. (cited on pages 52 and 56)