CZECH TECHNICAL UNIVERSITY IN PRAGUE

Faculty of Electrical Engineering

# DIPLOMA THESIS

Jiří Ulrich

## Fiducial Marker-Based Multiple Camera Localisation System

**Department of Computer Science**

Thesis supervisor: **doc. Ing. Tomáš Krajník, Ph.D.**

May, 2022

# ZADÁNÍ DIPLOMOVÉ PRÁCE

## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Ulrich** Jméno: **Jiří** Osobní číslo: **474426**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra počítačů**

Studijní program: **Otevřená informatika**

Specializace: **Umělá inteligence**

## II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

**Multikamerový lokalizační systém založený na detekci černobílých vzorů**

Název diplomové práce anglicky:

**Fiducial marker-based multiple camera localisation system**

Pokyny pro vypracování:

1. Learn principles of multi-camera geometry.
2. Learn about fiducial markers used in the mobile robotics.
3. Propose a set of key performance criteria to assess the markers' performance in relevant mobile robotics scenarios, where accurate 6DOF estimation over large-scale areas is required.
4. Evaluate the performance of the selected markers in a single-camera configuration.
5. Choose an appropriate marker detection method and design and implement its extension for multi-camera configurations.
6. Using the key performance criteria set in (3), evaluate the impact of the implemented extension.

Seznam doporučené literatury:

[1] T Krajník, et al.: A practical multirobot localization system. JINT 2014.
[2] P Lightbody, T Krajník, M Hanheide.: An efficient visual fiducial localisation system. ACM SIGAPP Applied Computing Review, 2017.
[3] Kim J. el at.: A new camera calibration method for robotic applications.
[4] Hartley, R.I. and Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge Uni press, 2004.

Jméno a pracoviště vedoucí(ho) diplomové práce:

**doc. Ing. Tomáš Krajník, Ph.D. centrum umělé inteligence FEL**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **28.01.2022** Termín odevzdání diplomové práce: ______________

Platnost zadání diplomové práce: **30.09.2023**

| | | |
|---|---|---|
| doc. Ing. Tomáš Krajník, Ph.D.<br>podpis vedoucí(ho) práce | podpis vedoucí(ho) ústavu/katedry | prof. Mgr. Petr Páta, Ph.D.<br>podpis děkana(ky) |

## III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

.

| | |
|---|---|
| Datum převzetí zadání | Podpis studenta |

## Declaration

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických princip ů při přípravě vysokoškolských závěrečných prací.

Prague, date May 20th, 2022 Jiří Ulrich

# Acknowledgement

I want to thank everyone that supported me during my work on the thesis. Special thanks to my supervisor Tomáš Krajník for the opportunity to choose such an exciting topic and learn new and challenging aspects of various disciplines. Thanks should also go to Tomáš Vintr for the moral support and endless encouragement to continue working. I am also grateful to Jan Blaha for the brighter moments during the work on the thesis. Many thanks to Kateřina Slabá for standing by me during the whole process. Also, I could not have undertaken this journey without all my friends and colleagues from the Chronorobotics Laboratory. The work was partially supported by the Czech Science Foundation project 17-27006Y and the EU FET Open programme under grant agreement No.964492 project 'RoboRoyale'.

## *Abstrakt*

V této práci je představen multikamerový lokalizační systém založený na referenčních značkách společně s upraveným algoritmem pro detekci černobílých kruhových referenčních značek. Představený lokalizační systém vychází z jednokamerového systému pro detekci referenčních značek v reálném čase s určením jejich pozice a orientace v 3D prostoru a rozpoznatelným variabilním kódováním identifikátorů. Tento multikamerový lokalizační systém postavený na levných a široce dostupných webových kamerách představuje masově dostupný lokalizační systém jako alternativu k aktuálně dostupným high-end systémům se specializovaným hardwarem a zdlouhavým nasazením. Za účelem posouzení výkonu je lokalizační systém testován proti původní jednokamerové metodě široce používané v oblasti mobilní a rojové robotiky. Vytvořili jsme simulovaná testovací prostředí umožňující dynamicky měnitelné simulované scénáře a také jsme shromáždili datovou sadu zachycující nasazení systému pro sledování pozice mobilního robotu při autonomní navigaci v exteriérech. Pro lepší správu testů byl připraven automatický evaluační a simulační nástroj.

## *Abstract*

In this thesis, the multi-camera localisation system based on fiducial markers is presented together with a modified algorithm to detect black-and-white circular fiducials. The introduced localisation system originates from the real-time single-camera fiducial marker system with an estimation of the position and orientation in the 3D space and a distinguishable encoded variable identification. This multi-camera localisation system built on cheap and widely available web cameras represents a publicly available and open localisation system as an alternative to the currently available expensive, closed systems using high-end cameras and specialised hardware requiring tedious deployment. In order to assess the performance, the localisation system is tested against the single-camera original method widely used in the fields of mobile and swarm robotics. We created simulated testing environments allowing dynamically changeable simulated scenarios, and we also collected a real-world dataset of an application on the mobile robot external localisation. An automatic evaluation and simulation framework was introduced to make the testing process more manageable.

# Contents

# List of Figures

# List of Tables

# 1 Introduction

Nowadays, the capabilities and utility of technologies have undergone an enormous evolution; especially, robot development is in the exponential era. Robots have become famous through massive improvements in material engineering, the accuracy of the manufacturing process and miniaturisation of electronics, and advances in the algorithms powering the robots. The robots managed to find their way into our daily lives without much disruption, primarily because they are in utility services, usually under the supervision of their masters or anyone passing by. However, the fully autonomous and independent behaviour is highly challenging to achieve while allowing the coexistence of the robots and people in the same environment.

In the case of a robot operating in a particular environment without constant supervision, it has to establish a plan to fulfil the required tasks. The robot needs to know the representation of the state space, a *map*, and the assigned goal in order to be able to plan. The map can be used to find a plan of getting from one state to another, and such actions of following a path constitute the *navigation* task. To know where the robot is on the map and to verify it is progressing on the path correctly, another area is addressed, the *localisation*. In many situations, those tasks are addressed jointly as Simultaneous Localisation and Mapping, *SLAM*.

In either the decoupled or joint approach, the crucial question to answer is how the robot can create such a representation of its surroundings. As the robot needs this knowledge in order to operate correctly, it has to be equipped with a way to sense the environment. The camera sensor is among the essential robotic sensors, and it can capture the environment's look in various image representations based on the detected light spectrum. Many image processing algorithms have been proposed to extract meaningful information from those images about what is in the image and how it can be utilised in robotics. One can use the found information for map building or for localising in an already built one. The information can span from a low to a high level of semantic understanding of what is the robot looking at, either significant groups of pixels, distinguishable patterns, or real-world objects. This thesis focuses on such patterns, *fiducial markers*, which are artificial landmarks enhancing the environment with reliable and accurate information about its position and, in many cases, also the orientation, which can be used for localisation. The fiducials are prominent among the other image *features* because of their clearly distinguishable characteristics, which make them easily detectable and localisable. Thanks to their low entry price, a printer and an off-the-shelf camera, they can be found in various fields and applications like augmented reality, reliable ground-truth for experiments evaluation, unmanned areal vehicles landing, swarm and multi-robot formations and tracking, and deployments where there is a need for additional supporting localisation information such as featureless or feature ambiguous environments.

However, as in the case of any measuring device, there are certain imperfections and preconditions which can swing the use of a camera from a success to a failure. First, there is a noise always present in an image that can have many forms, so not only the standard

Gaussian distribution as it is commonly assumed but all it depends on the illumination conditions and especially on the manufacturing quality and capabilities of a given camera chip. The next significant source of issues in the image processing is presented by the used lenses, which can cause a spacial warping, colour shifting, and misalignments requiring an additional set of methods to compensate them. A further drawback of a camera is the limited resolution, where one loses the precision with lower resolution and the field of view, which is bounded by the used lenses. One can opt for a camera with a very wide field of view, but at the same time, they would have to handle the more significant distortion or use expensive high-end lenses. Overall, there are many problematic aspects that one can encounter while deploying a fiducial marker localisation system; thus, it is essential to question how well the localisation system can address those.

This thesis introduces a localisation system using multiple cameras in configurations to cover large areas or improve the localisation performance. The localisation system is needed to support reliable estimation of the full 6 degrees of freedom for such a localisation system. On top of that, there should be an option to distinguish individual markers and provide a high detection rate even under a broad range of illumination conditions. However, the localisation system has to be able to provide sufficient real-time pose estimation. Therefore, computational performance is another restriction. The current state-of-the-art fiducial markers do not satisfy all the requirements, so that those additional modifications will constitute an improved fiducial marker detection method. Then, the overall performance is assessed to verify the capabilities of the extended system.

The thesis consists of five main sections. Firstly, the localisation and mapping in robotics are presented by the most popular state-of-the-art methods currently used. The methods are chosen from the field of camera-based localisation and vision geometry, which are suitable for external localisation over large areas. The second section evaluates the dominant fiducial markers in the single-camera application over a real-world dataset to gain a foundation for selecting the localisation system for further extension. The newly introduced multicamera system is presented in the third section, where the selection of the fiducial marker system is discussed, together with the algorithm for localisation and tracking using multiple cameras. The fourth section presents the collected datasets and the testing framework allowing reproducible and automated evaluation of given experiments. The fifth section evaluates the localisation system mainly for its accuracy and performance over several datasets compared to the base single-camera variant.

# 2 State-of-the-art in localisation and mapping

Once a mobile robot has to perform an autonomous task, it needs to know, represent, and understand the environment where it operates. In order to deliver the task, the robot has to follow partial steps to achieve the goal state, navigate, and know that the task has been achieved. In mobile robotics, the navigation task is about finding those partial steps, planning, and moving to specific locations or following a path. To plan, the robot has to have a map which is a simplified representation of an environment. Then, the robot needs to find itself within the map, localise, and assess the constitution regarding the goal state or the planned path. There comes the need to understand the surrounding, and for such, the robot utilises and evaluate equipped sensors. The received data are used for pose estimation with respect to a given coordinates frame. In situations when the localisation and map building problems are solved jointly, the problem is studied as Simultaneous localisation and mapping (SLAM). Because of the real-world dynamics, the incoming sensory data suffer from a naturally occurring noise and false reading, so the additional problem appears in terms of proper data evaluation and filtering. Therefore, it is crucial to address the data imperfection and introduce means of data processing and robust methods for the tasks above, as in the other case, the solutions might be corrupted and faulty.

## 2.1 Mapping

Capturing the appearance of the robot's environment can be utilised for building maps which are the abstract understanding of how the robot perceives the world. Basically, any information that describes a part of the surrounding can be considered a map. Pure recording of the raw sensory input maintains the exact appearance of the world; however, it is almost impossible to use such data as they are for higher-level decision making like planning, localisation, or searching for goals to explore or exploit. Therefore, a particular abstraction of the robot perception is required for meaningful mapping. There exist various approaches to map building, as each of them is beneficial in different applications. The robot does not have to build the map by itself, but it can be provided with a hand-crafted or generated one which can feature more complex information than the robot would be capable of acquiring. Based on the level of the data abstraction, the mapping techniques can be divided into topological or metric methods [1].

### 2.1.1 Metric maps

The metric maps record and maintain the real-world metric relations of the mapped environment, usually in the form of a grid with uniform resolution [2, 3]. The individual objectives, such as obstacles, are expressed by coordinates either in the world frame or in the discretised grid-map frame. The advantages of those maps are their straightforward implementation and modification because, in order to incorporate the sensory measurements, it is enough to know their transformation from the sensors to the current position of the robot within the map. As they represent a simple abstraction and discretisation over the robot observations, they can also be easily visualised and validated by people. When building the maps, it is essential to properly know the position and orientation of the sensors because any offset leads to not only incorrectly recording the new data but also can corrupt the already existing map. The metric maps are usually more complex and require more resources as even unimportant parts of the maps between several points of interest have to be kept in order to persist their mutual spacial relation.

**Occupancy grids** provide the information about the traversability of individual cells, usually in terms of the probability that the given cell is occupied, see Figure 1. The occupancy of individual cells is considered independent even though one can expect similarities of spacial close places in the real world. Every new measurement from the sensors has to be evaluated for possible obstacles. The traversability of every observable cell has to be updated based on the currently estimated occupancy state.

**Geometric maps** represents the robot's surrounding through simple geometric objects. The sensory data needs to be interpreted to form a line and other primitives, and therefore the cost of a more abstract and effective representation is the additional computational requirement. Those maps express the obstacles or objects, in general, more smoothly as they are geometry modelled. However, it also means that finer details might get lost as they would appear insignificant to the object fitting algorithm.

**Landmark maps** are even more simplified maps than the previous two approaches. They are usually built by recording only the position and orientation of known landmarks with respect to the global map coordinates. Such representation can be highly compact and efficiently created. However, with the additional abstraction, the robot must be provided with a dedicated algorithm for detecting and distinguishing individual landmarks. Also, those maps are highly dependent on the landmarks' observability, and in case of a change in the environment affecting the landmarks, the map might become useless.

### 2.1.2 Topological maps

In situations when the robot's pose does not have to be determined with high accuracy and the environment is easily distinguishable, the topological maps are more efficient [4, 5]. The efficiency is gained through a higher level of place representation. The places are transformed into a graph with the nodes as places and the edges as mutual relations between them as in Figure 2. Therefore they are more suitable for coarser maps and thus lower accuracy in localisation because the more accurate the map would be, the more nodes in the graph there would be. Thanks to the graph structure, planning or any graph-based algorithms are easier and faster to be applied. As the locations are represented as independent nodes, it is not necessary to express the robot's observations in the global map frame because it is enough to keep only a detailed description of the nodes, which can be specific to a given node. However, those maps become impractical if the individual locations are ambiguous because it is likely that the localisation would be highly incorrect due to imperfect sensory data.

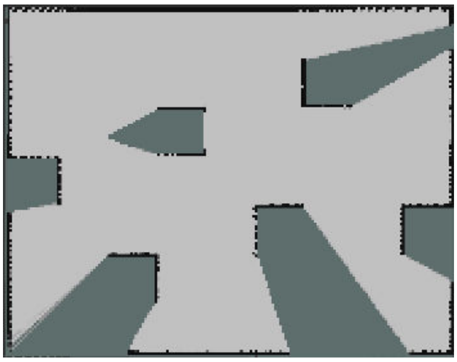


Figure 1: Occupancy grid map

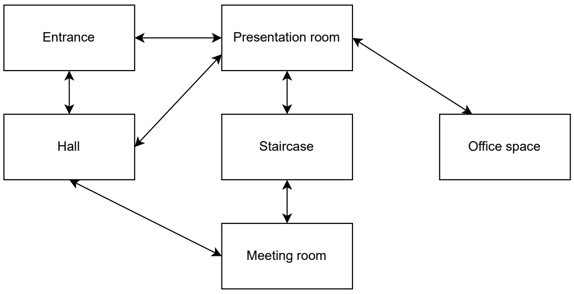


Figure 2: Topological map

### 2.1.3 Hybrid maps

A unique and more complex mapping approach is the hybrid map which integrates multiple different forms of maps into one. In [6], the authors combine the matric grid-based maps with the high-level topological maps. Although the grid-based mapping methods achieve high accuracy and a more consistent representation of the environment, their resource requirements increase quickly with the accuracy and the area they cover. However, they can be abstracted and split into individual smaller maps, and a topological map can be built on top of them, which benefits from the effectiveness in more complex planning or, in general, in the robot deployment over large spaces. Such hybrid map application is beneficial when a pure localisation in one or the other map would lead to incorrect pose estimation and thus the robot being lost.

## 2.2 Localisation

To localise means to estimate the robot's immediate position and orientation in a given frame using the available sensors. This task is specifically important when the robot needs to change the position but not only as a part of a static path following but rather when the destination is different with the time. Localisation approaches can be categorised into two sections depending on the frame of reference it is performed - relative or absolute. Other divisions are described based on the location of the sensors, whether it is equipped on the robot, onboard localisation, or whether they are not external localisation.

### 2.2.1 Relative localisation

When a robot estimates the pose with respect to its previous pose, it is called relative localisation or dead-reckoning. In this case, the pose is only considered as a change from the previous one. It can be accumulated to represent the pose in the frame of the beginning of the localisation, which is generally used rather than the first approach. As it is based on the estimation of the movement delta, the robot can only analyse the change in the incoming data from sensors, and thus, due to the accumulation of estimations, also the error is accumulated. Therefore, a certain method must be incorporated to overcome the error and correct it as the robot operates. However, thanks to evaluating only the small changes in the pose, the localisation is accurate from a short-term perspective. Moreover, it is often used as a source of differential change in localisation. Relative localisation is not memory demanding as it keeps just the previous estimation. In general, only the onboard sensors are considered for dead-reckoning.

#### Odometry

Odometry localisation is based on sensors for measuring the movement by encoders, and commonly they are either optical or magnetic. When a moving part of a robot is equipped with encoders, they count how many times the given part, wing, wheel, or belt, has turned around, even partially with a specific step. Together with the distance between individual steps, the overall moved distance can be obtained. As the number of signals to the encoders can grow quickly, the sensor control unit usually only provides the number or signally per a set interval. Therefore, the sensory output provides the robot's velocity, which must be integrated to establish the pose. Such an approach introduces the possibility of numerical instability when integrating over a large distance travelled or errors in estimation by delays in sensor readings. Moreover, when the parts are moving over various materials, they might slip and move more, so signals significantly different displacement than actually happened. So, the odometry cannot be entirely relied on as the only localisation estimation. However, it is accurate in the short term because it is directly based on the physical movement of the robot's motors. Thanks to its direct incorporation into the motors, they can be found in almost all current robots. To overcome the odometry drift in time, it can be joined with a different pose estimation for correction as in [7] where the authors are fusing it with a visual pose estimation.

#### Inertial localisation

In the case of using sensors measuring acceleration, thus accelerometer or gyroscope, we talk about inertial localisation. As the inertial sensors provide rotation readings, the derived localisation shows higher accuracy in the orientation estimation than the odometry, which can only approximate the orientation based on the differences between individual encoder readings. In [9], the authors present an accessible localisation framework using only

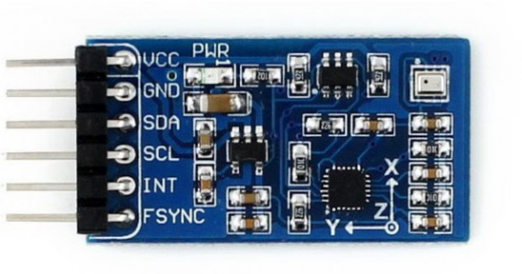


Figure 3: Combined kit of accelerometer and gyroscope [8]

the inertial sensors. The main drawback is position estimation because there have to be two integrations from acceleration to velocity and then into position. Thus, the localisation is even more sensitive to numerical instability or false reading from sensors because the error can significantly influence the odometry localisation. Thanks to the advances in technology, accelerometer and gyroscope can be manufactured together into an inertial measurement unit, *IMU*, providing a unified and reliable source of angular changes, see Figure 3.

**Accelerometers** provide robots with the knowledge of proper acceleration and its direction in space. However, they are easily affected by tilting or any change in the levelled position, also by the Earth's gravity. Therefore, they should be correctly calibrated when deployed on robots and even completely insulated in case of a rough environment to prevent errors. As the accelerometer measures instantaneously, the data might be indistinguishable from noise during slow movements.

**Gyroscopes** contrary to the accelerometers measure primarily the angular acceleration, which can be directly integrated into orientation estimation. Because of its high accuracy in orientation, it is commonly fused together with odometry's position estimation. Gyroscopes also need isolation from the surrounding in order to provide robust and accurate measurements, and so their enclosures have a bigger size. Therefore, it might limit their positioning or even deployment in general. Coriolis forces and also centrifugal forces have significant influence, and thus the reliability might decrease while the robots perform rapid changes in heading.

### 2.2.2 Absolute localisation

Absolute localisation, also called the kidnapped-robot localisation, is a task of establishing the current position and orientation with respect to a given global coordinate system which differs from the localised object's frame. Contrary to the relative localisation, no previous localisation estimation is incorporated. This approach benefits from the elimination of the likely introduced drift in integration over relative pose estimations because each pose is calculated independently on the previous one. Thanks to not relying on history, any sporadic mistake does not affect the future state. We can distinguish the kidnapped-robot problem into two categories based on the position of the sensors - onboard and external localisation. As onboard localisation is more suitable for relative localisation, additional information about the robot's surroundings and relations between existing frames must be provided. However, the external approach to localisation requires complete coverage of the operational space with sensors.

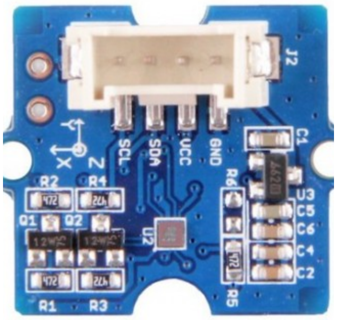


Figure 4: Digital compass [10]

Figure 5: South-pointing chariot [11]

### Compasses

Compasses represent an onboard sensor for obtaining the robot's direction regarding a specific reference place. Measuring the proper heading of the robot is especially important in localisation and also navigation tasks because a slight inaccuracy might lead to a larger turn, therefore significantly different and erroneous end position. In [12], the authors present the application of compasses for improving the estimation of orientation. As the sensor provides just the heading to the pole, it is not possible to transform it into a position estimate. Also, the measurements of the most common magnetic compasses would be influenced by strong electricity sources, metal constructions or even irregularities which naturally happen. Because of the tendency to be affected by the surrounding, it is not preferable indoors. The digital compasses are accessible sensors without moving components and are more durable and better for deployment in robotics, see Figure 4. Nevertheless, the idea to measure and track the heading for localisation dates back to the ancient Chinese south-pointing chariot, see Figure 5.

### Global positioning system

The Global positioning system (GPS) is an orbital localisation system formed from multiple satellites providing the observer data with their immediate location and current time. Because a sufficient number of satellites might not always be available, usually because of occlusion, the typical position estimation accuracy is approximately fifteen meters. However, local ground sources of supporting position information can be incorporated in order to increase the accuracy up to centimetres, and such a technique is called the differential GPS. As one has only to have a receiver for the emitted signal from orbit, there is no need for supporting hardware setup, and therefore, it makes it highly suitable for portable robots. The drawback of GPS deployment is the necessity of satellite visibility. Thus its usage indoors or underground is not possible due to the unavailability of the signal [13].

### Landmark localisation

Objects or patterns which stand out against their surrounding are called landmarks, and they can provide prominent information for localisation. Landmarks have a characteristic signature in the robot's sensory data so that they can be easily detected and recognised. They can have many different forms and types, but their analytical description is usually a priory known, which includes their visual appearance, space proportions, signal structure and many more aspects. The mutual spatial relation between the landmark and the robot can be established based on such an analytical description. The landmarks can vary significantly, so we can categorise them into artificial or natural classes depending on their source in an environment. We can approach the division from the point of view of emitted energy, thus either passive or active landmarks.

**Natural landmarks** are such patterns or objects that are serving a different purpose in the environment, but their characteristics are so distinguishable that they can be relied on. The widely used sensor, a camera, can be used for detecting high contrast environment areas like a window corner or edges [14]. Also, the position in the space can be sometimes recovered so that it can be used for localisation. When the natural landmarks are described by the vision sensors, they are primarily passive landmarks and are commonly referred to as image *features* [15, 16, 17, 18, 19, 20]. As mentioned, they are based on things that have originally had a different purpose, and therefore, nothing additional had to be added to the environment for localisation. Such an advantage can quickly turn into a disadvantage in situations when the environment is not constant, so the visibility or appearance of the features changes in time. The features are often more straightforward descriptions of image pixel patches as in Figure 6. Thus, there might be many of those across multiple images resulting in more complicated distinguishability and uniqueness.

**Artificial landmarks** are, on the other hand, mainly designed for straightforward detection and localisation based on known analytical characteristics. Contrary to the natural landmarks, the artificial ones are deliberately distributed over the operational space for easier localisation, especially in the case of a scarcity of natural ones. As almost anything that is added to the scene can be considered a landmark, there has been a wide variety in the appearance of mainly passive landmarks because manufacturing is considerably more manageable. One can think about QR-code like tags, reflexive pads or in the case of actively producing signals, and there are commonly used light or radio beacons as in [21]. In [22], the authors presents an ulstrasound indoor positioning system for robot tracking and localisation using active beacons as in Figure 7. Thanks to the prior knowledge of the landmark, the artificial approach succeeds in higher robustness and detection stability and reliability.

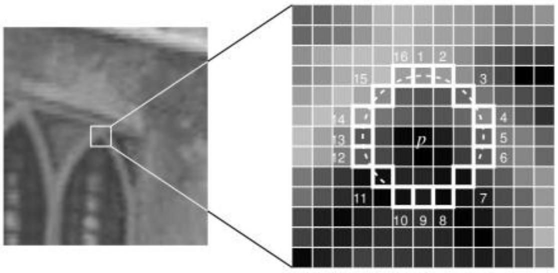


Figure 6: FAST corner detector [16]

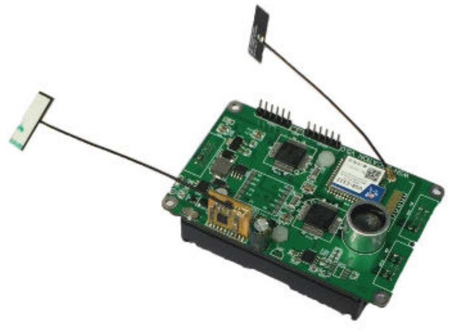


Figure 7: Ultrasonic beacon [22]

The idea of landmark localisation is to find the transformation between the immediately detected landmarks and the ones available in the map [23]. As the estimation of the landmark's pose is bounded to the mutual configuration with the robot, usually the perception, thus also the accuracy, is improved with a smaller distance and under a smaller observing angle. In many situations, some occlusions might happen, and therefore the landmark localisation is fused with other types of localisation like odometry to overcome the temporary lack of information. In [24], the authors present the below elementary solutions to the localisation task using landmarks. When the displacement from the individual landmarks is available, trilateration can be applied. However, the triangulation can provide the localisation estimation from the angles between the landmarks and the y-axis of the robot frame.

### External localisation systems

External localisation systems are complete out-of-the-box positioning and motion capture systems monitoring from outside the behaviour within the operational space. Their main domain is the excellent precision and high frequency of measurement. They usually serve as an evaluation framework for experiments because of reliable ground truth data or densely sampled motion analysis. Unfortunately, the hardware and technologies involved in the systems are often not open source and are actually at the high-end price spectrum. Apart from the limiting price aspect, they also suffer from lower flexibility because the systems have to be thoroughly calibrated every time there is a change in the setup. Another drawback of the vision-based localisation technology is that it commonly uses the infrared light spectrum, which means that there should not be other infrared light sources, resulting in the impossibility of deploying those systems outdoors.

**Vicon** is one of the most popular passive marker localisation systems for estimating both the position and orientation in various coordinates frames [25]. It is a vision-based system that consists of small markers which can reflect the infrared light which is emitted by the observing cameras, see Figure 8. The incorporated cameras have to be synchronised, so the captured images for localisation are reliable. The system firstly detects the reflec-

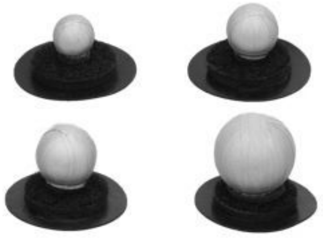


Figure 8: Vicon retroreflective markers [25]

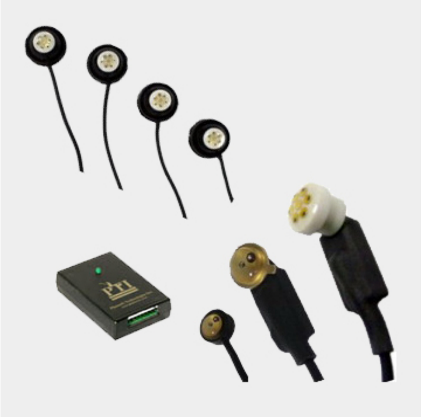


Figure 9: PTI Phoenix markers [26]

tive markers in the image coordinates, and then, using triangulation, the 3D position is estimated. As in [27], the Vicon system is used across a wide range of research fields, from robotics and augmented reality to biomechanical and gait analysis.

**OptiTrack** represents a passive motion capture and positioning system based on infrared light reflected by specialised markers equipped on the tracked desired targets [28]. This system is similar to the Vicon system because it incorporates multiple cameras and synchronised image capture to localise infrared targets in the space. In order to track and monitor intended objects, the retroreflective markers have to be visibly mounted to be captured by the high-resolution cameras. Because the high-level approach to the localisation is shared with the Vicon system, triangulation is used for the position estimation. However, the localisation is based on the markers' reflected light signal; therefore, it also suffers from the restrictions when deployed outdoors. OptiTrack's performance is comparable to the Vicon's one, as stated in [29], where the authors conclude that although the OptiTrack is priced lower, the localisation precision is comparable or slightly lower, which might still be acceptable in many fields.

**PTI Phoenix** is, contrary to the previous two systems, an active external localisation system [26]. The system again operates in the infrared light spectrum where the active LED markers emit specialised light signals as the markers can be distinguished based on the blinking frequency. Apart from the active LEDs, the overall system setup is similar to the others as it also requires synchronised image capture from multiple high-resolution cameras. Because the light source is in the operational space and not in the cameras, the environment is not flooded with infrared light. Therefore, there is a chance that the other infrared-sensitive sensors would not be blinded, as is shared among the passive systems. The main drawback of this system is the necessity of a power source for the LEDs, which introduces not only making sure that the batteries are charged but also thin and fragile wires that connect the LEDs with their control unit. In Figure 9, there are the necessary system components to be equipped on the tracked object.

## 2.3 Simultaneous localisation and mapping

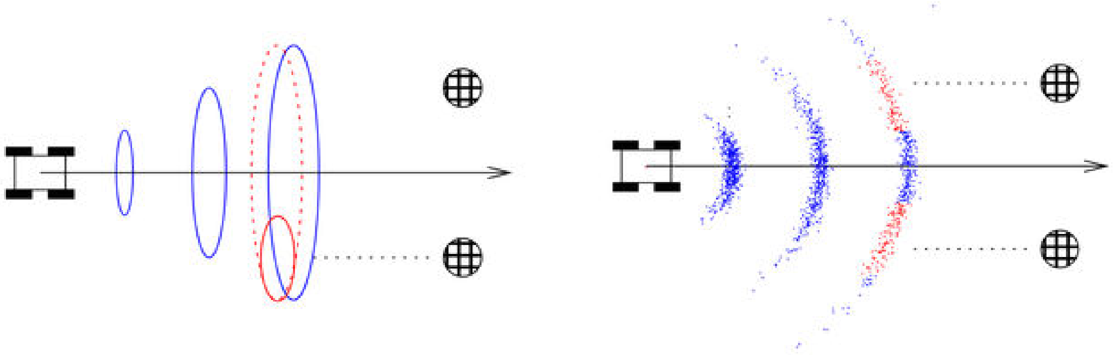
The task of building a map and, at the same time, localising is called simultaneous localisation and mapping (SLAM). This complex problem is the foundation of various robotic applications where the robot has to operate in a previously unvisited environment. In the beginning, the robot's position and the map are not known, and both of them have to be accurately estimated at the same time. Thus, it is essential to at first correctly interpret the current observation and add it to the map and then localise itself within the newly extended map. Then, when the robot moves, the motion model, which describes the probability of the new position, can be used to estimate the new position. The standard approach is based on derivations of the recurrent Bayes rule, so the imperfect measurements from sensors together with the robot's state transitions can be modelled jointly. In order to estimate the most probable map and pose, there are two main approaches, the extended Kalman filter and the particle filter [30] visualised in Figure 10.

### Extended Kalman filters

Extended Kalman filter (EKF) represents the belief of the map and pose by unimodal and multivariate Gaussian distribution. Therefore, the number of variables required to model the distribution is relatively small, however, only for small-sized maps. The underlying assumptions are the linearity of the state transition and sensor observation probabilities are linear with added Gaussian noise. Another assumption is the known observed feature correspondence with the ones on the map. The main disadvantage of the EKF approach is the dimensionality growth with the growth of the map. Thus, there is a computational limit on the size of a map. The restriction on the size of the environment can be overcome by either selectively choosing which observed features to add to the map or splitting the map into a set of smaller ones. In [31, 32], the authors presents the EKF-based SLAM effectivness in feature rich environments.

### Particle filters

Particle filters model the belief of the map and robot's position and orientation by multiple particles creating a probability density. With an incoming sensory measurement, each particle is drawn newly from a distribution with modified weights with respect to the sensor model and alignment with the map. Therefore, the particles cluster around the most likely solutions to the SLAM problem and form several hypotheses to choose from. The complexity of the filter increases with every new landmark added to the map because to achieve a high probability of such a landmark, it requires a high density of particles. Thus, in the pure form of this approach, it becomes quickly unpractical to build the map, and it is reasonable only to use it for robot localisation. The authors of FastSLAM [33] introduced an improvement to the resource requirements of particle filter by Rao-Blackwellization, which represents the landmarks as conditionally independent.



(a) extended Kalman filter

(b) Particle filter

Figure 10: Robot position estimation by SLAM filters [34]

### 2.3.1 Visual SLAM

Multiple different sensors have been fused together in the beginnings of the SLAM methods. However, the increasing accessibility and simplicity of onboard cameras opened the door for studying the SLAM methods based only on the visual information provided by the cameras. Contrary to the other sensors like laser range finders or GPS, the camera images are only a 2D projection of the surroundings. Therefore, obtaining the scale information important for accurate map building is substantially more challenging. We can divide the visual SLAM techniques into two groups depending on the image processing methods applied. Feature-based methods detect image features and use them for pose estimation and map building, and the direct methods process the whole image buffer, which might be beneficial in feature-less environments [35, 36].

#### MonoSLAM

In [37], the authors presented the first feature-based visual SLAM using only a monocular camera, thus called MonoSLAM. The underlying algorithm is the extended Kalman filter for the map and position estimation. To initialise the mapping, it is required to capture a priory known landmark or object in the space to establish the global frame. The used image descriptors are the SIFT features. The features are extracted from a single camera image, so the depth information cannot be directly recovered; therefore, it is estimated by tracking individual features in an image sequence. In order to correctly track the features, they have to be correctly detected and matched with the map. Therefore, the movement must be stable and slow enough to allow reliable tracking. Thus the algorithm assumes a uniform state transition model. Faster and sharper movements can be supported by using a higher frame rate, but it is in opposition to the resource restriction imposed by the growth of the map.

### LSD-SLAM

LSD-SLAM, or Large-Scale Direct Monocular SLAM, represents the second category of visual SLAMs; thus, it evaluates the whole incoming images directly from the monocular camera [38]. Direct image processing is utilised for both the localisation and also map building. As the image is exploited for information as a whole and not only a specific descriptor region, more information for localisation can be used. Compared to the previous direct SLAM methods, this approach addresses and corrects the scale drift occurring in large-scale maps. Contrary to the feature tracking in the image stream to assess the depth, the LSD-SLAM recovers the depth using stereo view comparison on a pixel level basis. It is more beneficial in feature-less or distinctive texture-less environments where it provides more accurate and robust pose estimation. In [39, 40], the authors present the modifications to support a stereo and omnidirectional camera, respectively.

### ORB-SLAM

Another popular representative of the feature-based visual SLAMs is the ORB-SLAM which uses the ORB image features instead of the SIFT as in the MonoSLAM [41]. Those features originate from the binary BRIEF descriptors with the addition of robustness to noise and changes in orientation. Because of their real-time detection capabilities, even on a CPU, they do not suffer from high resource requirements. The possibility of mapping large areas is achieved by expressing the maps as a co-visibility graph. Therefore, global relations are known, but the robot always operates on one local map. The drift correction through loop closure is not applied to the whole keyframe graph but only to the edges in a found spanning tree. The correction is then recalculated more quickly. Further evolved ORB-SLAM2 introduced in [42] extends the types of applicable cameras with stereo and even the depth cameras. Such extension supports the easier recovery of the map scale or improves the feature tracking.

### UcoSLAM

UcoSLAM is a special kind of visual SLAM as it fuses the fiducial marker pose estimation with the traditional feature-based localisation and mapping [43]. Originally in the work [44], the authors designed the SLAM to use the fiducial markers only instead of the image descriptors as in the aforementioned methods. However, in the last modification, the method can use both feature sources because otherwise, the environment would have to be enough populated with the markers. The used marker, the ArUco, reduces the uncertainty of the map scale of even the accumulated drift. As the markers have prior known characteristics, once detected, the map can be corrected to correspond with the metric relations obtained from the marker. Such SLAM feature is advantageous in applications with repetitive structure, thus with the high rate of false loop closure detection. Also, it supports proper relocalisation after the feature tracking has been lost.

## 2.4 Fiducial markers

Fiducial markers are artificial landmarks added to the environment in order to provide a stable and reliable source for measurements or a distinguishable anchor point in the image. Thanks to this property, the fiducials are highly used in robotics to support various localisation, tracking, and even navigation tasks[45, 46, 47]. Basically, the fiducials can be deployed to improve the onboard or external localisation whenever a camera is involved. Because they are designed to be easily detectable by image processing techniques, they commonly have a shape based on simple geometric objects, and their colour palette has high contrast. Therefore, there are usually black and white. In order to support their versatility, they are planar markers with variable sizes; thus, they can be attached and placed almost anywhere. In Figure 11, the fiducials are used for the unique identification of individual objects in the environment and for estimating their mutual positions, whereas in Figure 12, the markers is used for the bee queen tracking. An exhaustive evaluation of the performance of fiducial markers applied in the area of autonomous cars is presented in detail by the author in [48].

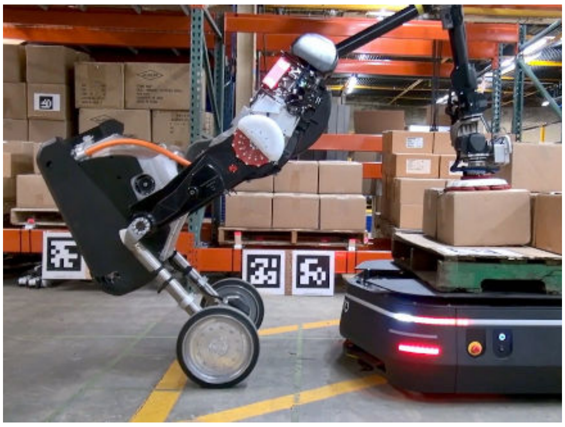


Figure 11: AprilTag markers as labels for warehouse robot [49]

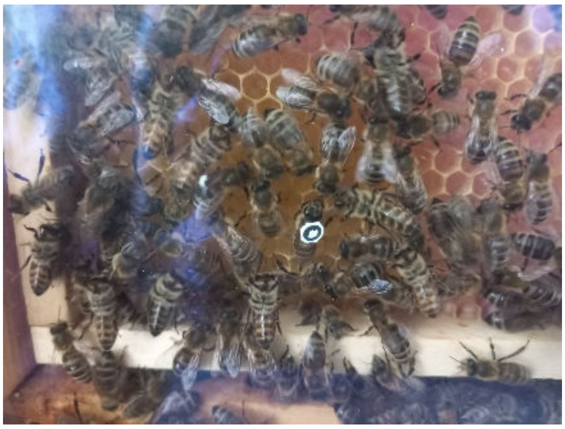


Figure 12: WhyCode marker for bee queen tracking

### AprilTag

AprilTag is one of the fiducial markers with a black square shape and black-and-white binary code inside [50, 51]. The fiducial features a unique identifier based on lexicographical binary encoding within the inner area of the square, see Figure 13. The binary code can be flexibly generated to produce sufficiently many markers while being resistant to false ID decoding by the scalable hamming distance between individual code words. The partial pose is estimated by the homography and extrinsic transformation, which provides the possible position and orientation of the fiducial. Also, the yaw rotation is recovered from the rotation of the 2D binary code obtained by thresholding the marker's inside area. Therefore, it is possible to estimate the full six degrees of freedom with respect to the camera coordinate frame. Even partially occluded markers can still be detected and even identified thanks to searching for all line segments in the image and then line fitting to form a closed region with four corners. If the region passes several checks for its squareness, then the ID is attempted to be estimated. An extension of the fiducial marker system is presented in [52] for estimating the camera calibration parameters. The authors of [53] relaxed the marker's shape to support a broader range of shapes and even allowed the embedding of custom content within the centre area. The marker design was further extended to applications with highly variable observing distance by introducing a recursive marker which consists of several differently sized markers.

### ArUco

ArUco is a fiducial similar to the AprilTag in terms of the shape and ability to estimate the full six degrees of freedom [54]. As the marker detection shares the approaches with the AprilTag, it also relies on the pose estimation using the four corners of the marker. However, when the marker is partially covered, the detection is not possible as there is no line reconstruction from partial line segments. On the other hand, the authors propose to utilise smaller markers close to each other; thus, the others could still be detected in case of occlusion. The used ID enoding is based on mixed-integer linear programming as introduced in [55]. A faster version of the detection algorithm relies on the evaluation of consecutive frames and on-the-fly selective limits on the marker's size and position as in [56]. In [48], the authors present an evaluation of the marker performance showing an unstable time required to detect the marker in an image which might become a bottleneck of a more complex localisation pipeline. The fractal variant based on ArUco has been presented in [57], and it introduces the resistance to occlusion and increases the detection distance. In Figure 14, one can see the similarity between the square-based marker designs.

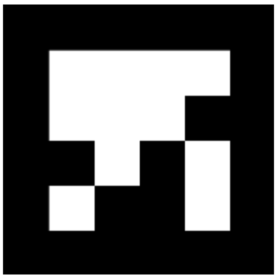


Figure 13: AprilTag

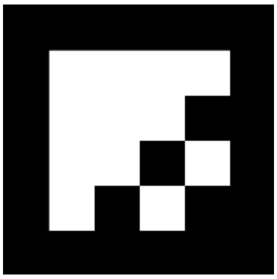


Figure 14: ArUco

### WhyCon

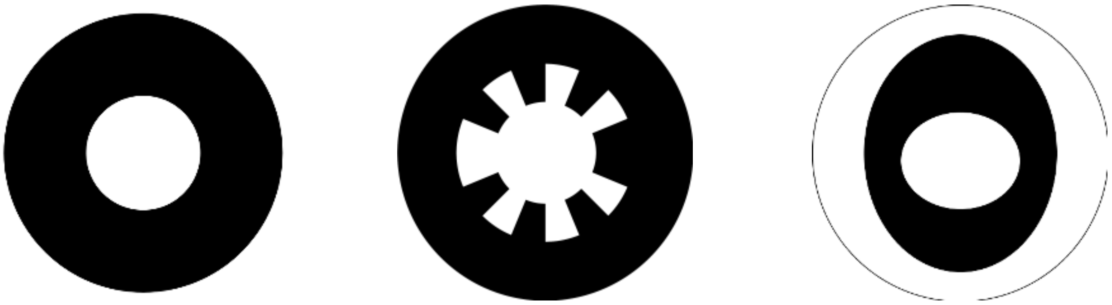
WhyCon represents a black-and-white annular fiducial marker which consists of two concentric circles with a flexible size ratio [58, 59]. For successful localisation, the outer black diameter together with the size ratio with the inner white area must be provided as those are the prior metric knowledge for scale estimation. As the design is relatively simple, the localisation can estimate up to five degrees of freedom because there is no information for determining the revolution around the marker's surface normal. The underlying geometric model can provide position estimation precision in a few millimetres. Therefore, its primary applications involved mainly the position estimation and tracking due to the fact that the time required to detect thousands of markers is in lower milliseconds. Such performance is the system's main advantage because it can smoothly run even on computationally restricted hardware [60]. Because of the marker's lack of distinguishability from each other and the impossibility of recovering the full six degrees of freedom, other marker systems evolved from it - WhyCode and SwarmCon as visualized in Figure 15.

### WhyCode

WhyCode is a circular fiducial marker which originates in the WhyCon marker and introduces solutions to the missing features of the original marker [61]. Because of the marker's similarity to the WhyCon, the core parts of the system are shared, especially the detection of the black-and-white inner and outer segments. Also, the 3D position and orientation estimation from the characteristics of the approximated conic section is based on the marker contour. In [62], the authors presented the WhyCode with a circular binary code which is inspired by the Necklace codes. The encoded binary sequence can be interpreted as an ID and uniquely generated for individual markers. Using the binary code, the rotation around the marker's surface normal can be estimated because the Necklaces are rotationally invariant sequences; therefore, the yaw rotation corresponds to the required shifts of the code until the lowest value is achieved. Unfortunately, the introduced IDs restrict the wide range of detection angles and lower the distance as the pattern is more complicated. In [59], further improvements of the pose estimation by resolving the ambiguity are presented.

### SwarmCon

SwarmCon is a specialised modification of the WhyCon marker localisation system, which adds the possibility of encoding unique marker ID and estimation of the rotation around the marker's surface normal [63]. The two concentric circles are generalised into ellipses with offset centres, and therefore, the yaw angle can be determined from the ellipses' semi-axes. The marker's ID is encoded into the centres' offset and rotation of the inner ellipse. Unfortunately, such an approach brings a restriction on the viewing angles, which are reduced to the almost perpendicular placement of the camera above the operational space. As of such restriction, the applications are mainly in mobile swarm robotics, where real-time tracking and minimal computational requirements are beneficial. Because of the specialisation for swarm experiments evaluation, the system offers an embedded coordinate system calibration.



(a) WhyCon (b) WhyCode (c) SwarmCon

Figure 15: WhyCon-based fiducial markers

### ChromaTag

ChromaTag [64] localisation system stands out in its specific usage of LAB colour space instead of black and white representation as in Figure 16. Even though the marker has a square shape, the detection is real-time and faster than other more traditional black-and-white square markers. Such performance is possible because it does not detect the edges in a thresholded grayscale image which can easily overwhelm the detection algorithm as those edges are naturally present in the environment. Instead, the detection is performed on the coloured image in the LAB colour space, which has the appropriate colour distribution suitable for immediate rejection of segments not belonging to the marker. However, due to the increased complexity of the marker's design, the detection distance and angles are more restricted than the other state-of-the-art markers.

### RuneTag

RuneTag markers are based on the circular markers approach; however, they utilise small dots to outline several concentric circles instead of solid coloured rings [65]. A specific dot pattern along the perimeter represents the unique encoded ID of a given marker, see Figure 17. The ID can be recovered even under significant occlusion thanks to a highly flexible error-correcting coding system. The fiducial design does not incorporate the inner area for detection or localisation purposes. Therefore, it is open to the placement of any additional helpful information for the image processing pipeline. However, as the pattern is based on small dots, they have to occupy a sufficient number of pixels in the image to be detectable; thus, the detection capabilities are lower than the other marker designs.

## 2.5 Camera-based localisation

All of the methods mentioned above require a sensory source of information as otherwise, the robot could not perceive its surrounding. With the development and minimisation of camera sensors, they have become more accessible and bearable for robots. Therefore, it

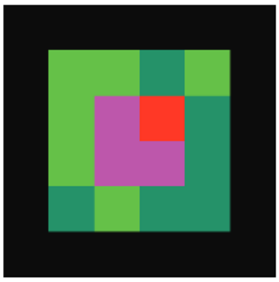


Figure 16: ChromaTag

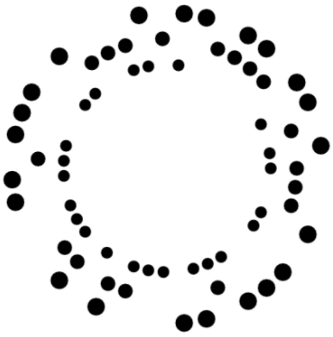


Figure 17: RuneTag

is essential to pay attention to this sensor as it is information-rich and can be used standalone for localisation. However, the geometry model is more complex compared to other sensors like laser range finders, as the information obtained is only the 2D projection of the surrounding world. Also, the captured image requires a significant amount of processing to detect and localise the features present.

### 2.5.1 Camera model

A camera sensor is a very complex and advanced device that can capture the observed environment's visual appearance. It is possible by recording the intensity of the light spectrum. The light intensity is measured by a special CMOS chip composed of many individual cells forming a square grid. As each cell provides only the intensity of the light and not the colour, there is a filter applied over the cells in order to allow only the light wavelengths of the desired colour. The most popular filter is the Bayer filter which is formed of a grid pattern with half the cells green, and the rest is half red and half blue because the human retina is the most sensitive to the green colour [66].

The camera chip cannot be used as is to capture an image because it would be overwhelmed by all of the surrounding light. It has to be enclosed, and the light rays have to be restricted in the way that only the rays reflected by the intended objects are measured. This approach was demonstrated in the early beginning of the cameras by the camera obscura, see Figure 19. It achieves it by directing the light through a tiny pinhole in a box-like device which allows the rays to form the image on the backside of the box as in the Figure 18. However, such an image has poor quality as it is blurry and dark due to the limited amount of light. When the pinhole is enlarged, more light can be captured, but the image becomes more and more blurry because the rays are not focused on a small point by the pinhole. This problem can be mitigated by adding lenses directing the light back to a single point.

The obtained image is a projection of the 3D world onto a 2D plane. When we want to relate the world point with its point in the image and vice versa, we have to be able to

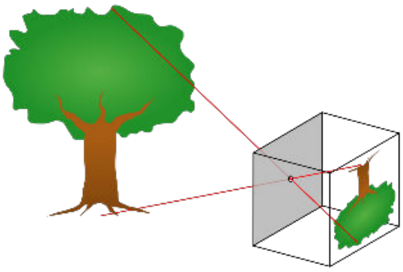
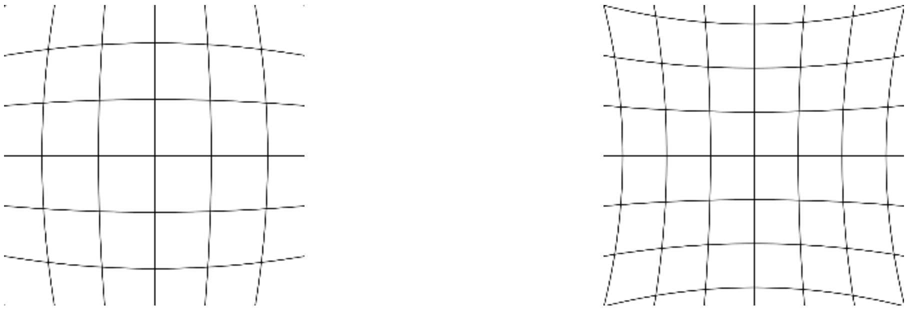


Figure 18: Pinhole camera [67]

Figure 19: Camera obscura [68]

model the camera projection. However, the introduced optic system for focusing the light makes it more complex even for using the simple thin lens model. Therefore, the pinhole camera model is used for its simplicity and linearity. However, it only approximates the actual camera, so the compound lenses that distort the resulting image are not reflected in the model.

The image must be adjusted to be as the pinhole camera would capture it. The adjustement is called *rectification* and it is described by the *distortion* parameters which compensate the captured picture element, *pixel*, position. There are two main types of distortion - radial and tangential. The radial distortion is caused by the uneven thickness of the lens, which causes the light bends differently around the edges and can be of two types - barrel and pincushion, see Figure 20. The tangential distortion is produced by improper assembly of the camera device when the capturing chip is not parallel to the lens. In order to determine the distortion parameters and estimate the parameters of the camera projection model, the camera has to be calibrated.



(a) Barrel [69]

(b) Pincushion [70]

Figure 20: Radial distortions

A 3D point in the camera's field of view can be projected onto the 2D image plane and assigned the image pixel coordinates. As the camera's coordinate system origin is usually at a different location than the world origin, we at first have to estimate the 3D rotation matrix $\mathbf{R}$ and the translation of origins $\mathbf{t}$ to transform the world frame to the camera frame. Then, we have to appropriately scale the coordinates so they would correspond to the pixel size, and the point would be lying in the image plane. Such transformation is denoted by matrix $\mathbf{K}$ below

$$\mathbf{K} = \begin{pmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}, \tag{1}$$

where $c_x, c_y$ are the coordinates of the optical center in pixels, $s$ is the skew coefficient in case of the axis are not perpendicular, $f_x, f_y$ are the focal lengths in pixels obtained by dividing the focal length in millimeters by the dimentions of the pixel expressed in the same unit. When the first rigid transformation and the projection are combined, they form the projective matrix $\mathbf{P}$ as

$$\mathbf{P} = \underbrace{\mathbf{K}}_{\substack{\text{intrinsic}\\\text{matrix}}} \underbrace{[\mathbf{R} \mid \mathbf{t}]}_{\substack{\text{extrinsic}\\\text{matrix}}} \tag{2}$$

Putting it all together, the pixel coordinates, $u, v$, of a world point, $(x_w, y_w, z_w)^T$, are calculated as

$$\begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} = \mathbf{P} \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \tag{3}$$

$$u = x_i/z_i \tag{4}$$

$$v = y_i/z_i \tag{5}$$

**Camera calibration** is a procedure to find the perspective projection matrix. The straightforward approach is the six-point algorithm which is based on six-point correspondences between the image point and the world points [71]. However, it is possible to recover only a non-zero multiple of the projection matrix as any introduced non-zero scale will cancel out in the projection to a 2D point. Also, the distortion parameters are not obtained, which are highly important, especially for cameras with small lenses. Another widely used approach is to capture multiple images of a planar pattern from different views [72, 73]. The used pattern can be anything, but we have to know the metric relations of it as in Figure 21. Because it does not require the correspondences, it is easier to use as measuring the exact position of a point in space is rather nontrivial. Also, this method can estimate the distortion parameters of the camera.

### 2.5.2 Camera pose estimation

The camera pose can be estimated from two views which observe the same points in the world coordinate frame. In other words, the cameras have to have an overlapping field of view. It can be applied for both an onboard localisation using the robot's camera and two cameras calibration for external localisation. In terms of the onboard localisation, the two

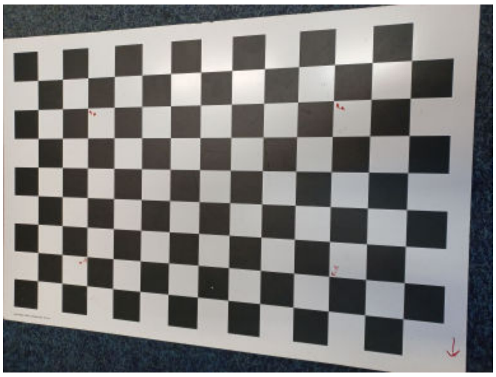


Figure 21: Camera calibration chessboard pattern

views can be even obtained from a single camera when considering the time to distinguish them. However, the robot's control has to be adjusted for it because an image loses quality under abrupt and sharp movements. The utilisation of two camera calibrations provides the transformation from one camera's coordinate frame to another; thus, their mutual position and orientation can be estimated.

The methods for multiple view geometry estimate the camera pose from a particular number of point pairs which are the mutual corresponding points in either the image or the world coordinate system. Therefore, most of the methods require additional knowledge in order to determine if the points belong to each other or not. We can divide the methods based on whether they need such information in advance of the pose estimation or are more general and can automatically find those correspondences.

Another point of view on the method categorisation is based on the coordinate frames of the observed points. There are methods which are more general and not bound directly to vision techniques, so they can be used for estimating the transformation between almost any standard metric system. However, sometimes, it might not be possible to use them for localisation because the data obtained from camera observations are only a projection from 3D to 2D space, so they are missing the depth and scale information. The pose estimation would be useless for any integration into a global coordinate system. The deployment of fiducial markers and other landmarks which can provide metric pose estimation can overcome this limitation. On the other hand, one can use methods designed to work directly with the perspective projection within and work over correspondence pairs in the image frame. They utilise the epipolar geometry to calculate the orientation and translation between cameras. It can be even used to obtain the projection of points from one camera into the other.

The epipolar geometry describes the geometry of two views [71]. When two cameras observe a place, various restrictions and relations can be imposed on their mutual pose and on the projected points they observe. The individual camera centres can be projected into each other as epipoles as in Figure 22. Also, as the 2D image point in one image can be understood as a perspective projection of a light ray cast to the camera, such a ray can also be projected into the other camera image and form an epipolar line through the epipole and the corresponding image point in the second camera. The relation of the epipolar transformation between the two images can be described by either the essential matrix, which embeds the mutual position and orientation and by the fundamental matrix, which is the essential matrix with the individual camera calibrations applied to it. Those two matrices can be further decomposed to obtain the sought second camera relative pose to the first one.

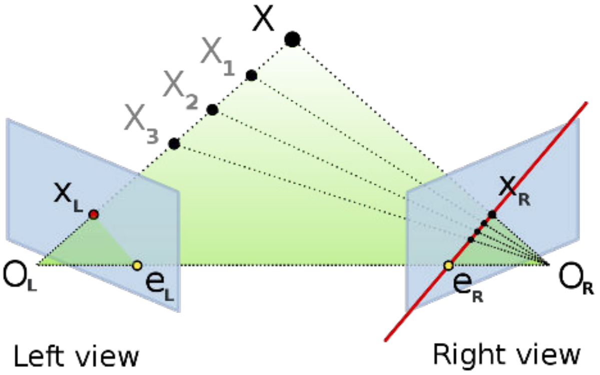


Figure 22: Epipolar geometry of two cameras [74]

### Known-correspondences methods

Many methods for transforming from one coordinate frame to another require knowing the correspondences between the individual points. The knowledge is necessary for adequately minimising a given criterion or establishing a set of equations to solve for the sought parameters. This precondition might appear straightforward, but it brings a new problem of correctly matching two sets of points, each in a completely different frame. To overcome the problem, one has to deliberately select the landmarks they will detect in the image so they would be distinguishable enough to be uniquely matched. It is usual that the points are obtained by appropriate image features or even by placing in the scene the fiducial markers, which can provide the position in both image and metric coordinate frames.

**Absolute orientation problem** describes the least-square method applied to obtaining the rotation and translation between two sets of matched points. In [75], the authors present a solution to this problem using the singular value decomposition for calculating the rotation matrix and then the difference between the datasets' means after rotation to obtain the translation estimate. The least-square minimization is over the space of 3D position vectors and the space of rotations of the corresponding dimensionality as follows

$$\mathbf{R}^*, \mathbf{t}^* = \underset{\mathbf{R}\in SO(3), \mathbf{t}\in\mathbb{R}^3}{\arg\min} \sum_i \|\mathbf{R}\mathbf{p}_i + \mathbf{t} - \mathbf{q}_i\|^2, \tag{6}$$

where the $\mathbf{R}^*$ and $\mathbf{t}^*$ are the transformation estimates, $\mathbf{p}_i$ and $\mathbf{q}_i$ are the matching points to be aligned.

**Essential matrix decomposition** is a technique to recover the cameras relative poses from image measurements [76, 77]. In the decomposition process, there is an ambiguity in the estimation of rotation and translation because any combination of them is valid under the epipolar constrain

$$\mathbf{p}^T\mathbf{E}\mathbf{q} = 0, \tag{7}$$

where $\mathbf{E}$ is the essential matrix and $\mathbf{p}$ and $\mathbf{q}$ are the uncalibrated image points, meaning they have been multiplied by their respective inverse camera calibration matrices. Thus, based on the possible combinations, there are four different solutions, among which there is the correct pose estimate, its twisted pair and their reflections. For assessing the proper pose, the individual poses have to be tested whether the observed points are located in front of both cameras.

### Unknown-correspondences methods

There are common situations when it is not possible to reliably assign the corresponding data point from one dataset to the other such as low-level image features, point clouds from laser range finder, and measurements with different capture rates. Therefore, the matching between datasets has to be approximated through their mutual closeness in a certain metric or by random sampling. Even though the methods applicable for matched data samples cannot be applied directly, they can be used as a quality measure in the following generalised methods.

**Iterative closest point** (ICP) is an algorithm widely used for geometric alignment of three-dimensional models [78]. The method minimises the euclidean distance between two sets of points. Firstly, it matches the point sets to obtain correspondences given euclidean or other metrics; then, it transforms one set of points while the other is kept unchanged. As it is an iterative numerical method, the termination condition can be a maximum number of iterations or a small enough error threshold value. The convergence can be speeded up by seeding the initial transformation from other robot sensors. In [79], the authors present a comparison of various method modifications, which can improve the convergence speed or the accuracy of the found transformation.

**Random sample consensus** (RANSAC) is a general iterative algorithm for finding parameters of a given model [80]. Compared to the ICP, it is more robust to outliers as it creates the corresponding data pairs by random sub-sampling of the datasets. At first, a minimal number of samples is drawn to estimate the sought parameters and the obtained model is evaluated over the whole dataset. Data points achieving error below a set threshold are considered as inliers and the rest as outliers. If the ratio of inliers to all data points is sufficiently large, the consensus is assumed to be reached. The MLESAC [81] further improves the fitted model by additionally maximising the likelihood of the model only on the inlier set.

# 3 Single-marker evaluation

There are many kinds of fiducial marker localisation systems, and each varies in the available features and the target users. Considering the robotic applications, the highly popular and verstile markers are the AprilTag, ArUco, and WhyCode [45, 82, 83, 84, 85, 46, 86, 87, 47]. We at first have to establish the critical aspects of the fiducial marker performance, so they can be then compared and evaluated for the marker system to be extended into a multi-camera system further. The pose estimation precision and accuracy are essential method characteristics to look for. However, it is also necessary to assess their computational speed. When deployed on a robot, the less resource-demanding the algorithm is, the longer the robot can operate on batteries or perform other tasks that would be suppressed in another way.

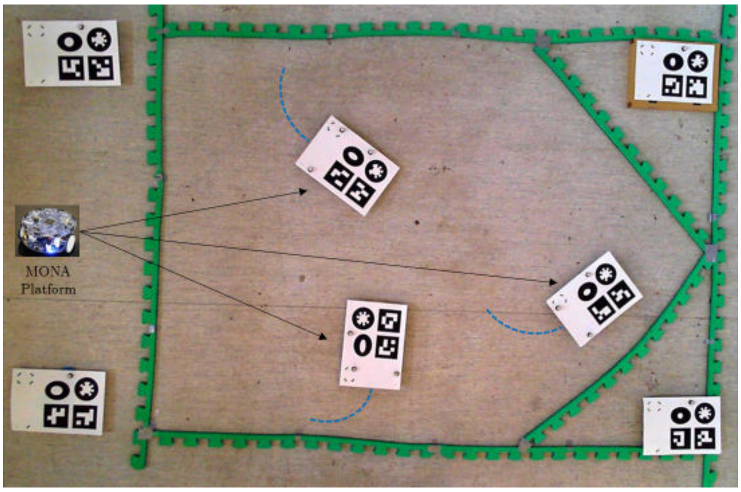


Figure 23: The real-world dataset of the swarm arena with three robots.

## 3.1 Real-world dataset collection

In order to approximate the typical marker application, a real-world robotic dataset has been collected at the University of Manchester. The dataset represents an application of the fiducials in swarm robotics, and so we obtain information about the performance under real-world conditions and camera imperfections. The dataset was recorded by an off-the-shelf web camera observing a swarm arena with three mobile robots moving around carrying a board with the evaluated fiducial markers, see Figure 23. The deployed mobile robots are the MONA [88, 89] swarm robots, which are accessible and easily modifiable robots based on a platform similar to the Arduino Mini. In the corners of the arena, other boards were placed with the markers to provide the coordinate system to align the estimated poses. The size of the individual markers is the same 8.4 cm. Thus, it is the size of the square-base markers' side and the diameter of the circle-based marker. The used ID range of each marker was chosen to contain approximately the same number of unique markers, so there would not be an advantage of choosing a smaller ID range to improve the detection. Thus, the AprilTag uses the $16h5$ encoding family, the WhyCode represents the ID by 8 bits, and the ArUco has $4 \times 4$ 2D code within. As all methods are designed to estimate the pose with the error in lower millimetres, we had to collect the ground-truth measurements by an even more precise device, the Vicon positioning and motion capture system.

The swarm arena has dimensions of 2 and 3 metres, and the recording camera is located around 3 metres above. The web camera does not suffer from significant lens distortion, and the resolution was set to FullHD with 30 frames per second. The distortion model and intrinsic parameters were obtained through the commonly used OpenCV library camera calibration toolbox. The recorded video stream and ground-truth measurements span over four minutes, which results in more than 7000 captured frames and more than 60000 ground-truth positions. The reference position data were captured at a significantly faster frame rate than the images; therefore, it required data stream synchronisation. The high measurement rate is expected for such high-performance localisation systems because it allows finer resolution of the observed movement. In order to align the two streams, the dataset collection started before any robot started driving around and continued even after the robots stopped because those significant moments can be easily spotted in the measured positions and the video stream. Once those two moments were identified, the individual frames could be matched with the nearest corresponding ground-truth position measurement.

## 3.2 Experimental evaluation

We evaluated the mentioned fiducial markers on the presented dataset for the estimation error in position and their execution time required to detect and localise the three moving and four stationary markers. It is beneficial to test the marker methods on the real-world data because it can demonstrate their robustness to the data imperfections. The dataset actually contains several sections where the markers are blurred due to the fast robot motion and sharp turns, which would not be straightforward to model in a simulated environment. Therefore, the evaluation criteria for selecting the marker for application in the multi-camera setup are the ones below.

## 3.3 Position estimation

In the beginning, we focus on the precision of the position estimation, which is essential in both external and also onboard localisation applications. For the purpose of the single-marker evaluation, thus understanding essential performance characteristics, we assess the position estimation using the summarised indicators in Table 1. We evaluated the mean

| | AprilTag | ArUco | WhyCode |
|---|---|---|---|
| Mean | 35.53 | 17.26 | 18.68 |
| Median | 35.35 | 17.45 | 18.98 |
| Std. dev. | 11.56 | 7.66 | 6.15 |

Table 1: Position estimation error [mm]

and median error estimation as the first value implies overall general performance, and the second one signifies the expected value with higher resistance to outliers. Another critical factor is the accuracy of the position measurements, which we express in the form of standard deviation. When examining the errors in the summary table, the AprilTag did not achieve convincing results; actually, it scored the worst. Even though the AprilTag shares similar detection concepts with the ArUco, the estimation errors differ significantly. The ArUco's results indicate higher precision than the WhyCode marker in mean and median comparison. However, the WhyCode might achieve higher accuracy than the ArUco.

## 3.4 Execution time

Another critical aspect of the fiducial marker localisation systems is the ability to process images at a high frame rate. The methods were tested on the same dataset, and the measured time is the duration of the detection and localisation function of the methods. The execution time was measured on the same machine Lenovo X280 with 16GB memory and the processor Intel Core i7-8550U and the operating system, Ubuntu 20.04 LTS, was without any additional load. In Table 2, the individual average durations over the whole dataset to process one image frame are listed together with the speedup relative to the AprilTag marker.

## 3.5 Marker selection

The performed evaluations do not provide a clear candidate for further extension into the multi-camera localisation system. We can either select the ArUco marker, which scored the best in the precision of the position estimation. However, on the other hand, the required time to process an image is related by the nature of the detection algorithm to the scene

| Marker | Time per image [ms] | Speedup [%] |
|---|---|---|
| AprilTag | 30.3 | N/A |
| ArUco | 30.2 | 0.2 |
| WhyCode | 0.9 | 3465.4 |

Table 2: The execution time of the fiducial marker methods. The speedup is relative to the AprilTag

composition as it requires more processing time in cluttered scenes. Another candidate to choose is the WhyCode marker, which does not achieve such a high precision as the ArUco but is more accurate in the estimated positions. The WhyCode can also localise the marker at a much higher frame rate than the other square-based markers. In mobile robotics, the computational resources are always limited either because of other parts of the localisation pipeline running on the robot or because a small power supply source and extensive computation would restrict the operational time of the robot. Taking into account the described setting, the marker for further extension is the WhyCode marker, as it can run at the highest frame rate and at the same time maintains a comparable performance to the ArUco marker. The thorough details of the aforementioned single-camera fiducial marker evaluation were presented in [59].

# 4 Multi-camera localisation system

In this section, the single-camera fiducial marker localisation system is further introduced with the necessary steps to transform it into a multi-camera localisation system. As presented in the previous sections, there is active research and development of many different fiducial markers. Therefore, in Section 3, we evaluated the characteristics and performance of the most popular state-of-the-art methods in order to gain the knowledge of their capabilities on real-world data. The most promising fiducial marker for the application in mobile robotics appears to be the WhyCode circular marker because of its real-time image processing and high localisation precision and accuracy. The marker has already been deployed in various scenarios of high frame-rate tracking in adverse conditions and also as an accessible and reliable ground-truth localisation system for robotic experiments evaluation.

The WhyCode marker is one of the circular patterns with foundations in the WhyCon marker. Therefore, it features minimal requirements like an off-the-shelf web camera and a basic office printer in order to achieve a few millimetre precision while being versatile to be embedded into complex image processing and robotic pipelines. The roots in the WhyCon system are noticeable in the low computational requirements, which was the primary design principle. Thus, despite robotic platforms' growing performance, the WhyCode is still significantly more efficient than the square fiducial markers. The legacy of the WhyCon is in the roundel detection stage presented in [58] and in [60] for the mobile robots swarms. As the detector module caches the estimated parameters from the preceding image frame on the fly, the binarization and seeding of the flood-fill segmentation are taken advantage of rather than evaluating every image pixel for the presence of a marker. The pattern detection is also accelerated by the cascade of relatively simple characteristics tests in order to reject the false positive detections at the early processing stages. The precondition that a marker would not drastically change its position in consecutive image frames, together with the marker parameter caching, allows the real-time tracking of even thousands of markers. Thus, the flood-fill algorithm searches only the local neighbourhood around the previous position, resulting in significantly fewer evaluated pixels.

The specific features of the WhyCode were introduced in [62, 61] where the authors describe the unique encoding system for individual marker identification with the possibility of recovering the missing sixth degree of freedom, the rotation around the marker's surface normal vector. The encoding of distinguishable IDs is performed by embedding a circular binary sequence along the perimeter in-between the black outer and white inner concentric marker segments, see Figure 24. The binary code is in the form of the Manchester encoding in order to improve the black and white area ratio of the marker as it is part of the cascade mentioned above of detection tests. Also, to ensure the uniqueness of the circular code, the binary values are inspired by the Necklace patterns [90] which are invariant to a rotation. Therefore, the decoding of the ID is not dependent on the starting sampling position on the circle, as the extracted binary code is rotated to find the lowest value corresponding to the sought ID. The amount of the binary code shifts to reach the lowest value determines

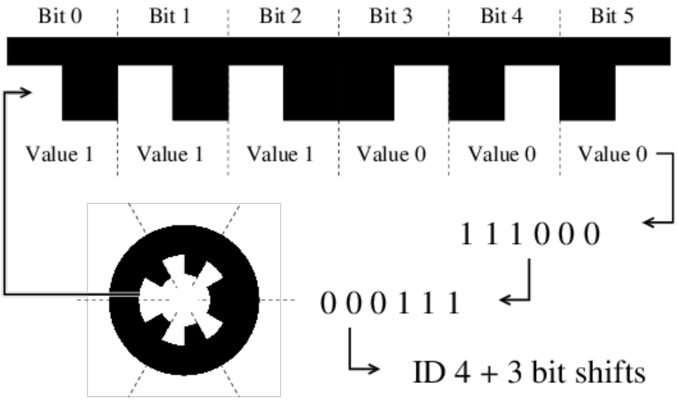


Figure 24: Manchester-encoded Necklace WhyCode ID scheme. The binary code is unrolled and rotated and based on the required shifts, the rotation is recovered [62].

the rotation around the marker's surface normal vector.

To unleash the full potential of the localisation system based on the fiducial markers, it needs to support simultaneous localisation in multiple cameras. The current system support for only a single camera scenario restricts the operational space significantly because it is highly dependent on the camera's field of view, which is traditionally in the range of 60° to 70° for the cameras approximated by the pinhole camera model. Of course, there are cameras capable of an even wider field of view. However, they introduce significant distortion to the image and require a different and more complex approximation camera model. This restriction can be partially overcome by observing the scene from a further distance or using smaller markers; however, in both cases, the available information for the precise marker localisation is reduced due to the lower amount of marker pixels. Therefore, it is beneficial to deploy multiple cameras, each observing a different part of the space with enough overlap between the scenes to support the extrinsic camera calibration procedure. There are two general configurations of their setup when deploying multiple cameras in terms of the captured area. Either they can be positioned to extend the field of view so the covered area would be maximized, or they can share as much observable scene as possible to combine the pose estimation by the individual camera so the estimation error would be lowered.

The following subsections describe the necessary steps to build the multi-camera fiducial-based localisation system. At first, we present the needed modifications of the WhyCode marker detection algorithm to remove the precondition of tracking and to remember the detection parameters. The possible extrinsic calibration approaches are discussed from the perspective of available information provided by the marker system. We introduce the possibilities to increase the methods of pose estimation performance by combining the localisation outputs of the individual cameras. Then, the system overview and design are described separately for the calibration and localisation functionality.

## 4.1 Marker modifications

The current version of the WhyCode localisation algorithm is not suitable for reliable simultaneous detection in multiple images. Such a claim originates in the image thresholding and segmentation method, tailored for single-camera tracking. The method achieves such a high frame rate in situations when the tracked markers do not disappear and reappear in the scene. The applied thresholding is adjusted on the fly from the previously measured marker brightness, and when a marker moves from one camera's field of view to another's, such statistics are not transferable, and it cannot be reliably passed from the previous camera as each camera sensor might observe the scene illumination differently. In terms of marker segmentation, an exhaustive flood-fill approach is used with a starting seed from the marker's previously known image pixel position. It is beneficial only in scenarios when it is known in advance how many markers there are in the scene and that they will not drastically change their position. Otherwise, the detection time is not stable and suffers from significant delays.

### 4.1.1 Thresholding

Two main approaches are available regarding the change in the used thresholding algorithm. Either one global threshold can be applied to the whole image, or each pixel can be thresholded based on a local characteristic [91]. The local adaptive methods have a significant disadvantage for our application, and that is their high dependence on the size of the local neighbourhood around the pixel they evaluate for thresholding. They are more suitable for images where the sought objects, meaning the foreground and background segments, are not expected to change their size, which is not our situation because the marker is expected to move freely in the scene. As the marker can move freely in the space, in situations when it would be close enough to the camera, thus occupying a large area of the image, the local pixel neighbourhood would be smaller than the width of the marker's black outer ring, so the inner ring part would be thresholded wrongly as the local characteristic would be obtained only from the black area. Therefore, it is more suitable to select the global thresholding approach for our purpose. The chosen method is the leading state-of-the-art Otsu algorithm for finding a threshold that maximises the inter-class variance between the foreground and background based on the image brightness histogram [92].

### 4.1.2 Segmentation

The image segmentation task groups together pixels with similar properties to form a general blob or a specific object with apriori known properties [93, 94]. One of the common approaches is to understand the image as a graph where each pixel is a node connected to its immediate four or eight neighbouring pixels. Each node has a specific label; in our thresholded binary image, it is foreground and background, and the nodes are grouped based on their label and presence in a connected component. It is also possible to directly segment a coloured image; however, those methods are more computationally demanding as they need to evaluate the colour information and update the segment characteristics with every newly processed pixel. In order to overcome the current limitations of the flood-fill method, we incorporated the Spaghetti Labeling [95, 96], which is based on an automatically generated directed acyclic graph producing a highly efficient connected component labelling algorithm. The method has been used because it allows scaling from four to eight neighbourhoods, and it is directly transformable into parallel image processing.

## 4.2 Extrinsic calibration

The camera calibration is formed of two main parts, the intrinsic and the extrinsic parameters. The intrinsic parameter matrix describes the coordinate scaling and the perspective projection onto the image plane. The extrinsic matrix $[\mathbf{R} \mid \mathbf{t}]$ represents the transformation of the camera coordinate frame from a particular global world frame. Such transformation does not necessarily have to relate to a world frame but also relatively to another camera coordinate system. Thus, we can construct the individual relative camera transformations to obtain marker poses in the one base camera frame. However, there has to be an overlap of the cameras' fields of view so there would be corresponding measurements in the images. Based on the information provided by the marker localisation system, there are two approaches to determine the relative transformation, either the 3D marker position or the respective image pixel coordinates.

### 4.2.1 Absolute orientation problem

Using the metric position estimation from the local localisation in one camera coordinate frame allows using of the least-square formulation to directly obtain the rotation $\mathbf{R}$ and translation $\mathbf{t}$. Assuming the relation of the $n$ marker positions $\mathbf{p}_i$ in one image and the respective positions $\mathbf{q}_i$ in the base image is as $\mathbf{q}_i = \mathbf{R}\mathbf{p}_i + \mathbf{t}$, we can find the sought rotation and translation as a solution to the following minimization

$$\mathbf{R}^*, \mathbf{t}^* = \operatorname*{arg\,min}_{\mathbf{R} \in SO(3), \mathbf{t} \in \mathbb{R}^3} \sum_{i=1}^{n} \|\mathbf{R}\mathbf{p}_i + \mathbf{t} - \mathbf{q}_i\|^2, \tag{8}$$

In [75], the authors describes a closed form solution to directly obtain the sought parameters through the singular value decomposition as follows

$$\mathbf{p}'_i = \mathbf{p}_i - \underbrace{\frac{1}{n}\sum_{i=1}^{n} \mathbf{p}_i}_{\overline{\mathbf{p}}}, \quad \mathbf{q}'_i = \mathbf{q}_i - \underbrace{\frac{1}{n}\sum_{i=1}^{n} \mathbf{q}_i}_{\overline{\mathbf{q}}} \tag{9}$$

$$\mathbf{H} = \sum_{i=1}^{n} \mathbf{p}'_i \mathbf{q}'^T_i = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{10}$$

$$\mathbf{R}^* = \mathbf{V}\mathbf{U}^T, \quad \mathbf{t}^* = \overline{\mathbf{q}} - \mathbf{R}^*\overline{\mathbf{p}} \tag{11}$$

### 4.2.2 Essential matrix decomposition

We can also use the image pixel positions of the local marker localisation to approach the extrinsic calibration from the epipolar geometry point of view. The 3D structure can be recovered from the essential matrix, which embeds the individual camera frame transformations from the world frame into them as $\mathbf{E} = \mathbf{R}_2[\mathbf{C}_2 - \mathbf{C}_1]\mathbf{R}_1$ where $\mathbf{R}_i, \mathbf{C}_i$ are the respective camera centres in the world frame and rotations to the camera frames as in the construction of the camera projection matrices. Thus, it is necessary at first to construct the essential matrix, thoroughly described in [76, 77]. The decomposition yields four sign combinations for the relative translation and rotation between the cameras. The world position vectors of the image points can be calculated using the camera projective matrix and the pixel coordinates. Then, individual solutions can be distinguished by the restriction on the world position of the used points as they have to be in front of the cameras and within their field of view. The world position vectors are tested by their reprojection onto the image using the camera projection matrices constructed from the four solutions pairs and the intrinsic matrices.

### 4.2.3 Method's approach

In our approach for the extrinsic calibration, the least-square absolute orientation problem is chosen to estimate the transformation between the camera frames. It uses the pose estimation directly in the local camera coordinate system and the marker ID for establishing the correspondences. Compared to the essential matrix decomposition, it is more efficient because the correct pixel position of a marker is strongly dependent on the estimated metric position of the marker. It is because the method requires the exact matching points, and thus we cannot use only the average pixel position of the roundel shape in the image obtained in the early detection phase. Actually, the pixel position of the marker centre is the reprojection of the obtained metric position onto the image plane. Then, it would require constructing and decomposing the essential matrix as described above. Thus, it would be not only more resource-demanding, but also it would still be highly dependent on the firstly established metric position of the marker in space. For such reasons, it is more beneficial to approach the extrinsic calibration through the least-square minimization of squared differences.

## 4.3 Localisation improvements

The fiducial marker localisation using multiple cameras can bring more advantages than only enlarging the field of view. It can also improve the accuracy and reliability of the detection. Given multiple detections of a marker in multiple cameras, it is possible to improve further the pose ambiguity resolution, which is in a single-camera approach based only on the circular binary code sampling and evaluating the code shape. Also, the possible estimation error might be reduced by fusing together the individual estimated poses. Another improvement is the higher resistance to marker occlusion because when a marker would be covered in one of the cameras, the other cameras at different positions could still detect the marker.

### 4.3.1 Ambiguity resolution

The marker's pose estimation is an ambiguous process as we are trying to recover the 3D pose information from the 2D projection. Of course, one could object that it can be mitigated with a higher resolution camera which would provide more information about the pixel properties of the conic section. However, it can be approached by the simultaneous localisation in multiple cameras because it is less likely that the ambiguity would be undecidable in all of the camera images. When the marker pose is obtained, the surface normal vectors are also recovered as parts of the solutions. The normal vector associated with the assumed correct pose estimation can be transformed into the frame of the other camera, which also detects the same marker. Then, the directions of the two normal vectors are compared. If they have similar orientations, the ambiguity in the estimation has been resolved at the first local stage successfully. In the case of different orientations, one of the estimates must be considered a correct one, and the other must be changed to the second ambiguous solution to match the direction of the base normal vector. To assess which normal vector is the right one, we propose to select the one which is the furthest from the image centre, the principal point, because such a marker is affected the most by the applied perspective, and thus it should provide the most information for the already existing ambiguity resolution based on the circular binary code. Then, the other detections from the other cameras are flipped based on the difference in their normal vector orientations compared to the chosen base normal vector.

### 4.3.2 Averaging estimations

Once all of the marker detections are in accordance with the above, the individual pose estimations can be fused together instead of selecting an arbitrary solution. Each detection and localisation from the cameras is burdened with an unknown level of uncertainty imposed by the image acquisition errors or imperfect processing of the image pixels. However, the corresponding marker poses can be transformed into the base coordinate frame and combined in order to distribute the estimation error among the other solutions. The approach is relatively straightforward in terms of position estimation, and all the position vectors can be averaged to find the most likely position estimation. Unfortunately, the orientation fusing of the fiducial is a little bit more complex. There are several approaches to the orientation averaging depending on the used representation and also the similarity or closeness of the individual orientations. The first option is to approximate the individual representation elements by their mean value. Such an approach is only an approximation, and it is not a robust nor correct solution to the problem; however, it may lead to a likely estimate when all of the orientations are close enough to each other. Based on the representation, whether we use a rotational matrix or a quaternion, we have to choose an appropriate method. In [97], the author presents two approaches to finding the mean 3D rotation matrix. However, the use of rotational matrices introduces unwanted singularities in expressing a rotation and also, and the construction of a rotation matrix from an axis-angle representation, which is the base representation obtained by the marker method, is more complicated. The quaternion averaging is thoroughly explained in [98], where the authors calculate the average of $n$ quaternions as the normalized eigenvector associated with the largest eigenvalue of the $4 \times 4$ matrix

$$\mathbf{Q} = \sum_{i=1}^{n} w_i \mathbf{q}_i \mathbf{q}_i^T, \tag{12}$$

where the $w_i$ is the possible weight of the i-th quaternion; however, the weights are normalized, to sum up to one. The quaternion approach to average the orientation of the multiple estimations is prefered due to the straightforwardness and the growing popularity in robotics as it is the primary representation of rotation in the Robot Operating System.

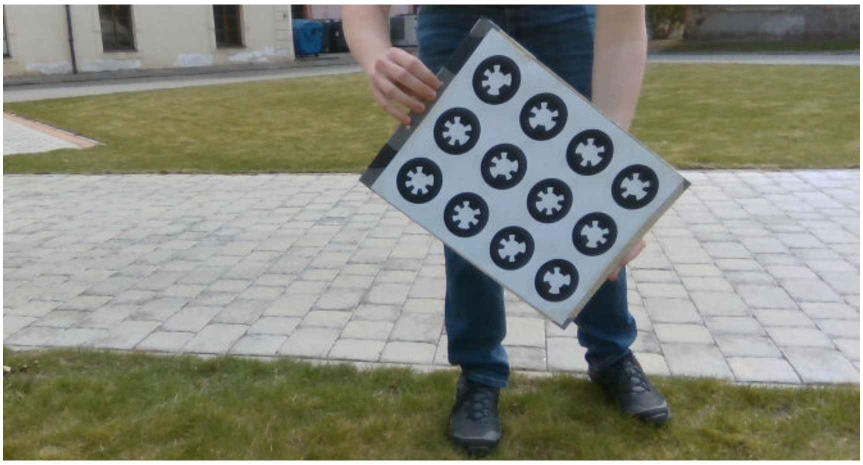


Figure 25: Calibration marker board for extrinsic calibration

## 4.4 System overview

The multi-camera localisation pipeline composes of three main parts, the local single-camera localisation, the extrinsic calibration and the actual marker localisation, see Figure 26. The initial stage for the pose estimation in the individual camera frames is shared for the two latter parts because it accepts the captured images from the cameras and localises the markers independently in each of them. It also provides both the possible solutions for every marker and an indication of which one is more likely, based on the local ambiguity decision. However, one has to ensure the image capture synchronization; otherwise, the markers could move in-between the captures; and therefore, the images would not represent the same marker pose leading to a possible decrease in precision. Then, the extrinsic camera calibration can follow unless the system is provided by the user with the transformation parameters acquired by different external methods. Because the calibration is performed by a marker board, see Figure 25, a supplemental ambiguity resolution can be applied because all the markers on the board must have similar directions of the surface normal vectors. Thus, the orientation shared among most markers is assumed to be the correct one and the markers not satisfying it have to change the pose estimation to the second one from the ambiguity pair. The markers are then matched based on their IDs to form correspondence pairs between the cameras, and the calibration is performed. The localisation procedure can no longer expect the common marker plane, so the ambiguity resolution cannot be applied immediately. At first, the assumed pose is transformed into the base camera frame. Then, multiple estimations of the marker orientation are compared, and if they do not correspond within a predefined threshold, the other pose estimate has to be selected. To determine which measurement is incorrectly resolved, we propose a comparison based on the marker position in the image because we expect the marker further from the image centre to be more affected by the perspective projection and thus provide better information for the initial sing-camera ambiguity resolution. As presented above, the results are averaged to estimate the marker's most likely pose.

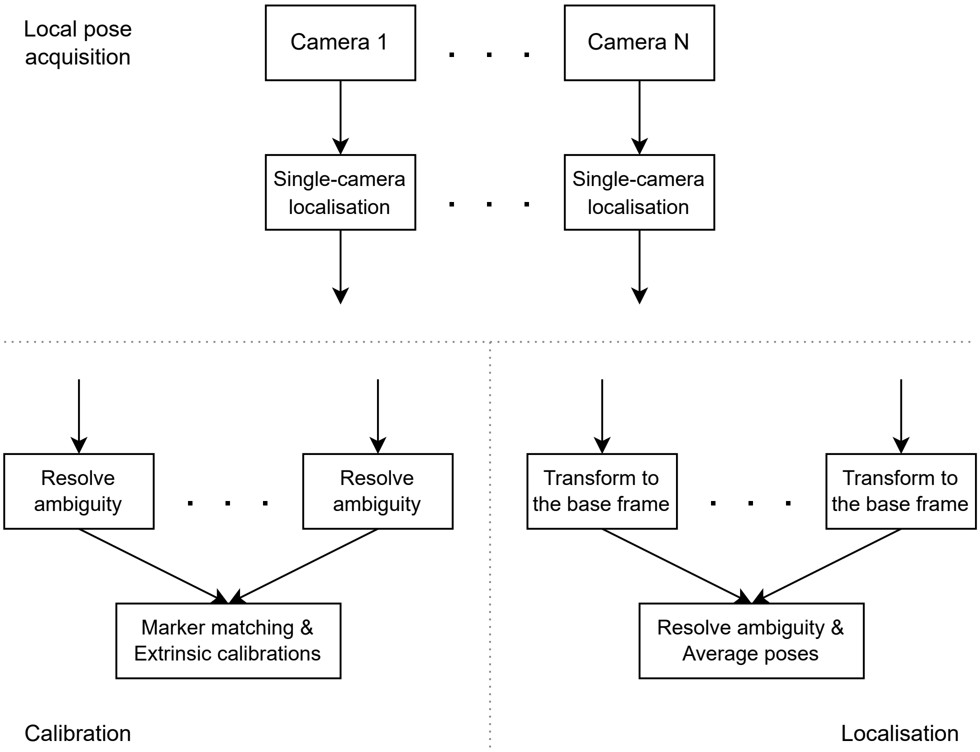

Local pose acquisition
Camera 1
Camera N
Single-camera localisation
Single-camera localisation
Resolve ambiguity
Resolve ambiguity
Marker matching & Extrinsic calibrations
Transform to the base frame
Transform to the base frame
Resolve ambiguity & Average poses
Calibration
Localisation

Figure 26: Multi-camera system overview

# 5 Datasets

Two types of datasets were collected to evaluate the localisation system performance after the extension and the modifications of the detection module. Because the fiducial marker can estimate the pose with high precision, it brings higher requirements on the ground-truth measurements because they need to be obtained with even higher precision. Therefore, the first dataset type was recorded in an synthetic environment that simulates the camera sensor and observed scene. It allows a flexible and repeatable data collection with a perfect image capture without noise. Also, the exact poses of individual elements are apriori known because it is computer-generated on demand. The second dataset we test the method on is a real-world robotic dataset to gain knowledge of the methods' behaviour in real conditions. However, it required additional methods to establish the mutual camera poses and measure the marker position precisely.

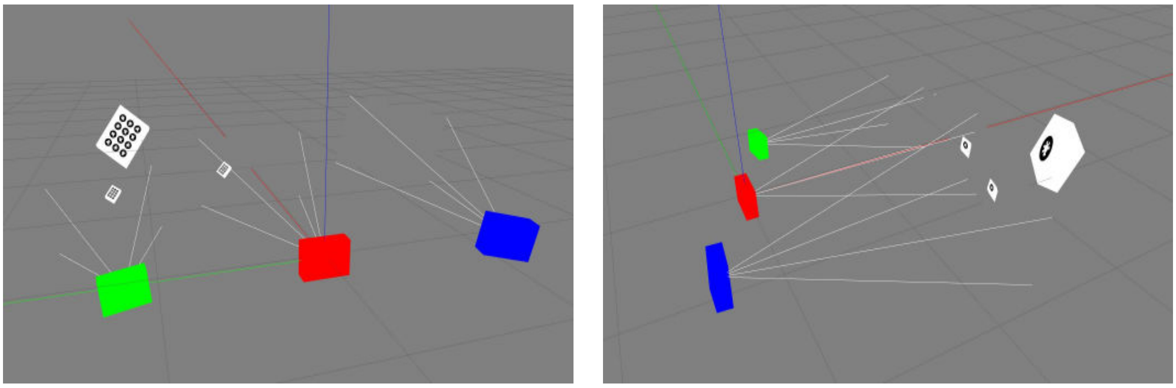
## 5.1 Synthetic dataset

The simulation to generate the synthetic dataset was performed using the Gazebo framework [99] which is one of the leading simulation platforms with a reliable graphics engine. It has an interface to the industry's popular Robot Operating System and provides comprehensible and flexible scene configuration with a wide range of available tools and sensor emulators. Two testing scenarios were created which correspond to the introduced possible multi-camera configurations. Thus one represents the extension of the field of view, and the other is to test the pose improvement when the scenes are highly overlapped. The data generation is programmatically coordinated through a central application that on-demand captures the images by individual cameras and then passes them to the localisation system. Also, the coordinator displaces the observed marker and records its ground truth and estimated position for further evaluation. Because we want to simulate a perfect environment, the used cameras are all the same, and they do not have any noise set, and their distortion is omitted. The size of the marker is set equally in both the simulated scenarios.

The world description originates in the empty base world with only minor changes. There is no directional light applied, and the ambient light is set to the maximum value in order to prevent any shadow casting over the markers. The camera sensor is provided through the standard Gazebo camera sensor with the additional gazebo_ros_trigger_camera plugin to capture the image when intended rather than at a fixed frame rate. The image resolution is set to 1280 × 720 pixels for all the cameras. The fiducial marker and the extrinsic calibration marker board are constructed by the fundamental visual element with box geometry and applied image of the marker as the PNG image. The WhyCode marker was generated with 7-bit encoding for both the localised marker and also the calibration board, where are twelve markers in a 4 × 3 matrix configuration.

### 5.1.1 Extend the field of view scenario

The world configuration consists of three cameras and one marker, as in Figure 27. The base camera frame is considered the coordinate frame of the middle camera, which is positioned at the centre of the world frame. The two other cameras are placed arbitrarily around it, also with different orientations to extend the field of view. The used marker has 20 cm in diameter, which is inspired by the maximal reasonable size possible to be printed on the A4 paper format. The marker poses are generated randomly by a uniform distribution in lateral direction ranging $[-3, 3]$ meters, the vertical $[-1, 1]$ meters, and the distance $[3, 5]$ meters. We also sampled the marker's orientation from a uniform distribution covering all rotations around the surface normal, and the other two angles were drawn from the same range $[-1, 1]$ radians. Evaluating the method on this dataset, we examine how well it can transform in-between the individual camera frames together with the performed modifications to the marker.



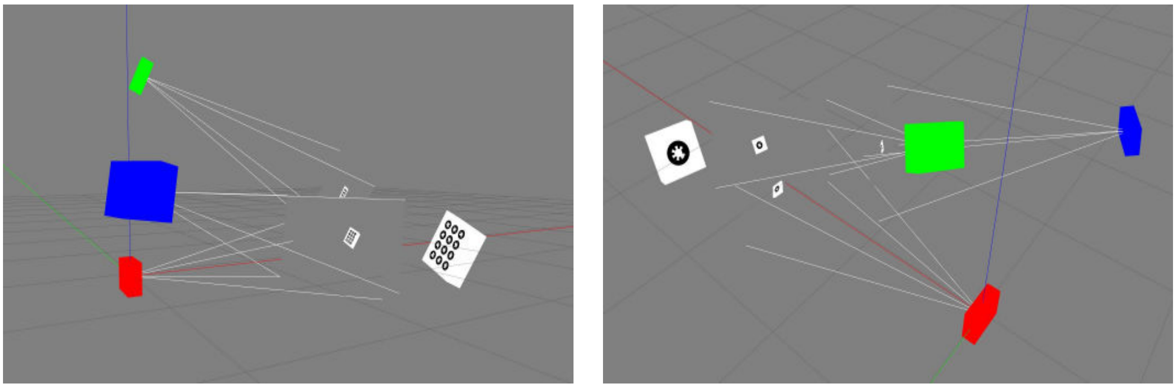
(a) Calibration procedure

(b) Marker localisation

Figure 27: Extend the filed of view scenario overview

### 5.1.2 Overlap scenario

In this scenario, the world setup is equivalent to the large-area scenario in terms of the used visual models and their parameters. However, the camera poses are different for the two cameras that are not at the origin of the world coordinate frame. They are positioned so their observations would highly overlap; thus, we can focus on the evaluation of the pose averaging capabilities as in Figure 28. The positions for the marker are sampled again from uniform distributions. The distance varies from 3 meters to 6 meters, and the horizontal and vertical displacement is in a range of $[-1, 1]$ and $[-0.5, 0.5]$ meters, respectively. In terms of the orientation, the rotations around the y and z world axes range $[-1, 1]$ radians, while the revolution around the normal vector covers the whole $2\pi$.



(a) Calibration procedure

(b) Marker localisation

Figure 28: Overlap scenario overview

## 5.2 Real-world dataset

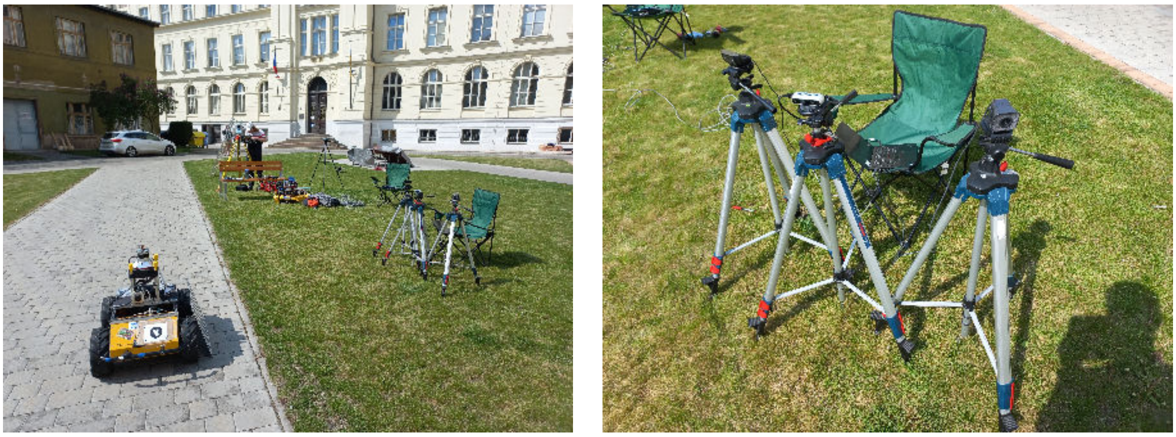
The real-world dataset was collected to evaluate our method's performance under conditions the simulated dataset cannot easily provide. The dataset represents a possible application of the proposed system for localisation in an outdoor environment over a large area, see Figure 29. The used robotic platform is the Husky A200 mobile robot equipped with additional sensors and the computational payload necessary for operation. In order to measure precisely the robot's position, the geodetic 360° crystal was mounted on top of the robot, and it was tracked by the Leica TS-16 positioning system, which provides the crystal position in the custom-defined 3D space with the precision below a millimetre. The measuring frequency were 8 Hz in average. As the data collection was done at the Czech Technical University in Prague's courtyard, the use of RTK-GPS for position measurements was not possible because of the closeness of the surrounding buildings.

The multi-camera system is highly heterogeneous as it consists of the Intel RealSense D435, Logitech C980, and Logitech C920 cameras positioned next to each other with a step of approximately 0.5 m and different orientations. The first two cameras recorded the video streams at the resolution of 1920 × 1080 and the latter one at 1280 × 720. As all the camera recording was done on one laptop, the frame rate was not stable, and occasional dropping of frames occurred. Also, the Logitech C920 camera suffered the most from the limited bandwidth and achieved only ten frames per second compared to the 30 frames of the other cameras. The cameras' intrinsic parameters were estimated by the widely used tool from the Robot Operation System framework, which relies on the computer vision library OpenCV. The used calibration board for extrinsic parameter estimation had markers of the same size 7.767 cm, and it featured 7-bit ID encoding. The localised marker was placed visibly on the robot passing by the cameras at approximately 3 m distance. The marker has 7.327 cm in diameter, and the ID is encoded into a 5-bit binary code.

In order to assure the correct synchronization of the captured images and the ground-truth position, a local NTP server was set up. Thus, the recording systems could label the data with synchronized time stamps. The individual camera streams were then manually checked and verified to assure proper synchronization. In terms of the total station measurements, they were obtained at a lower frame rate than the camera images, and therefore, the time alignment was performed in the nearest neighbour manner.

## 5.3 Data generation

The experiment evaluation, data generation, and recording were performed on the same machine based on the Ubuntu 20.04.4 LTS operating system with the Linux kernel 5.13. During the data gathering and evaluation, the Robot Operating System at version Noetic was used as it provides unified and widely used middleware for robotic platforms. The employed simulator for the synthetic experiments was Gazebo at version 11.10.2. In terms of the codebase, it was written primarily in C++11 and partially in Python 3.8. Some parts of the codebase depend on the OpenCV, which is a popular library for matrix operations and image processing, the version used was 4.2. The used computer is the Lenovo X280 laptop equipped with Intel Core i7-8550U CPU and system memory of 16 GB, and a storage device connected over M.2 PCIe interface.



(a) Overview of the used equipments and the experiment environment

(b) Cameras for recording. Left to right: Logitech C920, Intel RS D435, Logitech C980

Figure 29: Real-world dataset collection setup

# 6 Experiments

In this section, the proposed multi-camera system is evaluated on the aforementioned datasets to assess the performance compared to the single-camera method. The evaluation of the synthetically generated dataset provides insights into the theoretical capabilities of the presented system and whether the core idea leads us in a positive direction. However, it is essential to ask what is the method capable of under real-world conditions like imperfect intrinsic calibration, naturally occurring noise and image distortion. These are the main affecting elements which differentiate the evaluation results of the simulated and real-world experiments.

The evaluation starts with the synthetic dataset, where we focus on the pose estimation performance in the perfect environment, thus noise-free images with perfect information about the camera model. First, we focus on the synthetic scenario of overlapping scenes to extensively test the pose averaging influence on the position and orientation estimation. We evaluate the quality of the extrinsic calibration because we know them apriori from the simulator. We compare the 3D position estimation error of the presented method and the single-camera method, and also the orientation estimation separately. Those tests can provide us with an intuition of how well the current localisation capabilities combine with the estimated extrinsic transformation, which results in averaging the resulting individual poses. Further, we evaluate the extended field of view observable by the multi-camera configuration and assess the gain in area coverage thanks to the extended field of view. Then, the real-world dataset is used to obtain the position estimation error and understand how the method could behave in an actual deployment. Statistical tests were employed because we needed to compare the multiple and single-camera configuration errors.

## 6.1 Statistical evaluation

In order to statistically compare the obtained estimation errors and differentiate which are significantly smaller, we decided to employ the Student's t-test to compare the estimated mean error values. The number of error samples is large enough to provide sufficient information about the methods' behaviour. As we have enough samples, we can assume the normal distribution of the errors. To compare the two mean error values, the paired t-test is used where we assume that the random variable is the position or the orientation error tight to a specific image. Our primary hypothesis to test is whether the sample means equals or not. If rejected, we can proceed to one-sided interval testing to perform an ordering on the errors if even that hypothesis would be rejected. The confidence level of the performed tests is $p = 0.05$.

To perform the statistical testing, we have to compute the sample mean and sample standard deviation, which is straightforward regarding the position errors because the errors are in metres that can be just averaged. However, this approach cannot be directly applied to the measured orientation errors because they are angles which are circular variables; thus, a different approach had to be taken. Therefore, we used the directional statistics [100] methods to estimate the sample mean as follows

$$\bar{\theta} = \operatorname{arctan2}\left(\frac{1}{n}\sum_{i=1}^{n}\cos\theta_i, \frac{1}{n}\sum_{i=1}^{n}\sin\theta_i\right), \tag{13}$$

where $\theta_i$ is the orientation error, thus the shortest angle between the orientation samples, and $n$ is the number of samples. The deviation is calculated as follows

$$\begin{aligned} S &= \sqrt{-2\ln R} \\ R &= \left\| \frac{1}{n}\sum_{i=1}^{n}(\cos\theta_i + i\sin\theta_i) \right\|, \end{aligned} \tag{14}$$

where $n$ is the number of samples, $\theta_i$ is the orientation error, and $R$ is the norm of the averaged vector in the complex plane.

The angles for the evaluations represent the distance between the quaternions, the primary representation of the orientation in the systems. In [101], the authors explain that the difference between two quaternions is represented as the smallest rotation angle to transform from one to the other. The quaternion distance can be calculated as

$$\Phi(\mathbf{q_1}, \mathbf{q_2}) = \arccos|\mathbf{q_1} \cdot \mathbf{q_2}|, \tag{15}$$

where $q_1, q_2$ are unit quaternions and to restrict the possible negative results of the vector dot poduct in arccos, the absolute value restricts the $\Phi$ values from 0 to $\pi/2$ radians.

## 6.2 Overlap field of view scenario

The following experiment focuses on improving the pose estimation by averaging the local estimates of the individual cameras. It is an excellent representation of the second possible multi-camera configuration because they share most of their fields of view. The evaluation results can signify whether the position and orientation of the marker are being averaged correctly and whether the second level of ambiguity resolution is based on the correct assumption that the distance to the camera centre shall decide. The experiment is inspired by the application in the swarm robotic experiment evaluation because they are usually held in an indoor arena which can be surrounded by cameras.

To start with, the extrinsic transformation parameters were tested as the correct transformation is the key to estimating the pose as an averaged estimate correctly. The differences between the rotation and translation given to the synthetic simulator and the transformation obtained by the calibration procedure are presented in Table 3. The estimation errors appear to be sufficiently small apart from the translation between the camera 1 and 3 because 2.7 cm might be large enough to influence the other position estimates during the averaging phase.

| | Translation | | Rotation | |
|---|---|---|---|---|
| | Absolute [m] | Relative [%] | Absolute [rad] | Relative [%] |
| Cameras 1 ↔ 2 | 0.0098 | 0.7161 | 0.0011 | 0.3749 |
| Cameras 1 ↔ 3 | 0.0274 | 1.5777 | 0.0030 | 1.1184 |

Table 3: Extrinsic calibration errors of the overlap scenario

The detection stability can be evaluated even though the observed area is shared rather than enlarged. The single-camera failed to detect the marker in 31 cases out of 3000 uniformly sampled poses over the described configuration space. Those poses were significant in the extreme marker orientation; thus, the single-camera method was not able to reliably detect and localise the marker. However, the multi-camera system successfully detected the markers at all of the tested poses, which is the benefit of observing the same place from different viewing angles.

The main performance criteria of the presented system and the single-camera system, the position and orientation errors, are visualized in Figure 30. The position estimation improved significantly with the usage of multiple cameras, and the estimation is more accurate than using only one camera. The orientation error did not show a significant improvement over the single-camera system, and we could not reject the equality hypothesis.

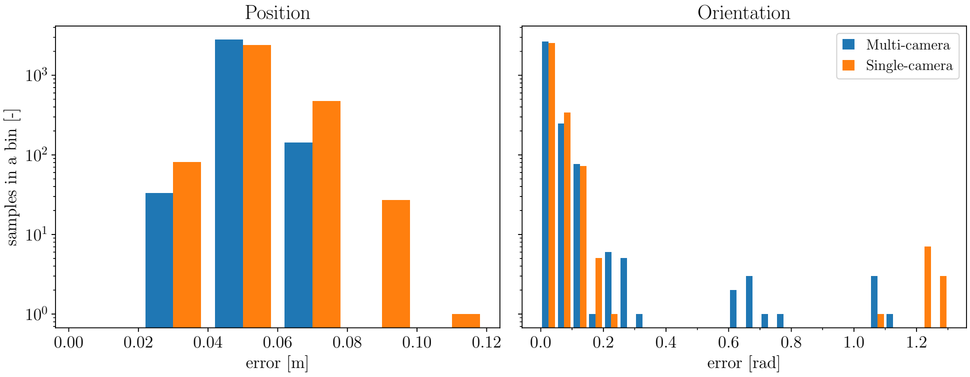


Figure 30: Histograms of the overlap scenario localisation errors with 0.02 m bins for position and 0.05 rad bins for angles. Vertical axis is in logarithmic scale of decadic base

## 6.3 Extended field of view scenario

In this tested scenario, both the multiple and single-camera systems are thoroughly examined for the position and also orientation estimation errors. Even though the environment is synthetic, the gained understanding of the behaviour of the methods can provide hints for further development and also about their maximal possibilities when deployed in

| | Translation | | Rotation | |
|---|---|---|---|---|
| | Absolute [m] | Relative [%] | Absolute [rad] | Relative [%] |
| Cameras 1 ↔ 2 | 0.0059 | 0.4647 | 0.0010 | 1.2589 |
| Cameras 1 ↔ 3 | 0.0167 | 1.5375 | 0.0019 | 1.5416 |

Table 4: Extrinsic calibration errors of the extended scenario

the real world. The experiment represents one of the two main possible configurations of the multi-camera system, which is to cover larger areas than one would be able with only the single-camera method. The difference between the expected and measured pose of the marker is a general performance measure which assesses all of the individual parts of the involved image processing, from the pixel thresholding to the ambiguity resolution.

We focus on the precision of the camera extrinsic calibration because the rest of the evaluation relies on it. After all, if the system cannot correctly estimate those parameters, the localised markers would be wrongly transformed to the base camera coordinate frame. In Table 4, there are summarized the differences between the expected extrinsic parameters provided to the simulator and the estimated one by the multi-camera system. The orientation estimation results in an insignificant difference, and even the position achieved difference small enough to be considered sufficient.

Further, we can evaluate the number of successful detections in the sampled space to demonstrate the benefit of deploying multiple cameras. We uniformly sampled 3000 random marker poses covering the introduced synthetic environment. The single-camera configuration failed to detect the fiducial in 467 cases, while the proposed system did not succeed only in 12 situations. Detecting 18% more marker configurations clearly demonstrates better space coverage.

The extended field of view scenario can also be utilized to test the performance of a hybrid situation when markers are located in the overlap between the cameras but not necessarily observed by all of them. We assume that the performance would follow the evaluation of the overlapping scenario, and thus in the overlapping region, the marker would be localised more precisely. Figure 31 presents the following important performance factors, the position and the orientation estimation error distribution of the compared methods. The visualized histogram of errors represents such marker poses when detected by the single-camera method and the multi-camera method. The multi-camera position estimation improved significantly, and the standard deviation of errors decreased by half of the single-camera value. The orientation estimation results are comparable to the single-camera estimation, with a slightly lower mean error but statistically undecidable. Therefore, in this camera configuration, one gains not only the better operational space coverage but also the improved pose estimation in the overlapped regions.

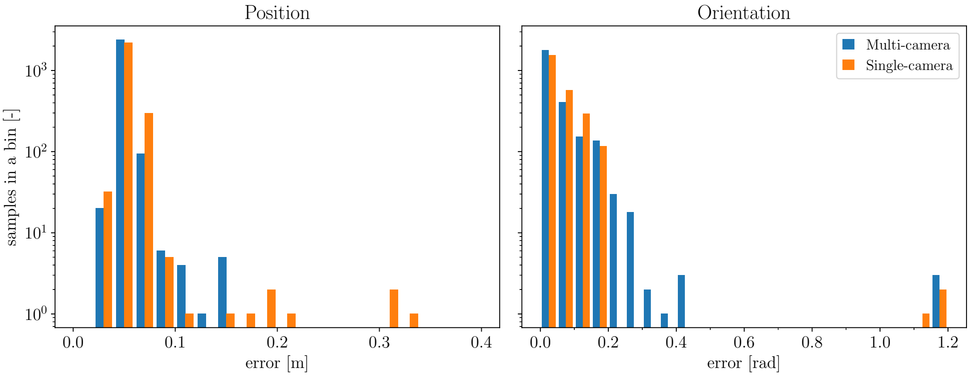


Figure 31: Histograms of the extended scenario localisation errors with 0.02 m bins for position and 0.05 rad bins for angles. Vertical axis is in logarithmic scale of decadic base

## 6.4 Real-world experiment

Evaluating the multi-camera system on the real-world dataset is crucial for assessing the performance under natural conditions that one would encounter when deploying the system. In this experiment, we focus only on the position estimation because the ground-truth measurement device could not estimate the orientation. However, it is not degrading the provided information of this evaluation because, in many situations, the position localisation is sufficient for a given application. The experiment extensively tested the system's ability to handle a highly heterogeneous system composed of three different cameras with dissimilar image resolutions.

### 6.4.1 Position estimation

The performance of the extended field of view configuration were discussed in the previous sections. We verified that the system could localise the marker by multiple cameras without losing the single-camera pose estimation performance. Thus, in the real-world experiment, we decided to focus on the part of the dataset where the individual cameras' fields of view overlap. We assume a similar improvement in the position estimation error as in the simulated scenarios. As the cameras were placed next to each other with 0.5 m step, we compared the position estimated by the middle camera with the fused estimated from all the cameras. Thus, we focused only on the overlapping regions where there was a single-camera detection, and at the same time, at least two of the cameras detected the marker in the multi-camera system. Evaluating such regions would allow us to verify the assumption that even in real-world conditions, the combination of more cameras leads to better estimations.

Table 5 presents the key position error distribution characteristics of the evaluated methods. The multi-camera achieved a significantly lower estimation error compared to the

| | Single-camera | Multi-camera |
|---|---|---|
| Median | 0.0685 | 0.0391 |
| Mean | 0.0664 | 0.0427 |
| Std. dev. | 0.0347 | 0.0265 |
| Min | 0.0022 | 0.0034 |
| Max | 0.1562 | 0.1130 |

Table 5: Position estimation errors in the overlapped regions of the real-world dataset. The listed values are in metres.

single-camera method. When examining the key characteristics, the overall performance of the presented method is better as the lower median error signifies even lower error omitting outliers, and the lower standard deviation implies higher accuracy. Unfortunately, the range of the measured errors was not reduced, but only their distribution improved.

### 6.4.2 Computational performance

Another critical performance measure is the computational performance to detect and localise the fiducial marker. For multi-camera localisation, processing the images reasonably fast and at a stable frame rate is essential. We evaluated the presented modified detection approach on the real-world dataset with HD images and compared it to the original method. We have to distinguish two situations, whether the marker is present in the image or not, because the original detection method takes advantage of locally tracking the marker. Therefore it does not have to process all of the image pixels. However, it suffers from the threshold selection procedure. Once a threshold is selected, the image is searched. If nothing is found, the threshold is modified, and the search starts again. Thus, to assess the execution time on the images, we let the aforementioned selection procedure run sixteen times to find the proper threshold.

We averaged the individual measured times based on the presence of the marker. The original method took 84.17 ms to output that no marker was present in the image. However, once the marker was found and the tracking could be initialised, it required only 0.49 ms to detect and localise the marker. The modified variant of the detection algorithm runs for 10.91 ms on images without the marker and 15.7 ms when it can find it.

## 6.5 Experimental results summary and discussion

To wrap up the results of individual extensive experimental evaluations on various datasets, each of the tested scenarios has to be considered because they represent the essential multi-camera configurations and applications. Judging the performance just on one of them might lead to a false understanding of the localisation system capabilities and reliability. If the system would establish the extrinsic transformation parameters correctly but then would not be able to localise the marker or vice versa, the whole reason for extending the single-camera method would be meaningless.

After thoroughly examining in performance outcomes of the experiments, we can conclude that the multi-camera localisation system based on the fiducial marker method managed to reliably and with a high precision estimate the marker pose over a large area. Also, all pose estimates were based on the estimated extrinsic calibration by the method itself rather than dividing the evaluation into separate stages and providing the system with transformations obtained through external tools. Two main spacial configurations of the cameras were evaluated. The highly overlapping scenario demonstrated the system's ability to improve the pose estimation significantly and even make the detection more reliable as the single-camera method suffered from poorer detection of highly rotated markers. The configuration for extending the field of view provided us with the opportunity to observe large areas without a loss in the estimation precision together with the hybrid functionality of improved localisation in the regions where the cameras observe the same scene. Thus, deploying a multi-camera localisation system is more beneficial than using only a single camera.

# 7 Conclusion

This thesis aimed to design and propose a new multi-camera localisation system based on the detection and localisation of fiducial markers. The system can process multiple image sources and localise a fiducial marker in them, and output the best estimate of the pose based on fusing the multiple estimates to achieve higher accuracy. The system also allows the extrinsic calibration of the individually calibrated cameras, which results in no other image processing tools to be incorporated in order to obtain additional parameters. The fiducial marker the system is built around is the state-of-the-art WhyCode marker which is a highly versatile and real-time localisation system using a circular black-and-white pattern for detection and the six degrees of freedom estimation. However, the WhyCode system can only process and localise the marker in a single camera. More importantly, it has to be provided with the number of markers in a scene in advance. The number of visible markers should not change. The flexible spacial configuration of the used cameras presents two main deployment setups; either the cameras can be positioned to have only minimal overlap; thus, the effective field of view would be maximised, or the observed scene can be shared among the cameras as much as possible which allows the pose estimates fusion and results in higher accuracy.

The limitation of the single-camera WhyCode of requiring the number of occurring markers has to be resolved prior to the multi-camera extension. Otherwise, only the configuration scenario with highly overlapping fields of view would be possible. Therefore, the detection core of the system was inspected, and necessary modifications were proposed. First, the pixel thresholding had to be changed from the local approach to the global one, and then also the segmentation had to undergo a similar scope change. Thus, the connected component labelling over the whole image replaced the local flood fill algorithm. Performing those modifications, the image is processed completely and therefore, the restriction of the apriori knowing number of markers to search for is overcome.

The next step was to find the extrinsic parameters of the individual cameras so the poses could be transformed into one base camera coordinate frame. The calibration is performed by a custom pattern formed from multiple different WhyCode markers on a calibration board which is moved around in the fields of view to establish mutual correspondence points between the cameras. Then, the relative rotation and translation of the cameras are estimated by the least-square minimization of the euclidean distance between the point correspondences.

The actual localisation from multiple cameras uses the pose estimates in the local coordinate frames of the used cameras. Each pose is then transformed into one based camera frame where they are compared and checked by the second level of ambiguity resolution because the individual transformed orientations should have the same direction. If not, the second from the ambiguous pair is selected. The transformed poses are then adequately averaged to provide a more likely estimate of the fiducial marker pose.

In order to evaluate the proposed modifications and the overall performance and capabilities of the multi-camera system, we evaluated it on three different datasets. The first two were generated artificially by a computer simulator, while the third was collected by real-world cameras and a ground-truth positioning system. The simulated datasets represent the two main system configurations, the extension of space coverage and the estimation improvement of the highly overlapped scene. Based on the experimental evaluation and comparison to the original single-camera WhyCode, the multi-camera system provided higher accuracy in the overlapping scenario and maintained comparable results in the other scenario. The real-world dataset represents a hybrid setup, thus extending the field of view and also providing reasonable overlap, whose evaluation showed that the system is also capable of desired performance even when using different off-the-shelf cameras integrated into one localisation system.

In future works, reducing the dependency on synchronized cameras could increase the system performance because delayed frames would not affect the pose averaging. One could approach it by introducing a motion model of the tracked object and estimating the pose through the extended Kalman filter. Another aspect to focus on is the computational requirement which grows with every added camera. Thus, it would be beneficial to transfer the algorithm to the GPGPU devices and take advantage of the parallel processing. Apart from the technical modifications, we noticed a higher level of false-positive detections whose rejection at earlier stages of single-camera localisation would decrease the required work in the multi-camera system. The integration of the presented system into the currently most popular frameworks in robotics, Robot Operating System 2 and OpenCV, would broaden the target group of potential users.

# 8 References

[1] Sebastian Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21–71, 1998.

[2] Hans P. Moravec. Sensor fusion in certainty grids for mobile robots. In *Sensor Devices and Systems for Robotics*, pages 253–276. Springer, 1989.

[3] Alberto Elfes. Sonar-based real-world mapping and navigation. *IEEE Journal on Robotics and Automation*, 3(3):249–265, 1987.

[4] David Kortenkamp and Terry Weymouth. Topological mapping for mobile robots using a combination of sonar and vision sensing. In *American Association for Artificial Intelligence*, volume 94, pages 979–984, 1994.

[5] Benjamin Kuipers and Yung-Tai Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and Autonomous Systems*, 8(1-2):47–63, 1991.

[6] Sebastian Thrun and Arno Bücken. Integrating grid-based and topological maps for mobile robot navigation. In *Proceedings of the National Conference on Artificial Intelligence*, pages 944–951, 1996.

[7] Frédéric Chenavier and James L Crowley. Position estimation for a mobile robot using vision and odometry. In *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pages 2588–2589. IEEE Computer Society, 1992.

[8] Waveshare. 10 dof imu sensor, low power. `https://www.waveshare.com/10-dof-imu-sensor-c.htm`, 2022. Accessed: 2022-05-01.

[9] Billur Barshan and Hugh F Durrant-Whyte. Inertial navigation systems for mobile robots. *IEEE Transactions on Robotics and Automation*, 11(3):328–342, 1995.

[10] Seeed Studio. Grove 3-axis digital compass module. `https://www.seeedstudio.com/Grove-3-Axis-Digital-Compass-V2.html`, 2022. Accessed: 2022-05-01.

[11] Wikimedia Commons. File:south-pointing chariot (science museum model).jpg — wikimedia commons, the free media repository, 2021. Accessed: 2022-05-01.

[12] Surachai Suksakulchai, Siripun Thongchai, D Mitchell Wilkes, and Kazuhiko Kawamura. Mobile robot localization using an electronic compass for corridor environment. In *IEEE International Conference on Systems, Man and Cybernetics*, volume 5, pages 3354–3359. IEEE, 2000.

[13] Tomáš Rouček, Martin Pecka, Petr Čížek, Tomáš Petříček, Jan Bayer, Vojtěch Šalanský, Daniel Heřt, Matěj Petrlík, Tomáš Báča, Vojěch Spurný, et al. Darpa subterranean challenge: Multi-robotic exploration of underground environments. In *International Conference on Modelling and Simulation for Autonomous Systems*, pages 274–290. Springer, 2019.

[14] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986.

[15] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[16] Edward Rosten, Reid Porter, and Tom Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):105–119, 2008.

[17] Tomáš Krajník, Pablo Cristóforis, Keerthy Kusumam, Peer Neubert, and Tom Duckett. Image features for visual teach-and-repeat navigation in changing environments. *Robotics and Autonomous Systems*, 88:127–141, 2017.

[18] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European Conference on Computer Vision*, pages 404–417. Springer, 2006.

[19] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571. IEEE, 2011.

[20] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European Conference on Computer Vision*, pages 778–792. Springer, 2010.

[21] AR Jimenez, F Seco, R Ceres, and L Calderon. Absolute localization using active beacons: A survey and iai-csic contributions. *Institute for Industrial Automation, CSIC Madrid*, 2004.

[22] Jun Qi and Guo-Ping Liu. A robust high-accuracy ultrasound indoor positioning system based on a wireless sensor network. *Sensors*, 17(11):2554, 2017.

[23] Margrit Betke and Leonid Gurvits. Mobile robot localization using landmarks. *IEEE Transactions on Robotics and Automation*, 13(2):251–263, 1997.

[24] Adam Chrzanowski and Gottfried Konecny. Theoretical comparison of triangulation, trilateration and traversing. *The Canadian Surveyor*, 19(4):353–366, 1965.

[25] Vicon — award winning motion capture systems. `https://www.vicon.com/`. Accessed: 2022-05-01.

[26] Pti phoenix technologies 3d motion capture systems. http://www.ptiphoenix.com. Accessed: 2022-05-01.

[27] Pierre Merriaux, Yohan Dupuis, Rémi Boutteau, Pascal Vasseur, and Xavier Savatier. A study of vicon system positioning performance. *Sensors*, 17(7):1591, 2017.

[28] Optitrack - motion capture systems. https://optitrack.com/. Accessed: 2022-05-01.

[29] Bruce Carse, Barry Meadows, Roy Bowers, and Philip Rowe. Affordable clinical gait analysis: An assessment of the marker tracking accuracy of a new low-cost optical 3d motion analysis system. *Physiotherapy*, 99(4):347–351, 2013.

[30] Josep Aulinas, Yvan Petillot, Joaquim Salvi, and Xavier Lladó. The slam problem: a survey. *Artificial Intelligence Research and Development*, pages 363–371, 2008.

[31] Andrew J Davison and David W Murray. Simultaneous localization and map-building using active vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):865–880, 2002.

[32] J Leonard and P Newman. Consistent, convergent, and constant-time slam. In *Proceedings of the 18th international joint conference on Artificial intelligence*, pages 1143–1150, 2003.

[33] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, et al. Fastslam: A factored solution to the simultaneous localization and mapping problem. *Innovative Applications of Artificial Intelligence Conference*, 593598, 2002.

[34] Tomáš Krajnık. *Large-scale mobile robot navigation and map building.* PhD thesis, Czech Technical University in Prague, 2011.

[35] Jorge Fuentes-Pacheco, José Ruiz-Ascencio, and Juan Manuel Rendón-Mancha. Visual simultaneous localization and mapping: a survey. *Artificial Intelligence Review*, 43(1):55–81, 2015.

[36] Takafumi Taketomi, Hideaki Uchiyama, and Sei Ikeda. Visual slam algorithms: a survey from 2010 to 2016. *IPSJ Transactions on Computer Vision and Applications*, 9(1):16, 2017.

[37] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007.

[38] Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *European Conference on Computer Vision*, pages 834–849. Springer, 2014.

[39] Jakob Engel, Jörg Stückler, and Daniel Cremers. Large-scale direct slam with stereo cameras. In *Proceedings 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1935–1942. IEEE, 2015.

[40] David Caruso, Jakob Engel, and Daniel Cremers. Large-scale direct slam for omnidirectional cameras. In *Proceedings 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 141–148. IEEE, 2015.

[41] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós. Orb-slam: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015.

[42] R. Mur-Artal and J. D. Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017.

[43] Rafael Muñoz-Salinas and Rafael Medina-Carnicer. Ucoslam: Simultaneous localization and mapping by fusion of keypoints and squared planar markers. *Pattern Recognition*, 101:107193, 2020.

[44] Rafael Munoz-Salinas, Manuel J Marín-Jimenez, and Rafael Medina-Carnicer. Spm-slam: Simultaneous localization and mapping with squared planar markers. *Pattern Recognition*, 86:156–171, 2019.

[45] Seongin Na, Yiping Qiu, Ali E Turgut, Jiří Ulrich, Tomáš Krajník, Shigang Yue, Barry Lennox, and Farshad Arvin. Bio-inspired artificial pheromone system for swarm robotics applications. *Adaptive Behavior*, page 1059712320918936, 2020.

[46] Dengqing Tang, Tianjiang Hu, Lincheng Shen, Zhaowei Ma, and Congyu Pan. April-tag array-aided extrinsic calibration of camera–laser multi-sensor system. *Robotics and Biomimetics*, 3(1):1–9, 2016.

[47] Jan Bacik, Frantisek Durovsky, Pavol Fedor, and Daniela Perdukova. Autonomous flying with quadrocopter using fuzzy control and aruco markers. *Intelligent Service Robotics*, 10(3):185–194, 2017.

[48] Patrick Irmisch. Camera-based distance estimation for autonomous vehicles. Master's thesis, Technische Universität Berlin, 2017.

[49] Boston Dynamics. Handle / otto integration. `https://www.youtube.com/watch?v=yVRAxpAjFrY`, 2020. Accessed: 2022-05-01.

[50] Edwin Olson. Apriltag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation*, pages 3400–3407. IEEE, 2011.

[51] John Wang and Edwin Olson. Apriltag 2: Efficient and robust fiducial detection. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4193–4198. IEEE, 2016.

[52] Andrew Richardson, Johannes Strom, and Edwin Olson. Aprilcal: Assisted and repeatable camera calibration. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1814–1821. IEEE, 2013.

[53] Maximilian Krogius, Acshi Haggenmiller, and Edwin Olson. Flexible layouts for fiducial tags. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1898–1903. IEEE, 2019.

[54] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel Jesús Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014.

[55] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Rafael Medina-Carnicer. Generation of fiducial marker dictionaries using mixed integer linear programming. *Pattern Recognition*, 51:481–491, 2016.

[56] Francisco J Romero-Ramirez, Rafael Muñoz-Salinas, and Rafael Medina-Carnicer. Speeded up detection of squared fiducial markers. *Image and vision Computing*, 76:38–47, 2018.

[57] Francisco J Romero-Ramire, Rafael Munoz-Salinas, and Rafael Medina-Carnicer. Fractal markers: a new approach for long-range marker pose estimation under occlusion. *IEEE Access*, 7:169908–169919, 2019.

[58] Tomáš Krajník, Matías Nitsche, Jan Faigl, Petr Vaněk, Martin Saska, Libor Přeučil, Tom Duckett, and Marta Mejail. A practical multirobot localization system. *Journal of Intelligent & Robotic Systems*, 76(3-4):539–562, 2014.

[59] Jiří Ulrich, Ahmad Alsayed, Farshad Arvin, and Tomáš Krajník. Towards fast fiducial marker with full 6 dof pose estimation. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, pages 723–730, 2022.

[60] Jan Faigl, Tomáš Krajník, Jan Chudoba, Libor Přeučil, and Martin Saska. Low-cost embedded system for relative localization in robotic swarms. In *2013 IEEE International Conference on Robotics and Automation*, pages 993–998. IEEE, 2013.

[61] Peter Lightbody, Tomáš Krajník, and Marc Hanheide. A versatile high-performance visual fiducial marker detection system with scalable identity encoding. In *Proceedings of the Symposium on Applied Computing*, pages 276–282, 2017.

[62] Peter Lightbody, Tomáš Krajník, and Marc Hanheide. An efficient visual fiducial localisation system. *ACM SIGAPP Applied Computing Review*, 17(3):28–37, 2017.

[63] Farshad Arvin, Tomáš Krajník, Ali Emre Turgut, and Shigang Yue. Cos$\phi$: artificial pheromone system for robotic swarms research. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 407–412. IEEE, 2015.

[64] Joseph DeGol, Timothy Bretl, and Derek Hoiem. Chromatag: A colored marker and fast detection algorithm. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1472–1481, 2017.

[65] Filippo Bergamasco, Andrea Albarelli, Emanuele Rodola, and Andrea Torsello. Runetag: A high accuracy fiducial marker with strong occlusion resilience. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 113–120. IEEE, 2011.

[66] Bryce E Bayer. Color imaging array, July 20 1976. US Patent 3,971,065.

[67] Wikimedia Commons. File:pinhole-camera.svg — wikimedia commons, the free media repository. `https://commons.wikimedia.org/w/index.php?title=File:Pinhole-camera.svg&oldid=561212900`, 2021. Accessed: 2022-05-01.

[68] Wikimedia Commons. File:1755 james ayscough.jpg — wikimedia commons, the free media repository. `https://commons.wikimedia.org/w/index.php?title=File:1755_james_ayscough.jpg&oldid=621088028`, 2022. Accessed: 2022-05-01.

[69] Wikimedia Commons. File:barrel distortion.svg — wikimedia commons, the free media repository. `https://commons.wikimedia.org/w/index.php?title=File:Barrel_distortion.svg&oldid=451292315`, 2020. Accessed: 2022-05-01.

[70] Wikimedia Commons. File:pincushion distortion.svg — wikimedia commons, the free media repository. `https://commons.wikimedia.org/w/index.php?title=File:Pincushion_distortion.svg&oldid=549555962`, 2021. Accessed: 2022-05-01.

[71] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision.* Cambridge University Press, 2003.

[72] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[73] Janne Heikkila and Olli Silvén. A four-step camera calibration procedure with implicit image correction. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1106–1112. IEEE, 1997.

[74] Wikimedia Commons. File:epipolar geometry.svg — wikimedia commons, the free media repository. `https://commons.wikimedia.org/w/index.php?title=File:Epipolar_geometry.svg&oldid=606461844`, 2021. Accessed: 2022-05-01.

[75] K Somani Arun, Thomas S Huang, and Steven D Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5):698–700, 1987.

[76] Richard I Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997.

[77] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770, 2004.

[78] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[79] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pages 145–152, 2001.

[80] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[81] Philip HS Torr and Andrew Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156, 2000.

[82] Seongin Na, Mohsen Raoufi, Ali Emre Turgut, Tomáš Krajník, and Farshad Arvin. Extended artificial pheromone system for swarm robotic applications. In *Artificial Life Conference Proceedings*, pages 608–615. MIT Press, 2019.

[83] Martin Saska, Tomas Baca, Justin Thomas, Jan Chudoba, Libor Preucil, Tomas Krajnik, Jan Faigl, Giuseppe Loianno, and Vijay Kumar. System for deployment of groups of unmanned micro aerial vehicles in gps-denied environments using onboard visual relative localization. *Autonomous Robots*, 41(4):919–944, 2017.

[84] Guo Zhenglong, Fu Qiang, and Quan Quan. Pose estimation for multicopters based on monocular vision and apriltag. In *2018 37th Chinese Control Conference*, pages 4717–4722. IEEE, 2018.

[85] Ju Wang, Chad Sadler, Cesar Flores Montoya, and Jonathan CL Liu. Optimizing ground vehicle tracking using unmanned aerial vehicle and embedded apriltag design. In *2016 International Conference on Computational Science and Computational Intelligence*, pages 739–744. IEEE, 2016.

[86] Ho Chuen Kam, Ying Kin Yu, and Kin Hong Wong. An improvement on aruco marker for pose tracking using kalman filter. In *2018 19th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, pages 65–69. IEEE, 2018.

[87] Mohammad Fattahi Sani and Ghader Karimian. Automatic navigation and landing of an indoor ar. drone quadrotor using aruco marker and inertial sensors. In *2017 International Conference on Computer and Drone Applications*, pages 102–107. IEEE, 2017.

[88] Farshad Arvin, Jose Espinosa, Benjamin Bird, Andrew West, Simon Watson, and Barry Lennox. Mona: an affordable open-source mobile robot for education and research. *Journal of Intelligent & Robotic Systems*, 94(3-4):761–775, 2019.

[89] Zheyu Liu, Craig West, Barry Lennox, and Farshad Arvin. Local bearing estimation for a swarm of low-cost miniature robots. *Sensors*, 20(11):3308, 2020.

[90] William YC Chen and James D Louck. Necklaces, mss sequences, and dna sequences. *Advances in Applied Mathematics*, 18(1):18–32, 1997.

[91] Mehmet Sezgin and Bülent Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13(1):146–165, 2004.

[92] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

[93] King-Sun Fu and JK Mui. A survey on image segmentation. *Pattern Recognition*, 13(1):3–16, 1981.

[94] Lifeng He, Xiwei Ren, Qihang Gao, Xiao Zhao, Bin Yao, and Yuyan Chao. The connected-component labeling problem: A review of state-of-the-art algorithms. *Pattern Recognition*, 70:25–43, 2017.

[95] Federico Bolelli, Stefano Allegretti, Lorenzo Baraldi, and Costantino Grana. Spaghetti labeling: Directed acyclic graphs for block-based connected components labeling. *IEEE Transactions on Image Processing*, 29:1999–2012, 2019.

[96] Federico Bolelli, Stefano Allegretti, and Costantino Grana. One dag to rule them all. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP, 2021.

[97] Maher Moakher. Means and averaging in the group of rotations. *SIAM Journal on Matrix Analysis and Applications*, 24(1):1–16, 2002.

[98] F Landis Markley, Yang Cheng, John L Crassidis, and Yaakov Oshman. Averaging quaternions. *Journal of Guidance, Control, and Dynamics*, 30(4):1193–1197, 2007.

[99] Gazebo. `http://gazebosim.org/`. Accessed: 2022-05-01.

[100] N. I. Fisher. *Statistical Analysis of Circular Data.* Cambridge University Press, 1993.

[101] Du Q Huynh. Metrics for 3d rotations: Comparison and analysis. *Journal of Mathematical Imaging and Vision*, 35(2):155–164, 2009.

# Appendix

## Data carrier content

In Table 6, the top level directories are listed with brief content description.

| **Directory name** | **Description** |
|---|---|
| calib_bag_imgs | Real-world dataset images for extrinsic calibration |
| cameras_bag_imgs | Real-world dataset images for marker localisation |
| report | Total station measurements and images time stamps |
| rosbag_reader | Tool to publish or decompose rosbags on demand |
| simulation_and_coordination | Gazebo simulation description of worlds and models.<br>Main programs to perform the experiments. |

Table 6: Data carrier content