The use of low cost optical technologies for the mapping of production environments

by

Marius Michael Greger

Thesis presented in fulfilment of the requirements for the degree of Master of Engineering in the Faculty of Engineering at Stellenbosch University. This thesis has also been presented at Reutlingen University in terms of a double-degree agreement.

Supervisor: Prof L Louw

Co-supervisor: Mr KH von Leipzig

Co-supervisor: Prof D Palm

April 2019

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification. This thesis has also been presented at Reutlingen University in terms of a double-degree agreement.

April 2019

Abstract

The importance of indoor positioning systems (IPS) and object recognition systems has increased in recent years, especially with regard to the digitization of processes and workflows. These systems enable a new kind of automation in the localization and identification of objects and play an important role in the digitalization of the industry. (Liu *et al.*, 2007) However, IPS are not widespread due to their cost issues.

In particular, small and medium-sized enterprises (SMEs), which are globally considered as drivers of the economy, are struggling with the digital transformation of their enterprises. Compared to large companies, they do not have the necessary resources and capacities to deal intensively with this topic. In addition, there is a lack of best practice experience and tangible examples of implementation in many areas.

The aim of this work is therefore to develop a precise, flexible and yet cost-effective IPS that can be used for a variety of application scenarios.

In this thesis, the underlying requirements for IPS in industrial environments are systematically analysed. Based on the knowledge gained, a concept is derived, on the basis of which a demonstrator is then developed.

In order to evaluate the solution with regard to the requirements, an extensive theoretical and experimental analysis is carried out on the designed demonstrator.

The result is a flexible, accurate and cost-effective IPS that can meet most of the requirements set out in this thesis. The accuracy requirement of less than 1 cm could be reached at working distances of less than 1 m and at larger working distances a lower accuracy of less than 5 cm was achieved. With this thesis, it was possible to

prove that high performance values can be realized with a cost-effective IPS that can cover a wide range of industrial applications.

Keywords

Optical Indoor Localization Systems, Cameras, Marker-based Tracking, Low Cost

Opsomming

Die belangrikheid van binnenshuise posisioneringstelsels (BPS) en voorwerpherkenningstelsels het die afgelope jaar toegeneem, veral ten opsigte van die digitalisering van prosesse en werkstrome. Hierdie stelsels maak 'n nuwe soort outomatisering in die lokalisering en identifisering van voorwerpe moontlik en speel 'n belangrike rol in die digitalisering van die bedryf. (Liu *et al.*, 2007). BPS is egter nie wydverspreid nie weens hul hoë koste.

Dit is veral klein- en middelgrootte ondernemings (KMO's), wat wêreldwyd beskou word as belangrike drywers van die ekonomie, wat sukkel met die digitale transformasie van hul ondernemings. In vergelyking met groot maatskappye het hulle nie die nodige hulpbronne en vermoëns om intensief met hierdie onderwerp te handel nie. Daarbenewens is daar 'n gebrek aan beste praktykervaring en tasbare voorbeelde van implementering op baie gebiede.

Die doel van hierdie werk is dus om 'n akkurate, buigsame en tog koste-effektiewe BPS te ontwikkel wat gebruik kan word vir 'n verskeidenheid toepassings scenario's.

In hierdie tesis word die onderliggende vereistes vir BPS in industriële omgewings sistematies ontleed. Gebaseer op die kennis sodoende opgedoen, word 'n konsep afgelei, waaruit 'n demonstrator dan ontwikkel word.

Ten einde die oplossing met betrekking tot die vereistes te evalueer, word 'n uitgebreide teoretiese en eksperimentele analise uitgevoer op die demonstrator.

Die resultaat is 'n buigsame, akkurate en koste-effektiewe BPS wat aan die meeste vereistes voldoen wat in hierdie tesis uiteengesit word. Die vereiste akkuraatheid van

minder as 1 cm kan bereik word op werkafstand van minder as 1 m en by groter werkafstande is 'n laer akkuraatheid van minder as 5 cm bereik. Met hierdie tesis was dit moontlik om te bewys dat hoë prestasiewaardes met 'n koste-effektiewe BPS gerealiseer kan word wat 'n wye reeks industriële toepassings kan dek.

Acknowledgements

At this point I would like to thank everyone from the bottom of my heart who accompanied, supported and endured me during the writing of this Master's thesis:

- First of all I would like to thank my supervisors Prof. Daniel Palm, Prof. Vera Hummel, Mr. Julian Hermann, Prof. Louis Louw and Mr. Konrad von Leipzig for their support, helpful suggestions, guidance and constructive criticism during the writing of this thesis.
- The staff of ESB LLF for always offering me a place, patience and willingness to help. Many thanks for the numerous discussions and ideas, which were an important input for this thesis. In particular, I would like to thank Mr. Vinu Vijayakumaran Nair for his patience and support.
- My family, for their continued love, support and encouragement
- My friends, for their support.

Thank You!

The Author

January 2019

Table of Contents

List of Figures

List of Tables

List of Acronyms

1D	1-Dimensional
2D	2-Dimensional
3D	3-Dimensional
AOV	Angle Of View
API	Application Programming Interface
AVG	Autonomous Guided Vehicle
CCD	Charge-Coupled Device
CFL	Compact Fluorescent Lamps
COMS	Complementary Metal-Oxide-Semiconductor
CPS	Cyber Physical System
CPU	Central Processing Unit
CSI	Camera Serial Interface
DD	Direct Detection
DIN	Deutsches Institut für Normung
EAN	European Article Number
FOV	Field Of View
fps	Frames Per Second
GNSS	Global Navigation Satelite System

GPS	Global Positioning System
IM	Intensity Modulation
IPC	Industrial Personal Computer
IPS	Indoor Positioning System
IR	Infrared
ISO	International Organization for Standardization
IT	Information Technology
LCD	Low Cost Digitalization
LCIA	Low Cost Intelligent Automation
LED	Light-Emitting Diode
LOS	Line of Sight
MP	Mega Pixle
NoIR	No Infrared
OIPS	Optical Indoor Positioning System
OpenCV	Open Source Computer Vision Library
PC	Personal Computer
SBC	Single Board Computer
SME	Small and Medium Enterprise
USB	Universal Serial Bus
VDE	Verband der Elektrotechnik Elektronik Informationstechnik
VDI	Verein Deutscher Ingenieure

VDMA	Verband Deutscher Maschinen- und Anlagenbau
VLC	Visible Light Communication
WD	Working Distance
WLAN	Wireless Local Area Network
Wi-Fi	Wireless Fidelity

Chapter 1

Introduction

1.1 Background

The continuous location of objects in industrial premises by means of indoor positioning systems (IPS) is one of the main enablers for the realization of diverse innovative application scenarios within the framework of Industry 4.0 (Aktas *et al.*, 2017).

The use of IPS thus enables a new degree of automation in object localization and represents an incremental component of future factory structures in which transparency and flexibility play a decisive role. Real applications can be found in numerous areas (Liu *et al.*, 2007).

Global navigation satellite systems (GNSS), such as GPS, are among the best-known localization systems: they enable to determine a position based on the determination of the transit time of signals transmitted by satellites. The positioning accuracy of about 5 meters is possible with GPS (van Diggelen and Enge, 2015). However, satellite-based positioning systems are very limited in their applications, as the signals transmitted cannot penetrate walls (Fritsche and Klein, 2009). Based on this limitation of GPS and the necessity of positioning in buildings, a new field of research has opened up in recent years, which investigates alternative methods for positioning within buildings.

These open up new possibilities in factories. Companies are already relying on driverless transport systems (AGVs) which, by automating work processes and acting autonomously, lead to significantly reduced error rates while at the same time increasing productivity. The core component of this system is the recording of their current position, which can be used to calculate the current route (Bubeck *et al.*, 2014): However, the areas of application extend far beyond those of the AGV, thus creating a new degree of transparency on the shop floor level, in which it is possible at any time to determine, where workpieces, load carriers or tools are located.

However, the manufacturing industry in particular is showing considerable pent-up demand in the implementation of concepts for data acquisition and evaluation (Zillmann, 2016).

Today, the corresponding industry 4.0 technologies are more frequently used by large companies, since they have the necessary means to evaluate technologies at an early stage and to easily assess their economic potential. SMEs, which make up the largest part of the global economy, have difficulties to meet the challenge and are still regarded as relatively cautious (Schröder, 2016).

As Köppe (2014) has already shown, there is still a need for further research in the field of indoor localization. Only a few systems are available on the market today that are either uneconomical, inflexible or not suitable for use in various environments. In addition, the achievement of the required accuracies as well as the short range of the available systems represent a challenge. In addition, rapid technological progress and the miniaturization of computing power mean that new possibilities for indoor localization are constantly emerging. In addition, the high cost factor often poses a great challenge for the implementation of digitization projects in the company. Frequently, there are also no concrete use cases, based on which an added value can be proven through the implementation. Furthermore, the majority of these systems are very specific and developed for a specific field of application and require IT specialists for implementation and maintenance. (Saam, Viete and Schiel, 2016)

1.2 Rationale of the research

In order to meet the requirements of industry 4.0, a high degree of adaptability of the production system is required (Bauernhansl, 2016). In the context of Industry 4.0, there is a focus on the so-called Smart Factory, in which people, machines, objects and products are intelligently networked with each other through the provision of hardware and software. This enables increasing automation and autonomous reaction to unpredictable events (Zillmann, 2016). For this purpose, embedded systems are integrated into objects, devices, buildings, transportation means as well as into production facilities and logistics components, whereby these become cyber-physical systems (CPS). These systems, also known as intelligent objects, can then communicate autonomously and decentrally with each other and optimize themselves. The result is a virtual image of reality, which is continuously updated with the help of real-time data. (Bauernhansl, Hompel and Vogel-Heuser, 2014)

Westkämper (2013a, 2013b) considers the ability to unambiguously identify and localize objects and the individuality of systems, processes, objects and products as well as the controllability of the increasing complexity to be particularly important.

A variety of technologies are available for the unambiguous identification and localization of objects, which differ in their degree of maturity and their performance. (Brena *et al.*, 2017; Lymberopoulos *et al.*, 2015; Sakpere, Adeyeye Oshin and Mlitwa, 2017). This thesis deals with the use of optical technologies for mapping production environments, relying on camera-based technologies for mapping. (Klopschitz *et al.*, 2010; Mautz and Tilch, 2011). The term *mapping* refers to the identification and positioning of an object.

1.3 Research problem statement and questions

The underlying problem of this thesis is that there is no known – and cost-effective (low cost) – solution on the market today, that enables the identification and localization of both homogeneous and heterogeneous objects in production environments. In general, there is no standard solution for this area. (Hirtle and Bahm, 2015; Springer, 2012; Zadganokar and Chandak, 2016). However, in the context of industry 4.0, it is repeatedly pointed out how important it is to increase transparency and that future production systems must be equipped with additional intelligence, which to a large extent insists on the identification and determination of objects in production systems (Dünkler, 2015). The question rises, for example, whether the use of conventional technology can already enable a sufficiently precise location to be determined, based on the application of technologies and processes from adjacent specialist areas. The following research questions are to be answered with the present work:

Which possibilities of optical object recognition are relevant in an industrial environment?

- Which applications of optical localization systems are there in industrial use?
- What requirements arise from the industrial use of optical localization systems?
- Which parameters influence the introduction of optical object recognition?
- What processes and technologies are optical localization systems based on?
- Are optical localization systems based on low cost components relevant alternative to conventional systems available on the market?

1.4 Research objectives

The proposed project investigates the application of image processing for the mapping of production environments to gain the knowledge for the development of a demonstrator by making use of low cost technologies.

The following objectives are pursued in this work:

I. Conduct a literature study on the following relevant topics:
   a. Industry 4.0
   b. Previous implementations of optical indoor localization systems
   c. Optical indoor localization and its use in production environments

II. Determination of the system components of an optical indoor localization system and their features.

III. Determination of the requirements for object recognition and indoor localization in production environments, based on the results from l

IV. Design and construction of a demonstrator

V. Verification of the demonstrator against the determined requirements

VI. Recommendation of the possible follow-up work based on the work presented in this thesis, which may be pursued in future.

1.5 Research contribution

This work contributes to the research area of image processing approaches for indoor localization and has the aim to develop a concept based on low cost components, which can be applied in industrial areas while providing sufficient use.

1.6 Research design

The research design of this thesis differs for each of the objectives, Table 1 provides a brief summary of the research design for each objective.

Table 1 Research design

<table>
<tr><th></th><th colspan="2">Objective l- lll</th><th>Objective lV- V</th></tr>
<tr><td>Research Philosophy</td><td colspan="3">Pragmatism</td></tr>
<tr><td>Research Approach</td><td colspan="2">Deductive
(Qualitative Data)</td><td>Inductive</td></tr>
<tr><td>Research Strategy</td><td colspan="2">Applied Research</td><td>Design Science</td></tr>
<tr><td>Data Acquisition</td><td colspan="2">Data, Documents (text, numerical, speech), Graphical modelling</td><td>Measurements, simulation and testing</td></tr>
<tr><td>Data Analysis</td><td>Thematic Data Analysis</td><td>Conceptual Description</td><td></td></tr>
</table>

After Saunders et al. (2009) the research design follows a philosophy of pragmatism. This is because both a deductive and an inductive approach are pursued.

To achieve the goals l-lll, a deductive approach is used, since applied research is conducted to analyse the relevant theory. Based on the literature studied, valuable conclusions are then drawn.

In order to meet the objectives lV and V, the approach is switched to inductive, herein design sciences are used to develop a prototype based on the findings.

1.6.1 Research methodology

This section describes the proposed methodology for achieving the individual objectives of this project.

In regard to Objective 1, the available literature is examined, this is in order to create a better understanding of the state of the art technologies and methods within industry. In addition, a market analysis is taken, to get further information of available systems and their characteristics and requirements.

In a next step, the literature is used to determine and compare appropriate technologies, methods, algorithms and software, therefore Objectives 2 and 3 can be fulfilled. Thereby requirements will also be derived and evaluated with industrial experts.

After the literature analysis has been carried out, a concept is derived. Based on this concept and the already obtained information, the necessary technologies are determined and a proof of concept is carried out. Afterwards, a demonstrator will be developed, for this reason so a more detailed technology selection will be taken. This is in order to fulfil Objective 4.

To fulfil Objective 5, the demonstrator will be implemented within the learning factory of the Reutlingen University.

1.7 Delimitations and limitations

In the context of this thesis a localization system is designed, which is intended exclusively for the use in the interior, thus should be able to be used for as broad a field of application as possible. The industrial environment represents the application domain of the solution to be developed. A production hall or a warehouse is therefore the typical application area for which the localization system is developed and verified.

The localization system should be able to detect the position of dynamic localization objects.

For the dynamic view, forklifts are regarded as the fastest moving localization object and therefore their usual speed in production facilities is used as a benchmark.

Basing on the previous chapters the main objective of this thesis is the development of a cost-effective overall solution. Solutions available on the market are too cost-intensive in many applications or for many users. This is especially true if there are a large number of localization objects in the localization system. Since the costs and the range of an optical localization system increase depending on the localization infrastructure used, the cost optimization of the costs per m^2 is defined as the optimization target.

Furthermore, only optical localization methods are considered in this thesis. These have already proven themselves in the field of logistics (Essati, 2013, 2013; Logivations, no date; TotalTrax, 2018) and are robust against reflections on metallic surfaces and scattering or absorption by liquids in comparison to radio-based locating methods (Mautz, 2012).

1.8 Ethical implications of the research

This study is based on data available in the public sector or produced by the author himself. Therefore, all relevant passages and the thoughts of others are highlighted by providing their proper source, as prescribed by the guidelines of the Stellenbosch University and Reutlingen University.

1.9 Thesis outline

The research project is divided into 4 sub projects, of which each has its own objectives.

	Subproject 1	Subproject 2
Subproject	**Definition of the Research Topic** ▪ Definition of the field of application ▪ Framework and restrictions ▪ First overview of possible technologies ▪ Literature analysis	**Technology Selection** ▪ Definition and research of requirements ▪ Detailed market analysis, determination of strengths and weaknesses ▪ Technology selection, based on the pre-defined terms ▪ Beginning of the development of a concept
Objective	Definition of the research topic and methodology	Technology selection and first steps towards the development of a prototype

	Subproject 3	Subproject 4
Subproject	**Building of a Prototype** ▪ Development of a concept for the implementation of the prototype into the existing infrastructure ▪ Procurement of all the required technology components ▪ Integration in software systems	**Validation** ▪ Result evaluation ▪ Assessment of the operation of the prototype ▪ Benefits- analysis
Objective	Concept development and Proof of Concept	Validation of the prototype, by using the pre-determined requirements.

Chapter 2

Literature Review

2.1 Country specific circumstances of South Africa and Germany

The objective of this thesis is the realization of a localization system based on low-cost components that should be a realistic alternative for as many companies as possible. From a global perspective, SMEs account for the largest share of companies. At the same time, however, it is the few large companies that have the financial strength to adapt innovations already at an early stage. Today, these large companies are also regarded as pioneers of the fourth industrial revolution. (Lichtblau *et al.*, 2015; Zillmann, 2016)

In Germany, 99.5% of companies are SMEs, in South Africa the figure is 91%, and in Germany they account for only 35% of total turnover, whereas in South Africa SMEs account for 34% of total turnover. There is no visible difference between the two countries in this respect. (Abor, J. & Quartey, P., 2010; Ahrens, 2018; Günterberg, 2012)

In both countries, SMEs are defined similarly. In South Africa, for example, SMEs belong to the "micro-enterprise" class, i.e. enterprises with up to 20 employees and an annual turnover of approx. 0.3 million, while in Germany, in this class, enterprises with up to 9 employees belong to an annual turnover of up to 2 million. (see Table 2 and Table 3). On the basis of this classification, it can be seen that SMEs in Germany have a much greater financial strength, e.g. for digitization projects.

Table 2 German SME definition (Günterberg, 2012)

Company size	Number of employees	Turnover €/year
micro	up to 9	up to 2 millions
small	up to 49	up to 10 millions
medium	up to 249	up to 50 millions

Table 3 South African SME definition (Abor, J. & Quartey, P., 2010)

Company size	Number of employees	Turnover €/year
micro	up to 5	up to approx. 0, 00945 millions
very small	up to 20	between approx. 0.0126 million and 0.0315 million, depending on the industry
small	up to 50	between approx. 0,126 million and 1,575 million, depending on the industry
medium	up to 200	between approx. 0,252 million and 3,150.0315 million, depending on the industry

Another common feature of both countries is that the majority of SMEs belong to the micro and very small enterprises.

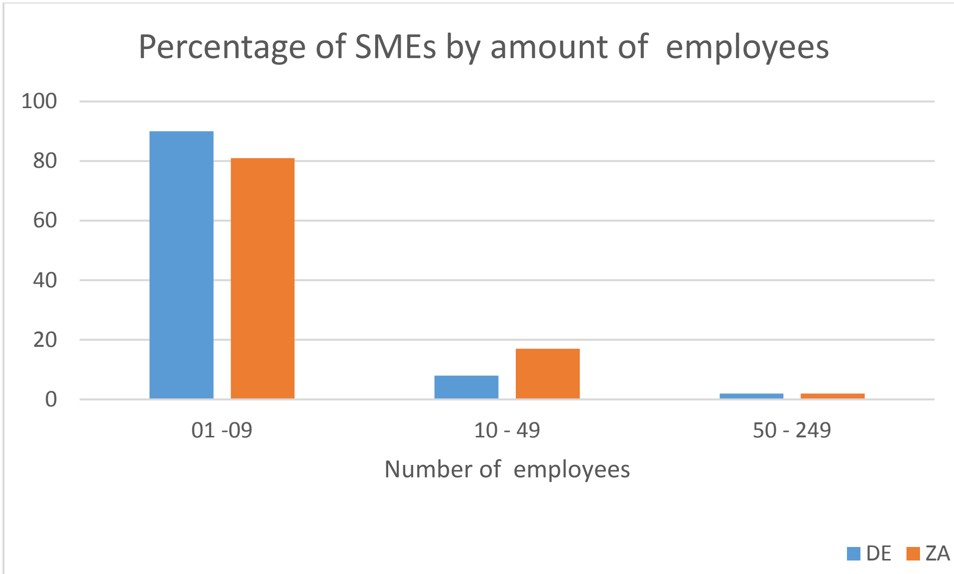


Figure 1 Percentage of enterprises according to the number of their employees

Figure 1 (National Small Business Chamber, 2016; Statistisches Bundesamt, 2017)

Wages also differ greatly between the two countries. According to the World Bank, Germany achieves a Gross National Income per capita of $51,680, while South Africa is only $13,090. (data.worldbank.org, 2017). Germany is considered a high-wage country whereas South Africa is considered a low-wage country.

However, in both countries SMEs are regarded as adopters who only use new technologies in a later phase. The pioneers are the large companies, as only these have the financial means to invest in research and development of new technologies. In both countries, the major challenges are inadequate access to resources (financing, employees, etc.), unclear economic benefits and missing use cases as well as high investment or operating costs and the low level of maturity of the required technologies.

Table 4 Country specific problems

Germany	South Africa
▪ Inadequate access to resources (financing, employees, etc.) ▪ Unclear economic benefits, use cases are still missing ▪ Missing standards and lack of data security ▪ Flexible organizational structures are required ▪ Qualification of employees ▪ High investment or operating costs ▪ Low level of maturity of the required technologies (Saam, Viete and Schiel, 2016; Schröder, 2016; Schwab, Sala-i-Martin and World Economic Forum, 2016)	▪ Weak currency and the high inflation rate ▪ High crime rate and unemployment ▪ Growing competition and low demand ▪ Rapid technological change ▪ Lack of access to appropriate funding ▪ Lack of institutional support ▪ Unnecessary complexity, influenced by the local bureaucracy ▪ Lack of education and training of personnel ▪ Restrictive labour relations (Abor, J. & Quartey, P., 2010; Emuze and Flanegan, 2014; Schwab, Sala-i-Martin and World Economic Forum, 2016; Small Enterprise Development Agency, 2016)

A key problem seems to be the cost and lack of financial strength of SMEs in both countries. In addition, the rapid technological development and the lack of experience with the implementation and handling of new technologies also pose a major challenge.

For these reasons, a solution suitable for SMEs should be cost-effective, easy to use and easy to operate.

2.2 Low Cost

Position determination plays an important role in intralogistics, becoming more and more important due to increasing digitalization level, as companies can no longer do without this data for internal process design and optimization in the future. The aim of this work is to show what possible applications there are available and to what extent a solution based on low cost components can do justice to them.

The idea of optimization based on low cost was very early taken up by Takeda. His concept pursues automation by means of cost-effective, easy to implement solutions that can be produced and integrated by the company itself. (Takeda and Klages, 2006)

This should result in productivity advantages through simple automation. This concept is particularly interesting for SMEs with limited financial resources.

Höning and Lorenz have already taken up Takeda's idea and implemented it in the digitization process. They see the goal of LCD as to automate elementary activities intelligently in a simple way, without displacing the human being from the center of the production process. Decisive factors here are the low investment costs and the fast and uncomplicated adaptation of the solutions. In addition, the resources that require specific production information at the respective workplace must be linked step by step. (Hönig and Lorenz, 2017)

In this context, it is assumed that a solution that corresponds to the LCIA concept may cause about 10-20% of the costs of similarly powerful solutions. (glossar.item24.com, no date; fml.mw.tum.de, 2012).

To be able to create a realistic cost goal a market analysis was carried out, based on available state of the art technologies. In Table 5 the results of this market analysis are shown. The information was collected due to actual research publications or the request of quotes, the data were collected in the end of 2017 and may have changed by the time the thesis was submitted. Table 5 is sorted after the applied technology and reached coverage.

Table 5 General information about State of the Art IPS (telocate via an quote request)

Manufacturer	Technology	Coverage [m²]	Accuracy [cm]	Sensors	Tags	Price	€/m ²
Pozyx (pozyx.io, no date)	UWB	30	10	4	5	999	33
KIO RTLS eliko.ee, no date)	UWB	33	30	4	1	2100	64
Localino v3.0 (localino.net, no date)	UWB	50	30	4	2	1099	22
Ubisense and Seco Granja, 2017)	UWB	160	15	6	10	26900	168
DecaWave (Jimenez Ruiz and Seco Granja, 2017)	UWB	300	30	3	1	925	3
telocate	UWB	400	20	6	4	4800	12
Bespoon (Jimenez Ruiz and Seco Granja, 2017)	UWB	880	10	1	6	1699	2
Marvelmind (marvelmind.com , no date)	ultrasonic	50	2	5	1	400	8

It is clear, that the market segment is very broadly positioned and that solutions are already being offered in various price classes and ranges.

The average cost is 39 €/m², with a range of up to 800 m² and accuracies of up to 2 cm.

However, these values should be considered with caution, as they apply under optimal conditions, which are often not given in the concrete application scenario. The average costs per square meter do not take into account the costs incurred for implementation in the operational environment and for integration into higher-level IT systems. Furthermore, it should be taken into account, that the purchase of additional tags and sensors can quickly increase the price of these systems.

If now these average costs of 39 €/m² are used for the determination of the value limit, then a solution, which is to do justice to the approach of low cost described in this chapter, may not exceed 20 % of these average costs. Which means that a low cost solution should not cost more than 7.8 €/m².

This value is used in the course of the thesis for the evaluation of the system components and the examination of the low cost requirement. Additionally, costs for installation and administration, maintenance and space costs may be incurred. Determining their monetary size does not seem to make sense at this point, but when selecting hardware, it is important to consider if it is easy to handle, that no expert knowledge in administration and maintenance is required, and that space costs are reduced to a minimum.

2.3 Industry 4.0 – Digitalization

Over the years the industry went through revolutions again and again. These industrial revolutions describe the rapid development of production techniques, which are triggered by technical development and scientific progress. In addition to the changes in industry, an industrial revolution also has an impact on society as a whole. (Pollert *et al.*, 2016) It should be noted that with every industrial revolution, the degree of complexity has also increased. Figure 2 summarizes the industrial revolutions and their characteristics. The first industrial revolution was characterised by the introduction of production facilities. This was followed almost 100 years later by the second industrial revolution with the introduction of the division of labour and mass production, followed in the 1970s by the third industrial revolution, characterised by the increasing automation of production and already the beginning of its digitisation.

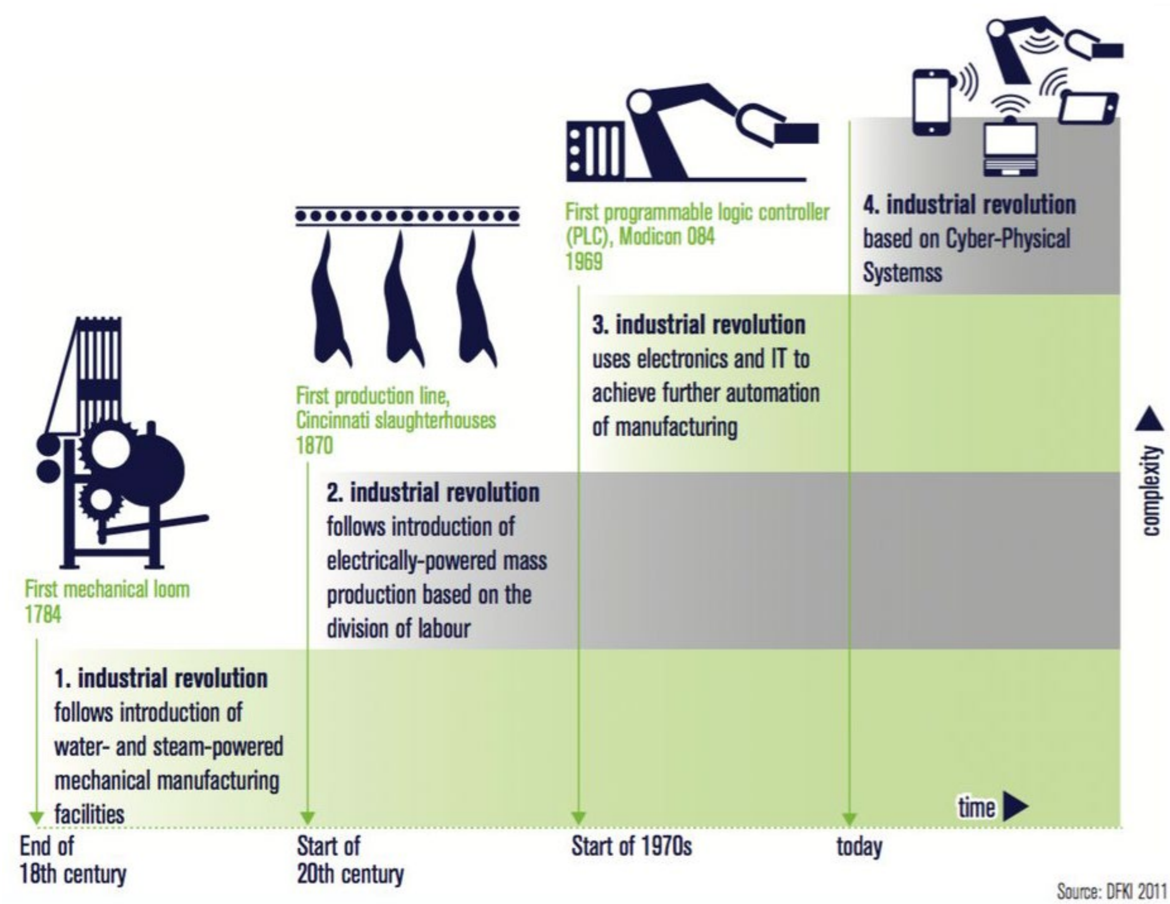


Figure 2 The industrial revolutions (Kagermann, Wahlster and Helbig, 2013)

This is also known as Industry 4.0, a term coined by Kagermann, Lukas & Wahlster (2011). There is still disagreement about the unambiguous definition of the term, so there are many definitions today. The most widely used definition comes from the Steering Committee of the Platform Industry 4.0 and describes the term as follows:

"The term industry 4.0 stands for the fourth industrial revolution, a new stage in the organization and control of the entire value chain over the life cycle of products. This cycle

is oriented towards the increasingly individualised wishes of customers and ranges from the idea, the order, the development and production, the delivery of a product to the end customer through to recycling, including the associated services". (BITKOM, VDMA and ZVEI, 2015)

Industry 4.0 is characterized in particular by highly flexible, highly productive and resource-conserving production, which is driven forward by increasing digitization and networking as well as by the development of the Internet of Things and its transfer from the private sector to production and its processes. This is necessary to meet the future challenges, the dynamic customer requirements and the resulting demand for personalized products as well as the shortened product life cycles. To do this, all participants in the production system must be equipped with new skills such as the ability to capture and respond to changes in their environment, intelligence and communication skills. Figure 2 gives an overview of the functional areas of Industry 4.0 and their important components. (Bauernhansl, 2016)

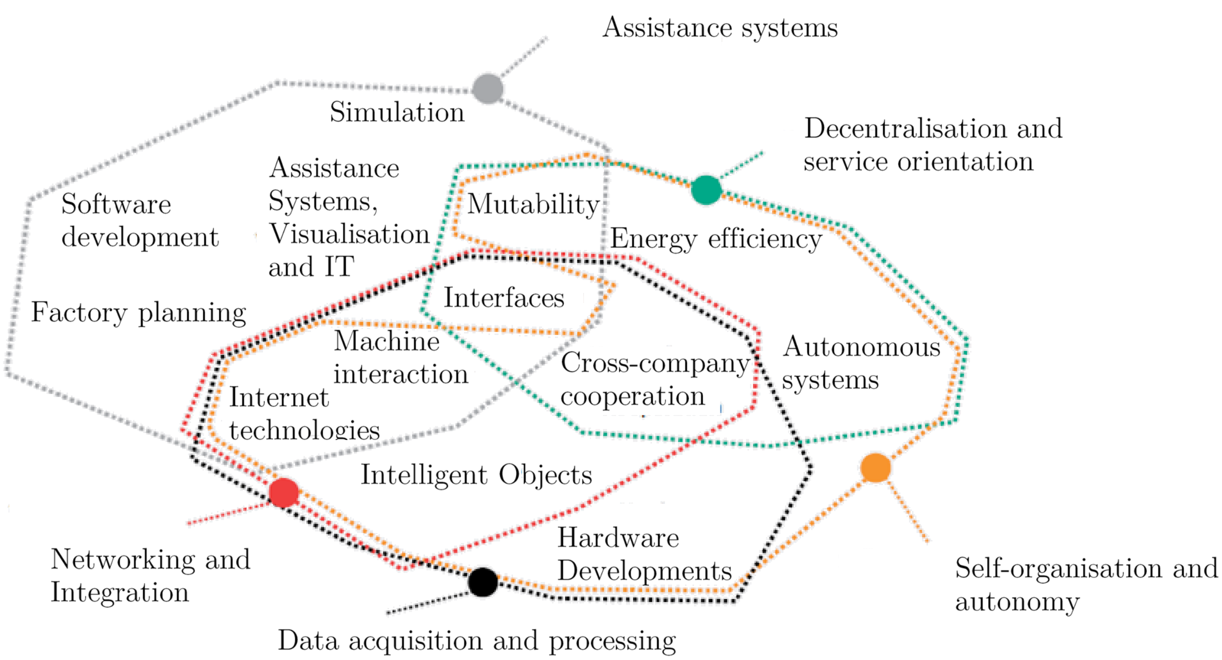


Figure 3 Functional areas of Industry 4.0 (Bauernhansl, 2016)

The use of cyber physical systems plays a decisive role in the functional areas described in Figure 3. These describe objects that become intelligent objects due to being equipped with sensors, actuators and a comprehensive communication option. (Bauernhansl, 2016)

Sensor data and the resulting information are central components of cyber-physical production systems. (Vogel-Heuser, 2014, p. 37). Real-time information, in particular on the status and current position of goods, machines and industrial trucks, is the decisive information basis (Bubeck *et al.*, 2014; Hermann, Pentek and Otto, 2016) and contributes to increasing flexibility and productivity in all processes. IPS can therefore be seen as the basic technology for the implementation of Industry 4.0 concepts.

2.4 Indoor Location Technologies

Basic terms are described in the following subchapter. An overview of the technologies used is given and the focused camera-based technologies are examined in more detailed way. The characteristics and application areas of these technologies will be shown and requirements for a concept design will be developed.

2.4.1 Definition of Indoor Location Technologies

Below the definitions of terms are given, that are commonly used in this work.

Position determination

The determination of position, location or localization, also localization of an object is the determination of the current location of an object or a person by means of a measuring method. (Anderl and Anokhin, Ol, Arndt, A., 2016; Caron *et al.*, 2017). According to DIN ISO 8373, the determination of the current location of a localization object is given if its position and orientation have been determined. (Deutsches Institut für Normung e.V., 2011)

Identification

Identification is the unambiguous recognition of an object by means of identification characteristics. A general definition is given in DIN 6763 as follows:

"Identification is the unambiguous and distinctive recognition of an object on the basis of characteristics (identification characteristics) with the accuracy specified for the respective purpose." (Deutsches Institut für Normung e.V., 1985)

Anderl, Anokhin & Arndt (2016) understand the identification of objects as the unique naming and identifiability using identification technologies or Internet Protocol addresses (IP addresses).

While the identification of an object is not a big challenge for humans, it is quite complex for computer-based systems. This is due to the fact that computers can recognize and process characteristics purely in the form of numbers or bits (sequences of numbers that describe a certain property). (Erhardt, 2008)

2.4.2 Technology overview

While GPS has been used for many years as a basic technology for the identification and localization of people and objects outdoors, its implementation in enclosed spaces is still considered challenging today. Although intensive research has been carried out in the field of indoor localization for some time, no standard has yet been able to establish itself on the market (Hirtle and Bahm, 2015; Springer, 2012; Zadganokar and Chandak, 2016). Basic technologies are based on Wi-Fi, Bluetooth, infrared, ZigBee, RFID, UWB or optical systems. Figure 4 gives an overview of the underlying technologies and their properties.

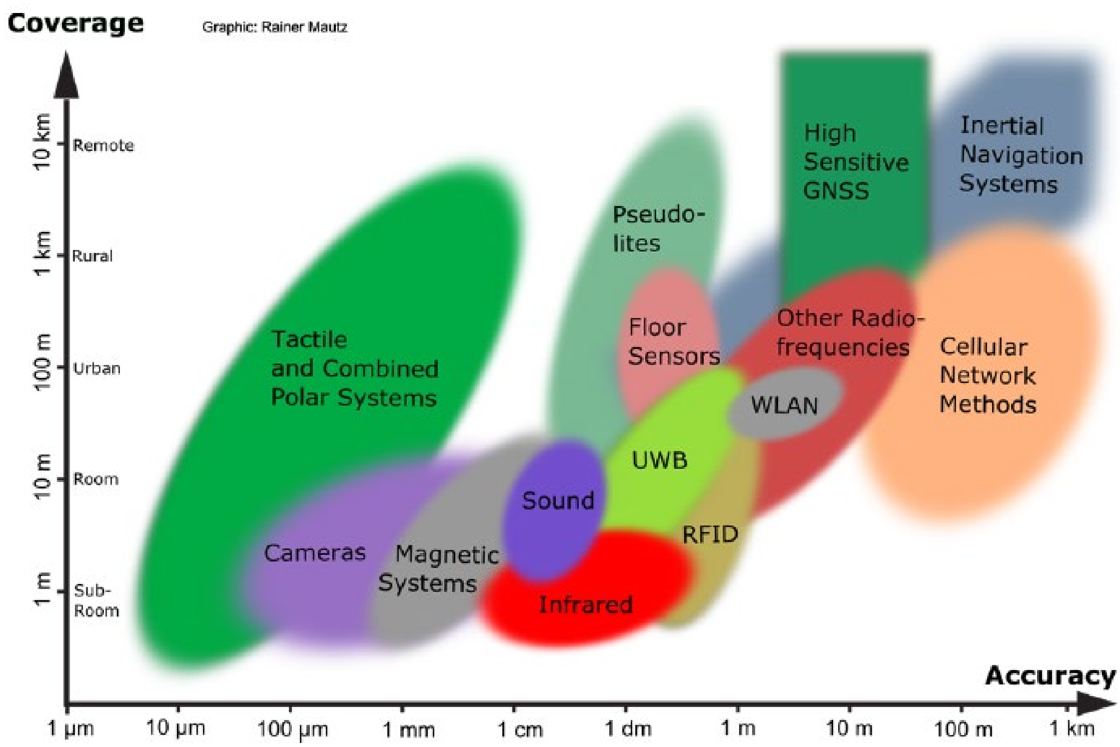


Figure 4 Performance of IPS technologies (Accuracy/Coverage) (Mautz, 2012)

Most of the presented technologies presented in Figure 4 rely on electromagnetic or mechanical (sound) waves, using different wavelengths.

One of the first IPS, the Active Badge System, was developed in 1992 at AT&T Laboratories in Cambridge (Youssef, 2017) is using infrared. Infrared-based IPS use different system architectures, such as systems with active beacons, thermal (natural) radiation or artificial light sources. The advantage of infrared is that it is not visible to the human eye.

Tactile and combined polar systems are only used for special scenarios, as they are very expensive but offer the highest accuracy within a few meters. This is why they have not been able to assert themselves on the mass market.

Sound-based IPS use mechanical waves and have been considered for positioning mobile robots or people. However, they are limited in their range and require additional nodes to cover larger areas; furthermore, they also suffer from frequency cages and are highly dependent on the temperature.

WLAN and WiFI-based IPS are becoming increasingly important as many companies are already using these technologies as part of their internal network.

A variety of different methods can be used to implement these technologies. Since the further integration of WLAN and WiFi seems to be very realistic in production environments the application of these technologies for indoor positioning seems promising. However, this technology is susceptible to interference and offers very low accuracy.

RFID-based IPS can be active or passive and has already found a high level of acceptance in retail. However, they suffer from low coverage, high costs or require a direct line of sight.

Ultra-wideband (UWB) seems to be one of the most promising technologies for IPS, as it offers high coverage and accuracy, but these technologies still suffer from relatively high infrastructure needs, leading to the emergence of relatively high costs.

Assisted GNSS as IPS is able to achieve the highest coverage of all proposed systems, but can hardly be applied in buildings, as the technology requires a direct line of sight to satellites, which is unlikely in buildings, and these techniques suffer from low accuracies and long positioning times. However, the development of highly sensitive receivers has already begun, which could make this approach even more applicable in the future.

IPS based on pseudolites are based on pseudo-satellites, which are essentially land-based beacons generating signals similar to those transmitted by GNSS. This technology offers a long range but still suffers from several unresolved sources of error, such as multipath, signal interference and costly and complex time synchronization tasks.

The use of other radio frequency technologies has also gained in importance in recent years, in particular, protocols such as ZigBee, Bluetooth, DECT, and mobile networks. They benefit from lower energy consumption, but also face the same challenges as other radio-based technologies. They also offer limited coverage.

Inertial navigation systems are based on inertial measurement and a processing unit and are usually fused with complementary sensors. For the positioning motion sensors (accelerometers), sensors that can measure angular velocity (gyroscopes) and/or a magnetometer are used. This technology is used in areas where the installation and maintenance of an external positioning infrastructure are not affordable.

Magnetic localization based on magnetic fields generated by alternating current (AC) or pulsed direct current (DC) coils.

Infrastructure-based positioning approaches that are not compatible with any other of the proposed technologies use the existing infrastructure or embed additional infrastructure.

OIPS using cameras are used in many different scenarios and are becoming a dominant technique for positioning applications that require high coverage with high accuracy. Numerous approaches and suggestions for their development can be identified in the literature.

Today, these different technologies enable localization in 2, 2.5 or 3 dimensions. The 2.5-dimensional localization describes cases in which the person or object to be localized takes place in several 2-dimensional levels and is located in a 3-dimensional building (Youssef, 2017).

IPS are used for the positioning of persons or objects either relatively or absolutely. In relative positioning, the position of the person or object to be located is relative to a reference point. In absolute position determination, the position is specified in a global coordinate system. (Youssef, 2017)

IPS has a very broad range of applications, for example, Fraunhofer IIS has developed a system for the real-time localization of industrial trucks with the accuracy of a metre. The *awiloc* system requires an industrial truck to be equipped with a smartphone. This then enables communication with several beacons distributed in the warehouse. Beacons are Bluetooth transmitters that are permanently connected to the infrastructure and serve as a reference point for position determination. The connected industrial trucks can then be identified and located via an intelligent web application. By evaluating the data generated in this way, the logistics concept can be improved. In addition, the search times for vehicles are eliminated as their position is always available in real time. Furthermore, the integration of the system enables new business processes for the user and autonomous systems. (iis.fraunhofer.de, 2018; iis.fraunhofer.de, 2018)

Systems available on the market consist of special hardware and software components, which often lead to increasing maintenance costs, limited scalability and rising costs. (Youssef, 2017)

For some years now, however, researchers have been focusing on the development of IPS that do not require additional hardware or make use of the existing infrastructure. This enables the increasing availability of Wi-Fi in more and more areas. (Youssef, 2017)

Sakpeter, Adeyeye-Oshin & Mitwa (2017) , Mautz (2012) and Liu et al. (2007) provide a detailed overview of the technologies available on the market. However, this paper focuses on optical technologies, which is why other available technologies will not be further investigated.

2.5 Characteristics and Requirements towards Indoor Location Systems in Industrial Environments

The wide field of application of indoor localization systems leads to a multitude of requirements. An overview of the requirements that can be found in the literature is given in Table 6. Here the authors have been assigned the mentioned requirements, whereby the most important requirements can be identified.

Table 6 Requirement determination

	(Günthner and Hohenstein, 2012)	(Hohenstein, 2014)	(Mautz, 2012)	(Liu *et al.*, 2007)	(Han, 2015)	(Malik, 2009)
Localization function						
Continuity		x	x			x
Output data		x	x			x
Required infrastructure		x	x			x
Localisation performance						
Accuracy	x	x	x	x	x	x
Coverage		x	x		x	x
Repeat accuracy		x		x		
Update rate		x	x		x	x
Overall system						
Scalability	x	x	x	x	x	x
Availability			x			
Robustness	x	x	x	x		x

In the following section, these requirement criteria are examined in more detail using a literature analysis.

2.5.1 Localization function

Continuity

Continuity describes the ability of a system to continuously repeat a certain function in time and space. Of particular importance is the spatial continuity of position determination. A localization system can be classified as continuous if the position determination is not limited to a few predefined waypoints within a surface. (Hohenstein, 2014)

Mautz (2012) notes that the continuity requirements are similar to those for the availability of a system. In this way, continuity can be distinguished from the update rate and categorised as an independent requirement criterion.

The localization systems described in the literature can be described as discontinuous except for a few examples. However, this is due to their current degree of maturity and not to their system architecture, so these can be classified as continuous in theory.

Output data

Output data describes the form of the output of a localization system. Depending on the system and the requirements, it can be in the form of 2D or 3D coordinates.

In addition to the output of coordinates, further information about the identified and located object can also be output. In some applications, the specification of the time (especially for tracking), the speed of travel or the spatial orientation plays a role here (Mautz, 2012).

The output data, which can be determined from the literature, can be divided into 6 categories.

1) xy position, named by the authors Hohenstein (2014), Lee and Song (2007), Barberis et al. (2014) and Liao et al. (2013).
2) xyz position, named by the authors Mustafah, Azman and Akbar (2012), Zhang et al. (2017) and Bergen et al. (2017).
3) xy position, alignment, named by the authors Zhang et al. (2015), Debski et al. (2015) and Saifizi, Hazry and M. Nor (2012).
4) xyz position, alignment, named by the authors Tilch and Mautz (2013), Kohoutek, Mautz and Wegner (2013), Boochs et al. (2010) and Ferrara et al. (2017).
5) xy position, alignment, coded data, named by the authors Mutak et al. (2008), Hagisonic (2016) and Babinec et al. (2014).
6) xy position, direction of motion and speed can be output by TotalTrax (2018).

Identification methods

The understanding of identification corresponds to the definition given in Chapter 3.2.1.1. In this context, the infrastructure required for position determination that goes beyond the location sensor and the location engine is described. Mautz distinguishes between systems that require an additional infrastructure in the form of markers and those that do not require an additional infrastructure (Mautz, 2012).

The majority of authors, Zhang et al. (2015), Debski et al. (2015), Mutak et al. (2008), Babinec et al. (2014), Hohenstein (2014), Lee and Song (2007), Barberis et al. (2014), Saifizi, Hazry and M. Nor (2012), TotalTrax (2018) as well as Hagisonic (2016) use passive 2D landmarks, which are integrated into the operational environment, in order to carry out a position determination. The position is determined in relation to the detected landmarks. Debski et al. (2015), TotalTrax (2018) and Lee and Song (2007) rely on retroreflective markers, which are able to reflect infrared light. This has the advantage that they are less susceptible to environmental influences and are located outside the spectrum visible to humans. Infrared light has a wavelength of about 0.78 µm to 1000 µm and is therefore relatively robust against light influences. Boochs et al. (2010) and Bergen et al. (2017) rely on active markers consisting of LED diodes.

Zhang et al. (2017) and Ferrara et al. (2017) also use 2D markers, but these do not serve as reference points in the environment, but directly on the object to be localized. Tilch and Mautz (2013) uses a device that actively projects laser beams at a known angle as reference points on the ceiling of a building. Kohoutek, Mautz and Wegner (2013) and Liao et al. (2013) rely on the recognition of natural environmental features and Mustafah, Azman and Akbar (2012) on the recognition of object features, so these approaches do not require any further infrastructure.

2.5.2 Localization performance

Accuracy

According to Liu et al., localization accuracy is one of the most important requirements for a localization system, but the authors also note that a high accuracy requirement is in direct conflict with other requirements. Accurate systems are often characterized by their small range and high costs. (Han, 2015; Liu *et al.*, 2007)

The localization accuracy is indicated by a numerical measure in a given metric unit and results from the deviation between nominal and actual position. Thus a circle diameter is given around the nominal position (real position). This defines a tolerance range in which an actual position (measured position) may be located (Hohenstein, 2014).

According to Günthner and Hohenstein (2012), the accuracy specification refers to the horizontal level, which means that the same calculation basis for accuracy also exists for three-dimensional coordinate systems consisting of x, y and z coordinates. The localization accuracy should be based on a confidence level of 95%, which means that 95% of all determined positions lie within the defined tolerance range. (Mautz, 2012).

In order to determine the accuracy Hohenstein (2014) uses a simplified error circle with a radius of r_{xy}.

$$rxy = \sqrt{(xshall - xis)^2 + (yshall - yis)^2} \quad (1)$$

In the literature, different accuracy requirements for localization systems can be found. These are due to the different application scenarios and specific requirements. The authors Boochs et al. (2010) and Zhang et al. (2017) reach accuracies of a few millimetres; TotalTrax (2018) and Hagisonic (2016), Debski et al. (2015), Liao et al. (2013), Ferrara et al. (2017) and Bergen et al. (2017) reach accuracies of less than 5cm; Zhang et al. (2015) and Barberis et al. (2014) reach accuracies of less than 10 cm. Saifizi, Hazry and M. Nor (2012) is able to reach an accuracy of less than 30 cm. Mustafah, Azman and Akbar (2012) only achieves an accuracy in the metre range.

Resolution

The resolution is the measure of the smallest distance between two adjacent measuring points, which can be determined by the localization system (Deutsches Institut für Normung e.V., 2011). Thus, it describes the smallest possible position changes of an object that can be determined by the localization system and influences the localization accuracy. Since the resolution is not a separate requirement, but an influencing factor on the localization accuracy, it will not be considered separately in the further course of this work.

Coverage

The coverage or range of an IPS is defined by the maximum distance between a camera system (location sensor) and the object in which object identification and position determination is still possible. It describes the spatial coverage of a localization system (Mautz, 2012) and is specified as Field of View (FOV) in optical systems.

Mautz (2012) divides systems into three different categories:

a) Local Coverage - systems that cover a defined, non-expandable area.

b) Scalable Coverage - systems in which the reach of the localization system can be increased by adding additional hardware. However, this increase must not result in a loss of accuracy.

   Sakpere et al. (2017) state that the majority of the systems available on the market can reach large ranges. However, this increase is associated with additional costs and an increase in the complexity of the overall system. In addition, the accuracy of these can also deteriorate.

c) Global Coverage, Systems based on astronomical navigation or the GNSS system achieve global coverage.

Hohenstein (2014) notes that the reported ranges of manufacturers and in the literature should be seen as a statement of tendency, as they would not be based on the same test scenarios. In general, ranges of more than 50 m are considered challenging (Al Nuaimi and Kamel, 2011). Hohenstein (2014) achieves a range of less than $10m^2$, which can be exceeded by Lee and Song (2007), who can achieve a range of 7.4x4.9m ($36.26m^2$) through the use of motor skills. The StarGazer system gives a range of up to 6.3m (area diameter) to achieve (Hagisonic, 2016). Zhang et al. (2015), Kohoutek, Mautz and Wegner (2013) and Boochs et al. (2010) can achieve ranges of up to $4m^2$.

Repeat accuracy

The precision indicates the repeatability of a measurement result, provided that the measurement is carried out under unchanged conditions. In the context of this thesis, precision describes the probability that the same accuracy will be achieved when a measurement is repeated. A high precession is given when there is a normal distribution of the measured values.

With regard to precession, Hohenstein (2014) notes that this should be 99.7%, as this enables a stable and reliable localization process.

System integrity

According to Mautz, system integrity describes the correctness of the output data and thus the reliability of a localization system. [Maut 2012]. The integrity requirement can be compared to the repeatability, since both aspects describe the reliability of a positioning system.

The only difference is the integrity risk, which describes the probability that a malfunction of the system will lead to an incorrect measurement result without being detected. Mautz (2012) also notes, however, that system integrity is a requirement criterion that is hardly considered in the literature.

This statement is also confirmed by the literature examined, none of the sources considered deals with the integrity parameter or mentions it as a requirement criterion. Based on this result and due to the similarity to the requirement criterion of repeatability, the system integrity is not further investigated in the further course of this work and is regarded as considered by the consideration of repeatability.

Update rate

The update rate is the frequency at which the position data can be determined or calculated by the system. It describes the period of time that elapses between a measurement and the completed evaluation of the measurement data in relation to the position of an object. In the literature, the terms latency time or delay or reaction time can be found.

Mautz (2012) divides the update rate into three different categories:

a) periodic, which means that the updating of the position data takes place at a regular interval, expressed for example in Hz or seconds.

b) on request, which means that the position data is updated by the user or by a remote device.

c) in the case of an event, this means that an update of the position data is triggered either by the occurrence of a particular event, such as the exceeding of a threshold value, for example the detection of a change in position.

With the exception of Barberis et al. (2014), only update rates that belong to category a) periodic can be identified in the literature considered (cf. TotalTrax (2018), Zhang et al. (2015), Debski et al. (2015), Mutak et al. (2008), Hagisonic (2016), Babinec et al. (2014), Hohenstein (2014), Lee and Song (2007), Tilch and Mautz (2013), Kohoutek, Mautz and Wegner (2013), Boochs et al. (2010), Mustafah, Azman and Akbar (2012), Zhang et al. (2017), Saifizi, Hazry and M. Nor (2012), Liao et al. (2013), Ferrara et al. (2017) and Bergen et al. (2017)). Zhang et al. (2015), Hagisonic (2016), Hohenstein (2014), Tilch and Mautz (2013), Mustafah, Azman and Akbar (2012), and Liao et al. (2013) reach update rates of less than 1 s.

Software complexity

According to Liu (2007), complexity refers to both hardware and software factors of a localization system. Liu, however, mentions software complexity (code complexity) as a more important factor, which refers to the complexity of the algorithms used. It has a special effect on latency, so the fast handling of complex code structures requires high computing power.

2.5.3 Requirements towards Indoor Localization Systems

Scalability

Hohenstein (2014) describes the relationship between the amount of localization structure used and the achievable number of localizable objects as the scalability of a system. According to Mautz (2012), scalability can be divided into categories.

a) not scalable

b) scalable by increasing the localization structure proportionally to the area

c) scalable with loss of accuracy

Although not all authors directly address the issue of scalability, its close connection to range makes it a critical requirement that should not be neglected. For the extension of the area in which localization or navigation is possible, the number of necessary markers is increased proportionally to the extended area. Hohenstein (2014) specifies the scalability to less than 100 system participants, this specification cannot be found in this form in any other of the other sources, but none of the other authors excludes this.

Availability

Availability is the percentage of time a system is available for use with the required accuracy and integrity. Availability is influenced by random factors such as software failures and hardware failures or by fixed factors such as planned maintenance or software updates. In principle, the availability of a system can be divided into three categories, although these depend strongly on the application. (Mautz, 2012)

- low availability < 95%
- regular availability > 99%
- high availability > 99,9%

Robustness

The robustness of a system means its protection against unwanted influences. These include, for example, the masking of system components, the occurrence of interference signals, protection against physical damage as well as functionality in different environments and resistance to changing environmental influences. Mautz (2012) also sees protection against unauthorized access as an aspect of robustness. Hohenstein (2014), speaks of the function of a localization system under changing conditions.

From the literature, several factors arise for the robustness of a system. The factors mentioned here influence these and should be considered. All authors agree that a visual contact must exist between the components of the localization system. Zhang et al. (2015), Mutak et al. (2008) and Babinec et al. (2014) cite the lighting conditions of the operating environment as an additional criterion. Boochs et al. (2010) also describes temperature fluctuations as problematic, but this in connection with the "active marker system" he uses. Bergen et al. (2017) also mentions the reflective properties of surfaces as a factor influencing the robustness of a system.

In order to achieve an increase in robustness, the majority of the authors agree that a camera calibration must be performed. (cf. Zhang et al. (2015), Babinec et al. (2014), Hohenstein (2014), Tilch and Mautz (2013), Kohoutek, Mautz and Wegner (2013), Boochs et al. (2010), Mustafah, Azman and Akbar (2012), Saifizi, Hazry and M. Nor (2012), Liao et al. (2013) and Ferrara et al. (2017)). In addition, Debski et al. (2015), Zhang et al. (2015), Mustafah, Azman and Akbar (2012) and Ferrara et al. (2017) cite greyscaling, the conversion of image data into grayscale images, as an improvement strategy. Zhang et al. (2015) also mentions a flexible exposure time as an advantage for being able to react to changing lighting conditions. Hohenstein (2014) and Bergen et al. (2017) can also increase the robustness of a system by increasing marker density.

2.5.4 Conclusion and summary

Some authors also name data protection and admissibility (Malik, 2009; Mautz, 2012), the requirement for the human-machine interface (Hohenstein, 2014; Mautz, 2012), the integration effort (Hohenstein, 2014; Malik, 2009) and adaptability (Hohenstein, 2014; Malik, 2009) as requirement criteria. These requirement criteria will not be considered in the further course of the work, as they either represent a special, case-related factor or are not assigned to the aspect of position determination.

- Data protection and permissibility, this requirement criterion deals with the protection of the privacy of persons in the operational environment as well as with official approvals.
- The requirements on the man-machine interface describe how the position information is reported and queried at the application device. In the literature, the human-machine interface represents a rather unimportant requirement and is independent of the localization system itself.

- The integration effort describes the effort that arises during and through the integration of a localization system. In principle, this can relate to both the localization object and the infrastructure and can be influenced by them. The integration effort includes in particular:
  - Installation effort, which refers both to the number of system modules to be installed and to the environment in which they are to be installed.
  - Power and communication connection, which can be wired or wireless. For the computing unit, however, it should be noted that these often have a relatively high power consumption and therefore require a direct voltage connection.
  - Measurement and calibration, this depends on the given localization system (Hohenstein, 2014; Malik, 2009).
  - Transformation costs based on necessary conversions and process changes.
  - Training for employees (with regard to maintenance and use) (Hohenstein, 2014; Malik, 2009).

  In the literature it is proposed to evaluate the integration effort in terms of the costs per square meter of developed localization area. For the investigation of different systems, however, this is not always useful and is more suitable for a case-by-case analysis, as the costs are strongly dependent on the concrete application. (Günthner and Hohenstein, 2012)

- Adaptability is bames by Hohenstein (2014) and describes the ability of a localization system to be extended by further sensor functions.

In the following, the metrics of the requirement criteria are summarized in a requirements catalogue, the choice made with regard to the requirement level is highlighted in colour. Basically, requirements for further localization applications can be derived on the basis of this catalogue of requirements, which is based on the principle of a morphological box.

2.6 Structure of Indoor Location Technologies

This chapter describes the basic elements of an optical localization system. Furthermore, it is divided into functional modules of such a system and the underlying architecture.

2.6.1 Functional modules of a Indoor Location System

Different technologies are used to implement localization systems in production environments. In sheet 5 of the DIN EN ISO/IEC 19762-5 guideline, a localization system is divided into three modules (International Standard, 2008). Accordingly, a localization system is subdivided into a transmitter for responding to a localization signal, which is transmitted by the exciter and received in the reader. The reader is also able to determine the position based on the received signals.

In his book "Wireless Positioning Technologies and Applications", Bensky (2008) also takes his cue from ISO 19762-5, but uses different terms for modules with the same content.

Malik (2009), describes the components of a localization system as tags, location sensors, location engine, middleware and application, which also conforms to ISO 19762.

Hohenstein (2014), also classifies the functional modules according to ISO 19762-5 for the design of a forklift localization system, but is strongly oriented on the definition according to Malik (2009), see Figure 5

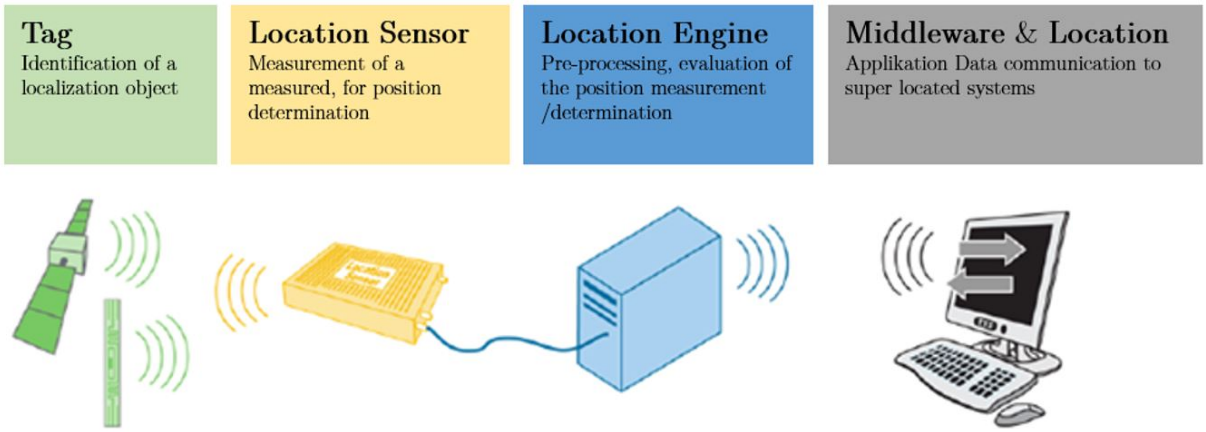


Figure 5 Functional models of an IPS according to Hohenstein (2014)

According to Hohenstein (2014), tags serve as reference points or for the unambiguous identification of an object to be located. The location sensor can have a uni- or bidirectional connection to the tag. The location sensor serves as transmitter and receiver in a bidirectional connection. However, the reception and interpretation of a signal is primarily the core function of the location sensor. The location engine is used for pre-processing the data received from the location sensor, which is then forwarded by the middleware to higher-level IT systems. The location application is the interface to the user.

Han (2015), applies Malik's (2009) approach, but divides a localization system into three main components.

- **Anchors / Location Sensors**

  Anchors are usually permanently connected to the infrastructure of the building and search the environment for tags, or can receive signals emitted by the tags. In terms of optical systems, anchors would be the location sensor.

- **Tags / Markers**

  Tags are fixed to the object to be located and are able to transmit signals which can be received by the anchor. In optical systems these are called markers, which can be used for identification and position determination or as a reference point.

- **Location Engine**

  The location engine is described by Malik as a computer unit on which software for processing the received signals is installed. This unit has access to all the recorded data of the anchors for position determination.

  The underlying distribution of the modules and their functionalities are identical. In the further course of this thesis, the middleware and the location application will not be examined in detail, as they do not represent the focus of this thesis.

- **Tags**

  Tags are objects that enable localization using an integrated localization technology. The size of tags should be selected so that they can be attached to the target object or person. According to Malik (2009), there are three different types of tags available, "Passive tags" which are characterized by their simple structure, their energy self-sufficient structure and low cost, "Semi-passive tags" which are equipped with an energy source and thus enable the integration of further sensors or an improvement of data transmission. In the sense of optical positioning systems, these could also be seen as retroreflective markers. Last but not least are "active tags", which also have an energy source but use it to communicate data autonomously. In optical systems, particularly active infrared markers are mentioned here.

- **Location Sensor**

  The location sensor is mostly a fixed device, connected to the infrastructure, which is able to identify the tags or to receive the signals transmitted by them.

- **Location Engine**

  describes the software, which enables the communication between tags and location sensors. This software is also able to carry out a location determination from the determined information. The location engine also passes this on to the middleware and application.

- **Middleware**

  is also a software component that is able to communicate both with the technology components of the localization system and with other business applications. The middleware makes it possible to make the location positions available to other systems in the company.

- **Application**

  represents the end-user application, interacts with the middleware, and provides users with the desired functionality. Malik (2009) describes the positioning of children in an amusement park as an application example. The application in this example would be the representation of the current positions via a web page to which parents would get access.

A further approach is provided by Schäfer, Schöttke and Hermann (2017), who dealt with optical space monitoring systems. For a first consideration he uses the VDI/VDE 2632 which describes the basics and terms of industrial image processing (Verein Deutscher Ingineure, 2010). Image processing systems consist of the following components, see Figure 6.

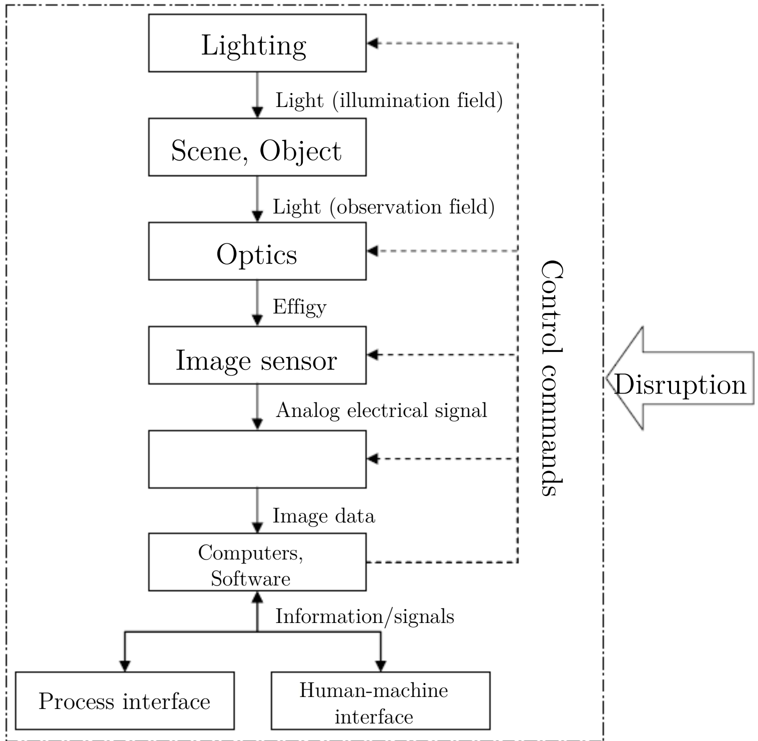


Figure 6 Concept proposed by VDI/VDE 2632

The scene represents the section of the environment, observed by the image processing system, in which the object to be examined or recognized is located. The object itself or a marker attached to it can serve as an exciter. The optical system describes the "reader" or location sensor (cf. Malik (2009). The image sensor and the digitization are also part of the location sensor. The computer with its software represent the "receiver", which can process the received data and forward it to other systems.

In industrial practice, systems can mainly be found that consist of a camera system, the frame grabber and the CPU, which has a communication interface to external automation systems. The camera system is able to display a physical image on an electronic sensor; the frame grabber enables communication between the camera system and the CPU. By means of the CPU the acquired data can then be further processed.

Schäfer, Schöttke and Hermann (2017) derives the following concept proposal from this. A camera system is used to view an area of application.

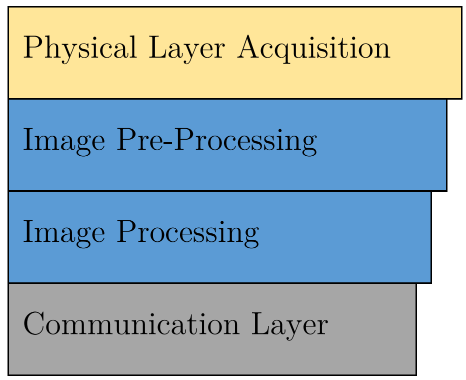


Figure 7 Concept proposed by Schäfer, Schöttke and Hermann (2017)

In comparison to the system structures described above, the use of tags is not always necessary in image processing; objects in image data can be identified by feature recognition and machine vision approaches. However, there are also a number of systems that take advantage of the benefits of using markers.

The system described by Schäfer, Schöttke and Hermann (2017) consists of the physical layer, which can be seen as a location sensor, since it is responsible for recording the scene and capturing the data necessary for position determination and identification.

Image pre-processing serves to pre-process the data obtained and facilitates its subsequent processing. Image processing is understood to mean algorithms that provide the functionalities necessary for the system, such as the identification of bodies in the image, evaluation of human-machine interactions, etc. The communication layer enables the communication with superior IT systems.

Considering the approaches mentioned it can be deduced that localization systems must consist of the following functional modules. First of all, a module is necessary which allows the unique identification of an object or a person. Tags, markers or distinctive object properties can be used for this. Secondly, a module is necessary which can interpret the identification features of an object described in the first module. The data collected in this way is then processed, interpreted and made available to higher-level systems by a computing unit. This third module represents the location engine. The location engine consists of a hardware component, which must be able to communicate with the Location Sensor and contains software, which enables the processing of the determined data. As a fourth module, a location application is necessary, which makes use of the collected data to generate added value for the user.

2.6.2 Architecture of a Indoor Location System

In addition to the functional modules of a localization system, the question of architecture remains open. The architecture is the structure of the system in the operational environment. Hohenstein (2014) uses the terms "self-location" and "external location" for this purpose.

In self-location, the localization system is attached to the object to be located itself. The position data are thus recorded directly at the object to be located.

In contrast to this, the mobile object is detected by a fixed optical system and its position is determined. As a rule, external positioning is only used if a clear identification of the located object is also possible. This is closely linked to the identification of objects.

Another consideration is the spatial relationship between location engine and location sensor. In addition to the basic principle of self- and external positioning, there is also an indirect form of self- and external positioning, which describes the case when location sensor and location engine are arranged spatially separately. The possible architectures of localization systems can be found in Table 7.

Table 7 OIPS architecture

	Location Engine	Location Sensor	Tag
External positioning	Infrastructure	Infrastructure	Localization object
Indirect external positioning	Infrastructure	Localization object	Infrastructure
Self positioning	Localization object	Localization object	Infrastructure
Indirect self positioning	Localization object	Infrastructure	Localization object

Finally, according to Hilty et al. (2012) the processing time of the position data plays a role for the architecture. This is referred to as synchronous and asynchronous positioning. Synchronous positioning is defined as the transmission and processing of position data in real time. In asynchronous positioning, the position data is stored and only transmitted at a later point in time for further processing.

2.7 Optical Indoor Location Technologies

2.7.1 Definition of Optical Indoor Location Technologies

In optical systems, cameras or other optical sensors are used to determine the position of people and/or objects within the system's field of vision. (cf. Klopschitz *et al.*, 2010; Mautz and Tilch, 2011) This thesis focuses on optical indoor positioning systems (OIPS), which are exclusively based on camera-based systems.

Different methods are used for this purpose. These can be classified according to Vossel's subdivision (2017) and Mautz & Tilch's classification (2011).

Table 8 Classification of OIPS

Methods of OIPS	
Marker-based	**Feature-based**
Methods making use of coded markers	Methods based on reference points from three-dimensional building models
Methods making use of projected goals	Methods making use of image interpretation
	Methods doing without reference points
	Methods making use of visible light
Methods that rely on other sensors	

Vossel (2017) speaks of optical tracking methods, which are regarded as synonymous with optical indoor localization, even if the tracking does not refer to the pure position determination, but to the continuous position detection and tracking of an object. Tracking not only records the position, but also the time at which an object is at a certain position. (Neumann, 2016)

Marker-based Tracking

As can already be deduced from the name, markers are attached to a person or object to be localized in marker-based tracking. According to Mautz (2012), these are characterized by a clear identifiability. For most markers, this is in the form of a unique pattern. One of the best-known examples of this is the OR code. If a marker is detected in the field of view of a camera and can be read out, the information read out can be used to draw conclusions about the object in the image and its position.

Vossel (2017) notes that the low demand for computing power is a particular advantage of these processes. He also notes that the use of monochrome markers has been particularly successful in the market due to their high contrast, which ensures optimum detection. Depending on the system, passive or active markers can also be used. For example, systems that make use of infrared light use markers with reflective surfaces, which can further improve marker recognition. Alternatively, active infrared markers can also be used, which are self-luminous.

A disadvantage of the use of markers is the necessity of their installation and production. In addition, the marker must be in the field of view of the camera to determine its position.

Mautz and Tilch (2011) go one step deeper in classifying these systems, so the following approaches can be assigned to marker-based systems:

- **Methods making use of coded markers**

  If the application environment places high demands on the positioning system, the use of coded markers represents a good alternative. Compared to optical positioning systems, which rely exclusively on the identification of natural features, marker-based methods are more robust against changing light conditions. In principle, all methods of marker recognition follow the same approach (Siltanen, 2012):

1. Image capture
    a. Recording of an intensity image
2. Pre-processing
    a. Low-level image processing
    b. Undisortion
    c. Line detection / line adjustment
    d. Detection of the corners of the marker
3. Detection of potential markers and discarding of obvious non-markers
    a. Fast rejection of obvious non-markers
    b. Fast acceptance test for potential markers
4. Identification and decoding of markers
    a. Template Adjustment (Template Marker)
    b. Decoding (data marker)
5. Calculation of the marker position
    a. Estimation of the marker position
    b. Iterative pose calculation for an exact pose

Nowadays there is a large number of coded markers available on the market. As an alternative to coded markers, infrared light-based systems can also be used: active infrared light sources and infrared light sensitive CCD cameras. Two different approaches are pursued, one is to use markers consisting of active infrared LEDs. (cf. Boochs *et al.*, 2010) the other approach would be the use of retroreflective tags, such as those used in the StarGazer system (Hagisonic, 2016).

- **Methods making use of projected goals**

  In addition to the use of permanently attached coded marks, there are also procedures that rely on the projection of reference points or patterns. These methods can be used if the fixed mounting of reference markers is undesirable or not possible. Here for example the projection of infrared light is used, since this is not visible for humans and the persons in the operational environment are not negatively impaired in their activity. The recognition of projected patterns is facilitated by their different colour, shape and brightness. The functional principle used in this case is active triangulation and can be used when the projected light replaces the optical path of a camera. A disadvantage of processes that depend on active lighting systems is that both the camera and the light source require a direct view of the same surface.

Feature-based Tracking

Feature-based tracking relies on the recognition of natural features of people or objects to identify and position them. Natural characteristics are known shapes and properties of an object to be localized. Methods based on feature-based tracking can follow different approaches. Some systems rely on the use of computer generated 3D models of the object. To do this, the camera must be calibrated so that the internal model matches the external model in terms of size and orientation. Camera systems available on the market are usually delivered with a calibration method.

A further approach is to examine the camera image for key features (natural features) of the object, which are then compared with information captured in advance.

The advantage of these systems is that they are independent of brands and therefore do not have to be produced or attached. If optimal conditions are given, these systems enable tracking of the target object from all sides and independent of camera position.

A disadvantage of these systems is their high dependence on external parameters such as lighting conditions and the visibility of the object. In addition, these systems are very computationally intensive and therefore place high demands on the computer components.

Mautz and Tilch (2011), classify these systems one level deeper, so the following can be assigned to the feature-based systems.

- **Methods based on reference points from three-dimensional building models**

  These methods are based on the identification of known objects in buildings, for which a three-dimensional image of the building and its equipment is created in advance and stored in a database. The system then compares the currently captured image information with the data from the database and thus determines the current position.

  Systems that make use of such a procedure have the advantage that the existing infrastructure does not have to be extended by sensors or markers. The information stored in the database is used as reference points. Thus, such systems have the potential to cover a large area without significant cost increases. However, the use of such a method is questionable if persons or objects are to be identified and localized in a constantly changing environment. Application examples for this process can be found, for example, in

  Böhm (2007), Hile & Boreillo (2006) and Kohoutek, Mautz and Wegner (2013). Even though these three projects use different technologies, the process is quite similar.

- **Methods making use of image interpretation**

  Methods based on image interpretation use a previously captured image sequence, which is then compared with the current camera image to determine the current position. The biggest challenge of this approach is to achieve real-time capability, as the localization process is very computationally intensive. In addition, such a system requires the use of independent reference sources, otherwise it is not possible to correct an error. (Mautz and Tilch, 2011) Similar to the methods, which are based on reference points from three-dimensional building models, these methods require a constant update of the image sequence captured for localization in continuously changing environments.

- **Methods doing without reference points**

  Reference pointless systems are used for the direct monitoring of position changes of objects and therefore do not require an external reference. In general, these systems rely on tracking mobile objects with one or more static cameras in real time. For this purpose, the cameras must be able to achieve high frame rates.

- **Methods that rely on references from other sensors**

  These procedures describe the class of hybrids. Systems belonging to this class use fusion with other measurement systems to determine the position of objects and persons. Hybrid sensor systems are used in different scenarios and can, for example, increase the accuracy, the area to be observed or the robustness of a system.

- **Methods making use of visible light**

  Processes that make use of visible light rely on Visible Light Communications (VLC). Compact Fluorescent Lamps (CFLs) and LED lamps are used for signal transmission, and transmission rates of up to 500 Mb/s can be achieved. (Grobe *et* *al.*, 2013; Ndjiongue, Ferreira and Ngatched, 2015).

By the spreading of these illuminants in indoor applications and their advantages compared to conventional illuminants, these have aroused the interest of researchers in recent years and could thus gain more and more importance in the research on indoor positioning topics. (Jovicic, Li and Richardson, 2013; Kim *et al.*, 2013).

In general, these methods are based on the modulation scheme of intensity modulation and direct detection (IM/DD). (Aminikashani, Gu and Kavehrad, 2016; Yin, Wu and Haas, 2015). The main components are a base station in the form of a light source (transmitter), an image sensor (receiver) or a mobile terminal and an LOS communication channel (Zhang, Chowdhury and Kavehrad, 2014).

Transmitters are mounted on ceilings in the form of light source (Hossen, Park and Kim, 2015) or attached to side walls in the form of a base station (Saab and Saab, 2016). The information sent by these can then be interpreted by the receiver and a position can be determined.

VLC positioning systems are based on the same functionality and only differ in hardware, technology or architecture (Hassan *et al.*, 2015).

Today, the implementations of Philips (Philips Systems, 2016), Qualcomm (Jovicic, Li and Richardson, 2013) and Acuity (Vogel, 2016) are well-known examples of applications. These show that VLC positioning systems are low maintenance, high reliability, scalability and accuracy (up to 10 cm (Halper, 2017))

However, it should be noted that VLC positioning systems can lead to varying degrees of complexity, accuracy and performance depending on the approach taken. However, they are generally considered to be scalable, accurate, reliable, simple and cost-effective. (Armstrong, Sekercioglu and Neild, 2013; Do, Hwang and Yoo, 2013).

Hospitals, airports, large shopping malls, museums and university buildings are considered areas of application for such systems. They are used for position determination and navigation both indoors and outdoors as well as in robotics and in vehicles.

2.7.2 Fields of Application of Optical Indoor Location Technologies

Industrial indoor localization is currently an important part of the industry's digitization initiative (Dubucq, Hinckeldeyn and Kreutzfeldt, 2018). For the use of indoor localization systems in production and logistics, a large number of applications are possible. An essential component here is the position information of goods or parts, since these process steps can be used to trace and analyse (Bildstein and Seidelmann, 2014).

The following is a list and brief description of these.

Table 9 Description of application scenarios

Area	Application scenario	Description
Logistics	Automatic goods recognition at goods arrival	Delivered goods can be automatically identified and posted directly at the arrival (Logivations, no date).
	Route inspection	Before a transport process is started, the system checks whether the route is blocked.
	Tracking of material flows, load carriers or workpieces	Material movements can be recorded with RTLS and passed on to higher-level systems for analysis purposes and process optimisation. (Pelka, 2014; Zsifkovits and Altendorfer-Kaiser, 2015)
	Quality management	Reduction of picking errors, packed goods are counted and compared with the packing list (Logivations, no date; Zsifkovits and Altendorfer-Kaiser, 2015).

	Navigation and localisation of mobile robots	Equipped with an RTLS, mobile robots can find their way around in industrial environments. The determination of their absolute position at any time is crucial for optimal navigation. (Siemens AG, 2018).
	Optimization of routes	Companies have to transport a large number of goods for further processing, the routes are often not optimal and empty runs occur. In addition, it can happen that the positions of goods or transport vehicles are not known.
	Inventory management	Stocks are often unnecessarily increased because the storage location cannot be determined. (Zsifkovits and Altendorfer-Kaiser, 2015).
	Tracking of intralogistics transport vehicles	Current position determination of transport vehicles (forklifts) in the internal transport process enables the complete tracking of goods and a more transparent process design. (The current warehouse location of goods can be determined by the continuous observation of the movements. (Hohenstein, 2014). By recording position data from transport systems, transport routes can be determined and optimised, and information about transported goods can be determined and conclusions drawn about their current position. (Zsifkovits and Altendorfer-Kaiser, 2015).
	Management of assembly and tool sets	Through the constant localisation of tools and the intelligent analysis of this data, it is possible to determine and optimize the utilization of these tools. (Lachmann, 2018b; Schmidt, 2011).
	Component search	Comprehensive localisation of components (Schmidt, 2011).
Production	Quality management	Reduction of assembly errors (Zsifkovits and Altendorfer-Kaiser, 2015).
	Intelligent tool control	Depending on the position of a tool (e.g. a screwdriver) relative to a workpiece to be machined, its control is automatically enabled. For example, the torque is adjusted depending on the position of the tool. (Zsifkovits and Altendorfer-Kaiser, 2015).

	Automation in production lines (Schmidt, 2011)	Modern production processes are highly automated, but the position data of workpieces or containers are often missing. This leads to a certain static in the flow of goods, as parts are not always where they are needed for the next processing step. This interrupts the automation chain and can cause delays in the production process. (Lachmann, 2018a).
Security	Access control	Monitoring of safety areas, increasing workplace safety. (Schmidt, 2011; Zsifkovits and Altendorfer-Kaiser, 2015)
	Anti-theft protection	Continuous presence check, inspection of a known object
	Evacuation management	In emergencies (e.g. fire, release of dangerous substances, danger of terrorism, etc.) companies must be able to determine whether employees are still in the building and ideally in which areas. (Lachmann, 2018c).
Overarching	Management of machines and vehicles (Schmidt, 2011)	The constant position determination of machines and vehicles enables the smooth design of production processes. This information plays a decisive role, especially in the case of increasing automation or even the introduction of autonomous system components.
	Operational optimization, by looking at the overall system of the factory	The position determination of every single batch, every tool and every employee enables extensive process optimization by means of analysis. (bespoon.com, no date).

In addition, the literature contains a large number of papers on cross-departmental applications which have not been developed for a specific task but can be used universally. An example is the fields of robotics and logistics (compare Appendix C). The main areas of application are the navigation of autonomous vehicles, human-machine collaboration and the tracking and positioning of transport vehicles, employees, goods and workpieces as well as tools. Within the scope of this work, these applications will not be examined in detail.

2.7.3 Classification of Application Scenarios and their Requirements

When selecting components, it is necessary to think about the usage scenario in advance. Since this work generally deals with the potential of a solution based on low-cost components, it would be desirable if a large number of different application cases could be covered. In the following, the use cases are examined for their requirements and subdivided into classes on the basis of these. These requirements were determined on the basis of a literature search based on scientific publications and transferred to the application cases. The results are presented in Appendix C, which gives an overview of the works of different authors as well as the area of application, coordinate reference and architecture of the system proposed in their work. Appendix E, presents a more detailed study of the requirements and their properties, which was applied for the classification.

In addition to the aspects identified and described in Chapter 2.5, which are based on general IPS requirements, aspects such as output data, power supply, required computing power, number of objects, positioning and frame rate play an important role for OIPS. Since most of these aspects have already been described in more detail or are closely related to those described in Chapter 2.5 only a brief characterization of these additional requirements is discussed in this chapter.

Architecture and Identification Methods

The architecture distinguishes between internal and external location. Self-location refers to applications in which the camera system itself is located on the object to be localized, but this is not the case with external location. In these applications, the camera system is not located on the object to be localized. For identification, a distinction is made between marker-based and feature-based methods.

Accuracy, coverage and image rate

For the accuracy, different classes were formed based on the literature search, these start in the sub-cm range, which is less than 1 cm and reach up to less than 5 dm. It is relatively difficult to make a concrete statement about the range, as it is limited by the number of cameras or subsystems and the focal length of the lens used. However, this can be extended by a high scalability.

The image rate is closely linked with the positioning and update rate, as this aspect is required for the distinction between application scenarios in which static or dynamic localization objects are in the point of interest. For the image rate (in frames per second, fps) only two classes are distinguished, these are less than 15 fps, greater than or equal to 15 fps. This is due to the fact that in some applications the objects are static or move very slowly or hardly at all. For dynamic observations, a frame rate of ≥ 15 fps is recommended, which should make it possible to locate even industrial trucks with sufficient accuracy.

Positioning and update rate

This indicates how long an image interpretation, identification and position determination may take. This can also be used to determine the real-time capability of the system. The update rate indicates in what interval a position determination must be possible, which means that it is carried out constantly.

Number of objects and scalability

The number of objects to be localized is difficult to be estimated, as there is a big difference between the fields of application, depended on the size and products of the enterprise. Based on the data found in the literature categories between smaller than 10 and greater than or equal to 50 localization objects have been set. Greater than or equal to 50 localization objects represents scenarios in which at least the localization of 50 objects is relevant, however no upper limit is given. Especially for marker-based approaches the fulfilment of this requirement seems to be unproblematic, since the availability of huge amounts of markers is given, while their production does not increase the costs of these approaches significantly.

The scalability should describe the extent to which the system is adaptable to the environment or can be extended or reduced again and whether this is necessary at all. In some cases it may be necessary to consider a large area, such as warehouse monitoring; in others, such as access control, it may be sufficient to be able to cover an important point and thus a small area.

Required computing power

The necessary computing power is seen in relation to the identification procedure. While marker-based localization causes relatively little computational effort due to simplified identification and intelligent algorithms, this is not the case with feature-based methods. In principle, these approaches require even more computing power, since the amount of data to be processed is larger. It should be mentioned that the computing power in the low cost range as well as in industrial image processing plays a decisive role for the costs of the system.

Power supply

The power supply can be active or passive. Active means that the system should be permanently connected to the mains and passive means that the power is supplied by a rechargeable battery. In general, image processing systems require an active power supply, as this is crucial for performance. In addition, a passive power supply increases the maintenance effort and thus also the costs. When tracking industrial trucks, for example, a passive supply is used according to Hohenstein (2014) but here the industrial truck supplies the system with energy. For this special application, however, this makes sense and can also be implemented without increasing the maintenance effort. In the case of systems based on external location, however, this can prove rather disadvantageous.

Output data

The output data describes the form in which the item data is relevant for the user. In most cases it is sufficient to determine the xy coordinate and the ID of the tag. Thus the exact position of an object can be determined.

After assigning the properties of the requirements to the use cases, these were divided into classes. Here it is to be noted that there are several possibilities of the realization for some use cases. In addition, it can happen that potential users implement the scenarios mentioned with different approaches.

However, the selection of the properties of the given requirements is based on an extensive literature study which can be found in Appendix E. With regard to the requirements for architecture and identification methods, the selection can be made on the basis of the given literature and the available system architectures. With regard to the other requirements no clear guideline is found in the literature, therefore different application scenarios were analysed and their characteristics were compared and logically matched with the application scenarios given in this work. For example, the application scenario of "route inspection" requires a system based on an external localization architecture that uses a markerless identification method, since objects that could block the route cannot be defined in advance. However, this scenario does not require high accuracy, since it only needs to know whether the route is free or occupied, and in terms of coverage only a small corridor needs to be covered, which can be very long. The requirement towards the frame rate is not demanding, the positioning rate should be higher and the update rate event-driven, while the number of objects should not be limited. The requirements of all application scenarios are derived in a similar logical way.

The following classes were created based on the two primary aspects architecture (external location/own location) and identification procedure (marker-based, feature-based):

Table 10 Class characteristics

Class	Architecture	Identification procedure
Class 1	External localisation	Marker-based
Class 2	External localisation	Feature-based
Class 3	Self-localisation	Marker-based
Class 4	Self-localisation	Feature-based

The use cases can be assigned to these classes.

Table 11 Classed application scenarios

	Class 1	Class 2	Class 3	Class 4
Automation in production lines	x	x		
Automatic goods recognition at goods arrival	x			
Inventory management	x			
Operation optimization	x	x		
Quality assurance (reduction of picking errors)	x			
Tracking of material flows, load carriers or workpieces	x	x		
Identification workplace	x	x		
Intelligent tool control		x		x
Component search	x	x		
Stock control	x			
Access control	x	x		
Anti-theft protection		x		x
Route inspection		x		
Optimization of routes	x	x	x	x
Tracking of intralogistics transport vehicles			x	
Navigation and localisation of mobile robots			x	x
Management of machines and vehicles	x	x	x	x
Overall	**12**	**11**	**4**	**5**

Class 1 can cover 12 out of 17 of the mentioned applications, followed by Class 2 with 11 out of 17. Due to the higher number of applications and thus the broader applicability as well as the fact that marker-based identification methods better meet the low cost requirements of this paper (cf. chapter 2.7.1), Class 1 (marker-based external location) is to be selected for further implementation in the following.

Due to the heterogeneous system requirements, the highest requirements of this class are selected for prototypical implementation and verification. The result is summarized in Figure 8.

Architecture	**External localization**	Self-Localization			
Identification methods	**Marker-based**	Feature-based			
Accuracy	**< 1 cm**	< 5 cm	< 10 cm	< 50 cm	< 100 cm
Coverage	< 1 m	< 5 m	**< 10 m**		
Image rate	< 15 fps	**≥ 15 fps**			
Positioning rate	**< 1 s**	< 15 s			
Update rate	On request	Event-driven	**Periodically**		
Number of objects	≤ 10	≤ 50	**≥ 50**		
Scalability	Low	Medium	**High**		
Required computing power	**Low**	Medium	High		
Power supply	**Active**	Passive			
Output data	Undefined	xy position	**xy position, ID**	xyz position	xyz position, ID
Orientation	**No**	Yes			

Figure 8 Required values

Chapter 3

Development and Construction of a Prototype

3.1 System Design

In the following, the experimental setup is described which corresponds to the criteria derived from the literature in the previous chapter for the application scenarios designated as Class 1 (marker-based external location). For the identification and localization of relevant objects, these are equipped with monochrome 2D markers. An embedded vision system consisting of a camera system and a computing unit is permanently installed on the ceiling in the operational environment. The optical sensor serves for the digital recording of the scene, which is examined on the computing unit for position determination on existing localization objects. Only relevant information is forwarded from the computing unit to higher-level systems.

Based on the position of the localization system, the current position of localization objects can then be determined. The position of the localization objects in the field of view of the localization system can be constantly determined by the continuous function of the localization system. This information forms the basis for the class 1 application scenarios described in Chapter 2.

Figure 9 shows the scenario for camera-based indoor localization in this paper as a starting point for the developed concept.

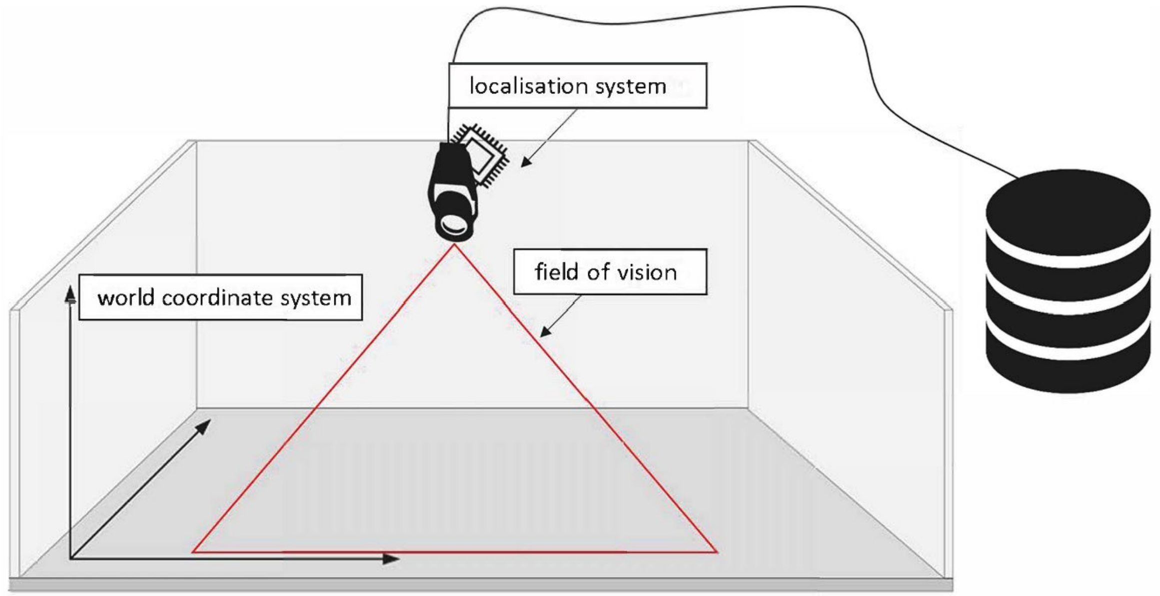


Figure 9 Proposed concept

3.2 Hardware Selection

Classic image processing systems consist of a camera body with a lens for image acquisition, a cable that connects the camera system and a computing unit. The computing unit is an industrial PC (IPC), which has a high computing power and is able to process images and generate the desired data from the information obtained by image processing software. (Rademacher, 2017)

However, the selected hardware components should be significantly less expensive than hardware suitable for industrial use. A subsystem requires a computing unit and a camera system. The wireless communication module required for later operation in practice is ideally integrated in the location engine, which reduces the total effort and thus the costs. Therefore, when selecting the computing unit, it is highly important to choose the widest possible selection of wireless technologies, which enable communication with different sensors.

3.2.1 Functional model

A computing unit and a camera system must be selected for the functional model. Since mutual compatibility with regard to hardware and software interfaces is a prerequisite for operating a camera on a processing unit, a processing unit is selected first and then a suitable camera.

3.2.1.1 Computing unit

When it comes to the well-founded selection of such a computing unit, there is a dilemma in the large selection of end customer products. However, many of the products available on the market differ only slightly from each other, often in areas whose benefits are unclear during the decision-making process. Conversely, a product rarely has all the desired properties. Analogous to multi-criteria optimization, there are therefore many solutions which are in principle incomparable with each other with regard to all criteria. (Pieper, 2017). An optimal selection therefore does not appear possible in advance. Due to the short product cycles for electronic end-customer products (Karwowski, 2011) the recommendation of a concrete product is of limited use, since the computational unit tested in this work might be outdated at the time of the publication. Therefore, a product class is selected that is particularly suitable for use in production environments. Due to the variety of products in this product class the system integrator has the maximum flexibility and future security, but is also spoilt for the choice.

Therefore, there is a large number of computational units to choose from. Hennessy and Patterson (2012) distinguish between five types of computational units and compare them based on the resulting costs and design goals. These five classes include personal mobile devices, desktops, servers, computing clusters and embedded devices (see Table 12). Due to their high cost, three out of five computational unit classes can be considered unsuitable in advance:

1. Server.
2. computer clusters, these are expensive to purchase and are also not suitable for simple integration into the infrastructure.
3. desktops, although laptops also belong to this class (Hennessy and Patterson, 2012) and these are significantly cheaper than the classes of servers and computing clusters, their use does not meet the low-cost requirements of the work.

Table 12 Comparison of computing units (Hennessy and Patterson, 2012)

Feature	Personal mobile device	Desktop	Server	Cluster	Embedded
System Costs	87-870 €	261-2,175 €	43,500-8,700,000 €	87,000-174,000,000 €	8.7-87,000 €
CPU Price	8.7-87 €	43.5-435 €	174-1,740 €	43.5-217.5 €	0.0087-87 €
Critical system design issues	Costs, Energy, Media Performance, Responsiveness	Price/performance ratio, energy, graphics performance	Throughput, availability, scalability, energy supply	Price/performance ratio, throughput, energy proportionality	Price, energy, application-specific performance

In principle, the use of mobile devices is a good option. They often already have an integrated camera. However, these are intended for manual (personal) photography. For this reason, the operating system does not support switching off the screen during image acquisition by default, which leads to high power consumption and thus reduces battery life. However, this problem is already successfully solved by some apps (apk.support, 2017). However, there are still limitations due to the built-in cameras, as their focus and zoom can be adjusted electronically (developer.android.com, 2018). The integration of a further camera or the exchange of the installed cameras is not planned. (Drees, 2017; Wölbert, 2015). In addition, robust and powerful mobile devices with costs of 500€ and more are comparatively expensive and therefore do not appear to be suitable for cost-effective and flexible image processing in production environments.

As an alternative to mobile devices, the technological development of recent years has enabled image processing solutions that in the past could only be implemented with high-class camera and PC platforms to also be built "embedded" (Rademacher, 2017).

Today, embedded devices are already used in a variety of application scenarios, for example in vehicle assistance systems. (Marwedel, 2011). This equipment class appears therefore suitable, since they possess apart from the cost advantage also the advantage of the energy efficiency. For the development of such embedded devices, there is a large number of hardware modules available that combine all the required components on a single board, so-called single-board computers (SBC). (Doerr, 1978). Low-cost representatives of this class therefore serve as hardware platforms for the functional sample and can be used directly in a later product for cost reasons. A comprehensive comparison of many SBCs can be found on Wikipedia ('Comparison of single-board computers', 2018).

3.2.1.2 Camera body

A large number of cameras are available on the market, which can be divided into different camera types, sensor types, sensor sizes, frame rates, lens mount options, resolutions, shutter types, interfaces and power supplies. A further selection criterion is the software availability or the possibility of software control. The latter depends on the manufacturer; drivers for Windows, Linux or Mac-based systems can be found from a large number of manufacturers.

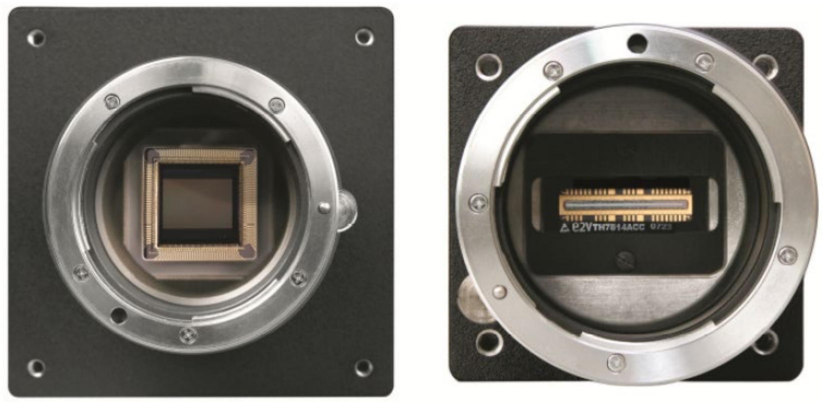
Camera types:

Basically, camera types can be distinguished by their image sensors, which are subdivided into on the one hand by their structure, which consists of a large number of light-sensitive sensor elements (pixels), which are either arranged in one line or as a matrix. On the other hand, they differ in functionality and production technology. (Naumann, 2014)

In the industrial environment, matrix (or area) or line scan cameras are used for the majority of applications. Matrix cameras have image sensors which consist of several rows and columns and thus consist of a flat, mostly rectangular arrangement of pixels. This arrangement of the pixels enables the imaging of a two-dimensional image. (Bloß and Bauer, 2000)

Matrix sensors are by far the most frequently used sensor types. Area cameras also include cameras from the private and commercial sectors. These include webcams, photo cameras, video cameras, etc. (Demant, Streicher-Abel and Springhoff, 2011)

Line scan cameras, on the other hand, only have image sensors that consist of a single line of pixels. Thus line scan cameras are only able to generate a two-dimensional image when the object or the camera moves. For example, cameras with a line sensor are often used to inspect workpieces on a conveyor belt. These are particularly advantageous if a workpiece to be inspected does not fit into the format of a normal image. In addition, line scan cameras are characterized by their high resolution and frame rate. (Bloß and Bauer, 2000; Demant, Streicher-Abel and Springhoff, 2011)



Matrix sensor Line sensor

Figure 10 Comparison between a matrix- and a line sensor(Demant, Streicher-Abel and Springhoff, 2011)

Industrial camera

So-called industrial image processing systems are used in the industrial environment. These consist of an industrial camera, lens, an external computing unit for image processing, image processing software and cabling for connecting the individual components.

Industrial cameras generally require an additional lens for image acquisition. This modular design is particularly advantageous because industrial image processing systems must be selected and adapted to the customer-specific application scenario. (industrie-kamera.de, no date)

In this class, cameras that are equipped with infrared filters (IR) and without (NoIR) are found as well. The advantage of the camera using no infrared filter is that a more sensitive sensor with very low luminosity is given. If the NoIR camera is equipped with additional infrared LED's, the camera can also be used in very poor lighting conditions. For applications that offer daylight conditions, however, cameras with IR filters are to be preferred.

They thus allow the user a high degree of flexibility, as they are offered in different versions and price classes. A special form of industrial cameras are the so-called board-level cameras, which do not have a housing and are characterized by their small and compact size. These are often used in embedded vision applications.

Commercial cameras

In addition to industrial cameras, there is also a large number of cameras from the commercial sector. Webcams are the best alternative among these, as they have a small design and their low power supply is sufficient.

Comprehensive comparisons of many webcams are available on the Internet (see e.g. Embedded Linux Wiki (elinux.org, 2018)). The disadvantage of using standard webcams, however, is the low resolution of the image sensor used, which is compensated by software interpolation.

In addition, the power supply also poses a challenge for some models. The most commonly used single board computer, Raspberry Pi, for example, requires an input voltage of 5 volts, which is achieved with a suitable power supply. Some webcam models work with higher voltages which would require a second power supply. In addition to voltage, power consumption is also an important criterion. Some webcams require a higher voltage than a Raspberry Pi can provide. In this case, an additional, externally powered USB hub must be used. (raspberrypi.org, no date)

In the non-commercial sector, image resolution is often only achieved by image interpolation. In the digital age this is the basis of image processing. It is a technique of supplementing by estimating image data from an existing set of discrete signals. In interpolation, a captured image is virtually enriched with a resolution of 640x480 pixels by inserting additional pixels on the software side, which apparently results in a higher resolution of 1280x1024 pixels. Therefore, the real image resolution has to be taken into account when using it. (Bannore, 2009) Figure 11 shows a simplified example of interpolation (Busch, 2009).

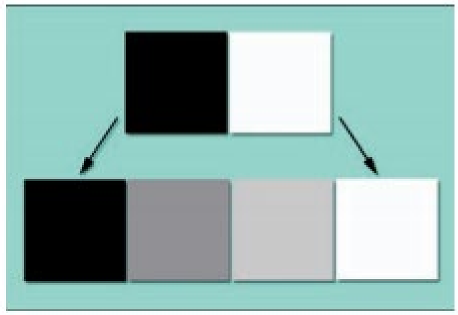


Figure 11 Software interpolation (Busch, 2009)

A further limitation of webcams is that they are intended for use at the end customer and therefore often do not offer the possibility of the simple exchange of individual components. Furthermore, webcams do not have their own intelligence and are used purely for recording images or videos. For the fulfilment of tasks a software is still needed, which runs on the selected computing unit and can interpret the pictures, in order to fulfil the set task. For these reasons, the use of a webcam in the context of this work will not be further investigated.

Sensor type

For a long time only Charge Coupled Device (CCD) image sensors were used for the sensor types, as there were no alternatives. However, with the rapid development of Complemenary Metal Oxide Semiconductor (CMOS) based image sensors, they are now a competitive alternative to them. Particularly with regard to pixel size and number, these sensors are no longer inferior to CCD sensors. At the same time, CMOS technology enables flexibility and performance increases at the same time as costs can be reduced. They have high photosensitivity, brightness dynamics (up to ~ 150 dB) and a wide permissible temperature range. (Bloß and Bauer, 2000; Demant, Streicher-Abel and Springhoff, 2011)

In addition, both sensor technologies are available as monochrome sensors and colour sensors. However, this has no influence on the price and is not relevant for the detection of monochrome markers.

Sensor size

The sensor size describes the optical format of the sensor used or its image diagonal. The size designation is historically based on those of television cameras for which so-called Vidicon vacuum tubes were used. Thus, the actual sensor dimensions do not correspond to the specifications. For example, a 1/2" sensor has an 8 mm diagonal instead of the expected 12.7 mm. It should also be noted that there is no fixed geometric relationship between the sensor sizes.

This is particularly important when selecting a suitable lens, as the image circle of the lens must at least correspond to the sensor size. Typical optical formats are listed in Table 13. (Demant, Streicher-Abel and Springhoff, 2011)

Table 13 Sensor sizes

Optical format [inch]	Width [mm]	Height [mm]	Diagonal [mm]
1/1	12.8	9.6	16.0
2/3	8.8	6.6	11.0
1/2	6.4	4.8	8.0
1/3	4.8	3.6	6.0
1/4	3.2	2.4	4.0

The sensor size has a large influence on the Field of View (FOV) of the camera system, and should be selected as large as possible. However this aspect is quiet difficult to handle, as only industrial camera bodies offering a wide range of different sensor sizes. The difficulty is to find a camera body of this class that can meet the requirements and be classified as low cost. In addition, the cost of a camera body increases with sensor sizes. Furthermore, lenses that are suitable for the use with large image sensors are more cost-intensive as well.

Mount

The mount is the interface between the camera body and the camera lens. Today there is a large number of different mounts available, which are characterized by their support dimensions and the maximum usable sensor size. The flange focal distance describes the distance between the contact surface of the lens and the image sensor. The most common mounts for machine vision applications are shown in Table 14.

Table 14 Lens mounts

	flange focal distance	Max. sensor size
F-Mount	45.600 mm	- 42 mm
C-Mount	17.526 mm	- 4/3“
CS-Mount	12.500 mm	1/1.8“ – 4/3“
S-Mount / M12	undefined	-1/.8“ – 2/3“

The interaction between the sensor size and the image circle of the lens is crucial. If the image circle is smaller than the sensor size, the image quality deteriorates. It is recommended to use a lens with an image circle of the same size or a slightly larger image circle, as this allows optimum utilization and leads to lower costs. For example, lenses designed for sensor sizes of 1" or 4/3" are significantly more expensive and should only be selected in combination with corresponding image sensors. (Czeranowsky, 2017)

If, for example, a camera body with a C-mount interface should be available and a CS-mount lens should be determined as optimal for cost reasons, it is also possible to use an adapter.

Resolution

The resolution depends on the pixel size of the sensor. The larger a pixel, the larger its light-sensitive area and thus also its sensitivity. In industrial applications, for example, pixel sizes between four and seven micrometres are used, since smaller pixels lead to a deterioration in image quality. An exception here are colour CMOS sensors, which still deliver acceptable results with a pixel size of up to two micrometres. Due to the constant further development of the sensors, a large number of different high-resolution sensors are available today. Classical sensors used in industrial image processing work with resolutions of 640x480 which corresponds to 0.3 megapixels (MP) and 4008x2672 pixels with 10 MP. There are also high-resolution image sensors for special applications (Demant, Streicher-Abel and Springhoff, 2011)

Table 15 Data of typical matrix sensors (Linß, 2018)

Resolution (H ×V)	Megapixel	Pixel size [µm²]	Sensor size [inch]	Frame rate
640×480	0.30	5.60×5.60	1/4	120
640×480	0.30	7.40×7.40	1/3	90
768×576	0.44	8.30×8.30	1/2	60
1024×768	0.80	4.65×4.65	1/3	30
1392×1044	1.30	4.65×4.65	1/2	20
1392×1044	1.30	6.45×6.45	2/3	30
1600×1200	2.00	4.40×4.40	1/1.8	25
2560×2048	5.00	3.45×3.45	2/3	15

However, it should be noted that a high resolution with a small image sensor means that the pixels are also very small and therefore require a good lightning of the scene to be captured for a high-quality image.

The required sensor resolution can be calculated via the field of view (FOV) and the required resolution per pixel. This takes into account that at least 2x2 pixels are required to detect the smallest detail. This results in the following formula.

$$Sensorresolution\ (min) = \left(\frac{FOV}{Resolution\ pro\ Pixle}\right) * 2 \qquad (2)$$

For example, a FOV of 3500mm and a size of 10x10mm of the smallest detail to be detected would require a sensor resolution of 700 pixels. For practical use, however, a much higher resolution than 2x2 pixels is recommended. (Naumann, 2014)

Frame rate

The frame rate describes the number of frames per second and is usually expressed in fps (frames per second) that can be captured by a camera system. It is essentially influenced by the exposure time, the time required to read an image and the time required to transmit it. (Demant, Streicher-Abel and Springhoff, 2011)

Today, a wide range of sensors is available, enabling frame rates from less than 10 fps up to greater than 100 fps. The maximum achievable frame rate is influenced by several factors. However, sensor technology and the number of pixels to be read have a particularly strong influence. It is not uncommon for the frame rate to decrease with increasing resolution. (Naumann, 2014)

This is shown as an example in Table 16 using a camera with a resolution of 5 megapixels. (Jones, 2017)

Table 16 Resolution and possible framerates

Resolution	Frame rate
2592x1944	1/6 <= fps <= 1
1920x1080	1 < fps <= 30
640x480	60 < fps <= 90

For use in production scenarios, cameras must be able to detect the position of moving objects, in particular, workpieces moving on assembly lines and industrial trucks. For example, assuming that an industrial truck moves at a maximum speed of 15 km/h, a frame rate greater than 15 frames per second is required to sufficiently determine its accurate position (Reinhart, 2017).

A frame rate of more than 15 frames per second is also a necessary criterion for the examination of goods transported on conveyor belts, as these transport goods at speeds of 0.15 - 2.5 m/s (0.54 - 9 km/h). (Heinrich, 2017)

Shutter type

The exposure of the sensor can basically be carried out according to two principles, the rolling shutter and the global shutter principle. Most of the cheaper sensors are based on the Rolling Shutter principle. In this principle all pixels are exposed with the same exposure time, but the exposure takes place for each line of the sensor one after the other. This can be disadvantageous when taking pictures of scenes with moving objects. (Naumann, 2014)

Energy absorption

The electrical connection plays an important role in image processing systems, since cameras only function reliably if there is a stable power supply and switching signals with good signal quality. Thus, a disturbed power supply in industrial image processing leads to malfunction of the camera system. (Demant, Streicher-Abel and Springhoff, 2011)

Communication interface

The communication interface of the camera system is used for data transmission to the computing unit. A large number of different interfaces are available on the market. The main interfaces are CameraLink, FireWire, Universal Serial Bus (USB), Gigabit Ethernet and CSI. (Demant, Streicher-Abel and Springhoff, 2011)

Table 17 Performance of communication interfaces(+/- medium, + good, ++ very good, - moderate, -- know) (Demant, Streicher-Abel and Springhoff, 2011)

	CameraLink	FireWire	USB 2.0	USB 3.0	Gigabit-Ethernet	CSI
Transfer rate	6800 Mbit/s	800 Mbit/s	480 Mbit/s	5000 Mbit/s	1000 Mbit/s	1250 Mbit/s
PC interface	-	--	++	++	++	--
Real-time capability	++	+	-		+	+
Availability of components	+/-	+	++	++	++	+/-
Industrial suitability	+	+	-	-	+	
Cost	-	+	++	++	+	++

Software availability

When it comes to software availability, it must be taken into account whether the necessary drivers are offered free of charge by the manufacturer on a cross-platform basis. If this is not the case, additional costs may arise or compatibility with the selected computing unit may not be possible.

3.2.1.3 Lens

In addition to selecting the camera module, a suitable lens must also be selected. Today, a large number of different lenses are available, such as entocentric or telemetric lenses, as well as wide-angle and fisheye lenses. In addition, there are also special lenses that are used for near infrared light.

The focal length of the lenses is decisive for categorization. As the focal length increases, the field of view decreases, but the zoom increases. For example, lenses with focal lengths of 100 mm are regarded as super telephoto lenses.

Often entocentric lenses are used for machine vision, these have a fixed aperture angle. This is similar to the human eye. Thus objects appear smaller in large distances and objects appear larger in small distances. (baslerweb.com, no date; Naumann, 2014)

The following Table 18 shows the basic classes available on the market for lenses and their properties.

Table 18 Lense classification (Schröder and Treiber, 2007)

Classification	Focal length at 2/3“	Further features
Supertelephoto	100 mm	▪ Long and heavy construction of the lens ▪ The light intensity is proportional to the lens diameter dEL. The camera system has an unfavourable centre of gravity with a larger lens diameter dEL.
Telephoto	35 mm	▪ Especially good light intensity, even if slightly reduced compared to the normal lens ▪ Relative neutral magnification of the captured scene ▪ Compromise in the positive and negative properties of a telephoto and normal lens
Normal	12 mm	▪ Corresponds to the horizontal angle of aperture of the human eye $\delta Mensch \approx 48$ ° ▪ - High optical manufacturing quality assumed: no distortions in the image ▪ Handy dimensions and high luminous intensity ▪ High degree of sharpness ▪ Neutral image of the captured scene
Wide angle	6 mm	▪ Barrel-shaped distortions in the edge area of the optical image require higher correction effort ▪ Lenses are less light intensive
Super wide angle (fisheye)	2 mm	▪ Extreme wide-angle lenses with $\delta > 110$ ° also known as fisheye lenses and have extreme barrel distortions.

In addition to these basic classes, lenses can also be distinguished by their focus; here there are four different classes of different lenses.

Table 19 Lens characteristics (Czeranowsky, 2017)

Lens type	Description	Example
fixed focus lens	Focus is not adjustable	Rare, old mobile phones
fixed focal length lens	Fixed focal length, focal length is not adjustable	Normal MV lens
varifocal lens	Focal length is adjustable	Lens for IP cameras
zoom lens	As varifocal, but remains focused when adjusting the focal length	SLR cameras

In order to select the optimum lens for an application, various factors must be taken into account, including the following:

Image circle diameter and sensor size

For the selection of the correct lens, a lens must be selected which at least corresponds to the optical format of the camera or is suitable for a larger sensor size. The common sensor sizes and their properties have already been shown in Figure 12.

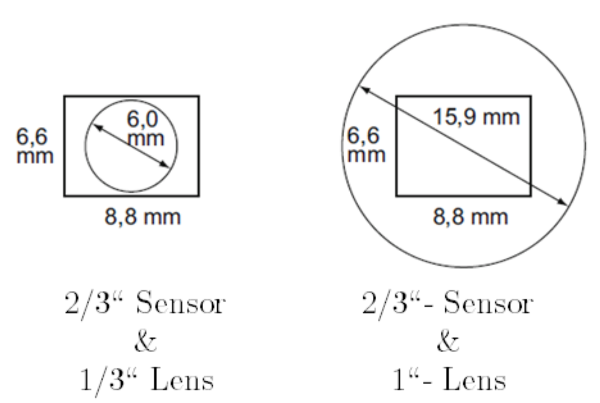


Figure 12 Connection between sensor and lens (Demant, Streicher-Abel and Springhoff, 2011)

The sensor size is important because it also determines the image circle diameter, the area of the sensor that is exposed evenly by the lens without shadowing at the edge - also known as vignetting. See Figure 13 in which the orange and red borders represent the sensor size of the camera body. A lens with an image circle diameter that is too small was selected for the orange area, which results in vignetting in the edge area of the image. (baslerweb.com, no date)

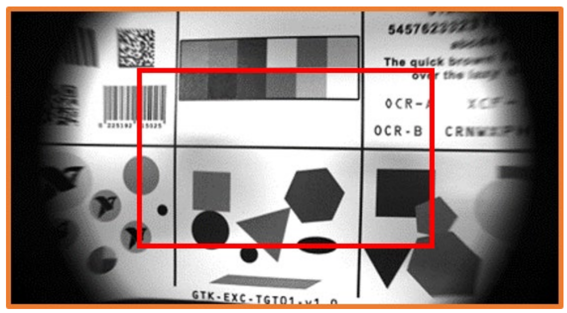


Figure 13 Vignetting caused by choosing the wrong lens (orange field), optimal picture (red field)(ni.com, 2018)

Manufacturers specify the image circle diameter similar to the sensor size. Ideally, a camera body with a sensor size of 1/4" is used together with a lens with an image circle diameter of 1/4". The use of a lens with a larger image circle diameter does not lead to any impairments, but the image angle would change, which in principle can even be advantageous. This is not always useful for cost reasons.

Focal length and sensor size

The focal length of a lens describes the distance f from the projection centre to the image plane, so the focal length determines how large the image section or angle of view is that can be imaged on the camera sensor used. It is determined by the sensor width, the object width and the working distance and indicated in (mm).

In terms of focal length, a distinction is made in principle between two lens types:

lenses with a fixed focal length and lenses with a variable focal length (zoom lenses) both differ particularly in the manufacturing costs, so zoom lenses are more cost-intensive. Lenses with fixed focal lengths are characterized not only by their lower cost, but also by their high light intensities (see section "Aperture and light conditions"). (lomography.de, 2011) The most common focal lengths are 4mm, 6mm, 8mm, 12mm 16mm, 25mm, 35mm, 50mm, 75 mm and 100mm.

Angular Field of View (AOV)

However, the focal length itself does not yet provide any information about the viewing angle of the camera system, but is still one of two decisive constants for its calculation. In addition to the focal length, the size of the sensor used is also required. This, in turn, is required to determine the Field of View (FOV) of the overall system. The AOV can be specified in three different forms: the horizontal AOV, the vertical AOV and the diagonal AOV. Manufacturers usually only specify the horizontal AOV.

The AOV can be determined as follows. (Kruegle, 1995)

$$AOV\ (H,V,D) = 2 * \arctan\left(\frac{Sensor\ (H,V,D)}{2 + f}\right) \qquad (3)$$

The determination of the AOV of super wide angle lenses is even more complex. The Formula 3 is particularly suitable for this purpose. (bobatkins.com, no date)

$$AOV\ (H,V,D) 4 * \arcsin\left(\frac{Sensor\ (H,V,D)}{4 + f}\right) \qquad (4)$$

During the technology selection different manufacturer data could be identified. These are always related to a given sensor size.

However, since it is possible that the selected camera body uses a smaller or larger image sensor, these data could not always be used for a comparison. Usually, however, information about focal length, mount, iris (F-number) and maximum image circle as well as optimum working distance can be found in manufacturers specifications. Using these values, the formula for calculating the AOV can then be used to convert the AOV to the sensor size of the camera body.

However, this formula should be seen as an approximation to the actual value, as it is based on the ideal case of a thin lens to determine the AOV. However, the actual values of commercially available lenses may differ slightly from these values due to design reasons. Table 20 gives an overview of typical focal lengths and sensor sizes. The calculated value is compared with the manufacturer's specifications (the values in brackets in the table). (Demant, Streicher-Abel and Springhoff, 2011)

Table 20 Interaction between focal length and sensor size using the example of common sizes (Demant, Streicher-Abel and Springhoff, 2011)

Sensor size								
f	1/4“		1/3“		1/2“		2/3“	
8,5 mm	23.91°	(24.02°)	31.53°	(31.87°)	41.26°	(42.09°)	54.74°	(56.49°)
16 mm	12.84°	(12.70°)	17.06°	(16.91°)	22.62°	(22.48°)	30.75°	(30.72°)
50 mm	4.12°	(4.13°)	5.50°	(5.48°)	7.32°	(7.32°)	10.06°	(10.06°)

Field of View (FOV)

Once the AOV has been determined, the field of view can be determined using this and the working distance. The same trigonometry is used for this. (Carr, 2016)

$$FOV\ (H,V,D) = 2 * (\tan\left(\frac{AOV}{2}\right) * WD \qquad (5)$$

A modified form of formula 5 is used to determine the FOV of super wide angle lenses. Based on the different calculation basis for the AOV, Formula 6 is used.

$$FOV\ (H, V, D) = 4 * (\tan\left(\frac{AOV}{4}\right) * WD \quad (6)$$

However, other approaches can also be found in the literature, for example Vision Doctor gives the following formula. (vision-doctor.com, no date)

$$FOV\ (H, V, D) = \left(\frac{WD}{f - 1}\right) * Sensor\ (H, V, D) \quad (7)$$

Demant, Streicher-Abel and Springhoff (2011) and Kruegle (1995) suggest the following approximation

$$FOV\ (H, V, D) = \left(\frac{WD}{f}\right) * Sensor\ (H, V, D) \quad (8)$$

It should be noted in the calculation bases mentioned that these refer to the ideal case and that the actual values for complex lens systems may deviate for design reasons (Demant, Streicher-Abel and Springhoff, 2011). However, practice has shown that these deviations are usually very small.

There are basically three ways to influence the FOV of a camera. The first and easiest way is to change the working distance. As the working distance increases, the FOV also increases automatically. The second possibility would be to use a lens with a different focal length and thus a different AOV. The third option would be to use a camera body with a different sensor size. Here the use of a larger sensor would also lead to an extension of the field of view.

Optimal is therefore the use of a lens with as large an AOV as possible, which, however, also has disadvantages. Lenses with small focal lengths have a high level of distortion. Furthermore, lenses with small focal lengths often cannot achieve the highest performance level, as difficulties can arise when using medium to large sensor sizes. (edmundoptics.com, no date)

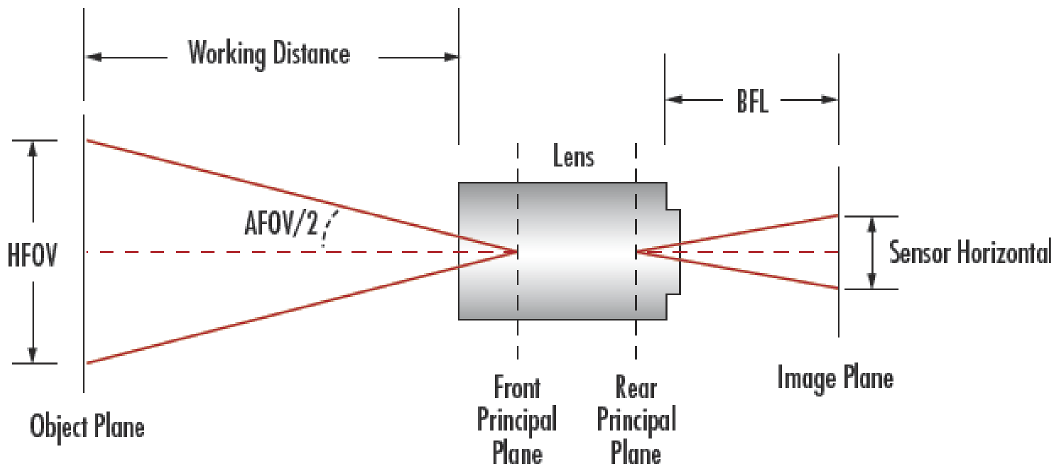


Figure 14 Function of a lens (edmundoptics.com, no date)

Table 21 illustrates the interaction of AOV, WD and sensor size for a lens with a focal length of 4 mm.

Table 21 Connection between sensor size ad f-number

Sensor size (1/4“) / 3,76x2,74mm			
AOV (H)	WD (1 m)	WD (2 m)	WD (3 m)
50.35°	940 mm	1880 mm	2820 mm

Sensor size (1/2,5“) / 5,8x4,3 mm			
AOV (H)	WD (1m)	WD (2 m)	WD (3 m)
71.88°	1450 mm	2900 mm	4350 mm

It is therefore important to choose the AOV as large as possible to optimize the FOV of the camera system, but at the same time the sensor size should not be neglected as this is ultimately responsible for the image of the scene.

Resolution

The supported resolution of the lens is just as important as the resolution of the camera sensor mentioned in Chapter 3.2.1.3. The acquisition of high-resolution images requires a high-resolution image sensor on the one hand and a lens that supports this resolution on the other. Therefore one should pay attention to this when choosing the lens. If a lens is selected that does not support the full sensor resolution, this leads to a decrease in image quality. (Czeranowsky, 2017)

Aperture and light conditions

The number of apertures (F-number) indicates the ratio of focal length to the diameter of the aperture and describes the opening of the aperture. It has a direct influence on image quality and brightness. Lenses with a high f-number have a smaller aperture and therefore allow less light through. Therefore, lenses with a low f-number are particularly advantageous in applications where no additional illumination is required. (baslerweb.com, no date) There are no major price differences between f-stops of 1.4 and 1.8, but if a smaller f-stop such as 1.2 is chosen, a significant jump in price can be expected. (Demant, Streicher-Abel and Springhoff, 2011)

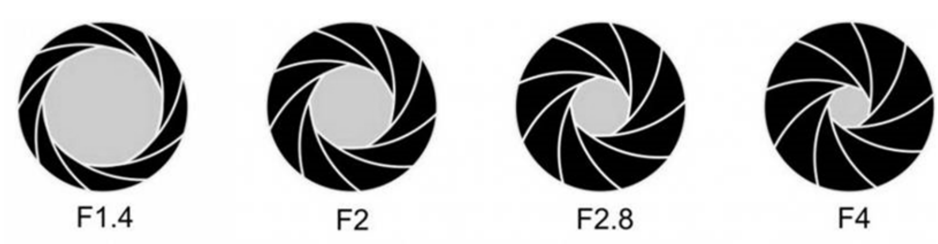


Figure 15 Difference of f numbers (Knaak, no date)

3.2.2 Technology Comparison and Selection

After the general description of the technology, the following subchapter will now provide the concrete technology selection.

3.2.2.1 Computing unit

In order to meet the requirements placed on the localization system, the computing unit must be able to evaluate the captured scene within a certain period of time and must not be too cost-intensive at the same time. For this purpose, the use of single board computers seems to be particularly useful. These have sufficient computing power and are particularly suitable for most application scenarios due to their compact design.

The Raspberry PI is one of the best known and cheapest single board computers. Especially the current model (Schmidt, 2018) is regarded as a particularly suitable technology for industrial image processing (Schaar, 2018). It has the following characteristics:

- relatively low costs, of approx. 35€.
- high availability of expansion options (especially the large number of available, cost-effective sensors) (raspberrypi.org, no date).
- small design, which allows integration in many places (raspberrypi.org, no date).
- frequent use and the extensive documentation available as a result. In addition, manufacturers of industrial cameras also offer drivers that connect their cameras in combination with the Raspberry Pi.
- high availability of powerful open source software and additional libraries such as the Open Source Computer Vision Library (OpenCV)
- continuous development and improvement of existing models
- low power consumption and low operating voltage of only 5 Volt

- wide application in research and application projects (Straub et al., 2013; Tso et al., 2013)
- easily as well as globally available even in large quantities (Kunow, 2016)
- large selection of communication modules for connection to higher-level IT systems. The latest module includes a Wi-Fi module (2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN), a Bluetooth 4.2 module which also supports Bluetooth Low Energy (BLE) and a Gigabit Ethernet interface. Other modules such as adapters for communication via mobile radio or Zig-Bee are also available. Despite the very compact design, the existing connections allow the integration of a large number of Linux-compatible communication modules. (raspberrypi.org, no date)

A further advantage is the large number of available cameras that have already been successfully integrated. In addition to the cameras specially developed for this board, conventional webcams or industrial cameras can also be connected.

3.2.2.2 Camera body & lens

Based on the characteristics determined in Chapter 3.2.1 and the associated influencing factors on the camera system, the requirements for the selection of camera body and lens set out result in Figure 16 and Figure 17. The selection of a CMOS sensor seems to be an obvious choice here, as these can be purchased at a lower price than CCD sensors based on their manufacturing process and guarantee a high image quality even at high frame rates. Here it is necessary to optimize the sensor size. A sensor size of 1" would be optimal. Based on the selection of the Raspberry Pi, only USB or CSI can be used as interfaces. To be able to use one of the other interfaces, an additional frame grabber would have to be integrated, which would significantly increase the integration and cost effort.

Camera body						
Camera type	**Matrix cameras**	Line scan cameras				
Sensor type	CCD	**CMOS**				
Sensor size	**1"**	2/3"	1/1.8"	1/2"	1/3"	1/4"
Lens mount	**C-Mount**	CS-Mount	S-Mount			
Resolution	0,3 MP	1 MP	**5 MP**	> 5 MP		
Frame rate	1 fps	< 15 fps	**< 30 fps**	< 60 fps		
Shutter type	**Global Shutter**	Rolling Shutter				
Power supply	**Integrated**	External				
Interfaces	FireWire 3.2 Gbit/s	Camera Link 680 Mbit/s	GigE 1000 Mbit/s	**USB 300 Mbit/s**	CoaXPress 6.25 Gbit/s	CSI 1.25 Gbit/s
Sensor	colour	**Monochrome**				
Software	**Free of charge**	Chargable				

Figure 16 Attributes of the optimal camera body

Lens						
Focal length (f)	Super telephoto	Telephoto	Normal	Wide angle	Super wide angle	
Sensor size	1”	2/3”	1/1.8”	1/2”	1/3”	1/4”
Aperture	1.4	2	2.8	4		
Lens mount	C-Mount	CS-Mount	S-Mount			

Figure 17 Attributes of the optimal lens

The camera systems shown in Table 22 could be determined on the basis of the requirements set. A list of all compared cameras is given in Appendix D.

Table 22 Camera Comparison

Typ	Arducam Camera body	Logitech C920	UI-3280CP-C-HQ Rev.2	BFS-U3-88S6M-C
Camera type	area scan camera	area scan camera	area scan camera	area scan camera
Sensor type	CMOS	CMOS	CMOS	CMOS
Sensor size	1/4"		2/3"	1"
Mount	CS		C	C
Resolution	5	3	5,04	8,9
Resolution / frame rate	2952 x 1944 / 15 fps	1920×1080 / 30 fps	2456 x 2054 / 36 fps	4096 x 2160 / 32 fps
Shutter type	Rolling		Global	Global
Power supply	integrated	integrated	integrated	integrated
Interface	CSI	USB 2.0	USB 3.0	USB 3.0
Software	free	free		
Price [€]	15	69,94	714	1895

Lens				
	CS-2,8-3MP 140		HF6XA-5M	eo-Lens
f	2.8	3.67	6	12.5
AOV (D)	79.61	78.00	85.02	88.60
Price [€]	16		208.25	595
Cost [€]	31	69.94	922.25	2490

However, the requirement for a large image sensor appears to be a challenge, as camera bodies that have a 1" image sensor are more expensive. In addition, no optimal lens could be identified either, because either the resolution of the lens is too low or the sensor size of the lens is inappropriate. Furthermore, no lenses with a focal length of less than 6 mm could be identified that would meet the set requirements.

It should also be kept in mind that even if such a camera system had been identified, a much lower pixel density per m^2 would be available for position determination at a resolution of 5 MP than with a smaller sensor size.

A camera system with a sensor size of 1", using a lens with a focal length of f 2.8 mm and a working distance of 7m, would theoretically allow an area of 762 m^2 to be viewed while a camera system with a sensor size of 1/4" would only allow an area of 60.75 m^2 to be viewed. If, however, both camera systems have a resolution of 5 MP, the camera system with the smaller image sensor would offer a pixel density approx. 12 times higher and thus theoretically a much higher accuracy.

However, the objective of this work is the use of low cost, state of the art technology components, to which a large part of the identified camera systems cannot be counted. For example, the UI-3280CP-C-HQ Rev.2 and BFS-U3-88S6M-C are hardware suitable for industrial use and therefore do not fall under the category of low cost.

The camera model of the Raspberry Pi can already be purchased for 15 € and represents a relevant technology component. Due to the wide range of components from third-party suppliers, this seems to be relevant for the application. Using the Arducam camera body, which is based on the technology of the Rapberrry Pi camera model, it would also be possible to mount CS-Mount lenses.

The requirement for the sensor size must be reconsidered due to the lack of availability of components which correspond to the general conditions of this work. The Arducam camera body seems to be a very promising alternative to meet these requirements.

If a smaller sensor size is used, the requirement on the mount can be waived. The use of a C-mount is particularly important only when sensor sizes of up to 4/3" are used. For sensor sizes between 1/1.8" and 4/3" the use of CS-Mount lenses is recommended. The low cost of these lenses also speaks for their use. (Czeranowsky, 2017)

Now the requirements for the shutter type have to be reconsidered. In principle, the use of a global shutter is recommended, as the entire image sensor is illuminated at the same time, which speaks particularly for the use with dynamic observations. With the rolling shutter, the illumination takes place line by line. However, a rolling shutter can also be used for dynamic viewing, in which case the exposure time must be adjusted. (baslerweb.com, no date)

The Raspberry Pi camera is able to adjust the exposure time and some other settings depending on the situation in order to capture an optimal image (Jones, 2017). The use of a Rolling Shutter does not seem to have a big disadvantage compared to a Global Shutter.

For the selection of the optimal lens, the initial requirements also change. It has already been discussed why a smaller image sensor is also relevant for use. With the adjustment of the sensor size, an adjustment of the mount is also possible at the same time, since this must be selected depending on the camera body, the selection of the optimal lens also depends on this.

Based on the adapted requirements, the market was examined for available lenses. Particular attention has been given to select a CS-Mount lens with the desired properties.

As already mentioned, the Arducam camera body is particularly suitable for the construction of the prototype. This body can be supplied in two variants. The first variant has a lens with a focal length of 4 mm and the second a lens with a focal length of 6 mm and is offered in a package for 30 €.

However, a focal length of 2.8 mm was set as a requirement, for this reason an additional lens was ordered. Lenses with a focal length of less than 2.8 mm and a CS mount could not be found on the market.

However, the camera module is also available with an M12x0.5 mount. Here a super wide angle lens with a focal length of 1.67 mm is installed directly by the manufacturer. This is offered for 30 €. Another alternative is the Raspberry Pi Camera version 1.3. This can be purchased for 15 € and has a fixed lens with a focal length of 3.6 mm. For test purposes and for the determination of the optimal system components both the Arducam camera body with the two mentioned lenses and a lens with a focal length of 2.8 mm as well as the standard camera module and the module with super wide angle lens were procured. The use of an industrial camera is excluded due to their high costs and the cost objective of this work, furthermore the use of webcams is excluded due to their disadvantages mentioned in chapter 3.2.1.2.2 for this application.

3.2.3 Marker Comparison and Selection

One of the most important aspects is the selection of an appropriate marker, as the marker enables the unambiguous identification of a localization object, as well as the determination of its position and orientation. The use of a system relying on coded reference markers seems to be the best solution, as these can be created very inexpensively and with low effort the user. Furthermore, it is advantageous that a large number of markers and software solutions for their recognition already exist. Thus, optical identification systems, which use markers, could prevail both in industrial use and in logistics and distribution. There they represent the link between information and material flow. Optical identification systems generally have a similar structure and consist of a data carrier (marker), a reading unit and an evaluation unit, which evaluates the recorded data, processes it and forwards it to a higher-level information system. Table 23 gives an overview of the markers used in industry. (Arnold *et al.*, 2008)

Table 23 Marker-types in industrial application

ID technology	**Optical Identification**		
Sensor principle	optoelectronic		
Data medium	Feature	1D-Codes	2D- Codes 3D-Codes Fiducials

Compared to the other identification methods described in Chapter 2.7.1, marker identification is simpler to perform and enables real-time position determination. In addition, the recognition algorithms are more robust and less susceptible to degraded image quality caused by image noise, for example, which poses a challenge to the feature recognition algorithms.

Another big advantage is the easy differentiation regardless of shape, colour, size or weight of an object. (Mulloni *et al.*, 2009; Mulloni, Seichter and Schmalstieg, 2011)

The choice of the optimal marker is an important component for marker identification. In principle, a distinction can be made between two different techniques. Feature-based markers (which are based on the image template matching) represent one alternative and code-based markers (which are based on the image code decoding) the other. The feature-based approach uses template markers, while code-based markers use decoding. (Rencheng *et al.*, 2011) Usually, both marker types are monochrome.

Template markers are usually black and white and consist of a simple image surrounded by a black border. Usually the recognition is done by matching them with known marker templates. Template markers represent an image of the marker to be recognized, which is stored in a database. The captured scene is then compared with this database. However, the matching process often poses a challenge because the size, position and orientation of the detected marker can lead to matching errors or the detection of another marker. (Tikanmäki and Röning, 2011)

Figure 18 Examples for template markers

With an increasing number of objects to be localized, the number of template markers in the database also increases, so the time required for marker identification increases with the number of objects to be localized.

Code-based markers are for example QR codes, fiducials, bar codes, etc. The simplest code-based markers are one-dimensional barcodes. These consist of parallel black bars built on a bright background, the coding of information takes place by adjusting the bar widths or the gaps between them. (Heinrich, 2017)

1D-Codes, the best known 1D codes are Code 128, Code 2/5 and Code 39. For example, Code 2/5 is only capable of storing numeric characters. Since in many applications a 1D code is used to store the delivery note number, package number, part number, tracking number and batch number, it has therefore been become widely used. Code 39 can store a total of 43 alphanumeric characters, but is rarely used due to its size. Code 128, on the other hand, can store up to 128 information characters. (Arnold *et al.*, 2008)

In order to select the correct 1D code, the application requirements must be recorded and weighed against each other; the decisive factor here is, for example (Heinrich, 2017):

- available space in the object and the information quantity
- shape of the object to which the code is to be attached.
- environmental influences such as damage and contamination
- existing reader
- material of the object

Advantages of 1D codes are:

- high standardization and global acceptance
- simple integration with high cost efficiency
- unique identification of objects

Disadvantages of the 1D codes are:

- low resistance to contamination
- necessity of direct visual contact and short reading distance

2D codes can be differentiated according to FIA-2010 into barcodes (or data markers) and reference markers (or ID markers / fiducial markers). The first 2D codes represent the barcodes developed in the late 1980s. In 2010, 70 different 2D codes already existed, which differ in their structure and storage capacity as well as their field of application. (Knuchel *et al.*, 2011)

Two-dimensional codes are differentiated between matrix codes, stack codes and composite codes (Arnold *et al.*, 2008). As the name suggests, 2D codes are two-dimensional in structure, which enables them to encode information vertically as well as horizontally, resulting in higher storage capacity and increasing fault tolerance (Adams, 2009).

- **Matrix codes** are characterized by polygonically, predominantly quadratic arranged data fields as well as by an existing orientation symbol. The coding of an information takes place by the use of set and not set data cells. Here the elements of the matrix code can consist of different geometrical forms, whereby the size of the code is variable. For a better readability of the codes they have a rest zone, which consists of a non-coded frame. A special feature of the codes is their error correction, which even allows the recognition of damaged codes (error corrections of up to 50%). (Overmeyer, 2014)

Figure 19 Examples of different Barcodes: Left: matrix code (Data Matrix code) / Centre: stacked barcode (PDF 417) / Right composite code (GS1 128)

- **Stacked barcodes** are also referred to as multi-line bar codes because they are based on the basic structure of one-dimensional bar codes. They consist of the arrangement of several barcodes in a horizontal direction, with the individual barcodes usually having a common start and stop character. These are used when the code length of a single bar code is no longer sufficient. (Overmeyer, 2014) Arnold et al. (2008) notes that a higher demand is placed on readers, which makes the use of more cost-intensive hardware necessary.

- **Composite codes** represent a new form of code structure. They consist of a combination of 1D code and 2D code (batch code) and are intended to complement the EAN system with a new, flexible form of 2D codes. (Arnold *et al.*, 2008)
- **Fiducial markers** a special form of 2D codes, which was specially developed for use in AR applications, are the reference markers (fiducial markers). While data markers are normally operated by a human reader, reference markers have been developed for automatic recognition and localization. They are able to guarantee this even with low resolution, poor illumination or suboptimal alignment of the marker. For this they have a lower storage rate than data markers (of up to 12 bits), but are also not designed for storing large amounts of information, but for determining the position of the camera. (Olson, 2011)

Among the most famous markers of this class are the ARTAg, the ArUco markers, the CALTAG and the AprilTag. These have a similar structure to matrix codes, but consist of a smaller matrix, because they do not require to store large amounts of data, but to offer a high degree of recognisability. Figure 20, for example, shows such a marker compared to a QR code.

Figure 20 Examples of different Markers: Left: AprilTag / Centre: QR-Code / Right Circular Data Matrix

As one can see, both markers consist of a matrix of black and white squares, which can be decoded by a corresponding software. The big advantage of the ID markers is their smaller matrix, which allows them to be present the square in bigger size, which in turn leads to better readability. Another alternative of the ID markers are the circular markers, but the square markers have prevailed over the circular markers in the frequency of use, as these are easier to handle (Siltanen, 2012).

A wide selection of these markers and the relevant literature on them is listed in Table 24.

Table 24 Examples for existing markers

Marker/ Tag	Reference
Hoffmann Marker System (HOM) IGD Marker SCR Marker	(Zhang, Fronz and Navab, 2002)
ARToolKit Plus Marker	(Fiala, 2005)
ReacTVision Marker	(Bencina, Kaltenbrunner and Jorda, 2005)
Circular Data Matrix Marker	(Naimark and Foxlin, 2002)
Projective Invariant Marker (PI-Tag) - 15	(Bergamasco, Albarelli and Torsello, 2013)
RuneTag	(Bergamasco *et al.*, 2011)
BlurTag	(Reuter, Seidel and Ihrke, 2012)
Cantag	(Rice, Beresford and Harle, 2006)
Furier Tag	(Sattar *et al.*, 2007)
d-touch – 8	(Costanza and Robinson, 2003)
ChromaTags	(Walters, 2015)

From the literature considered in Table 24 it can be concluded that the ARTag, ArUco and AprilTag libraries are among the most widely used. These seem particularly suitable for use and will be examined in more detail in the following section.

ARTag

The development of the ARTag was inspired by the ARToolKit and should offer a higher reliability and light sensitivity by using a more complex algorithm. Like the ARToolKit, it is based on an open source library and allows the determination of the orientation and position of a camera relative to detected markers. ARTag provided users with individual square markers in 2002. These consist of a black or white frame, with a 6x6 matrix inside. This matrix again consists of black and white cells, each of which represents one of 36 bit values between 1 and 0. The first 10 bits contain the marker ID, while the remaining 26 bits are used for error detection and alignment.

A disadvantage is the high computational effort for decoding the tags.

ArUco

ArUco markers are a form of matrix codes developed by the AVA Group of the University of Cordoba (Garrido-Jurado *et al.*, 2014).

They consist of a black border, inside which there is a binary matrix of black and white squares (bits). These serve for the unique identification of the markers, black squares encode a 0 and white squares a 1.

This binary matrix can be decoded using the ARUco library, which is cross-platform and open-soruce. Since the ArUco library does not provide users with a predefined set of markers, they can define the number of markers they want themselves. However, it should be noted that this also means that the performance of the decoding algorithm strongly depends on the number of generated markers.

ArUco has also been already integrated as a module in the Open CV library (Garrido-Jurado *et al.*, 2014). The library enables stable detection of markers of different sizes and distances in changing lighting conditions and backgrounds. (La Delfa *et al.*, 2016)

AprilTag

The AprilTag library was developed by Edwin Olson and can be used for a variety of tasks. It enables camera calibration and is suitable for robotics and augmented reality applications. It is used to determine the exact position, orientation and identification of a marker relative to a camera. The tags are also similar to matrix codes. As already mentioned, fiducial markers are designed for coding less data, in this case between 4 and 36 bits. This enables fast recognition and precise positioning of the markers. (Sagitov, Shabalina and Sabirova *et al.*, 2017)

Marker recognition consists of several steps:

1. searching for linear segments,
2. detecting squares,
3. calculating the position and orientation of the tag,
4. decoding the barcode.

The applied proceed is quite similar to that of the ARTag library and is superior to the ArUco library according to (Sagitov, Shabalina and Lavrenov *et al.*, 2017).

The AprilTag library seems to be particularly suitable for the application in the context of this work, since this was developed for the position determination of the markers in relation to the camera position and such a determination is pursued also here. In addition, it is based on an open source platform whose source code is available free of charge and can be extended if necessary. The markers are easy to produce. AprilTags are also able to deliver good results in low light conditions.

3.2.4 Outcome of Hardware Selection

The Raspberry Pi B 3+ as well as the selected camera systems are used as hardware. In addition, the Raspberry Pi also requires a storage medium, a power supply and a case. For optimal installation, a separate case was developed and manufactured. The housing consists of three Plexiglas plates, 12 spacers and 16 screws. It has been designed so that the camera module can be changed as easily and quickly as possible. Since concrete prices represent only a snapshot, the amounts are listed in Table 25 rounded up. This results in costs of less than 100 € for a subsystem.

Table 25 Components of the prototype and their costs

Module	Modell	Cost [€]
Computational unit	Raspberry Pi 3 B+	34
Camera	Raspberry Pi Camera	15
Lens	-	15
SD Card	Speicherkarte 32 GB	10
Power supply	USB-Ladegerät, 5 V, 2500 mA	12
Case	Einfaches Gehäuse	5
Mounting	2x Schrauben	1
Infrastructure	AprilTag	0
Total		**92**

The selected components basically meet the requirements of low cost, only the use of lenses with a focal length greater than 6 mm and a working distance of less than 3m would result in costs of more than 7.8 €/m². A uniform working distance could not be determined because it does not exist in the industry. However, the assumption of a working distance of 3m seems realistic as a reference. However, this does not represent a restriction but should only be a reference point. In general, however, heights between 6 and 12 m are common in industrial environments. (Hompel, Schmidt and Dregger, 2018; Martin, 2016; Pawellek, 2014)

3.2.5 Outcome of Software Selection

With the selection of the marker in Chapter 3.2.3, a decision was also made for the library for decoding and position determination. This is the AprilTag library.

This is implemented together with OpenCV on the Raspberry Pi. The programming language for customizations is Python.

Python is currently one of the most popular programming languages and has a mature ecosystem with easily accessible out-of-the-box packages and many sample programs in image processing. A large number of standard procedures and models are already implemented and available. With Python, rapid prototyping is also possible. (Adler, 2018)

The fact that Python can be learnt quickly and does not require any special training also speaks in favour of its application to SMEs. Python thus offers the optimal framework for project-based use.

Another central component of image processing is the three-dimensional interpretation of images. Based on this, the position of an object in space can be determined and its size measured. In order to realize this task, however, it is necessary that the parameters of the camera system are known. However, since this is often not the case, methods are needed to determine them. These also represent an important topic in image processing, as camera systems do not work ideally and distort images depending on the focal length. (Bradski and Kaehler, 2011) (see Figure 22). Thus the accuracy of a camera system also depends strongly on the calibration accuracy.

These distortions are usually particularly pronounced with low-cost cameras (Bradski and Kaehler, 2011). This is due, for example, to the fact that low-cost components are installed and manufactured in a relatively imprecise and low-cost production process. According to Thrun, Szeliski and Dahlkamp (2015), a low-cost camera, as shown in Figure 21, consists of a low-cost lens, a low-cost CMOS chip, which is imprecisely attached with an inferior adhesive. The left part of the picture shows the real state and the right part the ideal state. The lower part symbolizes the resulting distortion of the captured image. (Thrun, Szeliski and Dahlkamp, 2015)

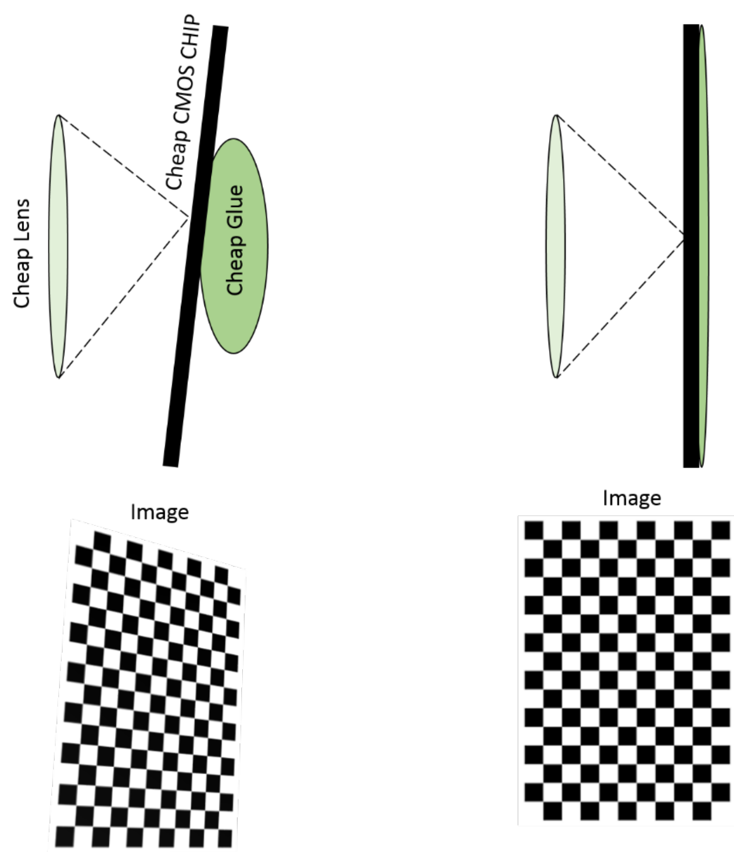


Figure 21 The cheap camera model (left) and its effects to the image

In order to counteract these distortions, the parameters of the lens and the image sensor are determined on the basis of the calibration and taken into account in the position determination.

The camera model according to Bouguet (2015) can be used for the calibration, this is based on the well known pinhole camera model (Zhang, 2000) as well as lens distortion (Heikkila and Silven, no date). The pinhole camera model describes the mathematical relationship between the three-dimensional world coordinate system and its projection onto the image plane. However, this cannot take lens distortion into account since the ideal pinhole camera model does not have a lens. However, since this is not the case in reality, this lens distortion must be considered separately.

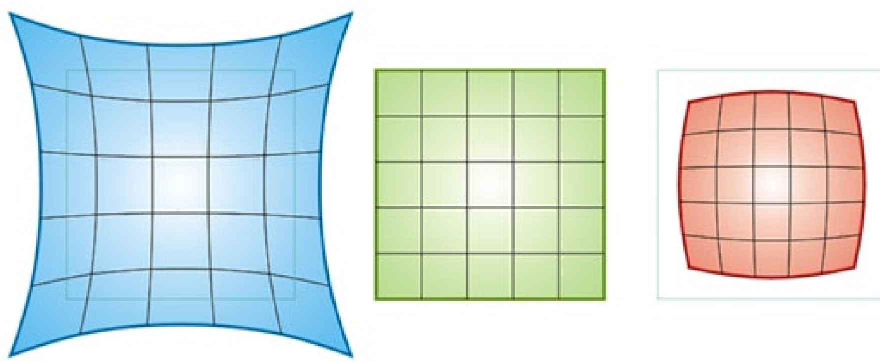


Figure 22 Different kind of image distortion

The parameters of the pinhole camera are displayed in a 4x3 matrix, which is called camera matrix. In this matrix the 3D world coordinates are displayed on the image plane (the image sensor). For this the extrinsic parameters, which are the coordinates of the camera in the 3D world coordinate system and the intrinsic parameters, which are the optical centre and the focal length of the camera, are needed (see Figure 23). These coordinates are then converted by the calibration algorithm into the coordinates of the camera matrix.

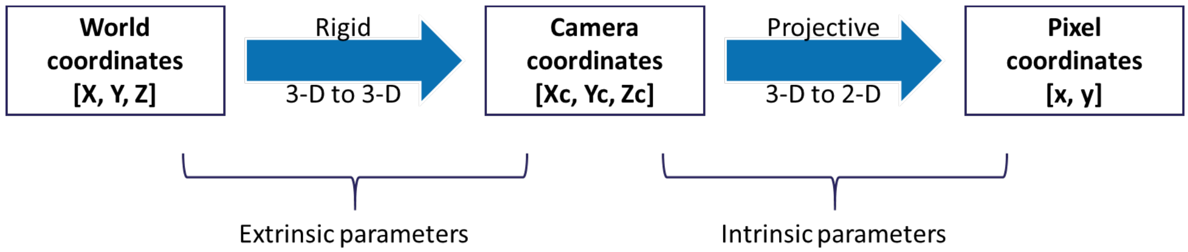


Figure 23 Camera calibration process (de.mathworks.com, no date)

Accordingly, a good calibration of the optical sensor is absolutely necessary for position determination. However, this calibration does not represent an additional requirement because this problem has already been extensively investigated (Albarelli, Rodolà and Torsello, 2010; Higuchi, Datta and Kanade, 2018)gated and therefore calibration models are already able to take into account the distortions that occur increasingly in practice with low-cost cameras. (Datta, Kim and Kanade, 2009; Heikkila and Silven, no date)

The software-based calibration process of a 2D camera is described by Bradsik and Kaehler (2011) in detail. Comparable calibration processes can be found in the OpenCV API documentation (2019) and in Datta, Kim and Kanade (2009).

Usually, calibration boards with known patterns and dimensions are used, such as a checkerboard matrix (see Figure 24). In some special applications, where accuracy plays a very important role, high-precision, expensive calibration standards are usually used. (Demant, Streicher-Abel and Springhoff, 2011), while printed checkerboard patterns are normally sufficient for lower requirements.

Besides the classical calibration with checkerboard patterns, there are also approaches which make use of ring calibration patterns (Albarelli, Rodolà and Torsello, 2010; Higuchi, Datta and Kanade, 2018). There is also the possibility of automatic calibration (Armstrong, 1996), but this requires a more powerful computing unit. Since the use of checkerboard patterns seems to be a widespread method and offers detailed documentation for implementation in OpenCV (docs.opencv.org, 2015), they shall also be used in this work.

3.3 Hardware Setup

As mentioned in chapter 3.2.4, the prototype consists of a processing unit, camera, lens, SD card, power supply, case, bracket and infrastructure (markers). The case is manufactured using a Trotec laser engraving machine and designed using Autodesk Fusion. Only the dimensions of the Raspberry Pi, the camera modules and the screws were needed (see Appendix F). In addition, the necessary libraries must still be installed on the Raspberry. The necessary installation and integration of the libraries is described below.

3.3.1 Software Integration and Optimization

The first step is to download the Raspbian operating system and install it on the memory card. To do this, the Raspbian image from the official website of the Raspberry Pi Foundation has been downloaded. To install the operating system, an additional tool is required which can create a bootable medium from the ISO file. In this thesis the "Win32 Disk Imager" and a Micro-SD card were used.

The following additional peripherals are required to start the Raspberry Pi:

- 5 Volt plug-in power supply + Micro-USB cable
- Network infrastructure (WLAN or Ethernet)
- optional: keyboard + mouse
- PiCam or USB camera

After switching on, the operating system is configured and the basic settings are adjusted. The required programs and libraries are then installed.

There are a number of open source and commercial solutions for image processing. OpenCV is most widespread in the free area. This is also very suitable, since libraries for 2D code recognition are already available. In addition, the selected cameras can be integrated relatively easily. OpenCV itself is programmed in C++, but offers APIs for several other programming languages.

OpenCV was installed according to the Abbas (2016) approach. OpenCV is compiled for the Raspberry Pi which takes some hours due to the low hardware performance.

To use OpenCV functions, AprilTag detection and MongoDB in a Python program, the following Python libraries are required:

Table 26 Necessary libraries

Programming library	Function
cv2	OpenCV-Python is a library which is used to solve computer vision problems with Python
imultis	Imultis is a package of convenience functions to make basic image processing functions such as translations, rotation, resizing, detection edges and much more easier with OpenCV
apriltag	AprilTag is a library which can be used to detect AprilTags, it contains a set functions and allows to determine the position of an tag
pymongo	PyMongo is a Python distribution for MongoDB and is the recommended way to work with this kind of databases in Python.
numpy	Numpy is the fundamental package for scientific computing with Python. It contains a lot of powerful functions.

An important software component is the camera calibration. For this a code based on the program code of Šmíd (2015) is developed, which enables the determination of the camera intrinsic parameters and image undistortion. For this purpose the focal lenght, the principal point and the radial disortion coefficients are calculated using a video analysis. First, two independent programs for video recording and calibration are created, which are later transferred to the main program. The basic procedure consists of the following steps:

1. Start video recording (recording time less than 30 seconds).
2. Move a printed chessboard matrix in the field of view of the camera.
3. End of video recording.
4. Calibration program calculates the disortion coefficients.

When calibrating, it should be noted that the disortion coefficients do not change when the resolution changes, but the camera matrix does. For this reason, the camera matrix must be scaled up or down according to the resolution.

3.3.2 Software Verification

For the verification of the software function, the hardware is put into operation and the code is tested for its basic function. The individual components are tested separately to find errors more quickly.

3.3.2.1 Video recording

To record a video the program video_cap.py is created. To make the program flexible, the following transfer parameters are kept variable:

- Recording time
- Resolution
- Images per second

3.3.2.2 Camera calibration

For the calibration a video must first be recorded via the program video_cap.py showing the chessboard matrix moving (see Figure 24).

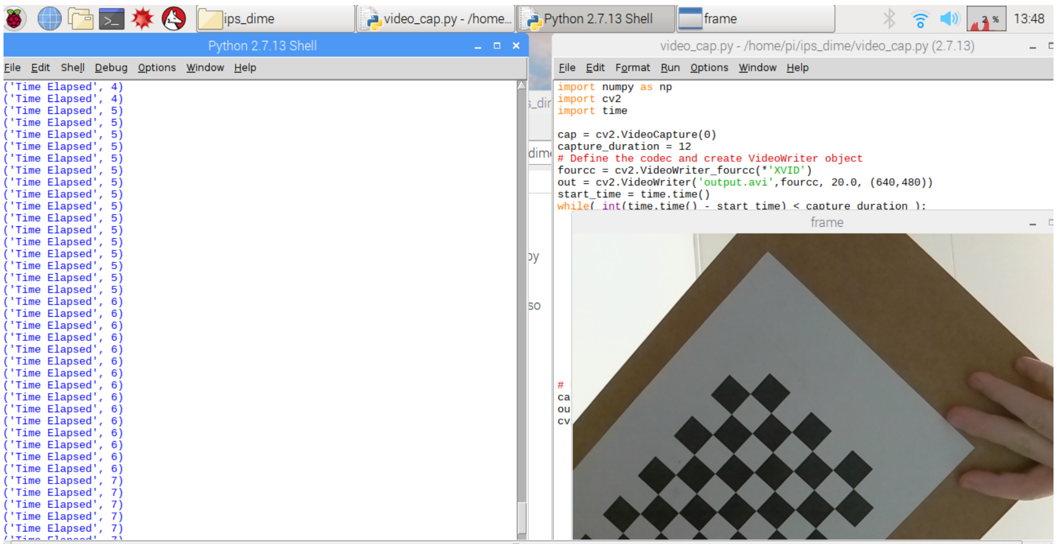


Figure 24 Example of chessboard matrix calibration within the video capturing process

After the video has been recorded the program calibrate.py can be started from the console. This starts the video interpretation (see Figure 25) and the calculation of the camera matrix (see Figure 26). When the calculation of the camera matrix is finished, the results are saved in the file "calibration_parametrs.yml".

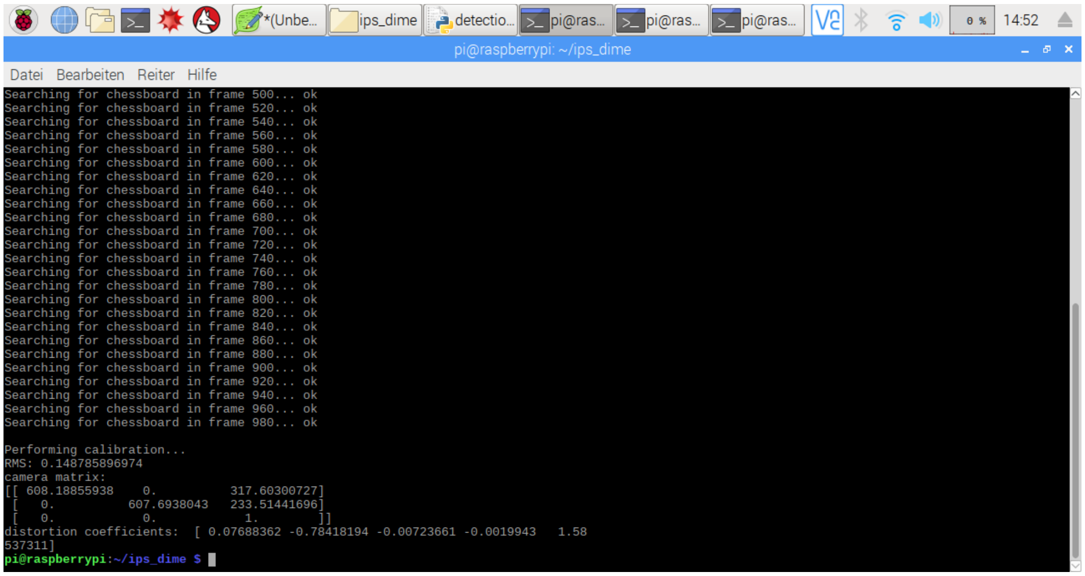


Figure 25 Example of the calculation of the calibration parameters

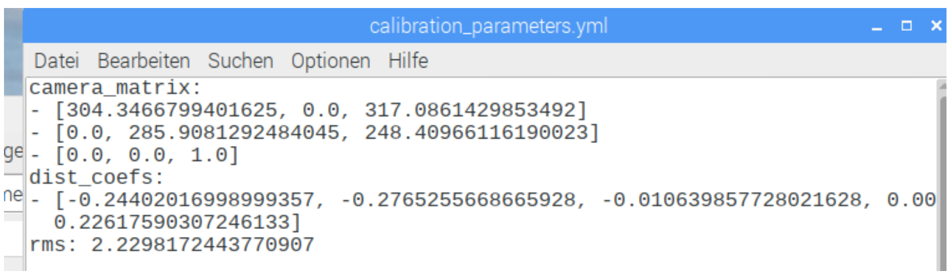


Figure 26 Example of calibration parameters

The camera matrix is loaded by the main program detection_apriltag.py during its execution, in order to be able to carry out the image optimization.

3.3.2.3 Marker Recognition and Detection

A new video is recorded in which an AprilTag is moved. The program detection_apriltag.py can now be called from the console. If there is now an AprilTag in the field of view of the camera system, its coordinates are determined and displayed in the console. Thus a first Proof of Concept could be accomplished. In the following chapter, the verification and optimization of the program is described and the methodology for determining the measurement accuracy of the camera system is presented. Furthermore, a database is installed and integrated, in which the measured values are stored.

Chapter 4

Verification of the Optical based Indoor Positioning System

4.1 Methodological Verification Approach

Two test setups were used for verification. Test setup 1 was developed in order to mount the camera system at a fixed height, and so make it possible to easily intervene in the system. In test setup 1, the camera is installed at a height of 1m, and markers can then be aligned on the base plate to determine its position. The markers are arranged in a matrix, each row of this matrix has a size of 10x10 cm. This makes it easy to determine the absolute positions and then compare them with the measured positions. This setup was used to check the properties of the cameras and compare the achieved accuracies. In addition, different resolutions and the influence of the illumination on the position values were determined. The code is structured as shown in Figure 27 and Figure 28.

Before starting the program, the user must determine whether a calibration is necessary or whether it has already been performed. If this has already been carried out, the determined and transferred parameters can be initialized directly. After a short waiting time, the video stream also starts directly (during this waiting time, the camera automatically adjusts some parameters such as the exposure time). Now the algorithm analyses frame by frame. For this purpose, the captured images are converted into grayscale images before the AprilTag detection algorithm examines them for existing markers. If a marker is found, its coordinates are determined, if there is no marker in the frame, the next one is analysed directly.

Once the coordinates have been determined, they are compared with the data in the database. If the position of the marker has not changed by at least the value defined in the threshold, the position of the marker is regarded as unchanged and the data in the database is not overwritten. The user has at any time the possibility to stop the program via the console; if this does not happen, the program can run endlessly.

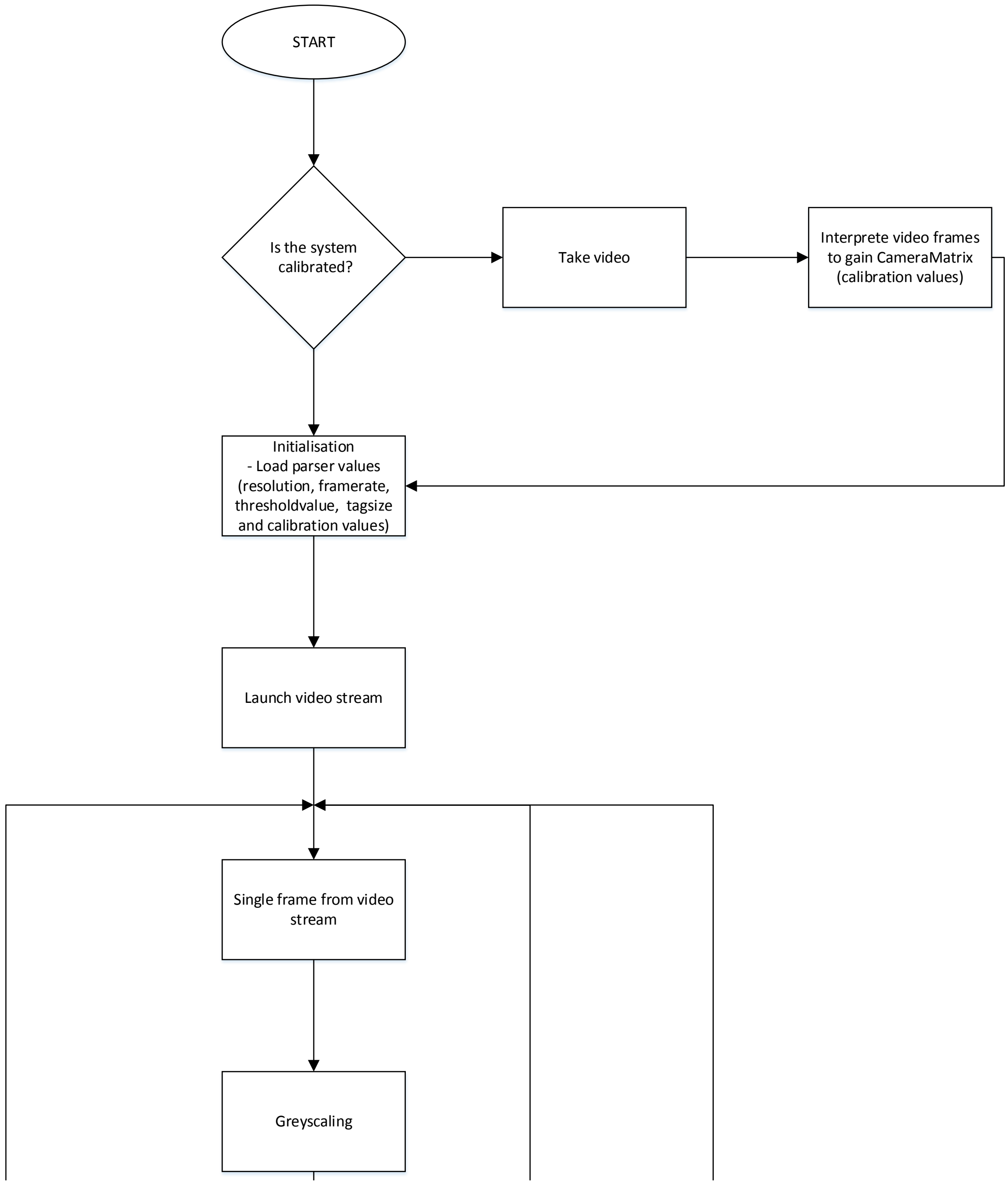


Figure 27 Flowchart of the program-code (1/2)

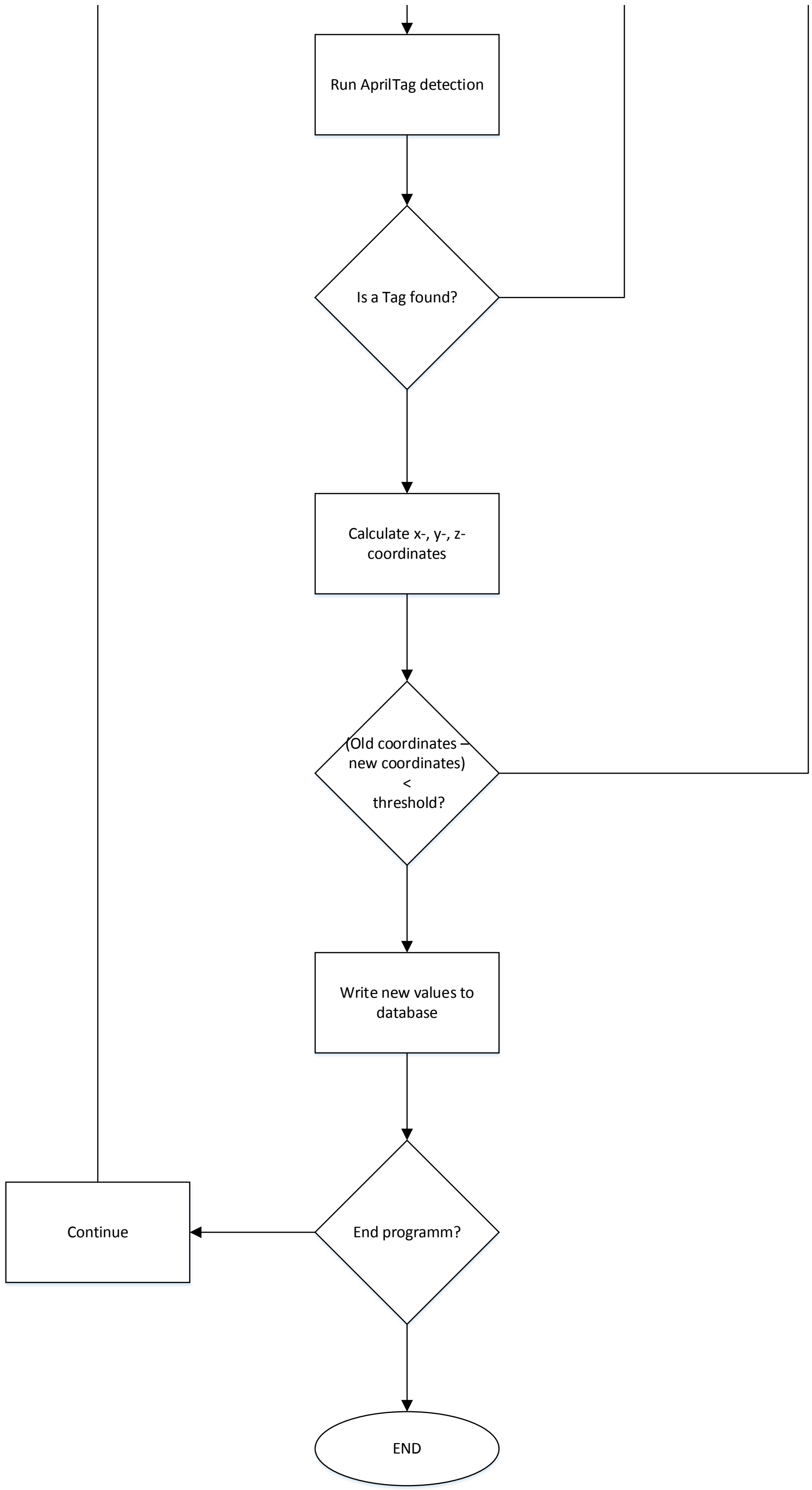


Figure 28 Flowchart of the program-code (2/2)

The first step in the verification process was to integrate the centre of the camera system into the program code. For this purpose, the following line was included in the main code.

```
cv2.line(overlay,(0, self.screenHeight//2), (self.screenWidth, self.screenHeight//2),(0,0,255),1)
cv2.line(overlay,(self.screenWidth//2, 0), (self.screenWidth//2, self.screenHeight),(0,0,255),1)
```

The idea was to halve the resolution in x- and y-direction to get the centre of the scene and based on it to display a crosshair, which can be used for the alignment of the markers. However, this proved to be incorrect, as the camera zero point could not be determined in this way.

After some research it could be determined that it is possible to determine the camera zero point differently. The following lines of code have been added to the AprilTag library for this purpose

```
for i, j in edges:
    cv2.line(overlay, ipoints[i], ipoints[j], (0, 255, 0), 1, 16)
cv2.putText(img = overlay, text = "(X:{:.4f}|Y:{:.4f}|Z:{:.4f})".format(tvec[0],tvec[1],tvec[2]), org= ipoints[0],fontFace = cv2.FONT_HERSHEY_DUPLEX,fontScale=0.6,color=(0,255,255))
cv2.putText(img = overlay, text = "ID: {}".format(tag_id), org= ipoints[2],fontFace = cv2.FONT_HERSHEY_DUPLEX,fontScale=0.5,color=(255,255,0))
```

This determines the centre of the image and draws a crosshair based on it. The crosshair is then faded in again during scene viewing and can therefore be used to align the markers.

Next, video recording and camera calibration were implemented in the main code.

Then a parser was integrated, which allows to change the transfer parameters during the execution of the main program via the console. Parameters for image resolution, frame rate, threshold, marker size as well as for video recording and camera calibration were integrated.

The functionality of the individual extensions was verified using Test Setup 1 by comparing the measurement results with the absolute values using the console's output data.

In the next step, the connection to a database was implemented. The NoSQL database MongoDB was used, because the document structure is easily changeable, which simplifies the development process.

The variables for database access were defined in lines 28 to 31 of the program, the database was accessed in lines 77 to 86, and the measurement results were written to the database in lines 166 to 182. This caused the problem that there was no comparison with the data in the database at this time. The threshold was set up for this purpose. This ensures that a new value is written to the database if the marker has shifted by the value stored in the threshold. The Device_number is stored in the database so that several system components can write to the same database at the same time without overwriting each other's values. The id of the marker, a timestamp, the time duration, device ID, resolution as well as the x-, y-, z-coordinates of the marker are written in the databank.

In general, the code is structured to allow endless viewing of the scene. The code can be found in Appendix A. Before executing the code, the camera driver should be loaded from the console using the command "sudo modprobe bcm2835-v4l2".

The console is opened via the icon marked in red (see Figure 29), first you have to klick on the icon in the red circle, the console will then open (marked in the red rectangular field).

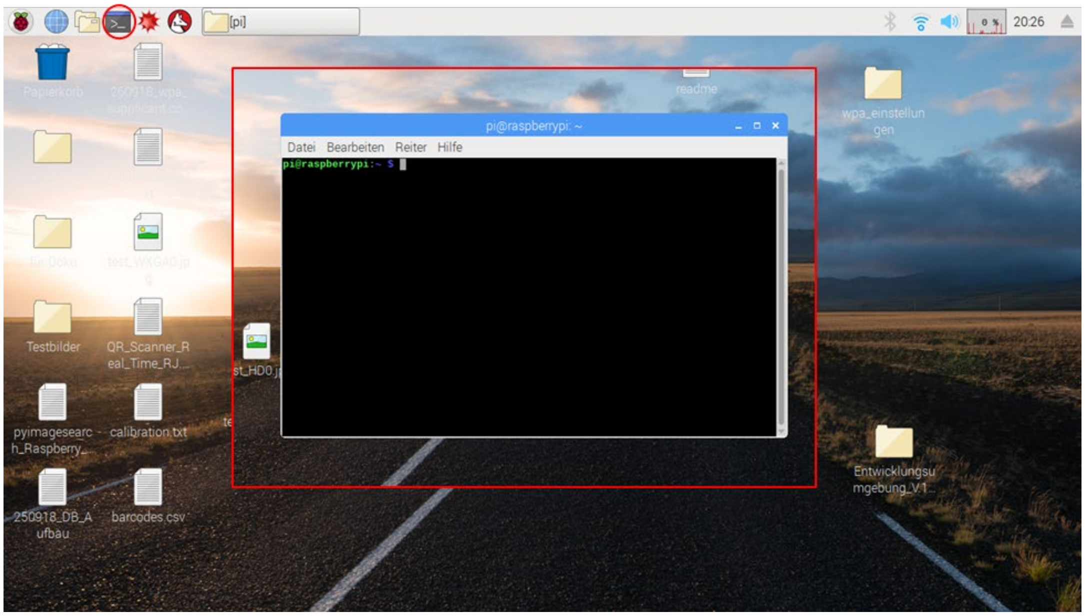


Figure 29 How to open the console

To execute the code, the camera driver must first be loaded; this is done with the command "sudo modprobe bcm2835-v4l2". The next step is to change to the target directory, which in this case is ips_dime. To do this, the command "cd ips_dime". Is to be entered. Now the program code with the command "python detection_apriltag_V.1.2.py -tag_size=0.08 can be executed.

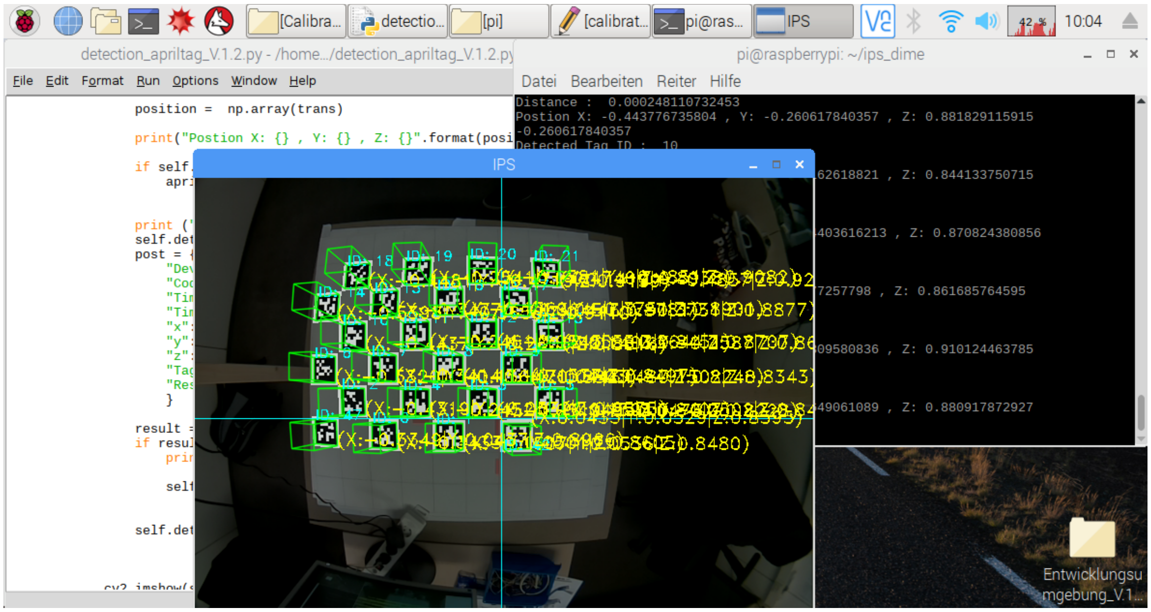


Figure 30 Example of the test setup of the accuracy measurement

Test setup 2 provides that the markers are located on a movable flat object. For this purpose, a movable board with fixable rollers was used. After the camera system was fixed, a distance measurement was made and the distances marked. During validation, the board was then moved back one metre.

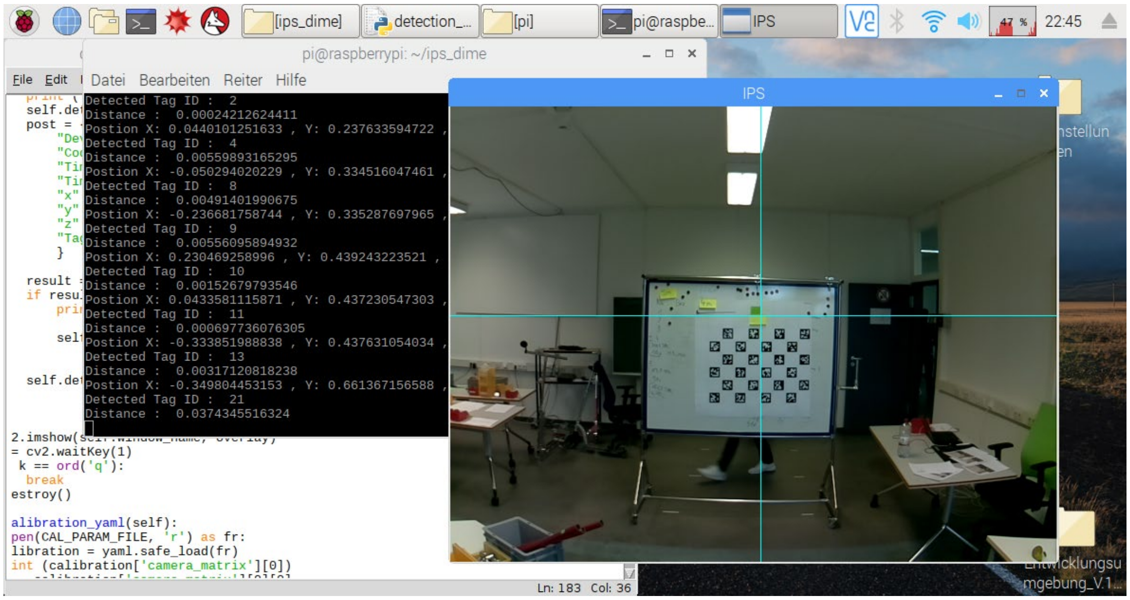


Figure 31 Example of the test setup for distance measurement

4.2 Verification Results

As already mentioned, the markers were executed on paper sheets measuring 90x90 cm for validation purposes. This are made up of 81 lines, each sheet measuring 10x10 cm. A total of 5 different sheets were made. They differ on the one hand in the size of the size and on the other hand in the number of printed markers. All markers are centrally aligned in their cells, thus an optimal position determination is possible, since the library determines the centre of the markers. The markers are aligned under the camera system to its zero point and the position is determined.

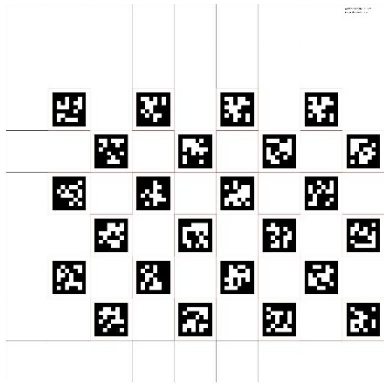


Figure 32 Verification "paper sheet"

4.2.1.1 Accuracy measurement

A total of four different measurement tests were carried out. The first test was to examine the five selected systems and determine basic parameters. These parameters include the FOV and the overall height of the individual systems. In principle, the systems differ only in the lens used. For the determination, the components were installed and then the overall height was measured. This is necessary to determine the working distance (see Figure 33).

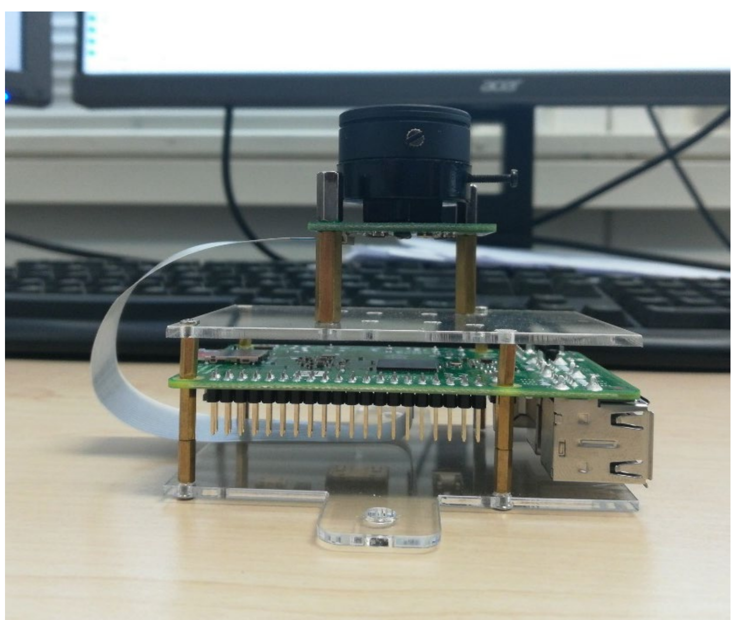


Figure 33 Prototype, with case and f 2,8 mm lens

In order to determine the FOV, the 5 camera systems were mounted in the test setup and a video stream was started. The FOV was measured with the help of a metre rule (see Figure 34 and Figure 35).

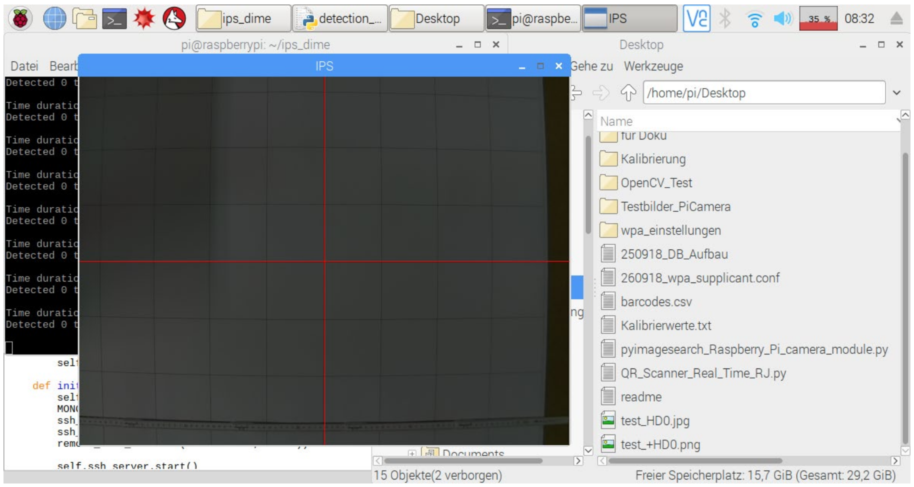


Figure 34 FOV measurement X

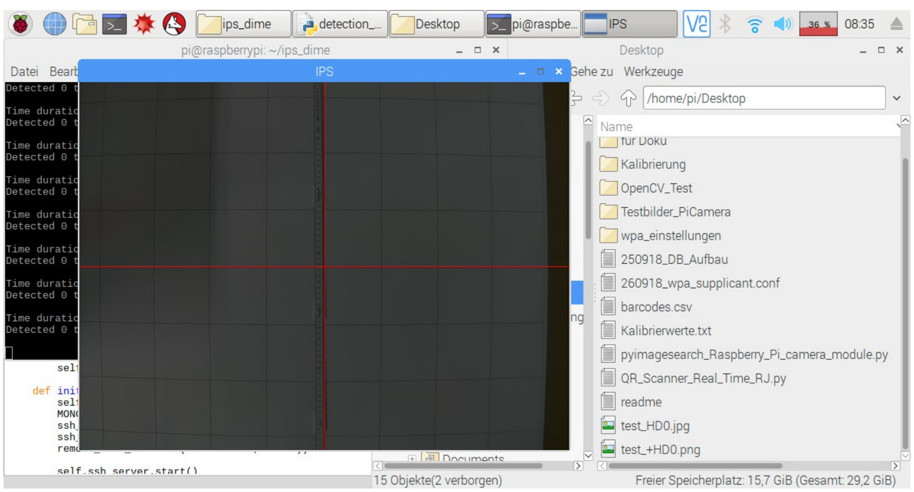


Figure 35 FOV measurement Y

The measurement results are shown in table 27.

Table 27 Comparison between the actual and calculated FOV

Focal length f [cm]	WD [cm]	Actual FOV (V) [cm]	Actual FOV (H) [cm]	Calculate d FOV (V) [cm]	Calculate d FOV (H) [cm]	Deviation (V) [cm]	Deviation (H) [cm]
1.67	91.00	256.00	167.50	232.88	160.85	23.12	6.65
2.80	94.00	132.00	88.00	120.86	90.64	11.14	2.64
3.29	92.52	100.00	73.00	92.52	69.39	7.48	3.61
4.00	92.60	86.50	64.00	83.34	62.51	3.16	1.50
6.00	95.64	55.00	40.00	57.38	43.04	2.38	3.04

It is noticeable that the target values deviate more strongly as the focal length decreases. However, it has already been pointed out that the formulas mentioned in Chapter 3,2.1 cannot exactly calculate the FOV and should rather be seen as an approximation. However, the table shows that these approximations are sufficient for the selection of the lenses, since the deviations between calculated and actual values remain within an acceptable range.

The aim of this work is to optimize the cost of the system while maintaining accuracy of the measurements. The focal lengths between 1.67 mm and 2.8 mm seem to be the best choice, as they can achieve a high coverage range.

Next, it must be clarified which accuracies can be achieved with the different focal lengths and whether the calibration is sufficient to compensate for the increasing distortion with decreasing focal length. For this purpose, the manufactured paper sheets were used. These do not only consist of a different number of markers but also of markers of different sizes.

In the following, the measurement results for the 5 lenses are shown, here the illustrations show the sheets and the positions of the individual markers on these. While the tables show the average deviation in Z as well as the X-Y level and the average latency (update rate).

The latency is between 0.04 seconds and 0.08 seconds, with the program running periodically or until a user abort command is received. Thus, the update rate requirement is considered met

All the following which are marked as Ø, are based on the mean of 10 measurements.

In terms of accuracy, Test Setup1 shows that with all objectives at the given working distance, the deviations are less than less than 1 cm and would therefore meet the accuracy requirement. The detection of several markers is also possible.

Table 28 Evaluation of the different system setups (1/2), (Number = amount of markers)

Focal length	Size x Number	Ø Deviation -Z [m]	Ø Deviation -X [m]	Ø Deviation -Y [m]
f 1,67 mm	10x12	-0.1027	-0.0032	-0.0014
	10x23	-0.1260	0.0016	-0.0005
	6x12	-0.0930	-0.0016	-0.0046
	9x12	-0.1000	-0.0022	-0.0033
f 2,8 mm	10x12	0.0082	-0.0033	-0.0008
	10x23	-0.0088	-0.0038	-0.0006
	6x12	0.0152	-0.0013	-0.0006
	9x12	0.0166	-0.0031	-0.0009
f 3,29 mm	10x12	-0.30222	-0.0006	0.0003
	10x23	-0.30811	-0.0018	-0.0002
	6x12	-0.28806	-0.0014	0.0003
	9x12	-0.29330	-0.0011	0.0001
f 4 mm	10x12	-0.1017	-0.0023	0.0005
	10x23	-0.1096	-0.0027	0.0002
	6x12	-0.0938	-0.0006	-0.0010
	9x12	-0.0953	-0.0008	0.0009
f 6 mm	10x12	0.1558	-0.0108	-0.0012
	10x23	0.1524	0.0010	-0.0005
	6x12	0.1574	0.0010	0.0003
	9x12	0.1569	0.0003	-0.0004

Table 29 Evaluation of the different system setups (2/2)

Focal length	Size x Number	Ø Latency [s]	Ø Deviation X-Y Level [%]
f 1.67 mm	10x12	0.05	2.87
	10x23	0.05	2.23
	6x12	0.04	3.37
	9x12	0.04	3.20
f 2.8 mm	10x12	0.06	0.21
	10x23	0.07	0.28
	6x12	0.05	0.21
	9x12	0.06	0.21
f 3.3 mm	10x12	0.06	0.88
	10x23	0.07	1.18
	6x12	0.05	0.67
	9x12	0.06	0.64
f 4 mm	10x12	0.06	1.10
	10x23	0.08	1.05
	6x12	0.05	0.70
	9x12	0.06	0.49
f 6 mm	10x12	0.06	6.71
	10x23	0.06	0.80
	6x12	0.05	0.54

Table 28 shows that all system configurations can maintain the required accuracy of less than 1 cm at an installation height of 1 m. It is noticeable that the marker size does not seem to have any influence on the accuracy. Only the determination of the Z-position of the working distance seems problematic. For the determination of this, the program calculates the given marker size and the one determined in the scene. In this area there still seems to be optimization potential.

The evaluation also showed that all system configurations are capable of detection in less than 1 s, as required (see Table 29). If the percentage average deviation in the Y-X level is considered, the system configuration with a focal length of 2.8 mm seems particularly advantageous.

A further limitation also results from the FOV of the individual system configurations. For example, the system configurations with the 4 mm and 6 mm lens could not detect all markers in the scene with the same setup. When using the 4 mm lens, 4 of 23 markers were not detected and when using the 6 mm lens, 15 of 23 markers were not detected due to the limited field of view.

4.2.1.2 Marker recognition at different resolutions

The results shown so far were achieved at a resolution of 640x480. However, the camera system is theoretically capable of much higher resolutions. However, the full resolution is not supported for dynamic viewing. Furthermore, it turned out that not every resolution is accepted by the system. After testing some test settings, the resolutions 640x480, 1024x768, 1600x1200 proved to be particularly suitable. The following test results could be achieved.

Table 30 Cohesion between resolution and accuracy

Resolution	Marker ID	Ø Deviation-X [m]	Ø Deviation-Y [m]	Ø Deviation-Z [m]	Ø Latency [s]
640x480	14	0.0094	0.0149	0.0879	0.0501
	18	0.0014	0.0209	0.0854	0.0501
	42	0.0062	0.0034	0.1606	0.0501
	47	0.0152	0.0051	0.1037	0.0501
1024x768	14	0.0073	0.0166	0.0839	0.1185
	18	0.0029	0.0259	0.0762	0.1185
	42	0.0055	0.0029	0.1551	0.1185
	47	0.0128	0.0047	0.0983	0.1185
1600x1200	14	0.0054	0.0177	0.0811	0.2266
	18	0.0041	0.0270	0.0741	0.2266
	42	0.0051	0.0038	0.1500	0.2266
	47	0.0113	0.0045	0.0967	0.2266

The markers with ID 14 and 18 have as large a distance as possible in X and Y direction from the camera zero point, the marker with ID 47 is on the X axis with a maximum distance along the Y axis, and the marker with ID 42 is on both the X and Y axes.

The deviation seems to be systematic, as shown in Figures 36 and 37. It is difficult to say whether the resolution affects the accuracy.

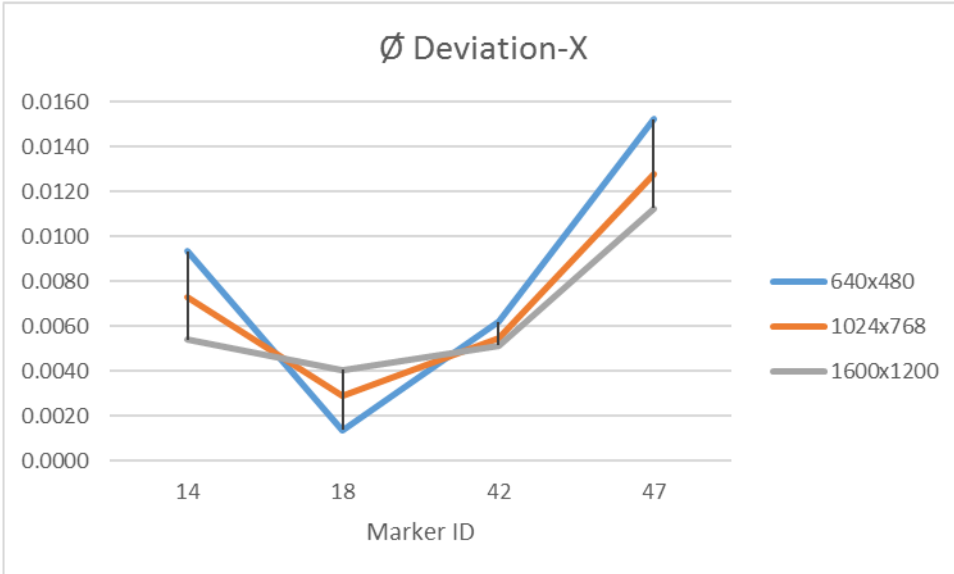


Figure 36 Cohesion between resolution and accuracy on X

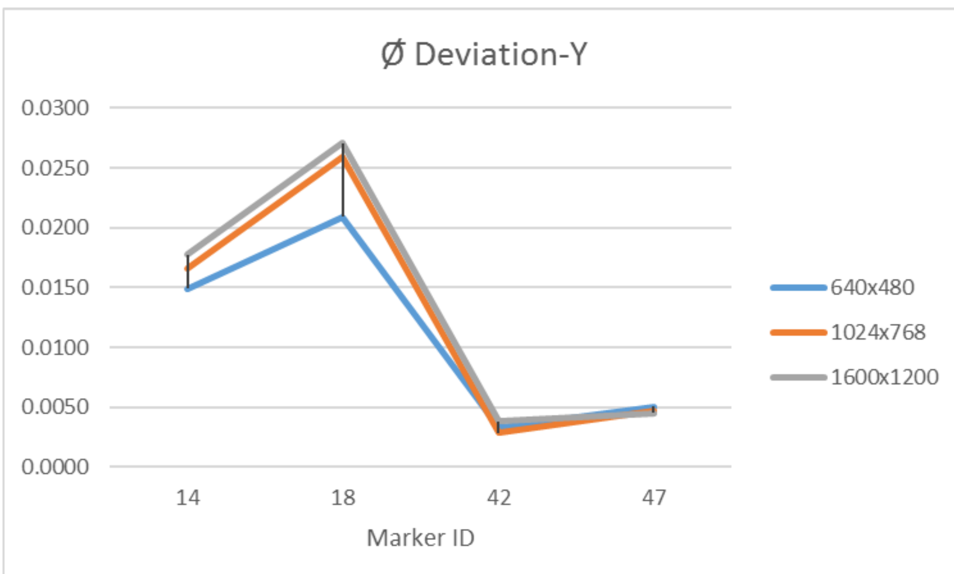


Figure 37 Cohesion between resolution and accuracy on Y

The resolution does not seem to have a big influence on the accuracy as expected, but further experiments should be carried out to prove this correlation. As expected, an increasing resolution has a negative effect on the update rate. It has been shown that the requirement for an update rate of less than 1 second is still met, so using a higher resolution is still an alternative.

4.2.1.3 Marker detection under different lighting conditions

In a further test, the measurement was carried out with active and passive lighting. Active lighting means that the measurement environment is illuminated by the ceiling lighting. In principle, it can be assumed in all the application scenarios mentioned in Chapter 2.7.2 that lighting conditions prevail that enable people to work comfortably and safely. In passive lighting, ceiling lighting was switched off and daylight was used purely to illuminate the scene. As Figure 38 shows, this did not play a decisive role in measuring accuracy. The system can therefore be regarded as robust under various lighting conditions.

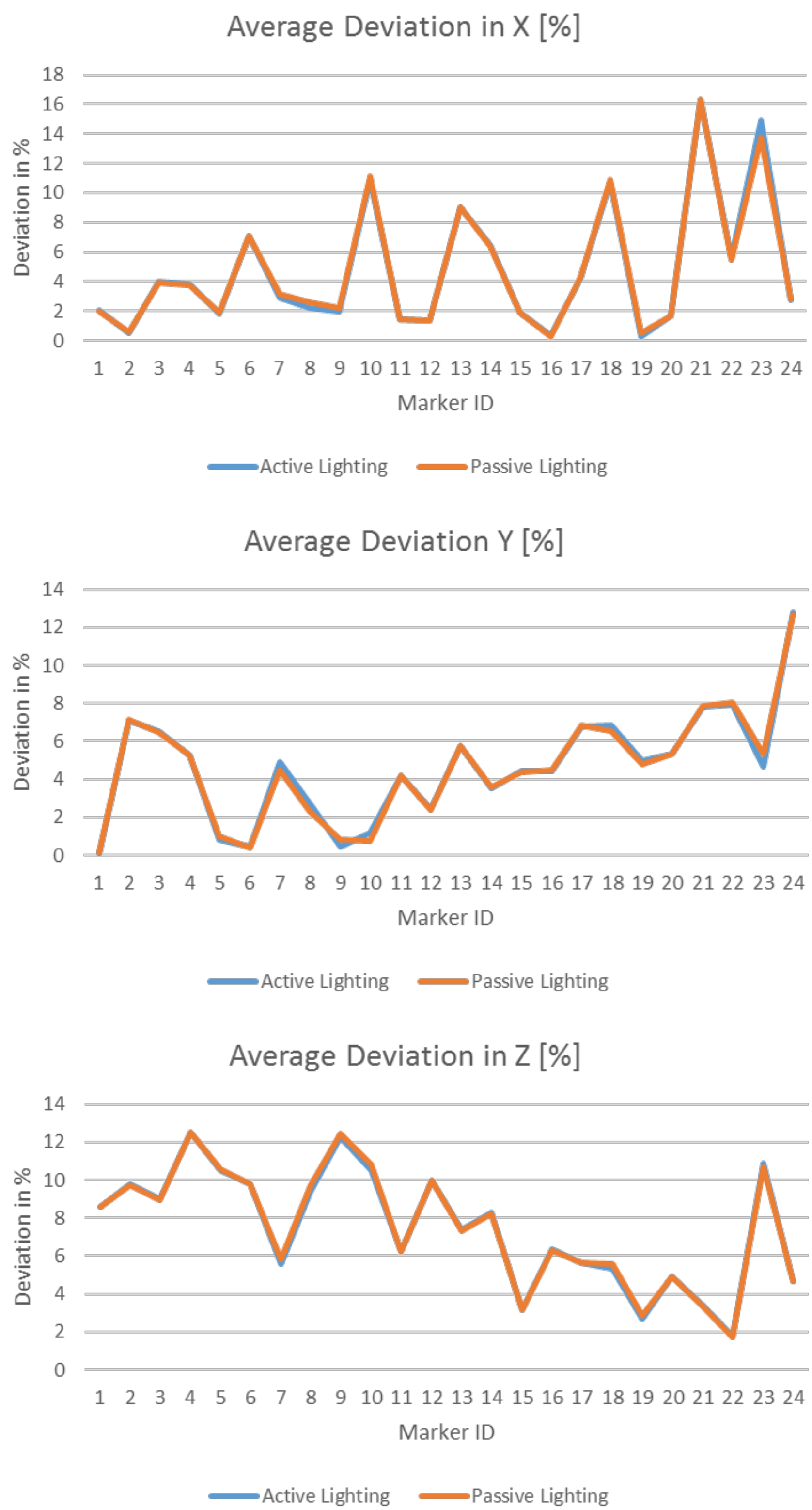


Figure 38 Impact of lighting conditions on the system performance

4.2.1.4 Marker detection at different working distances

Furthermore, the system was tested for its ability to recognize markers at different distances. As already stated, the lenses with focal lengths of 1.67 mm and 2.8 mm were particularly suitable for use. The test results for both focal lengths are shown in Table 31 below.

Table 31 Marker recognition at different distances and marker sizes

AprilTag							
Marker Size [cm]	Distance [cm]	Recognition (f 2.8)			Recognition (f 1.67)		
		Good	Poor	None	Good	Poor	None
10x10	100	x			x		
	200	x				x	
	300	x				x	
	400	x					x
	500		x				
	600			x			
12x12	100	x			x		
	200	x			x		
	300	x			x		
	400	x					x
	500		x				
	600			x			
15x15	100	x			x		
	200	x			x		
	300	x			x		
	400	x					x
	500	x					
	600		x				

The lens with a focal length of 2.8 mm is clearly superior to the lens with a focal length of 1.67 mm and is therefore recommended for system design. However, the question of the achievable accuracy of the varying working distances must still be clarified. Table 32 provides an evaluation of the measurement results for the different tested working distances and marker size

Table 32 Accuracy of the prototype at different distances and marker sizes

Marker Size [cm]	Distance [cm]	Ø Deviation-X [cm]	Ø Deviation-Y [cm]	Ø Deviation-Z [cm]	Ø Deviation X-Y Level [cm]
10	100	0.045336692	0.046715295	1.843024432	0.065224144
	200	0.635197073	0.565362183	7.78772336	0.850369095
	300	0.191244595	0.373599386	14.01016246	0.41979655
	400	0.943364986	1.208518913	22.15851181	1.53312206
	500	0.681118264	0.877805215	5.33794356	1.113721749
12	100	0.095690246	0.032001231	2.988227165	0.101415144
	200	0.573526694	0.516137245	0.51276913	0.772190444
	300	0.029968414	0.299977347	1.386916047	0.30217492
	400	0.888227642	1.129851304	6.490266254	1.437193974
	500	0.678624382	0.807397466	6.911135804	1.057050667
15	100	0.098364848	0.08355415	2.260144694	0.132129402
	200	0.554762599	0.448278523	7.586414197	0.71324729
	300	0.060004823	0.362995722	10.81618035	0.36814771
	400	0.949332726	1.106502734	19.27966114	1.458155493
	500	0.667206771	0.778614787	13.17024097	1.025413604
	600	0.342256271	0.871551439	30.76155959	0.936358462
Ø deviations based on n=10 measurements					

The average deviations of the accuracy vary for the different marker sizes and working distances as expected. Especially the average deviations of the Z- coordinates are quiet high and inconsistent, as they vary between 0.580 cm and 44.162 cm. This is due to the applied algorithm, which uses the marker size as an input parameter and compares it with the size of the marker detected to calculate the distance. The verification also showed that the deviations are not steadily increasing with the working distance. This connection cannot be explained at this point and needs further investigation.

The same connection appeared for the X- and Y- coordinates. For example, the average deviation in X was the smallest for all the marker sizes at a working distance of 3 m. This could be caused by the used lens, which might have its best focus point at a working distance of 3 m. The manufacturer is not providing such information and the used lens enables an infinite focus. At this point, the differences between the deviations, especially in regard to the working distance cannot be further explained and would need more research and testing. In Table 33 the standard deviations of the different working distances and marker sizes are given. There are also some differences between the standard deviations, which need further investigation.

Table 33 Standard deviations for the accuracy measurement at different working distances and marker sizes

Marker Size [cm]	Distance [cm]	SD-X [cm]	SD-Y [cm]	SD-Z [cm]	SD X-Y Level [cm]
10	100	0.0024128	0.0065985	0.0189155	0.0057366
	200	0.0103672	0.0070303	0.2801030	0.0117909
	300	0.0114273	0.0037089	0.1584343	0.0081358
	400	0.0060735	0.0057367	0.1593561	0.0078746
	500	0.0931901	0.0124506	0.8398209	0.0541050
12	100	0.0190043	0.0136329	0.2390526	0.0210402
	200	0.0353690	0.0095239	0.3736353	0.0198524
	300	0.0221008	0.0195181	0.8400634	0.0210768
	400	0.0048466	0.0039997	0.1207311	0.0050281
	500	0.0711215	0.0528022	5.3691555	0.0539624
15	100	0.0321005	0.0371112	0.1370570	0.0400805
	200	0.0033374	0.0008753	0.0551421	0.0023204
	300	0.0147707	0.0178846	1.3936015	0.0192813
	400	0.0388074	0.0441609	1.2278109	0.0531007
	500	0.0061733	0.0065319	0.2438767	0.0036848
	600	0.0068164	0.0102924	1.8749767	0.0112468
Standard Deviation (SD) based on n=10 measurements					

Overall the accuracy for all the different marker sizes and working distances, shows that an accuracy of less than 1 cm cannot be met. Therefore not all the application scenarios presented which have been assigned to category 1 can be covered by the introduced prototype. An accuracy of less than 5 cm can be fulfilled for the X- and Y- coordinates, therefore all the other application scenarios of this category can be fulfilled. This means that 11 out of 12 cases are covered by the prototype.

During the measurement data collection a systematic measurement error could be detected at a working distance of 5 m and a marker size of 15x15 cm. However, the cause for this could not be identified and, in addition, such an error did not occur in any of the other measurements.

In addition to Table 32 and Table 33, for which only used one marker for the measurements, another experiment was conducted using the paper sheets introduced in chapter 4.2.1.1.

The results are presented in Figure 39 and Figure 40, 23 markers with a size of 10x10 cm where placed and their position measured. Table 32 already shows that the deviations of the markers in X and Y position are different.

Figure 39 shows the average deviations per marker in relation to its X- coordinate in the coordinate system

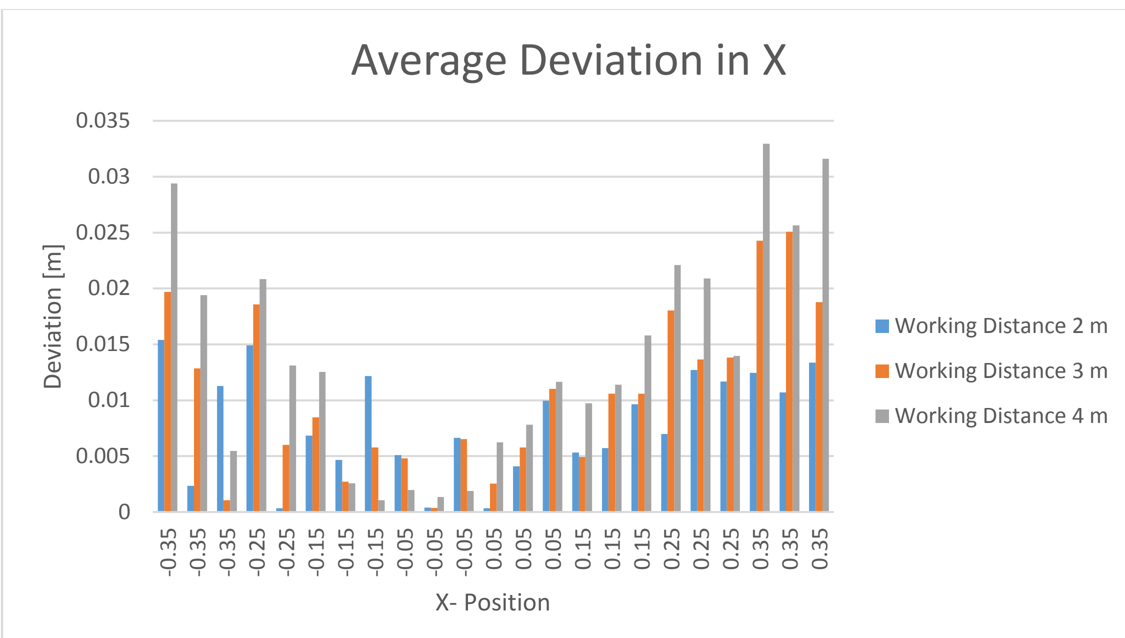


Figure 39 Average deviation in X (marker size 10x10 cm)

Figure 40 shows the average deviations per marker in relation to its Y -coordinate in the coordinate system

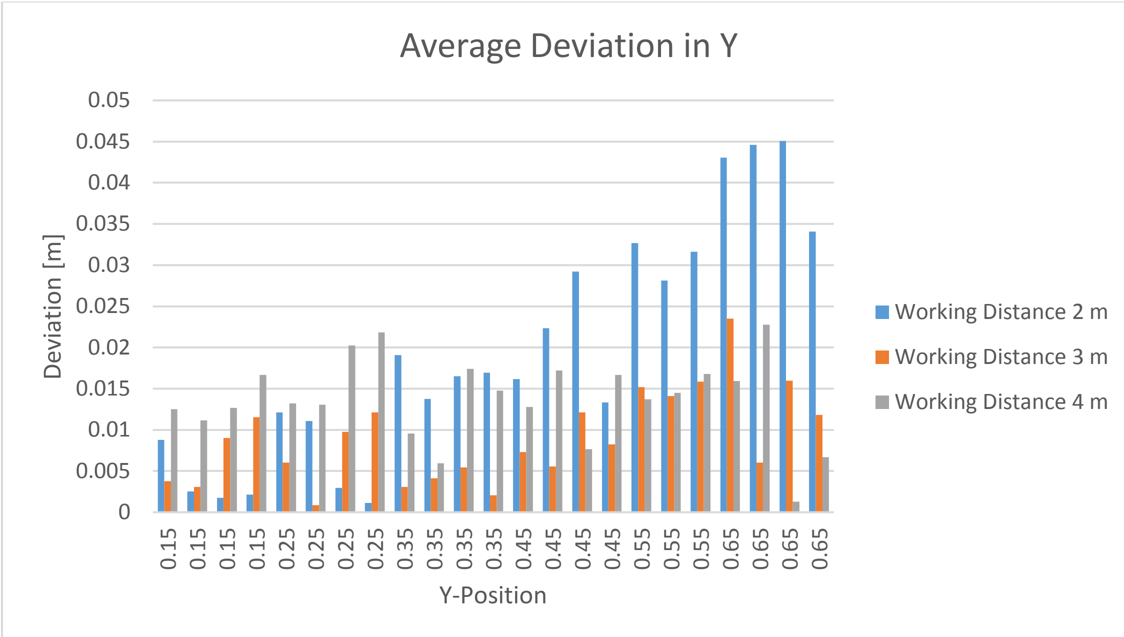


Figure 40 Average deviation in Y (marker size 10x10 cm)

The different distribution of the deviations is due to the fact that the camera zero point was set as highlighted with turquoise lines in Figure 41. Figure 39 and Figure 40 show that there is a tendency to get a higher deviation if the distance between the camera zero point and the absolute position

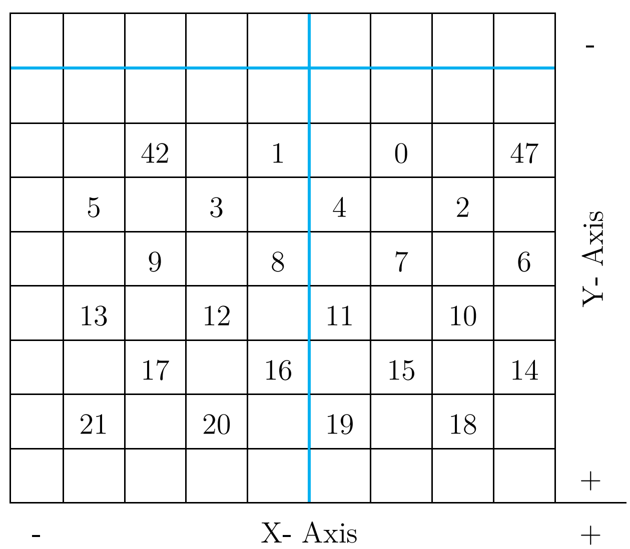


Figure 41 Coordinate system of the distance measurement, with a marker size of 10x10 cm

However this experiment also shows that the average deviations in both directions varies for the different distances and is not linearly increasing. As already discussed, this aspect would need more investigation, which was not able to be conducted in the given amount of time.

In general, determining the Z-position of the markers seems to be the greatest challenge and thus offers the greatest optimization potential.

4.3 Verification Summary

During the verification of the system an optimal code could be developed, which enables the calibration of the camera, a simplified installation of the system by the representation of the camera X- and Y- axes in the live image as well as a simplified operation due to the compact design and possibility to change parameters easily.

With regard to the hardware components, it was possible to verify that the computing unit and the program code met the requirements regarding the update rate. In addition, the developed case makes it easy to manufacture and assemble at low cost.

It could be determined that all possible system configurations can be used with a small working distance. However, a focal length of at least 2.8 mm is recommended as it provides high accuracy with a large field of view. The use of a super wide angle lens does not seem to be recommended, as it is no longer possible to detect markers as the working distance increases. In addition, the calibration of a super wide angle lens is also more challenging.

Furthermore, it was shown that an increasing resolution does not necessarily lead to an increasing accuracy, but to a worsening update rate. With regard to resolution, further testing should be carried out to make a more concrete statement. So it would be recommended to use the resolution in combination with an increasing distance.

By looking at the measurement, results with changing lighting it could be shown that the algorithm is not susceptible to changing lighting conditions. This is due to the camera body, which is able to automatically adjust some of its settings to the environmental influences.

Chapter 5

Summary, Conclusions and Recommendations

The structure of this work is designed in such a way that the most important knowledge bases for the specific problem were created first. Based on these, sound recommendations for the development of a prototype could then be made and implemented. Thus, it was also possible to contribute to the current state of knowledge. In the following chapter the structure of the work as well as the results will be summarized and recommendations for further research will be given.

5.1 Objectives of the thesis

The main research objective of this thesis was to show which possibilities of optical indoor position determination based on low cost technologies are relevant in the industrial environment.

For this purpose possible methods and technologies together with their fields of application and requirements as well as the influencing factors of system components on these requirements should be presented. Subsequently, the aim was to develop a prototype based on the identified requirements. Only components that can meet the low cost requirements should be used.

In order to achieve this goal, the relevant literature was analysed.

Here the relevance and topicality of OIPS could be shown particularly in the upheaval of the 4th industrial revolution. In addition, systems available on the market and systems presented in scientific publications were examined more thoroughly. Based on these sources and the available literature, the necessary main components of OIPS could then be determined and described in more detail.

A large number of applications could also be identified and classified. Based on this classification, a class could then be determined which covers the majority of applications. Furthermore, a prototype could be developed based on the characteristics of this class.

This prototype was then verified and optimized for use in various scenarios. The optimization is characterized on the one hand by the use of optimal hardware components and on the other hand by the development and improvement of a program code.

5.2 Theoretical overview

In the first part of this work, a literature study was carried out, on the basis of which the following points were examined.

- Approaches, structure and set-up as well as system components of OIPS
- Application scenarios and requirements of optical indoor positioning systems

In the second part of this work, a prototype was developed based on the data obtained. This was then verified in the third part of this work using a methodology designed for this purpose and constantly optimised for use during verification.

5.3 Research Methodology

The main objective of the research was examined in more detail in form of a literature review, on the basis of which important conclusions were drawn which are necessary for the construction of a prototype. The research approach is a mixed one, Chapters 2 and 3 follow an inductive approach, while Chapter 4 incorporates the experience gained here in the design of a prototype and thus leads to a deductive approach. The techniques used are the collection and evaluation of relevant literature in the form of reference books, scientific publications and publications by manufacturers in this field. In addition, design science was applied in the implementation and verification of the prototype (Table 34 gives an overview of the approach followed)

Table 34 Summary of the research methodology

Philosophy	Pragmatism
Approach	Inductive (Chapter 2 & 3) & Deductive (Chapter 4)
Strategy	Design science
Technique	▪ Data collection and analysis ▪ Experiments / case study

5.4 Summary of main findings

The following system requirements have been set: The architecture should be based on external positioning, as this facilitates the determination of the position of a large number of objects. Both points were achieved by using the AprilTag library and setting up the localization system. In addition, the software implementation requires a low computing power, so this requirement is also considered fulfilled. It also allows users the optional determination of the orientation of the markers.

The requirement of the accuracy of less than 1 cm can only be achieved at a very short working distance. At working distances between 2 m and 6 m, an accuracy of less than 5 cm could be demonstrated in this work. Nevertheless, an accuracy of less than 1 cm also represents a very high requirement, which cannot be achieved by most systems available on the market presented in Chapter 2.2. However, for many of the identified application scenarios an accuracy of less than 5 cm is sufficient. This requirement can be met by the system presented here.

A coverage of up to 10 m^2 is achieved by the proposed system components from a working distance of 3 m (see Table 35), so this requirement is considered to be fulfilled if one assumes a ceiling height of more than 3 m, which is quite realistic in common industrial buildings.

Table 35 Achievable coverage of the prototype (f 2,8 mm)

WD [m]	Coverage [m²]
1	1.24
2	4.96
3	11.16
4	19.84
5	30.99
6	44.63
7	60.75

The requirement regarding the frame rate can also be met. The maximum resolution is 1600x1200 pixels. The position determination rate of less than 1 s could be achieved by all configurations of the system and is therefore also considered fulfilled.

A high scalability also seems to be given, since the system can be extended with a localization component without much effort. However, it should be noted that the field of vision of already integrated localization components should be taken into account when implementing further localization components.

In the system presented here, the power supply is active. This means that the integration into the infrastructure may require additional effort due to the laying of additional cables. Alternatively, a passive power supply can also be used, but this is not recommended for the reasons given in Chapter 2.7.2. In addition, a passive power supply would lead to a considerable increase in maintenance costs and thus total costs.

The developed prototype of an OIPS is able to fulfil the requirements marked in Figure 42.

Architecture	External localization	Self-Localization			
Identification methods	Marker-based	Feature-based			
Accuracy	< 1 cm	< 5 cm	< 10 cm	< 50 cm	< 100 cm
Coverage	< 1 m	< 5 m	< 10 m		
Image rate	< 15 fps	≥ 15 fps			
Positioning rate	< 1 s	< 15 s			
Update rate	On request	Event-driven	Periodically		
Number of objects	≤ 10	≤ 50	≥ 50		
Scalability	Low	Medium	High		
Required computing power	Low	Medium	High		
Power supply	Active	Passive			
Output data	Undefined	xy position	xy position, ID	xyz position	xyz position, ID
Orientation	No	Yes			

Figure 42 Achieved values of the prototype

Thus, the requirements could be fulfilled for the most part. Achieving an accuracy of less than 1 cm does not seem possible with the current system components. Nevertheless, the presented prototype is a relevant alternative to IPS, which can be proven by comparing its performance to the performance of other Systems (see Table 36).

Table 36 Comparison to existing OIPS

Reference	Average Error [cm]			WD [m]
	x	y	z	
(Lee and Song, 2007)	2.38	4.63		2.20
(Saifizi, Hazry and M. Nor, 2012)	6.33			
(Zhang et al., 2015)	6.16	6.16		2.45
Presentet Prototype	0.76	2.05	15.40	2

Therefore, it was proven that with the use of state of the art, low cost components an OIPS could be developed, which is applicable for many different scenarios.

5.5 Limitations of the study

The limiting factors of this work are the lenses used and the camera body. The question of whether better results could be achieved by using higher quality lenses could not be considered in the context of this work.

The second limiting factor is the camera body used, even if it is equipped with different lenses and mounts, only one image sensor was used and tested in this work.

Furthermore, the results are limited by the measurement methods used.

5.6 Recommendations for future research

In the context of this work it could be shown that the optical indoor position determination represents a relevant alternative to other localization methods in the future. The developed prototype is not only inexpensive and easy to modify, it is also able to meet most of the requirements. By further improving the algorithms and the prototype, an improvement of the achieved results can certainly be achieved.

In a next step, the prototype shall be integrated into a productivity environment so that it can be tested in the individual application scenarios. In addition, the integration of several camera systems would be of high interest in order to validate the certified requirements. In addition, the development of a visualization software would be an option, which could be used to better represent the movement of localization objects.

In addition, the use of an infrared camera would be quite conceivable, since these could theoretically enable object recognition in very poor lighting conditions. However, the question arises whether this would provide the same results as the camera used in the context of this thesis in relatively good to good lighting conditions. The use of an industrial camera should also be considered. Even if it does not meet the low cost requirements, it is an interesting technology for comparison purposes. The same would speak for the use of a webcam, even if it works with interpolation, a collection of measurement data would be quite interesting for a comparison.

As an alternative to marker-based localization, it would also be a challenging issue to determine to what extent the prototype presented here is suitable for feature-based localization. Especially if one considers the fast technical progress it is quite conceivable that the next generation of SBC will be able to meet the high computing power required by feature-based localization methods.

References

Abbas, O. (2016) *Compile OpenCV 3.2.0 + OpenCV Contrib for Python on Raspberry Pi,* 2018. Available at: https://github.com/Tes3awy/OpenCV-3.2.0-Compiling-on-Raspberry-Pi (Accessed: 12 January 2018).

Abor, J. & Quartey, P. (2010) 'Issues in SME development in Ghana and South Africa', *Issues in SME development in Ghana and South* (39), pp. 215–228.

Adams, R. (2009) *Specifications For Popular 2D Bar Codes*. Available at: http://www.adams1.com/stack.html (Accessed: 20 December 2018).

Adler, J. (2018) *Die passende Programmiersprache für Machine Learning wählen,* 2018. Available at: https://www.searchenterprisesoftware.de/meinung/Die-passende-Programmiersprache-fuer-Machine-Learning-waehlen (Accessed: 2 December 2018).

Ahrens, W. (2018) *KMU in Deutschland*. Available at: https://www.hotelier.de/lieferanten/dienstleistungen/kmu-in-deutschland (Accessed: 12 December 2018).

Aktas, I. *et al.* (2017) *Funktechnologien für Industrie 4.0*. Available at: http://www.industrialradio.de/Attachments/Funktechnologien_Industrie_4.0_Web.pdf (Accessed: 4 November 2018).

Al Nuaimi, K. and Kamel, H. (2011) 'A survey of indoor positioning systems and algorithms', *2011 International Conference on Innovations in Information Technology, 2011 International Conference on Innovations in Information Technology (IIT)*, Abu Dhabi, United Arab Emirates, 25-27 April: IEEE, pp. 185–190. doi: 10.1109/INNOVATIONS.2011.5893813

Albarelli, A., Rodolà, E. and Torsello, A. (2010) 'Robust Camera Calibration using Inaccurate Targets', 16.1-16.10. doi: 10.5244/C.24.16

Aminikashani, M., Gu, W. and Kavehrad, M. (2016) 'Indoor positioning with OFDM Visible Light Communications', *2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), 2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, NV, USA, 9-12 January: IEEE, pp. 505–510. doi: 10.1109/CCNC.2016.7444832

Anderl, R. and Anokhin, Ol, Arndt, A. (2016) 'Effiziente Fabrik 4.0 Darmstadt – Industrie 4.0 Implementierung für die mittelständige Industrie', in Sendler, U., Anderl, R. and Dumitrescu, R. (eds.) *Industrie 4.0 grenzenlos.* (Xpert.press). Berlin: Springer Vieweg, p. 121.

Armstrong, J., Sekercioglu, Y. and Neild, A. (2013) 'Visible light positioning: A roadmap for international standardization', *IEEE Communications Magazine*, 51(12), pp. 68–73. doi: 10.1109/MCOM.2013.6685759

Armstrong, M.N. (1996) *Self-calibration from image sequences*. Dissertation. University of Oxford.

Arnold, D. *et al.* (2008) *Handbuch Logistik*. Berlin, Heidelberg: Springer.

Atkins, B. (no date) *Field of View - Rectilinear and Fishye Lenses*. Available at: http://www.bobatkins.com/photography/technical/field_of_view.html (Accessed: 11 January 2019).

Babinec, A. *et al.* (2014) 'Visual Localization of Mobile Robot Using Artificial Markers', *Procedia Engineering*, 96, pp. 1–9. doi: 10.1016/j.proeng.2014.12.091

Bannore, V. (2009) *Iterative-Interpolation Super-Resolution Image Reconstruction: A Computationally Efficient Technique*. (Studies in Computational Intelligence, 195). Berlin: Springer.

Barberis, C. *et al.* (2014) 'Experiencing Indoor Navigation on Mobile Devices', *IT Professional*, 16(1), pp. 50–57. doi: 10.1109/MITP.2013.54

Basler AG (no date) *n richtigen Durchblick: So finden Sie das passende Objektiv zu Ihrer Kamera*. Available at: *Für de* https://www.baslerweb.com/de/vision-campus/vision-systeme-und-komponenten/das-richtige-objektiv-finden/ (Accessed: 5 January 2019).

Basler AG (no date) *Kameraauswahl – Wie finde ich die richtige Kamera für mein Bildverarbeitungssystem?* Available at: https://www.baslerweb.com/de/vision-campus/vision-systeme-und-komponenten/kamera-auswahl/.

Bässmann, H. (no date) *Industrie-Kameras*. Available at: http://www.industrie-kamera.de/ (Accessed: 15 October 2018).

Bauernhansl, T. (2016) *Industrie 4.0 Entwicklungsfelder für den Mittelstand*. Stuttgart. Available at: https://www.ipa.fraunhofer.de/content/dam/ipa/de/documents/Publikationen/Studien/Studie_Industrie_40_Entwicklungsfelder_fuer_den_Mittelstand.pdf (Accessed: 28 June 2017).

Bauernhansl, T., Hompel, M. ten and Vogel-Heuser, B. (eds.) (2014) *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung, Technologien, Migration*. Wiesbaden: Springer Vieweg (SpringerLink).

Bencina, R., Kaltenbrunner, M. and Jorda, S. (2005) 'Improved Topological Fiducial Tracking in the reacTIVision System', *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops, 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, San Diego, CA, USA, 20-26 June 2005: IEEE, p. 99. doi: 10.1109/CVPR.2005.475

Bensky, A. (2008) *Wireless positioning technologies and applications*. (GNSS technology and applications series). Boston, Mass: Artech House.

Bergamasco, F., Albarelli, A. and Torsello, A. (2013) 'Pi-Tag: A fast image-space marker design based on projective invariants', *Machine Vision and Applications*, 24(6), pp. 1295–1310. doi: 10.1007/s00138-012-0469-6

Bergamasco, F. *et al.* (2011) 'RUNE-Tag: A high accuracy fiducial marker with strong occlusion resilience', *CVPR 2011, 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Colorado Springs, CO, USA, 20-25 June: IEEE, pp. 113–120. doi: 10.1109/CVPR.2011.5995544

Bergen, M.H. *et al.* (2017) 'Design and Implementation of an Optical Receiver for Angle-of-Arrival-Based Positioning', *Journal of Lightwave Technology*, 35(18), pp. 3877–3885. doi: 10.1109/JLT.2017.2723978

BeSpoon (no date) *Use Case Fabrik: Bringen Sie Intralogistik auf die nächste Stufe*. Available at: http://bespoon.com/de/zoom-on-2/fabrik/ (Accessed: 12 December 2018).

Bildstein, A. and Seidelmann, J. (2014) 'Industrie 4.0-Readiness: Migration zur Industrie 4.0-Fertigung', in Bauernhansl, T., Hompel, M. ten and Vogel-Heuser, B. (eds.) *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung, Technologien, Migration.* (SpringerLink). Wiesbaden: Springer Vieweg, pp. 581–597.

BITKOM, VDMA and ZVEI (2015) *Umsetzungsstrategie Industrie 4.0: Ergebnisbericht der Plattform Industrie 4.0*.

Bloß, H. and Bauer, N. (2000) *Leitfaden zur bildgebenden Sensortechnik*. (Vision).

Böhm, J. (2007) 'Orientation of Image Sequences in a Point-based Environment Model', *Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007), Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007)*, Montreal, QC, Canada, 21-23 August: IEEE, pp. 233–240. doi: 10.1109/3DIM.2007.38

Boochs, F. *et al.* (2010) 'Increasing the accuracy of untaught robot positions by means of a multi-camera system', *2010 International Conference on Indoor Positioning and Indoor Navigation, 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Zurich, Switzerland, 15-17 September: IEEE, pp. 1–9.
doi: 10.1109/IPIN.2010.5646261

Bouguet, J.Y. (2015) *Camera Calibration Toolbox for Matlab,* 2015. Available at: http://www.vision.caltech.edu/bouguetj/calib_doc/ (Accessed: 11 December 2018).

Bradski, G.R. and Kaehler, A. (2011) *Learning OpenCV*. (Software that sees). Beijing: O'Reilly.

Brena, R.F. *et al.* (2017) 'Evolution of Indoor Positioning Technologies: A Survey', *Journal of Sensors*, 2017(6, article 359), pp. 1–21. doi: 10.1155/2017/2630413

Bubeck, A. *et al.* (2014) 'Vom fahrerlosen Transportsystem zur intelligenten mobilen', in Bauernhansl, T., Hompel, M. ten and Vogel-Heuser, B. (eds.) *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung, Technologien, Migration.* (SpringerLink). Wiesbaden: Springer Vieweg.

Busch, D.D. (2009) *Digital SLR cameras & photography for dummies*. 3rd edn. (--For dummies). Hoboken, N.J: Wiley.

Caron, C. *et al.* (2017) 'Indoor Positioning', in Shekhar, S., Xiong, H. and Zhou, X. (eds.) *Encyclopedia of GIS.* Cham: Springer International Publishing, pp. 1011–1019.

Carr, D. (2016) *How to calculate Field of View in photography*. Available at: https://shuttermuse.com/calculate-field-of-view-camera-lens/ (Accessed: 10 January 2019).

CME Services (2017) *Secret Video Recorder by CME Services APK,* 2017. Available at: https://apk.support/app/com.aes.secretvideorecorder (Accessed: 12 December 2018).

'Comparison of single-board computers' (2018), 2018. Available at: https://en.wikipedia.org/wiki/Comparison_of_single-board_computers (Accessed: 15 November 2018).

Costanza, E. and Robinson, J. (2003) 'A region adjacency tree approach to the detection and design of fiducials', in *Vision, Video and Graphics (VVG).*

Czeranowsky, C. (2017) *Basler Whitepaper: Welche Objektive gibt es und wie wählen Sie as geeignete Objektiv für eine Kamer?* Basler Whitepaper.

Datta, A., Kim, J.-S. and Kanade, T. (2009) 'Accurate camera calibration using iterative refinement of control points', *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, Kyoto, Japan, 27 September - 4 October: IEEE, pp. 1201–1208. doi: 10.1109/ICCVW.2009.5457474

Debski, A. *et al.* (2015) 'Open-source Localization Device for Indoor Mobile Robots', *Procedia Computer Science*, 76, pp. 139–146. doi: 10.1016/j.procs.2015.12.327

Demant, C., Streicher-Abel, B. and Springhoff, A. (2011) *Industrielle Bildverarbeitung: Wie optische Qualitätskontrolle wirklich funktioniert.* 3rd edn. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg.

Deutsches Institut für Normung e.V. (1985) *Nummerung Grundbegriffe*.

Deutsches Institut für Normung e.V. (2011) *Begriffe der Qualitätssicherung und Statistik*. Berlin: Beuth Verlag.

developer.android.com (2018) *Camera.Parameters,* 2018. Available at: https://developer.android.com/reference/android/hardware/Camera.Parameters (Accessed: 12 December 2018).

Do, T.-H., Hwang, J. and Yoo, M. (2013) 'TDoA based indoor visible light positioning systems', *2013 Fifth International Conference on Ubiquitous and Future Networks (ICUFN), 2013 Fifth International Conference on Ubiquitous and Future Networks (ICUFN)*, DA NANG, Vietnam, 2-5 July: IEEE, pp. 456–458. doi: 10.1109/ICUFN.2013.6614860

Doerr, J. (1978) 'Low-cost microcomputing: The personal computer and single-board computer revolutions', *Proceedings of the IEEE*, 66(2), pp. 117–130. doi: 10.1109/PROC.1978.10862

Drees, C. (2017) *Facebook bastelt jetzt auch am modularen Smartphone*. Available at: https://www.mobilegeeks.de/news/facebook-bastelt-jetzt-auch-am-modularen-smartphone/ (Accessed: 15 December 2018).

Dubucq, A., Hinckeldeyn, J. and Kreutzfeldt, J. (2018) 'Einsatz von Location Based Services in Produktion und Intralogistik', *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb*, 113(5), pp. 337–341. doi: 10.3139/104.111909

Dünkler, R. (2015) *»Industrie 4.0«: Drahtlose Kommunikation, Lokalisierung und RFID als Basistechnologien*. Nürnberg. Available at: https://www.coburg.ihk.de/files/1_duenkler_basistechnologien_industrie_4.0.pdf (Accessed: 6 July 2017).

Edmund Optics (no date) *Understanding Focal Length and Field of View*. Available at: https://www.edmundoptics.com/resources/application-notes/imaging/understanding-focal-length-and-field-of-view/ (Accessed: 15 December 2018).

Embedded Linux Wiki (2018) *RPi USB Webcams,* 2018. Available at: https://elinux.org/RPi_USB_Webcams (Accessed: 14 October 2018).

Emuze, F. and Flanegan, D. (2014) 'Exploring Transformation Challenges: Limits to the Growth of Building Contractors in South Africa', pp. 135–143.
doi: 10.32738/CEPPM.201411.0014

Erhardt, A. (2008) *Einführung in die Digitale Bildverarbeitung: Grundlagen, Systeme und Anwendungen*. Wiesbaden: Vieweg+Teubner / GWV Fachverlage GmbH Wiesbaden.

Essati, A. (2013) 'Staplerortung und -navigation', *IT Logistics* (1), pp. 38–39.

Ferrara, P. *et al.* (2017) 'Wide-angle and long-range real time pose estimation: A comparison between monocular and stereo vision systems', *Journal of Visual Communication and Image Representation*, 48, pp. 159–168.
doi: 10.1016/j.jvcir.2017.06.008

Fiala, M. (2005) 'Comparing ARTag and ARToolkit plus fiducial marker systems', *IREE International Worksho on Haptic Audio Visual Environments and their Applications, 2005, IREE International Worksho on Haptic Audio Visual Environments and their Applications, 2005*, Ottawa, Canada, Oct. 1, 2005: IEEE, pp. 147–152.
doi: 10.1109/HAVE.2005.1545669

Fraunhofer-Institut für Integrierte Schaltungen IIS (2018) *awiloc®: Die Lokalisierungstechnologie für Städte und Gebäude*. RSSI - Received Signal Strength Indication. Available at: https://www.iis.fraunhofer.de/de/ff/lv/lok/tech/feldstaerke/rssi.html (Accessed: 12 January 2019).

Fraunhofer-Institut für Integrierte Schaltungen IIS (2018) *Ortung von Flurförderzeugen*. Available at: https://www.iis.fraunhofer.de/de/ff/lv/lok/proj/indo.html (Accessed: 12 January 2019).

Fritsche, C. and Klein, A. (2009) 'On the Performance of Hybrid GPS/GSM Mobile Terminal Tracking', *2009 IEEE International Conference on Communications Workshops, 2009 IEEE International Conference on Communications Workshops*, Dresden, Germany, 14-18 June: IEEE, pp. 1–5. doi: 10.1109/ICCW.2009.5207984

Garrido-Jurado, S. *et al.* (2014) 'Automatic generation and detection of highly reliable fiducial markers under occlusion', *Pattern Recognition*, 47(6), pp. 2280–2292. doi: 10.1016/j.patcog.2014.01.005

Grobe, L. *et al.* (2013) 'High-speed visible light communication systems', *IEEE Communications Magazine*, 51(12), pp. 60–66. doi: 10.1109/MCOM.2013.6685758

Günterberg, B. (2012) *Unternehmensgrößenstatistik: Unternehmen, Umsatz und sozialversicherungspflichtig Beschäftigte 2004 bis 2009 in Deutschland, Ergebnisse des Unternehmensregisters (URS 95)*. Bonn. Available at: https://www.ifm-bonn.org/statistiken/unternehmensbestand/ (Accessed: 12 December 2018).

Günthner, W.A. and Hohenstein, F. (2012) 'Anforderungen und Fähigkeiten gegenwärtiger Stapler – Lokalisierung', in Schatzmann, A. (ed.) *Nikarchos II: Epigrammata.* Göttingen: Vandenhoeck & Ruprecht, pp. 17–124.

Hagisonic (2016) *User's Guide: Localization system StarGazerTM for Intelligent Robots*. Available at: http://ftp.k-team.com/KheperaIII/KSG/StarGazer_Guide_02.0904.16(English).pdf.

Halper, M. (2017) *Acuity pads its indoor-positioning push with shopping cart trackers*. Available at: https://www.ledsmagazine.com/articles/2017/04/acuity-pads-its-indoor-positioning-push-with-shopping-cart-trackers.html.

Han, G. (2015) *Development of an Indoor Location Tracking System for the Operating Room*. Dissertation. Technische Universität München.

Hassan, N.U. *et al.* (2015) 'Indoor Positioning Using Visible LED Lights', *ACM Computing Surveys*, 48(2), pp. 1–32. doi: 10.1145/2835376

Heikkila, J. and Silven, O. 'A four-step camera calibration procedure with implicit image correction', *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1106–1112. doi: 10.1109/CVPR.1997.609468

Heinrich, M. (2017) *Transport- und Lagerlogistik: Systematik, Planung, Einsatz und Wirtschaftlichkeit*. 10th edn. Wiesbaden: Springer Fachmedien Wiesbaden GmbH.

Hennessy, J.L. and Patterson, D.A. (2012) *Computer architecture: A quantitative approach*. 5th edn. San Francisco: Morgan Kaufmann Publishers/Elsevier.

Hermann, M., Pentek, T. and Otto, B. (2016) 'Design Principles for Industrie 4.0 Scenarios', *2016 49th Hawaii International Conference on System Sciences (HICSS), 2016 49th Hawaii International Conference on System Sciences (HICSS)*, Koloa, HI, USA, 5-8 January: IEEE, pp. 3928–3937. doi: 10.1109/HICSS.2016.488

Heuel & Löher GmbH & Co. KG (no date) *Localino v3.0 Kit 3D*. Available at: http://www.localino.net/shop/consumer/16/localino-v3.0-kit-3d (Accessed: 12 January 2019).

Higuchi, M., Datta, A. and Kanade, T. (2018) *Software Package for Precise Camera Calibration,* 20 May. Available at: https://www.ri.cmu.edu/project/software-package-for-precise-camera-calibration/.

Hile, H. and Borriello, G. (2006) 'Positioning and Orientation in Indoor Environments Using Camera Phones', *Fourth IEEE International Conference on Computer Vision Systems (ICVS'06), Fourth IEEE International Conference on Computer Vision Systems (ICVS'06)*, New York, NY, USA, 4-7 January: IEEE, pp. 32–39. doi: 10.1109/MCG.2008.80

Hilty, L.M. *et al.* (eds.) (2012) *Lokalisiert und identifiziert: Wie Ortungstechnologien unser Leben verändern*. Zürich: vdf (TA-SWISS, 57).

Hirtle, S.C. and Bahm, R.B. (2015) 'Cognition for the Navigation of Complex Indoor Environments', in Karimi, H.A. (ed.) *Indoor wayfinding and navigation.* Boca Raton, FL: CRC Press.

Hohenstein, F.C. (2014) *Systementwurf und Umsetzung einer funktionsintegrierenden Gabelstaplerlokalisierung für eine wandlungsfähige und effiziente Transportausführung*. Disseration. Technische Universität München.

Hompel, M. ten, Schmidt, T. and Dregger, J. (2018) *Materialflusssysteme: Förder- und Lagertechnik*. 4th edn. (VDI-Buch).

Hönig, H. and Lorenz, B. (2017) 'Low-Cost-Digitalisierung in der Produktion', *ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb*, 112(12), pp. 895–898. doi: 10.3139/104.111832

Hossen, M.S., Park, Y. and Kim, K.-D. (2015) 'Performance improvement of indoor positioning using light-emitting diodes and an image sensor for light-emitting diode communication', *Optical Engineering*, 54(3), p. 35108. doi: 10.1117/1.OE.54.3.035108

International Standard (2008) *Information Technology - Automatic Identification and Data Capture (AIDC) Techniques - Harmonized Vocabulary, Part 5: Locating Systems.* Berlin: Beuth Verlag.

item Industrietechnik GmbH (no date) *Low Cost Intelligent Automation*. Available at: https://glossar.item24.com/glossarindex/artikel/item/low-cost-intelligent-automation.html (Accessed: 12 January 2019).

Jimenez Ruiz, A.R. and Seco Granja, F. (2017) 'Comparing Ubisense, BeSpoon, and DecaWave UWB Location Systems: Indoor Performance Analysis', *IEEE Transactions on Instrumentation and Measurement*, 66(8), pp. 2106–2117. doi: 10.1109/TIM.2017.2681398

Jones, D. (2017) *Picamera: Camera Hardware,* 25 February. Available at: https://picamera.readthedocs.io/en/release-1.13/fov.html (Accessed: 12 November 2018).

Jovicic, A., Li, J. and Richardson, T. (2013) 'Visible light communication: Opportunities, challenges and the path to market', *IEEE Communications Magazine*, 51(12), pp. 26–32. doi: 10.1109/MCOM.2013.6685754

Kagermann, H., Lukas, W.-D. and Wahlster, W. (2011) 'Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution', *VDI-Nachrichten* (13).

Kagermann, H., Wahlster, W. and Helbig, J. (2013) *Recommendations for implementing the strategic initiative INDUSTRIE 4.0.: Final report of the Industrie 4.0 Working Group*. Germany.

Karwowski, W. (2011) *Human factors and ergonomics in consumer product design: Methods and techniques*. (Ergonomics design and management).

Kim, H.-S. *et al.* (2013) 'An Indoor Visible Light Communication Positioning System Using a RF Carrier Allocation Technique', *Journal of Lightwave Technology*, 31(1), pp. 134–144. doi: 10.1109/JLT.2012.2225826

Klopschitz, M. *et al.* (2010) 'Visual tracking for Augmented Reality', *2010 International Conference on Indoor Positioning and Indoor Navigation, 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Zurich, Switzerland, 15-17 September: IEEE, pp. 1–4. doi: 10.1109/IPIN.2010.5648274

Knaak, K. (no date) *Die Tiefenschärfe – Das Hervorheben der Motive*. Available at: http://www.segefo.de/kameraeinstellung/tiefenschaerfe (Accessed: 10 December 2018).

Knuchel, T. *et al.* (2011) '2D-Codes: Technologie und Anwendungsbereiche', *WIRTSCHAFTSINFORMATIK*, 53(1), pp. 49–52. doi: 10.1007/s11576-010-0255-x

Kohoutek, T.K., Mautz, R. and Wegner, J.D. (2013) *Fusion of building information and range imaging for autonomous location estimation in indoor environments* (2).

Köppe, E. (2014) *Lokalisierung sich bewegender Objekte innerhalb und außerhalb von Gebäuden*. Dissertation. Freien Universität Berlin.

Kruegle, H. (1995) *CCTV surveillance: Video practices and technology*. Boston: Butterworth-Heinemann.

Kunow, A. (2016) *Raspberry Pi Verkaufszahlen: Mini-PC mit 10 Millionen Nutzern*. Available at: https://www.pc-magazin.de/news/raspberry-pi-verkaufszahlen-starter-kit-release-preis-3196825.html (Accessed: 5 December 2018).

La Delfa, G.C. *et al.* (2016) 'Performance analysis of visualmarkers for indoor navigation systems', *Frontiers of Information Technology & Electronic Engineering*, 17(8), pp. 730–740. doi: 10.1631/FITEE.1500324

Lachmann, J. (2018a) *Auslastungsanalyse von Arbeitsgeräten*. Available at: https://www.infsoft.com/de/anwendungsbeispiele/articleid/328/auslastungsanalyse-von-arbeitsgeraeten (Accessed: 12 December 2018).

Lachmann, J. (2018b) *Indoor Tracking mit Display-Beacons in der Logistik*. Available at: https://www.infsoft.com/de/anwendungsbeispiele/articleid/277/indoor-tracking-mit-display-beacons-in-der-logistik (Accessed: 12 December 2018).

Lachmann, J. (2018c) *Management und Tracking von externen Mitarbeitern*. Available at: https://www.infsoft.com/de/anwendungsbeispiele/articleid/312/management-und-tracking-von-externen-mitarbeitern (Accessed: 12 December 2018).

LEAN:log (2012) *Low Cost Automation*. Available at: http://www.fml.mw.tum.de/leanlogwiki/index.php?title=Low_Cost_Automation&oldid=3209 (Accessed: 12 January 2019).

Lee, S. and Song, J.-B. (2007) 'Mobile robot localization using infrared light reflecting landmarks', *2007 International Conference on Control, Automation and Systems, 2007 International Conference on Control, Automation and Systems*, Seoul, South Korea, 17-20 October: IEEE, pp. 674–677. doi: 10.1109/ICCAS.2007.4406984

Liao, X. *et al.* (2013) 'Computer vision system for an autonomous mobile robot', *Proceedings of SPIE - The International Society for Optical Engineering*.

Lichtblau, K. *et al.* (2015) *Industrie 4.0 Readniness IMPULS Studie Oktober 2015*. Achen, Köln.

Linß, G. (2018) *Qualitätsmanagement für Ingenieure*. 4th edn. München: Hanser Carl.

Liu, H. *et al.* (2007) 'Survey of Wireless Indoor Positioning Techniques and Systems', *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 37(6), pp. 1067–1080. doi: 10.1109/TSMCC.2007.905750

Logivations (no date) *Kamerabasierte Objekterkennung: Deep Machine Learning in Verbindung mit Computer Vision*. Available at: https://www.logivations.com/de/land/pdf/KameraBasierte_Objekterkennung.pdf.

Lomography (2011) *Variable oder Festbrennweite: Zoom vs Prime. Lichstärke gewinnt!* Available at: https://www.lomography.de/magazine/74656-variable-oder-festbrennweite-prime-vs-zoom-lichtstaerke-gewinnt.

Lymberopoulos, D. *et al.* (2015) 'A realistic evaluation and comparison of indoor location technologies', *Proceedings of the 14th International Conference on Information Processing in Sensor Networks - IPSN '15, the 14th International Conference*, Seattle, Washington, 13-16 April. New York, New York, USA: ACM Press, pp. 178–189. doi: 10.1145/2737095.2737726

Malik, A. (2009) *RTLS for dummies*. (--For dummies). Hoboken, N.J: Wiley.

Martin, H. (2016) *Transport- und Lagerlogistik*. Wiesbaden: Springer Fachmedien Wiesbaden.

Marvelmind Robotics (no date) *Starter Set – HW v4.9 + IMU*. Available at: https://marvelmind.com/product/starter-set-hw-v4-9-imu-plastic-housing/ (Accessed: 12 January 2019).

Marwedel, P. (2011) *Embedded system design: Embedded systems foundations of cyber-physical systems*. 2nd edn. (Embedded systems): Springer.

MathWorks (no date) *What Is Camera Calibration?* Available at: https://de.mathworks.com/help/vision/ug/camera-calibration.html (Accessed: 1 December 2018).

Mautz, R. (2012) *Indoor positioning technologies*. Habilitation. ETH Zurich.

Mautz, R. and Tilch, S. (2011) 'Survey of optical indoor positioning systems: Guimaraes, Portugal, 21 - 23 September 2011', pp. 1–7. doi: 10.1109/IPIN.2011.6071925

Mulloni, A. *et al.* (2009) 'Indoor Positioning and Navigation with Camera Phones', *IEEE Pervasive Computing*, 8(2), pp. 22–31. doi: 10.1109/MPRV.2009.30

Mulloni, A., Seichter, H. and Schmalstieg, D. (2011) 'Handheld Augmented Reality Indoor Navigation with Activity-Based Instructions', in *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, pp. 211–220. Available at: http://dl.acm.org/citation.cfm?id=2037373.

Mustafah, Y.M., Azman, A.W. and Akbar, F. (2012) 'Indoor UAV Positioning Using Stereo Vision Sensor', *Procedia Engineering*, 41, pp. 575–579. doi: 10.1016/j.proeng.2012.07.214

Mutka, A. *et al.* (2008) 'A low cost vision based localization system using fiducial markers', *IFAC Proceedings Volumes*, 41(2), pp. 9528–9533. doi: 10.3182/20080706-5-KR-1001.01611

Naimark, L. and Foxlin, E. (2002) 'Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker', *Proceedings. International Symposium on Mixed and Augmented Reality, IEEE and ACM International Symposium on Mixed and Augmented Reality*, Darmstadt, Germany, 30 Sept.-1 Oct. 2002: IEEE Comput. Soc, pp. 27–36. doi: 10.1109/ISMAR.2002.1115065

National Instruments (2018) *Berechnen der Objektivbrennweite und der Sensorauflösung einer Kamera*. Available at: http://www.ni.com/product-documentation/54616/de/.

National Small Business Chamber (2016) *2016 National Small Business Survey: A National Small Business Chamber (NSBC) Initiative*.

Naumann, H. (2014) *Handbuch Bauelemente der Optik: Grundlagen, Werkstoffe, Geräte, Messtechnik*. 7th edn. München: Hanser.

Ndjiongue, A.R., Ferreira, H.C. and Ngatched, T.M.N. (2015) *Visible Light Communications (VLC) Technology*.

Neumann, K. (2016) *Ultra-Wideband Positioniersystem für die Schwerindustrie*. Dissertation. Rheinisch-Westfälische Technische Hochschule Aachen.

Olson, E. (2011) 'AprilTag: A robust and flexible visual fiducial system', *2011 IEEE International Conference on Robotics and Automation, 2011 IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 9-13 May: IEEE, pp. 3400–3407. doi: 10.1109/ICRA.2011.5979561

OpenCV (2015) *Camera Calibration*. Available at: https://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html (Accessed: 11 November 2018).

OpenCV Dev Team (2019) *Camera Calibration and 3D Reconstruction: OpenCV 2.4.13.7 documentation,* 2019. Available at: https://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html (Accessed: 12 January 2019).

Overmeyer, L. (2014) 'Automatisierung in der Materialflusstechnik', in Grote, K.-H. and Feldhusen, J. (eds.) *Dubbel.* Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 1719–1725.

Pawellek, G. (2014) *Ganzheitliche Fabrikplanung: Grundlagen, Vorgehensweise, EDV-Unterstützung*. 2nd edn. (VDI-Buch). Berlin: Springer Vieweg.

Pelka, M. (2014) *(Indoor)-Lokalisation*. Available at: https://docplayer.org/26329710-Indoor-lokalisation.html.

Philips Systems (2016) *Philips indoor positioning whitepaper*. White Paper. Available at: http://www.lighting.philips.com/main/systems/lighting-systems/indoor-positioning (Accessed: 12 January 2019).

Pieper, M. (2017) *Mathematische Optimierung: Eine Einführung in die kontinuierliche Optimierung mit Beispielen*. (essentials). Wiesbaden: Springer.

Pollert, A. *et al.* (2016) *Duden Wirtschaft von A bis Z: Grundlagenwissen für Schule und Studium, Beruf und Alltag*. 6th edn. (Duden). Berlin: Dudenverlag.

Pozyx BVBA (no date) *Pozyx developer's kit*. Available at: https://www.pozyx.io/store/detail/3 (Accessed: 12 January 2019).

Rademacher, T. (2017) *Embedded Vision: Neues Konzept mit neuen Anwendungen*. Basler White Paper.

Raspberry Pi Foundation (no date) *Camera Module*. Hardware Documentation. Available at: https://www.raspberrypi.org/documentation/hardware/camera/README.md (Accessed: 10 December 2018).

Raspberry Pi Foundation (no date) *Power Supply*. Available at: https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md (Accessed: 11 January 2019).

Raspberry Pi Foundation (no date) *Raspberry Pi 3 Model B+: Specifications*. Available at: https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/ (Accessed: 12 December 2018).

Reinhart, G. (2017) *Handbuch Industrie 4.0: Geschäftsmodelle, Prozesse, Technik*. München: Hanser.

Rencheng, S. *et al.* (2011) 'The Design of a New Marker in Augmented Reality', in *2011 International Conference on Economics and Finance Research,* 4th edn. Singapore.

Reuter, A., Seidel, H.-P. and Ihrke, I. (2012) 'Blurtags: spatially varying psf estimation with out-of-focus patterns', in *In 20th Int. Conf. on Computer Graphics, Visualization.*

Rice, A.C., Beresford, A.R. and Harle, R.K. (2006) 'Cantag: An open source software toolkit for designing and deploying marker-based vision systems', *Fourth Annual IEEE International Conference on Pervasive Computing and Communications (PERCOM'06), Fourth Annual IEEE International Conference on Pervasive Computing and Communications (PERCOM'06)*, Pisa, Italy, 13-17 March 2006: IEEE, pp. 12–21. doi: 10.1109/PERCOM.2006.13

Saab, S.S. and Saab, K.K. (2016) 'A Positioning System for Photodiode Device Using Collocated LEDs', *IEEE Photonics Journal*, p. 1. doi: 10.1109/JPHOT.2016.2611441

Saam, M., Viete, S. and Schiel, S. (2016) *Digitalisierung im Mittelstand: Status Quo, aktuelle Entwicklungen und Herausforderungen*. Mandelheim. Available at: https://www.kfw.de/PDF/Download-Center/Konzernthemen/Research/PDF-Dokumente-Studien-und-Materialien/Digitalisierung-im-Mittelstand.pdf (Accessed: 28 June 2017).

Sagitov, A. *et al.* (2017) 'ARTag, AprilTag and CALTag Fiducial Marker Systems: Comparison in a Presence of Partial Marker Occlusion and Rotation', *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics, 14th International Conference on Informatics in Control, Automation and Robotics*, Madrid, Spain, 26-28 July: SCITEPRESS - Science and Technology Publications, pp. 182–191. doi: 10.5220/0006478901820191

Sagitov, A. *et al.* (2017) 'Comparing fiducial marker systems in the presence of occlusion', *2017 International Conference on Mechanical, System and Control Engineering (ICMSC), 2017 International Conference on Mechanical, System and Control Engineering (ICMSC)*, St.Petersburg, Russia, 19-21 May: IEEE, pp. 377–382. doi: 10.1109/ICMSC.2017.7959505

Saifizi, M., Hazry, D. and M. Nor, R. (2012) 'Vision Based Mobile Robot Navigation System', *International Journal of Control Science and Engineering*, 2(4), pp. 83–87. doi: 10.5923/j.control.20120204.05

Sakpere, W., Adeyeye Oshin, M. and Mlitwa, N.B.W. (2017) 'A State-of-the-Art Survey of Indoor Positioning and Navigation Systems and Technologies', *South African Computer Journal*, 29(3) (53pp). doi: 10.18489/sacj.v29i3.452

Sattar, J. *et al.* (2007) 'Fourier tags: Smoothly degradable fiducial markers for use in human-robot interaction', in *Fourth Canadian Conference on Computer and Robot Vision*, pp. 165–174.

Saunders, M., Lewis, P. and Thornhill, A. (2009) *Research methods for business students*. 5th edn. Harlow: Financial Times Prentice Hall.

Schaar, D. (2018) *Raspberry PI: Zukunftsmodell in der Bildverarbeitung*. Available at: https://www.industrielle-automation.net/raspberry-pi-zukunftsmodell-in-der-bildverarbeitung/ (Accessed: 5 December 2018).

Schäfer, S., Schöttke, D. and Hermann, P. (2017) 'Optische Raumüberwachungssysteme in wandelbaren Umgebungen der Smart Factory', in Pinnow, C. and Schäfer, S. (eds.) *Industrie 4.0 - Safety und Security - mit Sicherheit gut vernetzt: Branchentreff der Berliner und Brandenburger Wissenschaft und Industrie.* (DIN). Berlin: Beuth Verlag GmbH, pp. 41–54.

Schmidt, D. (2011) 'GeoCEP - RTLS und GIS unterstütztes Event Processing', in Strobl, J. (ed.) *Angewandte Geoinformatik 2011: Beiträge zum 23. AGIT-Symposium Salzburg.* In: Strobel, J. (Ed.). Angewandte Geoinformatik 2011: Beiträge zum 23. AGIT-Symposium Salzburg. Berlin: Wichmann.

Schmidt, H. (2018) *Raspberry Pi programmieren mit C/C++ und Bash: Mehr als 50 Programme rund um Foto, Video & Audio*. (#makers do it). München: Hanser.

Schröder, C. (2016) *Herausforderungen von Industrie 4.0 für den Mittelstand*. Bonn (Accessed: 16 June 2017).

Schröder, G. and Treiber, H. (2007) *Technische Optik*. 10th edn. (Kamprath-Reihe). s.l.: Vogel Business Media.

Schwab, K., Sala-i-Martin, X. and World Economic Forum (2016) *The global competitiveness report 2016-2017: Insight report*. Geneva: World Economic Forum.

Siemens AG (2018) *Real-Time Locating Systems (RTLS) in der Produktion: Für ein digitales Unternehmen in Bewegung*. White Paper.

Siltanen, S. (2012) *Theory and applications of marker-based augmented reality [Markkeriperustaisen lisätyn todellisuuden teoria ja sovellukset] // Theory and applications of marker-based augmented reality*. (VTT science, 3). Espoo: VTT.

Small Enterprise Development Agency (2016) *The small, medium and micro enterprise sector of South Africa* (Accessed: 12 January 2019).

Šmíd, M. (2015) *Camera intrinsic parameters calibration from chessboard video sequence*. Available at: https://github.com/smidm/video2calibration (Accessed: 11 January 2019).

Springer, T. (2012) 'MapBiquitous: An Approach for Integrated Indoor/Outdoor Location-Based Services', in Zhang, J.Y., Wilkiewicz, J. and Nahapetian, A. (eds.) *Mobile Computing, Applications, and Services: Third International Conference, MobiCASE 2011, Los Angeles, CA, USA, October 24-27, 2011. Revised Selected Papers.* (Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, 39).

Statistisches Bundesamt (2017) *Anzahl der Unternehmen* in Deutschland nach Beschäftigtengrößenklassen im Jahr 2016*. Available at: https://de.statista.com/statistik/daten/studie/1929/umfrage/unternehmen-nach-beschaeftigtengroessenklassen/.

Straub, J. *et al.* (2013) 'OpenOrbiter: A Low-Cost, Educational Prototype CubeSat Mission Architecture', *Machines*, 1(1), pp. 1–32. doi: 10.3390/mach1010001

Takeda, H. and Klages, C. (2006) *LCIA - Low Cost Intelligent Automation: Produktivitätsvorteile durch Einfachautomatisierung*. 2nd edn. Landsberg am Lech: REDLINE.

The World Bank Group (2017) *GNI per capita, PPP*. Available at: https://data.worldbank.org/indicator/NY.GNP.PCAP.PP.CD (Accessed: 12 December 2018).

Thrun, S., Szeliski, R. and Dahlkamp, H. (2015) *CS 223-B: Introduction to Computer Vision – Lecture 2 Lenses and Camera Calibration*. Vorlesungsfolien. Stanford University. Available at: https://slideplayer.com/slide/7552326/.

Tikanmäki, A. and Röning, J. (2011) *Markers - toward general purpose information representation.* University of Oulu.

Tilch, S. and Mautz, R. (2013) 'CLIPS – a camera and laser-based indoor positioning system', *Journal of Location Based Services*, 7(1), pp. 3–22. doi: 10.1080/17489725.2012.688643

TotalTrax (2018) *The Sky-Trax System™: Vehicle Tracking and Fleet Management*. Available at: http://www.totaltraxinc.com/smart-forklift-solutions/the-sky-trax-system/ (Accessed: 12 December 2018).

Tso, F.P. *et al.* (2013) 'The Glasgow Raspberry Pi Cloud: A Scale Model for Cloud Computing Infrastructures', *2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops, 2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops (ICDCSW)*, Philadelphia, PA, USA, 8-11 July: IEEE, pp. 108–112. doi: 10.1109/ICDCSW.2013.25

van Diggelen, F. and Enge, P. (2015) 'The World's first GPS MOOC and Worldwide Laboratory using Smartphones', in *Proceedings of the 28th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2015).* Tampa, Florida, pp. 361–369.

Verein Deutscher Ingineure (2010) *Industrielle Bildverarbeitung Grundlagen und Begriffe*. Berlin: Beuth.

Vision Doctor (no date) *Optische Grundlagen - Berechnung der Optik*. Available at: https://www.vision-doctor.com/optische-grundlagen.html (Accessed: 10 January 2019).

Vogel, K. (2016) *Illuminating the In-Store Experience: LED-Based Indoor Positioning Services Benefit Shoppers and Retailers*. Available at: https://blog.nema.org/2016/02/26/illuminating-the-in-store-experience/.

Vogel-Heuser, B. (2014) 'Herausforderungen und Anforderungen aus Sicht der IT und der Automatisierungstechnik', in Bauernhansl, T., Hompel, M. ten and Vogel-Heuser, B. (eds.) *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung, Technologien, Migration.* (SpringerLink). Wiesbaden: Springer Vieweg, pp. 37–48.

Vossel, C. (2017) 'Konzeption und prototypische Umsetzung einer Augmented-RealityLösung zur Unterstützung qualitätssichernder Maßnahmen in der industriellen Produktion', in Pinnow, C. and Schäfer, S. (eds.) *Industrie 4.0 - Safety und Security - mit Sicherheit gut vernetzt: Branchentreff der Berliner und Brandenburger Wissenschaft und Industrie.* (DIN). Berlin: Beuth Verlag GmbH, pp. 219–234.

Walters, A. (2015) *ChromaTags: An accurate, robust, and fast visual fiducial system*. Available at: https://austingwalters.com/chromatags/ (Accessed: 20 December 2018).

Westkämper, E. (2013a) 'Integration in der digitalen Produktion', in Westkämper, E., Spath, D. and Constantinescu, C. (eds.) *Digitale Produktion.* Berlin, Heidelberg: Springer Berlin Heidelberg.

Westkämper, E. (2013b) 'Zukunftsperspektiven der digitalen Produktion', in Westkämper, E., Spath, D. and Constantinescu, C. (eds.) *Digitale Produktion.* Berlin, Heidelberg: Springer Berlin Heidelberg.

Wölbert, C. (2015) *Fairphone 2: Android-Smartphone in Modulbauweise*. Available at: https://www.heise.de/newsticker/meldung/Fairphone-2-Android-Smartphone-in-Modulbauweise-2690732.html (Accessed: 20 November 2018).

Yin, L., Wu, X. and Haas, H. (2015) 'Indoor Visible Light Positioning with Angle Diversity Transmitter', *2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall), 2015 IEEE 82nd Vehicular Technology Conference (VTC Fall)*, Boston, MA, USA, 6-9 September: IEEE, pp. 1–5. doi: 10.1109/VTCFall.2015.7390984

Youssef, M. (2017) 'Indoor Localization', in Shekhar, S., Xiong, H. and Zhou, X. (eds.) *Encyclopedia of GIS.* Cham: Springer International Publishing, pp. 1004–1010.

Zadganokar, H. and Chandak, M. (2016) 'Object Localization Analysis Using BLE: Survey', in Unal, A., Nayak, M. and Mishra, D.K. (eds.) *Smart Trends in Information Technology and Computer Communications: First International Conference, SmartCom 2016, Jaipur, India, August 6–7, 2016, Revised Selected Papers.* (Communications in Computer and Information Science).

Zhang, H. *et al.* (2015) 'Localization and navigation using QR code for mobile robot in indoor environment', *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Zhuhai, 6-9 December: IEEE, pp. 2501–2506. doi: 10.1109/ROBIO.2015.7419715

Zhang, W., Chowdhury, M.I.S. and Kavehrad, M. (2014) 'Asynchronous indoor positioning system based on visible light communications', *Optical Engineering*, 53(4), p. 45105. doi: 10.1117/1.OE.53.4.045105

Zhang, X. *et al.* (2017) 'Stereo vision based autonomous robot calibration', *Robotics and Autonomous Systems*, 93, pp. 43–51. doi: 10.1016/j.robot.2017.04.001

Zhang, X., Fronz, S. and Navab, N. (2002) 'Visual marker detection and decoding in AR systems: A comparative study', *Proceedings. International Symposium on Mixed and Augmented Reality, IEEE and ACM International Symposium on Mixed and Augmented Reality*, Darmstadt, Germany, 30 Sept.-1 Oct. 2002: IEEE Comput. Soc, pp. 97–106. doi: 10.1109/ISMAR.2002.1115078

Zhang, Z. (2000) 'A flexible new technique for camera calibration', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11), pp. 1330–1334. doi: 10.1109/34.888718

Zillmann, M. (2016) *Smart Factory - Wie die Digitalisierung Fabriken verändert: Transformation von der Werkshalle bis zur Unternehmensleitung*. Mindelheim (Accessed: 7 June 2017).

Zsifkovits, H.E. and Altendorfer-Kaiser, S. (eds.) (2015) *Management logistischer Informationsflüsse: 3. Wissenschaftlicher Industrielogistik-Dialog in Leoben (WiLD)*. Mering: Rainer Hampp Verlag. Available at: https://www.wiso-net.de/document/EBOK,AEBO___9783957101426134.

Appendix A Source code for the realisation of the prototype

```python
#!/usr/bin/env python
# coding=utf-8

''''Demonstrate   Python wrapper of C apriltag library by running on camera frames.'''

from __future__ import division
from __future__ import print_function

from imutils.video import VideoStream

from video_cap import Capture_Video
from calibrate import Calibrate_Camera
from sshtunnel import SSHTunnelForwarder
from argparse import ArgumentParser

import cv2
import apriltag
import time
import pymongo
import pprint
import os
import numpy as np
import sys
import yaml
import imutils ######used for resizing the output frame###### PRUEFEN



MONGO_HOST = "134.103.112.214" #Variablen für dei Datenabnk
MONGO_DB = "dime"
MONGO_USER = "mongo"
MONGO_PASS = "mongo"

CAL_PARAM_FILE = "calibration_parameters.yml" #Variable für das leden der Kalibrierparameter

FRAME_WIDTH = 640   #
FRAME_HEIGHT = 480  #STANDARDEINSTELLUNGEN DER KAMERA
FPS = 15            #

DIST_THRESHOLD = 0.050 #Einstellung des Thresholds, bzw. des Abstandes ab welchem ein Positionswechsel erfasst wird

TAG_SIZE = 0.12 #Angabe der Taggröße // Werte werden in Meter ausgegeben

#ID_SIZE_MAPPING = {0: 0.12, 1 : 0.05} # Key value pair

DIME_IPS = "1001" #Angabe der ID des RaspberryPi
```

```


class IPS:
    def __init__(self,options):
        self.options = options
        self.init_camera()
        self.init_database_client()
        self.init_visualisation()
        self.init_apriltag_detector()

    def init_camera(self):
        self.vs = VideoStream(usePiCamera=True,
                              resolution=(self.options.width,self.options.height),
                              framerate=self.options.fps).start()
        time.sleep(2.0)

        #self.cap = cv2.VideoCapture(int(0)) #der Wert 0 steht für die PiCam / der Wert 1 steht für die Webcam
        #self.cap.set(cv2.CAP_PROP_FRAME_WIDTH, self.options.width);
        #self.cap.set(cv2.CAP_PROP_FRAME_HEIGHT, self.options.height);
        #self.cap.set(cv2.CAP_PROP_FPS, self.options.fps)

        ##self.cap.set(cv2.CAP_PROP_MODE, ) #backend-specific value indicating the current capture mode
        ##self.cap.set(cv2.CAP_PROP_BRIGHTNESS,  )
        ##self.cap.set(cv2.CAP_PROP_CONTRAST, )
        ##self.cap.set(cv2.CAP_PROP_SATURATION, )
        ##self.cap.set(cv2.CAP_PROP_EXPOSTURE, )

        self.threshold = self.options.threshold
        self.screenHeight = int(self.options.height) #Einstellung des Fadenkreuzes
        self.screenWidth = int(self.options.width) #Einstellung des Fadenkreuzes

    def init_database_client(self):
        self.ssh_server = SSHTunnelForwarder(
        MONGO_HOST,
        ssh_username=MONGO_USER,
        ssh_password=MONGO_PASS,
        remote_bind_address=('127.0.0.1', 27017))

        self.ssh_server.start()
        self.client = pymongo.MongoClient('127.0.0.1', self.ssh_server.local_bind_port)
        self.db = self.client[MONGO_DB]

    def init_visualisation(self):
        self.window_name = 'IPS'
        cv2.namedWindow(self.window_name)

    def init_apriltag_detector(self):
        self.tag_size = self.options.tag_size
        self.detected_tags = {}
        self.detected_tags_old = {}
        self.params = self.read_calibration_yaml()
        self.detector = apriltag.Detector(searchpath=[os.getcwd()])

    # This function returns true if
    def check_for_position_change(self,tag_id,current_position):

        try:
```

```
            dist = np.linalg.norm(self.detected_tags_old[tag_id]-self.detected_tags[tag_id]) #berechnungsgrundlage für *1
            print("Distance : ",dist) #absolute distanbe between the last measured and current position *1

            if dist > self.threshold:
                print("Moved update DB")
                return True

            else:
                return False

        except Exception as e:
            print("Tag id {} is new ",tag_id)
            return True


    def run_detection_loop(self):

        while True:

            start = time.time()

            try:
                frame = self.vs.read()
                ##frame = imutils.resize(frame, width=400) #resizing the frame
                print(frame.shape)

            except:
                continue

            gray = cv2.cvtColor(frame, cv2.COLOR_RGB2GRAY) #Greyscaling

            detections, dimg = self.detector.detect(gray, return_image=True) #Erkennung wird durchgefuehrt
            end = time.time()
            time_duration = end-start
            print ("Time duration {}".format(time_duration)) #gibt an wie lange die Ermittlung der Koordinaten eines Tags gebraucht hat

            num_detections = len(detections)
            print('Detected {} tags.\n'.format(num_detections))

            if self.options.precal == 1:                                 #wird gebraucht

                overlay = frame // 2 + dimg[:, :, None] // 2    #um die Einstellung des
            else:                                                        #Messboard's vornehmen
                overlay = frame                                          #zu können

            cv2.line(overlay,(0, self.screenHeight//2), (self.screenWidth, self.screenHeight//2),(0,0,255),1) #zeichnet das rote Linie auf der x-Achse
            cv2.line(overlay,(self.screenWidth//2, 0), (self.screenWidth//2, self.screenHeight),(0,0,255),1)  #zeichnet eine rote Linie auf der y-Achse
            apriltag._draw_crosshair(overlay, self.params, self.tag_size)   #zeichnet ein Fadenkreuz, dessen ausgangspunkt der Mittelpunkt der Kamera ist

            for i, detection in enumerate(detections):#iterating through each detection, Die erkannten Codes werden der Reihenfolge nach abgearbeitet und deren Pose bestimmt
```

```

                    pose,e1,e2 = self.detector.detection_pose(detection,self.params,self.tag_size)#Berechnung der pose jeder einzelnen detection (Erkennungen)
                    #pose,e1,e2 = self.detector.detection_pose(detection,self.params,ID_TAG_MAPPING[detection.tag_id])  #Use this for multiple tag sizes in same picture,
                                 #But define tag_id -> size mapping in ID_SIZE_MAPPING variable
                    trans = pose[0:3,3]

                    position =  np.array(trans)

                    print("Postion X: {} , Y: {} , Z: {}".format(position[0],position[1],position[2]))

                    if self.options.precal == 1:
                        apriltag._draw_pose(overlay,self.params,self.tag_size,pose,detection.tag_id) #Einzeichnen des Framworks und des Textes im LIvebild


                    print ("Detected Tag ID : ",detection.tag_id)#Ausgabe der erkannten Marker ID
                    self.detected_tags[detection.tag_id] = position
                    post = {
                        "Device_number": DIME_IPS, #Ausgabe der fest vergebenen IP des Gerätes
                        "Code_id": detection.tag_id,
                        "Timestamp": str(int(round(time.time() * 1000))),
                        "Time_duration": str(time_duration),
                        "x": position[0],
                        "y": position[1],
                        "z": position[2],
                        }

                    result = self.check_for_position_change(detection.tag_id,position)

                    if result:
                        print("Updating Datenbank")
                                                    ###
                        self.db.ips_2.insert(post)  ### < hier wird angegeben, in welche collection meienr Datenbank geschrieben wird >###
                                                    ###

                    self.detected_tags_old[detection.tag_id] = position



                cv2.imshow(self.window_name, overlay)
                k = cv2.waitKey(1)
                if k == ord('q'):
                    break
            self.destroy()

        def read_calibration_yaml(self):
            with open(CAL_PARAM_FILE, 'r') as fr:
                calibration = yaml.safe_load(fr)
                print (calibration['camera_matrix'][0])
                fx = calibration['camera_matrix'][0][0]
                fy = calibration['camera_matrix'][1][1]
                cx = calibration['camera_matrix'][0][2]
                cy = calibration['camera_matrix'][1][2]
```

```
                cam_params = [fx,fy,cx,cy]
                print("params fx,fy,cx,cy :",cam_params)
                return cam_params
        def destroy(self):
            print("Releasing all the resources")
        self.cap.release()

        cv2.destroyAllWindows()
        sys.exit(0)


    if __name__ == '__main__':
        parser = ArgumentParser(description='Indoor positioning system help')


        parser.add_argument('--width', metavar='width',type=int, nargs='?',  default=FRAME_WIDTH,       # übergabeparameter für die
                            help='width of frame to be captured')                  # horizontale-Pixeleinstellung
        parser.add_argument('--height', metavar='height',type=int, nargs='?', default=FRAME_HEIGHT,         # übergabeparameter für die
                            help='height of frame to be captured')                     # vertikale-Pixeleinstellung
        parser.add_argument('--fps', metavar='fps',type=int, nargs='?', default=FPS,    # übergabeparameter für die
                            help='frame rate to be captured')  # FPS einstellung
        parser.add_argument('--threshold', metavar='threshold',type=float, nargs='?', default=DIST_THRESHOLD,   # übergabeparameter für die Einstellung des Thresholds, dieser wird
                            help='distance threshold for controlling the databank upload rate')                 # genutzt um nur ueue Positionen in die Datenbank zu schreiben

        parser.add_argument('--tag_size', metavar='tag_size',type=float, nargs='?', default=TAG_SIZE,   # übergabeparameter für die Einstellung der
                            help='size of the apriltag')                  # verwendeten Markergröße
        parser.add_argument('--capture', metavar='capture',type=int, nargs='?', default=0,  # übergabeparameter für den Start der
                            help='duration for video capture for calibration')      # Kalibrierung
        parser.add_argument('--precal', metavar='precal',type=int, nargs='?', default=1,        # übergabeparameter für die Einstellung des
                            help='enable or disble visualisation  for precalbration')          # Messboards (die Markererkennung wird hierfür ausgeschaltet)
                  # wird der Wert 1 übergeben, ist die Messung aktiv


        options = parser.parse_args()
        print((options))

        if (options.capture>0):
            Capture_Video(options.capture, options.width, options.height)
            Calibrate_Camera()
            sys.exit(0)
```

```
        ips_1 = IPS(options)

        ips_1.run_detection_loop()
```

Appendix B Overview and classification of the application scenarios

Application scenarios and their characteristics for use in SMEs

Application scenarios	Architecture	Required infrastructure	Accuracy	Coverage	Frame rate	Update rate	Latency	Number of objects	Scalability	Computing power	Power supply	Output data	Accuracy class	Classes
Automation in production lines	outside-in	markers	< 1 cm	< 5 m	≥ 15 fps	periodically	< 1 s	≥ 50	medium	low	active	xy position, ID	1	Class 1
Inventory management	outside-in	markers	< 10 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	low	active	xy position, ID	2	
Inventory control	outside-in	markers	< 10 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	low	active	xyz position, ID		
Quality assurance (Reduction of picking errors)	outside-in	markers	< 10 cm	< 1 m	< 15 fps	periodically	< 1 s	≥ 50	low	low	active	xy position, ID		
Identification work station	outside-in	markers	< 10 cm	< 1 m	< 15 fps	periodically	< 1 s	≥ 50	low	low	active	xy position, ID		
Operational optimisation	outside-in	markers	< 50 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	low	active	xy position, ID	3	
Tracking of material flows, load carriers or parts	outside-in	markers	< 50 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	low	active	xy position, ID		
Management of machines and vehicles	outside-in	markers	< 50 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	low	active	xy position, ID		
Automatic fabric recognition at goods receipt	outside-in	markers	< 50 cm	< 5 m	< 15 fps	event-driven	< 1 s	≥ 50	low	low	active	xy position, ID		
Component search	outside-in	markers	< 50 cm	< 10 m	< 15 fps	on request	< 1 s	≥ 50	high	low	passive	xy position, ID		
Optimisation of routes	outside-in	markers	< 100 cm	< 5 m	≥ 15 fps	periodically	< 1 s	≤ 50	medium	low	passive	xy position, ID	4	
Access control	outside-in	markers	< 100 cm	< 5 m	< 15 fps	periodically	< 1 s	≤ 10	low	low	active	undefined		
Automation in production lines	outside-in	none	< 1 cm	< 5 m	≥ 15 fps	periodically	< 1 s	≥ 50	medium	high	active	xy position, ID	1	Class 2
Identification work station	outside-in	none	< 10 cm	< 1 m	< 15 fps	periodically	< 1 s	≥ 50	low	high	active	xy position, ID	2	
Intelligent tool control	outside-in	none	< 10 cm	< 1 m	< 15 fps	event-driven	< 1 s	≤ 10	medium	high	active	xyz position		
Operational optimisation	outside-in	none	< 50 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	high	active	xy position, ID	3	
Component search	outside-in	none	< 50 cm	< 10 m	< 15 fps	on request	< 1 s	≥ 50	high	high	passive	xy position, ID		
Tracking of material flows, load carriers or parts	outside-in	none	< 50 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	high	active	xy position, ID		
Management of machines and vehicles	outside-in	none	< 50 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	high	active	xy position, ID		
Route inspection	outside-in	none	< 50 cm	< 5 m	< 15 fps	event-driven	< 15 s	≥ 50	high	high	passive	xyz position		
Optimisation of routes	outside-in	none	< 100 cm	< 5 m	≥ 15 fps	periodically	< 1 s	≤ 50	medium	high	passive	xy position, ID	4	
Access control	outside-in	none	< 100 cm	< 5 m	< 15 fps	periodically	< 1 s	≤ 10	low	high	active	undefined		
Anti-theft protection	outside-in	none	< 100 cm	< 1 m	< 15 fps	periodically	< 1 s	≤ 10	low	high	active	undefined		
Navigation and localisation of mobile robots	inside-out	markers	< 10 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≤ 10	high	low	passive	xy position	1	Class 3
Management of machines and vehicles	inside-out	markers	< 50 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	low	active	xy position, ID	2	
Tracking of intralogistics transport vehicles	inside-out	markers	< 50 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≤ 10	high	low	passive	xy position, ID		
Optimisation of routes	inside-out	markers	< 100 cm	< 5 m	≥ 15 fps	periodically	< 1 s	≤ 50	medium	low	passive	xy position, ID	3	
Navigation and localisation of mobile robots	inside-out	none	< 10 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≤ 10	high	high	passive	xy position	1	Class 4
Intelligent tool control	inside-out	none	< 10 cm	< 1 m	< 15 fps	event-driven	< 1 s	≤ 10	medium	high	aktive	xyz position		
Management of machines and vehicles	inside-out	none	< 50 cm	< 10 m	≥ 15 fps	periodically	< 1 s	≥ 50	high	high	aktive	xy position, ID	2	
Optimisation of routes	inside-out	none	< 100 cm	< 5 m	≥ 15 fps	periodically	< 1 s	≤ 50	medium	high	passive	xy position, ID	3	
Anti-theft protection	inside-out	none	< 100 cm	< 1 m	< 15 fps	periodically	< 1 s	≤ 10	low	high	aktive	undefined		

Appendix C Analysis of Areas of Application and used Technologies

Reference	Area of Application	Coordinate Reference	Architecture
(TotalTrax, 2018)	Logistics	Reference from Deployed Coded Targets	Self-positioning
(Zhang *et al.*, 2015)	Robotics	Reference from Deployed Coded Targets	Self-positioning
(Debski *et al.*, 2015)	Robotics	Reference from Deployed Coded Targets	Self-positioning
(Mutka *et al.*, 2008)	Robotics	Reference from Deployed Coded Targets	Self-positioning
(Hagisonic, 2016)	Robotics	Reference from Deployed Coded Targets	Self-positioning
(Babinec *et al.*, 2014)	Robotics	Reference from Deployed Coded Targets	Self-positioning
(Hohenstein, 2014)	Logistics	Reference from Deployed Coded Targets	Self-positioning
(Lee and Song, 2007)	Robotics	Reference from Deployed Coded Targets	x
(Tilch and Mautz, 2013)	Unconstraint	Reference from Projected Targets	Self-positioning

Reference	Area of Application	Coordinate Reference	Architecture
(Kohoutek, Mautz and Wegner, 2013)	Unconstraint	Reference from 3D Building Models	Self-positioning
(Boochs *et al.*, 2010)	Robotics	Systems without Reference	External positioning
(Barberis *et al.*, 2014)	Unconstraint	Reference from Deployed Coded Targets	Self-positioning
(Mustafah, Azman and Akbar, 2012)	Robotics	Systems without Reference	External positioning
(Zhang *et al.*, 2017)	Robotics	Reference from Deployed Coded Targets	External positioning
(Saifizi, Hazry and M. Nor, 2012)	Robotics	Reference from Deployed Coded Targets	x
(Liao *et al.*, 2013)	Robotics	Reference from Deployed Coded Targets	x
(Ferrara *et al.*, 2017)	Unconstraint	Reference from Deployed Coded Targets	External positioning
(Bergen *et al.*, 2017)	Unconstraint	Reference from Deployed Coded Targets	Self-positioning

Appendix D Camera comparison

Type	**Raspberry cameras**	**Raspberry cameras**
	Raspberry Pi Camera V1.3	RPI WWCAM
Camera type	Area scan camera	Area scan camera
Sensor type	CMOS	CMOS
Sensor size	1/4"	1/4"
Mount	M12x0,5	M12x0,5
Resolution	5	5
Frame rate/Resolution	2952 x 1944 / 15 fps	2952 x 1944 / 15 fps
	1920×1080 / 30 fps	1920×1080 / 30 fps
Shutter type	Rolling	Rolling
Power supply	Integrated	Integrated
Interface	CSI	
Software	free	
Price [€]	10	30
Lens		
f	3.6	1.67
AOV (D)	64.01	169.40
FOV (D)		
Price [€]		
Costs [€]	**10**	**30**

Raspberry cameras	**Raspberry cameras**	**Raspberry cameras**
Arducam	Arducam	Arducam camera body
Area scan camera	Area scan camera	Area scan camera
CMOS	CMOS	CMOS
1/4"	1/4"	1/4"
CS	CS	CS
5	5	5
2952 x 1944 / 15 fps	2952 x 1944 / 15 fps	2952 x 1944 / 15 fps
1920×1080 / 30 fps	1920×1080 / 30 fps	1920×1080 / 30 fps
Rolling	Rolling	Rolling
Integrated	Integrated	Integrated
CSI	CSI	CSI
free	free	free
15	15	15
LS-2716	0612-3MP-A	CS-2,8-3MP 140
4	6	2.8
58.72	41.11	77.57
		16
15	**15**	**31**

Webcam	**Industrial cameras**	**Industrial cameras**
Logitech C920	BFS-U3-122S6M-C	BFS-U3-88S6M-C
Area scan camera	Area scan camera	Area scan camera
CMOS	CMOS	CMOS
	1.1"	1"
	C	C
3	12.3	8.9
1920×1080 / 30 fps	4096 x 3000 / 23 fps	4096 x 2160 / 32 fps
	Global	Global
Integrated	Integrated	Integrated
USB 2.0	USB 3.0	USB 3.0
free		
69,94	2295	1895
	M111FM08	eo-Lens
3.67	8	12.5
78.00	83.50	88.60
	590	595
69.94	**2885.00**	**2490.00**

Industrial cameras	**Industrial cameras**	**Industrial cameras**
GS3-U3-41 C6C-C 3.0	UI-3280CP-C-HQ Rev.2	Basler ace acA2440-35um
Area scan camera	Area scan camera	Area scan camera
CMOS	CMOS	CMOS
1"	2/3"	2/3"
C	C	C
4.2	5.04	5.00
2048x2048 / 90 fps	2456 x 2054 / 36 fps	2448 x 2048 / 35 fps
Global	Global	Global
	Integrated	Integrated
USB 2.0	USB 3.0	USB 3.0
1450	714	950
V0814-MP	HF6XA-5M	
8	6	
79.68	85.02	
390	208.25	
1840	**922.25**	**950**

Industrial cameras	**Industrial cameras**	**Industrial cameras**
UI-1240LE-C-HQ	UI-3240LE-C-HQ	UI-3270LE-C-HQ
Area scan camera	Area scan camera	Area scan camera
CMOS	CMOS	CMOS
1/1.8"	1/1.8"	1/1.8"
C/CS	C/CS	C/CS
1.31	1.31	3.17
		2056 x 1542 / 57 fps
1280 x 1024 25,8 fps	1280 x 1024 60 fps	
Global	Global	Global
Integrated	Integrated	Integrated
USB 2.0	USB 3.0	USB 3.0
283.9	280.5	433.5
M118FM06	-	BM4018S118
6	-	4
66.10	-	94.72
-	-	
219		69
502.9	**280.5**	**502.5**

Industrial cameras	**Industrial cameras**	**Industrial cameras**
caspa	Basler ace acA1920-25um	MER-030-120UM
Area scan camera	Area scan camera	Area scan camera
	CMOS	CCD
1/3"	1/3.7"	1/4"
	C	C
,36	2	,32
752 × 480 / 60 fps	1920 x 1080 / 25 fps	656 x 492/120 fps
	Rolling	
	External	Integrated
Camera Flex Ribbon Cable	USB 3.0	USB 2.0
75	375	2302.93
		Computar H0514-MP2
		3.5
		65.47
		289.95
75	**375**	**2592.88**

Appendix E Literature-based derivation of requirement aspects

	Localization function	
	Continuity	Output Data
(TotalTrax, 2018)	x	xy position, alignment, speed
(Zhang *et al.*, 2015)	x	xy position, alignment
(Debski *et al.*, 2015)	x	xy position, alignment
(Mutka *et al.*, 2008)	x	xy position, alignment, coded data
(Hagisonic, 2016)	x	xy position, alignment, coded data
(Babinec *et al.*, 2014)	x	xy position, alignment, coded data
(Hohenstein, 2014)	x	xy position
(Lee and Song, 2007)	x	xy position
(Tilch and Mautz, 2013)	x	xy position, alignment
(Kohoutek, Mautz and Wegner, 2013)	x	xyz position, alignment
(Boochs *et al.*, 2010)	x	xyz position, alignment
(Barberis *et al.*, 2014)	x	xy position
(Mustafah, Azman and Akbar, 2012)	x	xyz position
(Zhang *et al.*, 2017)	x	xyz position
(Saifizi, Hazry and M. Nor, 2012)	x	xy position, alignment
(Liao *et al.*, 2013)	x	xy position
(Ferrara *et al.*, 2017)	x	xyz position, alignment
(Bergen *et al.*, 2017)	x	xyz position

Localisation Performance			
Accuracy	Range		Repeat Accuracy
	Category	[m^2]	
< 5 cm	Scalable Coverage	x	x
< 10 cm	Scalable Coverage	3	x
< 5 cm	Scalable Coverage	x	x
x	Scalable Coverage	x	x
< 5 cm	Scalable Coverage	< 10	x
x	Scalable Coverage	x	x
< 50 cm	Scalable Coverage	< 10	99,7%
< 50 cm	Local Coverage System	36	x
< 5 cm	Scalable Coverage	x	x
< 50 cm	Local Coverage System	4,5	x
mm	Local Coverage System	4	x
< 10 cm	Scalable Coverage	x	x
mm	Local Coverage System	x	x
< 5 cm	Local Coverage System	x	x
< 30 cm	Scalable Coverage	x	x
< 5 cm	Scalable Coverage	x	x
< 5 cm	Scalable Coverage	x	x
< 2 cm	Scalable Coverage	x	x

			Requirements towards ILS	
Update Rate			Scalability	
				Objects
periodical	x	in real-time	x	x
periodical	< 1s	in real-time	x	x
periodical	x	x	x	x
periodical	2- 3 s	x	x	x
periodical	< 1s	x	x	x
x	x	x	x	x
periodical	< 1 s	in real-time	range depended	< 100
periodical	x	x	x	x
periodical	30 Hz	x	x	x
periodical	x	near real time	x	x
periodical	x	x	x	x
on request	x	in real-time	x	x
periodical	< 10 HZ	in real-time	x	x
periodical	x	x	x	x
periodical	< 1s	in real-time	x	x
periodical	< 1s	in real-time	x	x
periodical	x	in real-time	x	x
periodical	x	x	x	x

(Indoor Location Systems)				
Availability	Infrastructure		Robustness	Security
x	2D-landmarks	x	LOS, Illumination	x
x	2D-landmarks	12x12 cm	LOS, Illumination	x
x	2D-landmark, IR-reflecting	x	LOS	x
x	2D-landmarks (LARICS)	15 cm, r	LOS, Illumination	x
x	2D-landmarks, IR-reflecting	x	LOS	x
x	2D-landmarks (ArUco)	16x16 cm	LOS, Illumination	x
x	2D-landmarks	x	LOS	x
x	2D-landmark, IR-reflecting	x	LOS	x
x	Projected laser reference points	x	LOS, reflection	x
x	Natural features	x	LOS	x
x	Active LED system	x	LOS, temperature	x
x	2D-landmarks	8x8 cm	LOS	x
x	Natural features	x	LOS	x
x	2D-Marker	x	LOS, occlusion	x
x	Circular-landmarks	x	LOS	x
x	Natural features	x	LOS, changing environments	x
x	2D-marker	40x60 cm	LOS	x
x	Active LED system	x	LOS, reflection	x

Appendix F Case and mounting constructions

Appendix F contains displays the case and mounting construction of the case for the Raspberry Pi 3 B+ and the Raspberry Pi Camera (as well as the Arducam camera body. On the left side the of Figure 43 the top of the cover can be seen. The two holes on the outer centre are used to attach the case to a celling, for this purpose standard screws can be used. Both the spacers as well as the screws are used to combine the Raspberry Pi 3 B+ with the bottom part. The spacers to the right of the top part of the case are used to mount the Raspberry Pi and to provide the necessary space for the mounting of the top part of the cover (right).

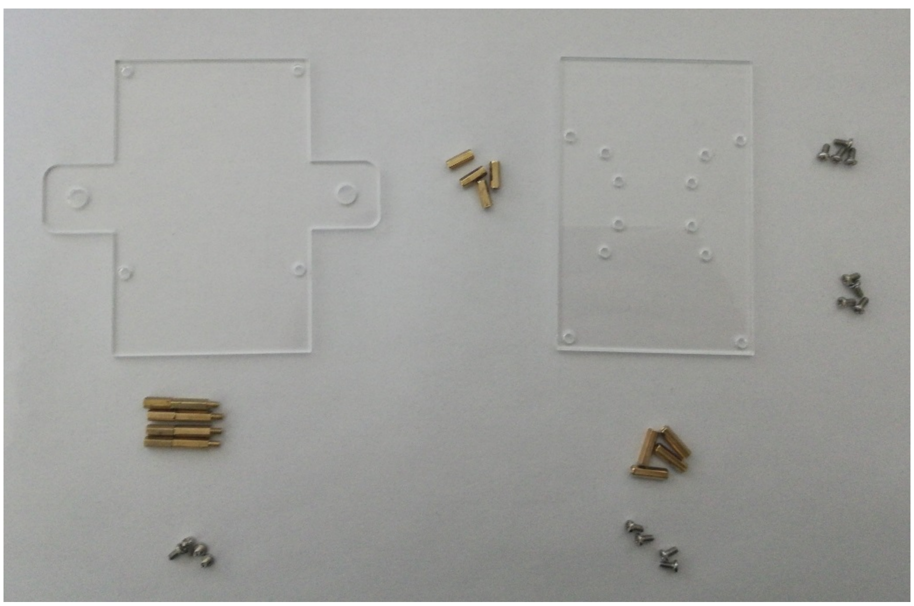


Figure 43 Top (left) and bottom (right) cover piece for the Raspberry Pi 3 B+

On the left side the bottom piece is shown, which is used to mount the camera body to the top part of the case. The spacers are again used to provide the necessary space between the hardware components. Figure 44 shows how the parts are fitting together.

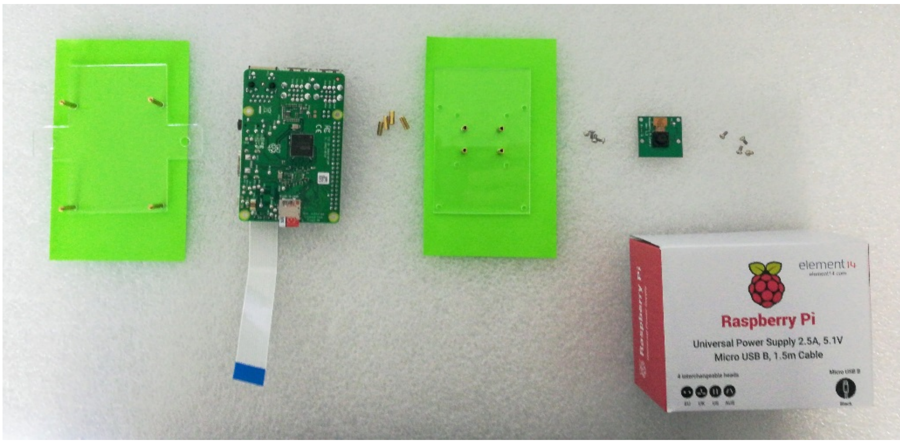


Figure 44 Cover and hardware components, shown in the order in which they are mounted.