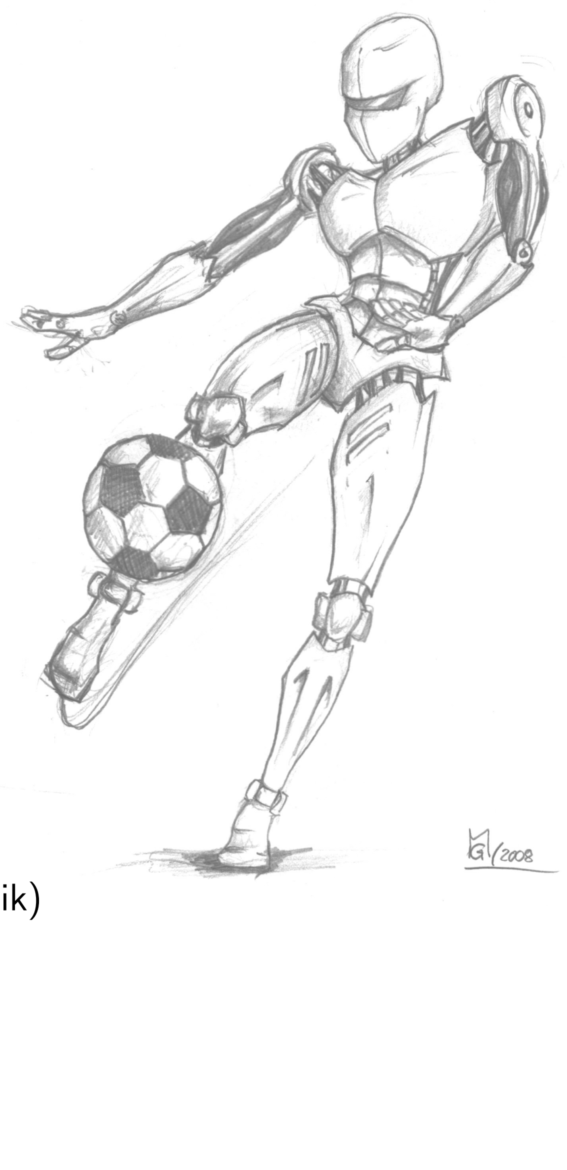
Mobile Robots in Dynamic Environments – State Estimation and Simulation

Tim Laue



Kumulative Dissertation
zur Erlangung des Grades eines
Doktors der Ingenieurwissenschaften – Dr.-Ing. –

Vorgelegt im Fachbereich 3 (Mathematik und Informatik)
Universität Bremen

18. Oktober 2010

Datum des Promotionskolloquiums: 20. Dezember 2010

Gutachter

Prof. Dr. Bernd Krieg-Brückner (Universität Bremen)
Prof. Dr. Hans-Dieter Burkhard (Humboldt Universität zu Berlin)

Cover illustration courtesy of Martin Görner, 2008

Abstract

This thesis presents works addressing scientific challenges of the domain of autonomous mobile robots in general and the RoboCup competition in particular. The two major areas of research are state estimation in uncertain, dynamic environments as well as the simulation of these environments.

Different efficient, robust, and thus practically usable approaches for state estimation have been developed, mainly for self-localization but also for tracking objects in the robot's environment. In general, compensation for very limited computing resources and sensor equipment that is far from perfect is sought. Furthermore, some domains – such as the RoboCup competition that served as testbed for most approaches presented in this thesis – demand the robots to operate at their limits, causing additional uncertainties that need to be coped with. These are the major scientific challenges of this work.

The second focus of this work is the simulation of mobile robots and their environment. For this purpose, a modernized version of the simulator SimRobot has been developed, regarding graphics and the modeling of dynamics. Further work encompasses the simulation of common artifacts in images of fast moving cameras as well as the learning of parameters determining the simulated world's physical behavior to realistically execute robot motions. These approaches are the virtual counterparts to the state estimation's sensor and motion models.

Especially in RoboCup, but also in other areas of science, progress is achieved in small, iterative steps. In robotics, this is often attributed to the performance of the available resources. In the course of the development of the major contributions of this thesis, a series of mostly theoretical works about the long-term aims and developments in robot soccer has been published. These works are presented in a prefacing chapter for the purpose of a broader introduction into the RoboCup domain.

The thesis closes with applications of the presented approaches beyond the scope of RoboCup. Different works regarding state estimation as well as simulation have been successfully applied in the areas of rehabilitation robotics as well as Ambient Assisted Living.

In addition to publishing at conferences, workshops, and in journals, the public release of software and data is an important aspect of scientific work as it enables others to comprehend the results and to base their own works upon them. Therefore, the appendix contains – in addition to all publications – the descriptions of all releases of software components that have been developed in the course of this thesis.

Zusammenfassung

Die vorliegende Dissertation präsentiert Arbeiten, die sich wissenschaftlichen Herausforderungen der Domäne autonomer, mobiler Roboter im Allgemeinen sowie der RoboCup-Wettbewerbe im Besonderen widmen. Die zwei Hauptthemenbereiche sind die Schätzung von Zuständen in unsicheren, dynamischen Umgebungen sowie die Simulation dieser Umgebungen.

Es wurden verschiedene effiziente, robuste und dadurch anwendbare Ansätze zur Zustandsschätzung, hauptsächlich zur Selbstlokalisierung, aber auch zur Modellierung anderer Objekte in der Umgebung eines Roboters entwickelt. Im Allgemeinen müssen dabei stark limitierte Ressourcen sowohl bzgl. der sensoriellen Ausstattung als auch der Rechenleistung der Roboter kompensiert werden. Des Weiteren werden die Roboter in einigen Domänen – wie beispielsweise in Wettbewerben wie dem RoboCup, der als Testumgebung für die meisten der hier präsentierten Arbeiten dient – im Grenzbereich betrieben, so dass zusätzliche Unsicherheiten eine Herausforderung darstellen. Dies sind die zentralen wissenschaftlichen Herausforderungen dieser Arbeit.

Der zweite Schwerpunkt dieser Arbeit ist die Simulation von mobilen Robotern und deren Umgebungen. Hierzu wurde eine hinsichtlich Grafik und physikalischer Modellierung zeitgemäße Version des Simulators SimRobot entwickelt. Weitergehende Arbeiten umfassen die Simulation weit verbreiteter Artefakte in Bildern schnell bewegter Kameras sowie das Lernen von Parametern der physikalischen Beschaffenheit der Welt zur realitätsnahen Ausführung von Roboterbewegungen. Sie sind die virtuellen Gegenstücke zu den Sensor- und Bewegungsmodellen der Zustandsschätzung.

Insbesondere im RoboCup, aber auch in anderen Bereichen der Wissenschaft, vollzieht sich der Fortschritt in kleinen, iterativen Schritten. In der Robotik ist dies oftmals bedingt durch die Leistungsfähigkeit der zur Verfügung stehenden Ressourcen. Während der Entwicklung der in dieser Dissertation vorgestellten Ansätze ist eine Reihe von schwerpunktmäßig theoretischen Arbeiten entstanden, die sich mit langfristigen Zielen und Entwicklungen des Roboterfußballs auseinandersetzen. Im Rahmen einer breiteren Einführung in die Domäne RoboCup werden sie in einem einleitenden Kapitel vorgestellt.

Die Arbeit schließt mit Anwendungen außerhalb des RoboCups, in den Bereichen Rehabilitationsrobotik sowie Ambient Assisted Living. Einzelne Arbeiten aus den Bereichen Zustandsschätzung und Simulation konnten erfolgreich übertragen werden.

Neben dem Publizieren bei Konferenzen, Workshops und in Zeitschriften ist zudem die Veröffentlichung von Daten und Software ein wichtiger Aspekt wissenschaftlicher Arbeit, da somit eine direkte Nachvollziehbarkeit gewährleistet werden kann und die Basis für weiterführende Arbeiten anderer Forscher geschaffen wird. Neben allen Publikationen sind daher die Beschreibungen der Veröffentlichungen der durch diese Arbeit entstandenen Software-Komponenten angehängt.

Contents

1 Introduction 1
1.1 Autonomous Mobile Robots . . . 1
1.1.1 A Short State of the Art . . . 2
1.1.2 Uncertainties in Robot State Estimation . . . 3
1.1.3 Robot Simulations and the Reality Gap . . . 5
1.1.4 Robot Competitions . . . 6
1.2 RoboCup – Dynamic Testbed for Autonomous Systems . . . 7
1.2.1 Constitution and Objectives . . . 8
1.2.2 RoboCup Domains Today . . . 8
1.2.3 RoboCup Soccer in Detail . . . 9
1.2.4 The Scientific Significance of RoboCup . . . 11
1.3 Contributions of this Dissertation . . . 12

2 General Contributions to RoboCup 15
2.1 Participation in RoboCup Competitions . . . 15
2.1.1 Standard Platform League – Four-legged . . . 15
2.1.2 Humanoid League . . . 16
2.1.3 Standard Platform League – Two-legged . . . 17
2.1.4 Small Size Robot League . . . 17
2.2 Inter-League Transfer . . . 18
2.3 (A) Vision for 2050 . . . 18
2.3.1 Challenges and State of the Art . . . 19
2.3.2 Robustness through Context . . . 20
2.3.3 Experimental Evaluation . . . 21
2.4 Physical Interaction in Human-Robot Soccer . . . 22
2.4.1 Safety for Humans and Robots . . . 22
2.4.2 Increased Kicking Performance Through Joint Elasticity . . . 23
2.5 Contributions of the Corresponding Publications . . . 23

3 State Estimation 25
3.1 Self-Localization in RoboCup – Prerequisites . . . 25
3.1.1 Robot Platforms . . . 25
3.1.2 The Environment . . . 26
3.1.3 Vision Systems . . . 27
3.2 State of the Art . . . 28
3.2.1 Probabilistic Filters . . . 29
3.2.2 Self-Localization in RoboCup . . . 29

3.2.3 Monte-Carlo Localization 30
3.3 Sensor Models 31
3.3.1 Multi-Cue Localization 32
3.3.2 State-based Sensor Models 33
3.3.3 Particle Depletion and Sensor Resetting 34
3.3.4 Parameter Optimization 34
3.4 Pose Extraction 35
3.5 Applications for Multi-Modal Robot Pose Estimates 36
3.5.1 Entropy-based Active Vision 36
3.5.2 Potential Field-based Behavior Specification 37
3.6 Evaluation of Self-Localization 37
3.6.1 Ground Truth 37
3.6.2 Quantitative Evaluation 38
3.6.3 Qualitative Evaluation 39
3.7 Object Tracking 40
3.7.1 Ball Tracking 40
3.7.2 Robot Tracking 40
3.7.3 Ultrasound Obstacle Modeling 41
3.8 Contributions of the Corresponding Publications 42

4 Simulation **45**
4.1 State of the Art 45
4.2 Development of SimRobot 46
4.2.1 History 47
4.2.2 Current State 48
4.2.3 The RoSiML Language 49
4.3 Simulating Image Distortions 49
4.4 Excursion: Motion Design for Walking Robots 50
4.5 Optimizing Simulation Parameters 51
4.6 Contributions of the Corresponding Publications 52

5 Beyond RoboCup **55**
5.1 Rehabilitation Robots 55
5.1.1 Rolland – A Smart Wheelchair 55
5.1.2 iWalker – An Intelligent Walker 56
5.2 Indoor Navigation 56
5.2.1 Mapping and Localization 57
5.2.2 Evaluation Support for Shared Control Experiments 58
5.2.3 Autonomous Navigation and Way-finding Assistance 58
5.3 Outdoor Localization in Road Networks 59
5.4 Simulating an Ambient Assisted Living Environment 60
5.4.1 The Bremen Ambient Assisted Living Lab 60
5.4.2 Interactive Evaluation of Configurations 61
5.5 Contributions of the Corresponding Publications 61

6 Conclusion **63**

List of Publications by the Author **65**

References **75**

A Released Software **89**
A.1 Released RoboCup Code 89
A.1.1 The GermanTeam 89
A.1.2 B-Human 90
A.2 SimRobot 90
A.3 SSL-Vision 90
A.4 GMapping Localization Library 91

B Accumulated Publications **93**
B.1 (A) Vision for 2050: The Road Towards Image Understanding for a Human-Robot Soccer Match 94
B.2 (A) Vision for 2050 – Context-Based Image Understanding for a Human-Robot Soccer Match 101
B.3 Tracking of Ball Trajectories with a Free Moving Camera-Inertial Sensor . 120
B.4 Foul 2050: Thoughts on Physical Interaction in Human-Robot Soccer . . . 133
B.5 Kick it with Elasticity: Safety and Performance in Human-robot Soccer . . 141
B.6 Getting Upright: Migrating Concepts and Software from Four-Legged to Humanoid Soccer Robots 157
B.7 Particle Filter-Based Self-Localization Using Landmarks and Directed Lines 164
B.8 Particle Filter-based State Estimation in a Competitive and Uncertain Environment 173
B.9 Efficient and Reliable Sensor Models for Humanoid Soccer Robot Self-Localization 185
B.10 Optimizing Particle Filter Parameters for Self-Localization 193
B.11 Pose Extraction from Sample Sets in Robot Self-Localization - A Comparison and a Novel Approach 205
B.12 SSL-Vision: The Shared Vision System for the RoboCup Small Size League 211
B.13 Entropy-based Active Vision for a Humanoid Soccer Robot 223
B.14 A Behavior Architecture for Autonomous Mobile Robots Based on Potential Fields 235
B.15 Integrating Simple Unreliable Perceptions for Accurate Robot Modeling in the Four-Legged League 247
B.16 SimRobot - A General Physical Robot Simulator and its Application in RoboCup 255
B.17 Real-time Simulation of Motion-based Camera Disturbances 266
B.18 Automatic Parameter Optimization for a Dynamic Robot Simulation . . . 278
B.19 SimRobot - Development and Applications 291
B.20 Gait Optimization on a Humanoid Robot using Particle Swarm Optimization 299
B.21 A Robust Closed-Loop Gait for the Standard Platform League Humanoid 306
B.22 Kicking a Ball – Modeling Complex Dynamic Motions for Humanoid Robots 314

B.23 Controlling an Automated Wheelchair via Joystick/Head-Joystick Supported by Smart Driving Assistance 327
B.24 Navigating a Smart Wheelchair with a Brain-Computer Interface Interpreting Steady-State Visual Evoked Potentials 333
B.25 iWalker - An Intelligent Walker providing Services for the Elderly 341
B.26 Particle Filter-based Position Estimation in Road Networks using Digital Elevation Models 345
B.27 Modeling and Simulating Ambient Assisted Living Environments – A Case Study 351

Chapter 1

Introduction

After the early beginnings in the 1960s, industrial robots have already become a standard tool for factory automation. Being situated in well-defined, fixed environments, executing precisely defined tasks, they are able to operate efficiently without the need for sophisticated sensing capabilities and self-generated models of their environment. In contrast, mobile robots still have not reached this level of applicability and distribution. They are rather a subject of intensive research than a product, particularly when operating in imprecisely perceivable environments that might include previously unknown or even moving objects.

This chapter provides a short overview about the current state of the art of mobile robots and the specific problems – in particular regarding uncertainty – that need to be addressed. In addition, a popular testbed for autonomous mobile robots – the *RoboCup* – is presented as it is the domain in which most works of this dissertation are settled. Finally, the dissertation's content is outlined shortly and set in relation to the state of the art.

1.1 Autonomous Mobile Robots

Current actual mobile robots perform a variety of tasks, use different kinds of actuators for motion, and differ in their sensorial equipment. Especially the latter directly influences the complexity of the tasks a system can perform. Sensor measurements are used to compute effectually complex estimates of the spatial configuration. Thus, the estimation process has to deal with uncertainty, imprecision, and failures as well as with false positives in order to provide robust and reliable results. On the other hand, robot simulations – which often replace real robots during the development process – have to deal with the opposite problem: reproducing the uncertainties and bothersome details of the real world to provide reasonably realistic input data and output actions for the robot's control software.

Due to the huge variety of current robot systems, different approaches are hard to compare in general since they are often developed for specific scenarios. In recent years, different competitions for mobile robots have been established. These competitions provide complex but standardized environments to achieve a level of comparability and to foster the development of robust and reliable systems.

1.1.1 A Short State of the Art

The mobile counterpart to the standard industrial robots are automated guided vehicles. This kind of robot is, in general, used for logistical tasks in assembly or storage facilities. As the environments they operate in have already been structured and are not necessarily accessible for humans, special alterations and assumptions for their navigation can be made. From the first systems in the 1950s – which followed simple lines on the ground – to the current state of the art *Kiva System* [WURMAN et al., 2008] – which operates hundreds of robots along virtual tracks on a special floor –, an adaption of the environment to the robots instead of an adaption of the robots to the environment has remained the most efficient strategy in industrial contexts.

A second important domain for the application of autonomous systems is the field of domestic service robots. In general, the operational environments – e. g. households or office spaces – are rather unstructured and at least partially unknown to the robot. Up to now, the commercially most successful robot class for such a domain is the one of the autonomous vacuum cleaner (there are also robots for similar tasks such as lawn mowing). The leading product is the *Roomba* which was introduced by the company *iRobot* in 2002 [JONES, 2006]. For reasonably executing such cleaning tasks, only a low behavioral and state estimation complexity is necessary. Thus, current products are based on behavior-based approaches, i. e. the driving direction is initially random and changes only as a direct result of sensor events. Due to the inability of identifying specific objects in their environment, these systems are neither able to estimate their position within a given area nor to track any other objects, e. g. in order to plan any reactions in advance. This design prevents the execution of any complex tasks such as driving to a predefined position but provides the necessary robustness to operate in unstructured environments.

A similar simplicity can be found in most entertainment robots such as the *RoboSapiens* [TILDEN, 2004], a humanoid toy robot that is able to execute simple programs. One significant exception in this domain is the Sony *AIBO* robot [FUJITA and KITANO, 1998], a robot pet (cf. Fig. 1.1c) that was launched in 1999 and discontinued in 2006. This system is able to recognize objects and to perform multi-modal interaction sequences with a user. As this robot was used in context of research presented in this thesis, a detailed technical description will be given in Sect. 3.1.1. Its designated successor was the *Sony QRIO* [ISHIDA et al., 2001], a small humanoid robot based on similar technologies. Until its discontinuance, this robot has only been used in research contexts. Among others, it has been enabled to operate safely in arbitrary household environments [SABE et al., 2004].

Besides the previously described systems, autonomous mobile robots are mainly a subject of intense research rather than any kind of established product. Currently, a huge variety of systems performing different tasks in different environments exists. A famous example is the humanoid robot *ASIMO* by Honda [HIROSE and OGAWA, 2007], shown in Fig. 1.1a. Originally developed for research about bipedal walking, this system is now able to climb stairs, to act as a waiter, or even to run dynamically. There are also many research prototypes built upon the basic *Pioneer* system [MOBILEROBOTS INC, 2010]. These robots are used for a huge variety of experiments such as exploration, multi-agent cooperation, or Simultaneous Localization and Mapping (SLAM). The currently most sophisticated walking robot is *BigDog* [RAIBERT et al., 2008] by Boston Dynamics. It is intended

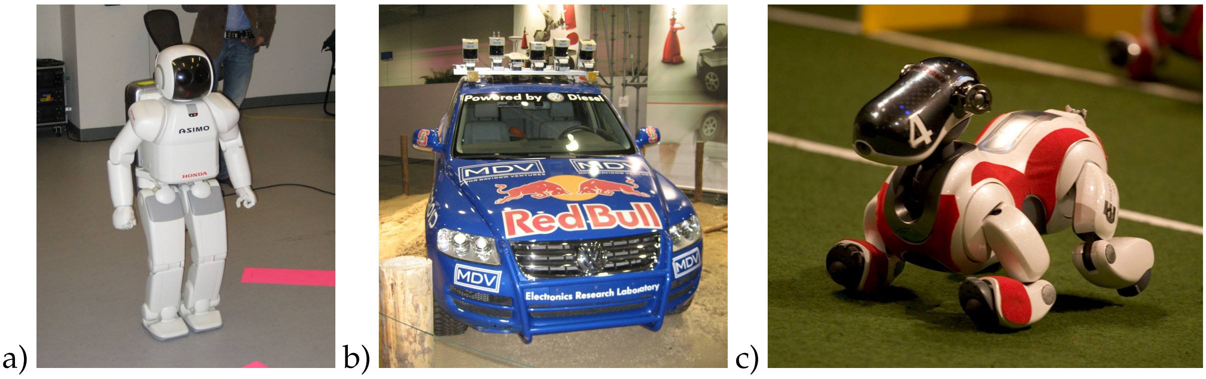


Figure 1.1: Examples of prominent autonomous mobile robots: a) The humanoid robot ASIMO by Honda, b) The autonomous car Stanley from Stanford University, which won the DARPA Grand Challenge in 2005, and c) The AIBO ERS-7 by Sony in a robot soccer setting.

to be used as a pack mule on rough terrain. Although defining the state of the art in four-legged locomotion, it is so far only able to execute simple behaviors such as following a marked human. In addition to these well-known robots, a vast number of – mostly unique – other system exists. Some of them will we described in detail in the course of this work, such as the *Nao* by Aldebaran Robotics (Sect. 3.1.1) or different rehabilitation robots (Sect. 5.1).

1.1.2 Uncertainties in Robot State Estimation

Having established the previously described diversity of systems – mostly operating in research laboratories –, one major questions arises: What is the major problem preventing robots from entering our daily life? Given a technical point of view, the increase of uncertainty from well-defined industrial environments to less structured domains appears to be one crucial issue. Therefore, this as well as the following section will highlight the specific problems real as well as simulation systems have to deal with.

As mentioned above, systems successfully operating in rather unstructured environments do exist but are limited in the complexity of tasks they are able to perform. Most high-level tasks at least demand the ability of self-localization within an environment. In addition, it might also be necessary to find or track other objects that have to be manipulated.

Even in a static environment, given a known starting position, a robot needs to integrate extrinsic sensor measurements for reliable self-localization. A pure accumulation of the performed motion offsets – also referred to as odometry or dead reckoning – is insufficient as uncertainties – caused by slippage, sensor imprecision, or even slight collisions – accumulate over time and thereby invalidate the robot's position estimate. This is true for all kinds of freely moving ground robots, wheeled as well as legged ones.

In robot systems, different sensors with different strengths and weaknesses are applied. Currently, the most common extrinsic sensors are cameras and laser range finders. Also commonly used are ultrasonic and infrared sensors for distance measurement as well as different kinds of touch sensors for determining contact. A variety of uncertainties and limitations has to be considered when dealing with sensor measurements; quite common ones are:

- *Imprecision.* Sensor measurements in the real world always contain errors of a certain magnitude. Hence, perceived sizes and distances can never be perfectly precise. In general, these errors are already inherent in physical characteristics of the applied measurement method. In addition, imprecision is caused, for instance, by a too low sensor resolution or the motion of the measuring device (cf. Sect. 4.3 and Fig. 1.2a) or the measured object.

- *Limited Coverage.* A sensor cannot fully cover a robot's whole environment. The opening angle as well as the possible measurement range are limited. Therefore, each measurement can only represent a section of the relevant area.

- *False Negatives.* It is not guaranteed that all objects within the area covered by a sensor are actually measured. These false negatives might already be caused by simple occlusions. Optical sensors are especially irritable to changing lighting conditions, reflections, or some special surface characteristics. Often, information provided by a sensor is too sparse for detection algorithms to reliably extract the desired features.

- *False Positives.* Almost the same reasons that cause false negatives are also responsible for the opposite: the measurement of objects that are actually not present at the determined position.

- *Perceptual Aliasing.* The information provided by a single sensor measurement is in general limited regarding the indistinguishability of similar objects. For example a camera image does not contain any depth information and thus a huge object that is far away appears equal to an object of the same shape that is close but small.

A detailed overview of sensor as well as motion models is given by [THRUN et al., 2005], who also provide probabilistic models for different uncertainties.

For most estimation tasks, a single perception – and often even a single sensor – cannot reliably provide the needed information, e. g. a robot's pose in an indoor environment, due to the described effects. Therefore, the integration of different measurements over time is necessary. One exception – often used in outdoor applications – are Global Navigation Satellite Systems (GNSS) such as the Global Positioning System (GPS) that directly provide a position. At a smaller scale, similar systems also exist for indoor tasks. Nevertheless, these systems again introduce a dependency on an external infrastructure.

Obviously, adding more sensors can enable a robot to estimate its state more precisely and reliably. However, for mobile systems, the possible payload has to be taken into account. Not only the additional sensors have to be carried but also the computing units for processing the provided data as well as the batteries providing the power for the computers and the sensors. This might only be a minor problem for large systems such as autonomous cars (cf. Sect. 1.1.4 and Fig. 1.1b) but is absolutely crucial when working, for instance, with small humanoid robots (cf. Sect. 3.1.1). For this kind of platforms, efficient algorithms that depend on only a few sensors are needed.

a) b)

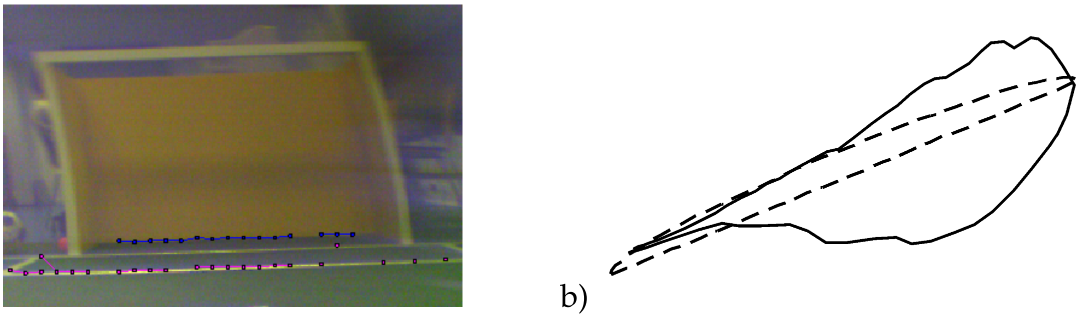


Figure 1.2: Two typical examples of sensor and motion uncertainty in the robotics context: a) A part of an image taken by the head-mounted camera of a humanoid soccer robot. Due to the low resolution and the camera's fast motion, object edges are blurry and have not been detected reliably by the vision software, as indicated by the pink and blue lines. In addition, the camera's rolling shutter causes a distortion of the whole image. b) Comparison of the commanded (dashed) and real (solid) trajectory of an AIBO robot's joint during walking. Such a divergence is the result of different effects, e. g. the inertia and backlash of the gears or slippage of the foot on the ground.

1.1.3 Robot Simulations and the Reality Gap

When working with robots, a simulation is often of significant importance. It enables the evaluation of different alternatives during the design phase of robot systems and may therefore lead to better decisions and cost savings. Furthermore, it supports the process of software development by providing a substitute for robots that are currently not at hand (e. g. broken or in use by another person) or not able to endure extensive experiments such as the learning tasks described in Sect. 4.4.

Currently, a variety of simulators is available to robot system developers. An overview of the state of the art and a comparison are given in Sect. 4.1.

One characteristic trait of all simulations is that they have to deal with the so-called *Reality Gap*, i. e. a simulation is only capable to approximate reality to a certain degree of precision but can never be perfect. This affects all aspects of simulations: the level of detail as well as the characteristics of sensors and actuators. For many applications, this gap is of minor relevance as long as the simulated robot system performs in a reasonable way.

In some domains, for instance in robot soccer competitions (cf. Sect. 1.2), actuator performance is a crucial aspect. Through applying optimization algorithms, impressive results regarding the velocity of walking robots have been achieved in recent years (cf. Sect. 4.4). One common attribute of these algorithms is the strong exploitation of the environment's features, i. e. certain characteristics of the motors or the properties of the ground in this case. This leads to control trajectories that strongly differ from the resulting trajectories of the real robot joints, as shown in Fig. 1.2b. For robot simulations, in particular when working with legged robots that have a high number of degrees of freedom, this requires a proper parametrization to simulate actuators that behave similar to real ones. Otherwise, the simulated robot might not only behave unrealistically but could fail completely by constantly stumbling and falling.

Similar problems arise for the calculation of realistic measurements for the simulated sensors. For many applications, missing details are not relevant and the interaction between the robot software and the simulated world might not be affected. Nevertheless, the characteristics of some sensors matter to an extent that leads to a significant mismatch between simulation and reality. Such a mismatch is relevant in two cases: the first and naïve case is that the robot control software is initially developed only in a simulator, and it does not handle sensor distortions that are not present in the simulation, but surely will be in reality. In the other case, the more experienced developer would develop the software on the real robot and in simulation side by side. However, software tuned for real sensor readings – taking into account some of the effects described in Sect. 1.1.2 – may perform worse or even fail if confronted with unrealistically simple sensor readings coming from a simulator, which would render the simulator as a rather useless tool. Hence, it is important that a simulation generates sensor distortions that are actively handled by the robot control software.

1.1.4 Robot Competitions

Comparing and benchmarking research results in robotics is a difficult issue. In general, results refer to one specific – and often even unique – robot system and have been achieved within an environment selected (or even created) by the developers. Furthermore, the tasks carried out by the robots might be specific for this special environment. It often remains unsure how the presented works might perform in a different setting. This makes the verification of certain claims impossible and a detailed comparison with other research results hardly realizable as they are in general based on different prerequisites. A detailed overview of current problems is given in [DEL PÔBIL, 2006].

One possible approach to overcome these shortcomings is publishing and using standardized data sets such as the ones provided by the *RAWSEEDS* project [BONARINI et al., 2006] or the Radish repository [HOWARD and ROY, 2010]. These data sets enable a direct comparison of the quality of different approaches and are very popular in domains such as Simultaneous Localization and Mapping. Of course, the offline processing of data sets blends out some aspects of robotics, e. g. the interaction with the environment or the demand of real-time processing with only limited computing power.

An increasingly popular alternative is participating in robot competitions that provide standardized test beds for different robotic systems. Such competitions force researchers to operate their robots outside their lab in a different environment at a scheduled time [BEHNKE, 2006].

Over many years, one highly regarded project of the Artificial Intelligence community – and thus some kind of a predecessor of current robot competitions – was to develop powerful chess programs and finally even beat the human world champion. This problem was solved in 1997 when IBM's *Deep Blue* won against Garry Kasparov [CAMPBELL et al., 2002]. However, the problems addressed by current robot competitions are different ones. In contrast to chess, which is set in a static, discrete environment, the systems have to act in real-time in a dynamic, continuous space (cf. [KITANO et al., 1997]) and are subject to the uncertainties described in Sect. 1.1.2.

One of the first popular robot competitions was *Micromouse* [ALLAN, 1979] in which autonomous robots have to navigate a maze. This competition started in the late 1970s and is still active today. The *AAAI Mobile Robot Competition and Exhibition* is held together with the US National Conference on AI since 1992 [DEAN and BONASSO, 1993]. Every year, it features different tasks that the robots have to accomplish, e. g. "Office Delivery" or "Office Cleanup". These tasks change over the years, a comprehensive overview of this competition is given by [BALCH and YANCO, 2002].

The *DARPA Grand Challenges* are prize competitions for driverless vehicles, funded by the US Defense Advanced Research Projects Agency. These competitions – organized as single events and not held regularly – had many participants from renowned universities and gained a huge media coverage. The first two challenges were held in 2004 and 2005, each requiring the vehicles to drive on country roads along an over 200 km long course through the Mojave Desert. As in 2004 no vehicle was able to successfully complete the course, the challenge in 2005 was scheduled as a repetition and finally accomplished by multiple teams [BUEHLER et al., 2007]. The winning car is depicted in Fig. 1.1b. The succeeding event, the *DARPA Urban Challenge* [BUEHLER et al., 2010], provided the scenario of a small city and required the participating vehicles to perform not only autonomous navigation but also to safely conduct traffic maneuvers in presence of other moving objects on the streets. For all these challenges, no strict constraints in terms of size (the vehicles ranged from motorcycles to trucks), sensorial equipment, or computing power have been imposed.

The currently probably most regarded and largest (in terms of participants) annual competition is the *RoboCup*, which is the domain of most works presented in this dissertation and which thus will be described in detail in the following Sect. 1.2. For this competition, a quite similar – but much smaller – counterpart exists in the form of the *Federation of International Robot-soccer Association (FIRA)* which was funded in 1997 (after having the first competition in 1996) [FIRA, 2010].

A detailed overview and comparison of Micromouse, RoboCup, and the AAI robot competition is given in [BRÄUNL, 1999]. Another more recent comparison of RoboCup and the DARPA Grand Challenge is given by [BEHNKE, 2006]. A comprehensive list of different robot competitions and their current dates is maintained by [RAINWATER et al., 2010].

1.2 RoboCup – Dynamic Testbed for Autonomous Systems

RoboCup – originally called *"The Robot World Cup Soccer Games and Conferences"*– is an international research and education initiative. Its goal is to foster artificial intelligence and robotics research by providing a standard problem where a wide range of technologies can be examined and integrated [ROBOCUP FEDERATION, 2010]. This section provides a short overview of its objectives, progress, and current state with a focus on those parts of the competition that are relevant to this thesis.

1.2.1 Constitution and Objectives

The concept of soccer-playing robots was published for the first time by [MACKWORTH, 1993]. In 1995, after a two-year feasibility study, an announcement was made on the introduction of the first international conferences and football games. At this point in time, the name *RoboCup* was introduced:

> "RoboCup is a task for a team of multiple fast-moving robots under a dynamic environment."
>
> *[KITANO et al., 1995]*

In the following years, the idea became reality during various workshops and demonstrations at major robotic conferences. The first official RoboCup workshop was held in Osaka in 1996 as a part of the *International Conference on Intelligent Robots and Systems*. The workshop was associated with preliminary competitions by real robots as well as simulated ones. Finally, the first RoboCup competition was held during the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI-97) in Nagoya, Japan, as part of the conference's special program. The overall state of the art for this event is summarized in [NODA et al., 1998].

Soon after establishing the RoboCup competition in 1997, the RoboCup Federation proclaimed its ambitious long term goal.

> "By mid-21st century, a team of fully autonomous humanoid robot soccer players shall win the soccer game, comply with the official rule of the FIFA, against the winner of the most recent World Cup."
>
> *[KITANO and ASADA, 1998]*

This goal can be considered to be a new benchmark project in artificial intelligence, comparable to the efforts made to beat the human world champion in chess. It is also a main difference to similar competitions such as the FIRA that do not offer this kind of long-term vision. The research presented in Sect. 2.3 an Sect. 2.4 directly refers to this 2050 goal.

Currently, RoboCup competitions take place every year. Within a defined set of different sub-domains and leagues, incremental steps towards this big goal are made. A rapid and remarkable progress has been observed during the first decade of these robot competitions, regarding the number and variety of sub-competitions (as described in the following section) as well as regarding the number of teams. The last RoboCup in 2010 in Singapore had more than 3000 participants from over 40 countries.

1.2.2 RoboCup Domains Today

As mentioned before, the RoboCup competition is split up into different domains, each providing a different scenario and consisting of a set of leagues. Currently, RoboCup has four major domains:

RoboCup Soccer can be considered as the original domain of RoboCup since in 1997, the initiative started with only three different soccer leagues [NODA et al., 1998]. This domain is also the scenario for the works presented in this thesis, thus it is described in more detail in Sect. 1.2.3.

RoboCup Junior is the educational branch of RoboCup. It provides different competitions – partly inspired by the senior competitions – for pupils to foster the interest in robotics already at school level. The idea originated from the LEGO lab at the University of Aarhus in Denmark and was published for the first time by [LUND and PAGLIARINI, 1999]. The first official competitions at RoboCup world championships were held in 2000, a comprehensive overview of the first years of this domain is given by [SKLAR et al., 2003].

RoboCup Rescue provides standard problems for Search and Rescue in disaster scenarios. It was motivated by problems arising from the enormous destruction through earthquakes, e. g. by the one in Kobe, Japan in 1995. The domain was proposed by [KITANO et al., 1999], intending to extend “RoboCup to a socially significant real world domain”. After a first demonstration in 2000, the simulation league started at RoboCup 2001 in Seattle, the robot competition followed in 2002 in Fukuoka [ASADA et al., 2003]. By using official NIST (National Institute of Standards and Technology) arenas [JACOFF et al., 2002] as standard environment, RoboCup Rescue features detailed quantitative performance metrics for a rather unstructured environment.

RoboCup@Home extends RoboCup towards the emerging field of domestic service robots. It consists of a set of benchmark tests that are used to evaluate the robots’ abilities and performance in a realistic home environment setting. RoboCup@Home was proposed by [VAN DER ZANT and WISSPEINTNER, 2006] and has been a part of RoboCup competitions since 2006 [VISSER and BURKHARD, 2007]. The progress during its first years is summarized by [WISSPEINTNER et al., 2010].

1.2.3 RoboCup Soccer in Detail

The RoboCup started as a soccer-only competition, featuring three different leagues: the Simulation League, the Small Size Robot League, and the Middle-Size Robot League [NODA et al., 1998]. Over the years, new soccer leagues – namely the Humanoid League and the Standard Platform League – were added and already existing leagues split up into more specific competitions. Within RoboCup, each league focuses on specific areas of research.

In the original *Simulation League*, two teams of eleven autonomous software programs (called agents) each play soccer in a 2D virtual soccer environment represented by a central server. This simulation provides a rudimentary physical model, generates noisy input data for the agents’ sensors and executes the action commands generated by the agents. This league serves as a testbed for multi-agent collaboration and learning algorithms, as e. g. outlined by [STONE et al., 2005] at the example of playing keep-away soccer. In 2004, the *3D Simulation League* was additionally introduced [LIMA et al., 2005] and has since then been run in parallel to the original competition. It features a more realistic simulation based on rigid body dynamics [KÖGLER and OBST, 2004] which allows the usage of humanoid robot models. As such a simulation demands high computational resources, currently no full 11 vs. 11 matches can be conducted. The 3D Simulation League can be considered to be a link between the original 2D Simulation League and the RoboCup

Figure 1.3: Example of a robot soccer match: The RoboCup 2006 Small Size League final between 5dpo from Portugal and the CMDragons from the USA.[1]

leagues featuring real humanoid robots [MAYER et al., 2007]. It is also closely related to the simulation research presented in this thesis (cf. Sect. 4.1 as well as Sect. 4.2.3).

The *Small Size Robot League (SSL)* was founded as entry robot league as it limits the size – and thus the costs – of the robots (the maximum diameter is 18 cm and the maximum height is 15 cm) and allows off-board processing by external computers as well as a global vision system. From 2010 on, the latter is even provided by the organizers, allowing the participants to focus on research other than computer vision. Parts of this standard system are a contribution of this thesis (cf. Sect. 3.6.1). The robots – five per team – have to be constructed by the participants and in general possess fast omnidirectional drives as well as special kicking and dribbling mechanisms. Due to the almost perfect model of the robots' whole environment and the efficient actuation, the league is able to feature a faster and tactically more sophisticated gameplay than other robot leagues. Therefore, the SSL can be considered as a link between these leagues and the Simulation League, having quite similar research topics. In Fig. 1.3, a typical match of this league is depicted. We participated in the SSL with the team *B-Smart*, as described in detail in Sect. 2.1.4.

The *Middle Size Robot League (MSL)* was the first RoboCup league in which fully autonomous real robots participated. Being mechanically similar to SSL robots – scaled to a diameter of up to 50 cm –, MSL systems have on-board computers and can be equipped with almost arbitrary sensors. Thus, the MSL is probably the soccer league with the most computing power per robot and the least restrictions on sensorial equipment. The current state of the art is the usage of multiple cameras of which one is directed to a special mirror providing omnidirectional sight, allowing, for instance, quite specialized localization approaches such as [LAUER et al., 2006]. Within RoboCup, the MSL is the league with the most elaborated vision systems, having been the first league playing under natural

[1]Image copyright: TZI and Messe Bremen

lighting conditions [VISSER and BURKHARD, 2007] and now having a field without artificial landmarks. In the beginning, the field size was similar to today's other soccer robot leagues (cf. Sect. 3.1.2), but in the meantime it has grown to impressive $18m \times 12m$, providing space for 6 vs. 6 matches using a standard-sized soccer ball. The RoboCup naming scheme might indicate the existence of a *Large Size Robot League*, but so far, no according plans have been published.

The *Humanoid League* was introduced in 2002 (after exhibitions in the two previous years), making the RoboCup the first competition of humanoid robots ever [ASADA et al., 2003]. The robots are divided into different size classes, namely the *KidSize*, the *TeenSize*, and the *AdultSize*. Depending on the class, robot teams play soccer matches or perform a penalty shootout-like competition. In general, the participants have to construct their own robots that are required to have a human-like body and human-like sensors, so omnidirectional vision or laser range finders are not permitted. As for the KidSize class, construction kits or even fully assembled robot bodies are commercially available, the construction of the larger humanoid robots – which have to withstand collisions and falls – is an area of intensive research. All robots have to operate autonomously, therefore many research areas overlap with other robot leagues, e. g. vision, self-localization, and action-selection. Additionally, the Humanoid League fosters research in the area of bipedal motions, as described in detail in Sect. 4.4. We participated in this league with the teams *BreDoBrothers* and *B-Human* (cf. Sect. 2.1.2) and used it as a base for research. Thus, detailed descriptions of the robot platforms and the environment will be given in Sect. 3.1.1 and Sect. 3.1.2 respectively.

The *Standard Platform League (SPL)* is a software competition using real robots. As the league's name suggests, every team has to use the same standard robot to avoid a distortion of competition by excessive investments in superior hardware, the so-called "arms race". After first demonstrations in 1998, the league started in 1999 [CORADESCHI et al., 2000] using the aforementioned Sony AIBO robot as platform, intermediately giving the league the name *Sony Four-legged Robot League*. After Sony discontinued the production of this robot in 2006, the RoboCup Federation selected the *Nao* [GOUAILLIER et al., 2008] by Aldebaran Robotics as new standard platform, starting in 2008. As the Nao is almost compatible to the Humanoid KidSize class, both leagues share almost the same research goals. The SPL has been the major domain for the works presented in this thesis. That is why the environment and the robot platforms are also described in Sect. 3.1.1 and Sect. 3.1.2 respectively. The close relations to the Humanoid League are discussed in Sect. 2.2. We participated in the SPL with different teams of which descriptions will be given in Sect. 2.1.1 and Sect. 2.1.3.

1.2.4 The Scientific Significance of RoboCup

As mentioned in Sect. 1.1.4, the RoboCup is an established event for benchmarking and comparing different robot systems and approaches. Such a competition does not only help to determine quality in regards of precision and correctness but also in regards of robustness and efficiency as the systems are situated in a competitive, dynamic environment. The setting requires acting in real-time, even in situations that have not been foreseen in detail during the development phase.

The environments in the different domains and leagues do not remain static but change over the years. The particular problems of each scenario become incrementally harder over time to continuously foster research and to avoid any overspecializations. Some examples of such changes (and their intended progress) are: the removal of artificial lighting (more flexible vision systems), the removal of closed field borders (more intelligent object recognition, cf. Fig. 3.2), enlargement of fields (higher robot velocities and more intelligent team play), removal of artificial features (more intelligent state estimation), more players per team (more complex cooperation).

In addition to the competitions, each RoboCup is accompanied by a conference – the *RoboCup Symposium* – for the presentation and discussion of scientific contributions. The proceedings of these conferences are published in Springer's *Lecture Notes in Artificial Intelligence* series afterwards. Established already at the first RoboCup [Kitano, 1998], this approach has been retained to the present [Baltes et al., 2010].

In addition to the annual RoboCup symposia, workshops about closely related topics are held in conjunction with major conferences. In its first years, the whole RoboCup was sometimes co-located with major international AI conferences, e. g. the *International Joint Conferences on Artificial Intelligence (IJCAI)* 2001 in Seattle, USA. Current examples are the *Workshop on Humanoid Soccer Robots* [Zhou et al., 2009] – which is regularly held at the *Humanoids* conference – or the workshop on *Agents in Real-Time and Dynamic Environments* [28] at the 9th international conference on *Autonomous Agents and Multiagent Systems (AAMAS-2010)* in Toronto, Canada.

Over the past years, the number of RoboCup-related publications at major robotics conferences also increased, documenting the common acceptance of this research community. Two examples originating from this thesis are [6] at the *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007)* and [10] at the *4th European Conference on Mobile Robots - ECMR'09*.

1.3 Contributions of this Dissertation

The contributions of this thesis cover different areas of robotics research – mainly state estimation and simulation – but have one common theme: the integration of uncertainty in dynamic environments.

As most of the presented works are settled in the RoboCup domain, Sect. 2.1 presents our teams that successfully participated – and won several international and national titles – in different RoboCup leagues and thereby provided the base for this research. As different leagues have partially overlapping areas of research, contributions to the inter-league exchange of approaches and software are described in Sect. 2.2. An intensive involvement in a research domain also requires to formulate and to assess future goals, independent of the current state of the art's constraints, which in the case of RoboCup primarily foster iterative, small research steps. Therefore, bearing in mind the RoboCup Federation's 2050 goal, analyses regarding the middle-term feasibility of a full-featured soccer vision system (cf. Sect. 2.3) and the long-term problems of physical interaction arising in human-robot soccer matches (cf. Sect. 2.4) have been conducted.

The major part of this thesis describes contributions to the field of state estimation for legged soccer robots (cf. Chap. 3). As such robots possess only very limited computing resources and have to rely on quite noisy and limited perceptions, efficient and reliable sensor models have been developed to perform self-localization (cf. Sect. 3.3) and object tracking (cf. Sect. 3.7). The self-localization implementation is based on the popular *Monte-Carlo Localization* approach [FOX et al., 1999] and thus able to represent multimodal probability distributions. To efficiently extract the robot's pose from the according sample set, a new approach has been developed (cf. Sect. 3.4). This approach is also able to extract multiple hypotheses of which the contributions to an active vision system and a potential field-based action selection can make use (cf. Sect. 3.5). The overall self-localization approach has been subject to a detailed quantitative evaluation (cf. Sect. 3.6.2) using a vision-based tracking system (cf. Sect. 3.6.1). In addition, the self-localization has successfully demonstrated its quality in different competitions (cf. Sect. 3.6.3).

The counterpart of the state estimation is the simulation of mobile robots and their environments as described in Chap. 4. The contributions in this field are based on the *SimRobot* application of which a contemporary version has been developed in the course of this thesis as described in Sect. 4.2. To reflect the uncertainties that occur in the real world and are accordingly handled by a robot's control software, contributions to the simulation of image disturbances (cf. Sect. 4.3) and to the optimization of the parameters determining the simulated world's physical behavior (cf. Sect. 4.5) have been made. Especially the latter is closely related to the contributions made to the field of robot motion modeling and optimization that are described in a separate, preceding section (cf. Sect. 4.4).

Results and experiences from this thesis have been transferred beyond the RoboCup scope to the fields of rehabilitation robotics and Ambient Assisted Living (cf. Chap. 5). This includes indoor localization and navigation for mobility assistants (cf. Sect. 5.2), position estimation in road networks based on digital elevation models (cf. Sect. 5.3), and the evaluation of furniture configurations in an Ambient Assisted Living scenario through simulation (Sect. 5.4).

The conclusion in Chap. 6 shortly summarizes these results and provides an outlook on open issues that could be addressed in the future.

In addition to all publications, several software components have been developed and released to the public in the course of this thesis. These include code releases of RoboCup teams (cf. Sect. A.1), the simulator *SimRobot* (cf. Sect. A.2), *SSL-Vision* – the standard vision system of the RoboCup Small Size Robot League (cf. Sect. A.3), and the *GMapping Localization Library* (cf. Sect. A.4). The respective relations to my research are pointed out in the according subsections.

Finally, all my publications are cumulated in Appendix B. For each publication, the *List of Publications by the Author* (cf. pp. 65) contains my share in percent as well as a short description of the particular contribution.

Chapter 2

General Contributions to RoboCup

RoboCup can be seen from different perspectives. On the one hand, it is an annual competition demanding incremental progress in terms of research as well as system maintenance and upgrading. On the other hand, it is a long-term project that requires to look ahead and to anticipate future developments that are not compliant to the current setting and state of the art.

This chapter describes the RoboCup teams I participated in, which provided the base for the research presented in this thesis. In addition, works regarding the RoboCup's future perspective – complementing and substantiating aspects of existing roadmaps by [KITANO and ASADA, 1998] and [BURKHARD et al., 2002] – are presented.

2.1 Participation in RoboCup Competitions

During the past years, I successfully participated in several RoboCup competitions as a member of different teams that won the world championship four times and also the German Open – one of the largest and most significant regional RoboCup competitions – four times. A complete summary of all awards is given in Tab. 2.1. This section briefly describes all teams, my particular activities, and the respective relations to this thesis. From my point of view, it is important that research results are actually used in such competitions to avoid academic research and competitions drifting apart.

2.1.1 Standard Platform League – Four-legged

The *GermanTeam* was founded as the German national team in 2001. It was a joint team of researchers and students from the Humboldt Universität zu Berlin, the Universität Bremen, the Technische Universität Darmstadt, and the Universität Dortmund (until 2005). Until the discontinuation of the league in 2008, it was one of the most influential and successful teams regarding awards (cf. Tab. 2.1) as well as scientific publications [GERMANTEAM, 2008]. For regional competitions, each university had its own local team. Our team were the *Bremen Byters*.

I was involved in these teams from 2001 to 2007, conducting research regarding self-localization (cf. Sect. 3.3), robot tracking (cf. Sect. 3.7), and potential field-based behavior control (cf. Sect. 3.5.2). The corresponding software components have been used by the team during competitions. Furthermore, the GermanTeam was the first main application of the revised SimRobot (cf. Sect. 4.2) and the AIBO was used as an instance for the

Year	Competition	League	Team	Rank
2004	RoboCup 2004	Four-legged	GermanTeam	1^{st}
2005	RoboCup German Open	Four-legged	Bremen Byters	3^{rd}
	RoboCup 2005	Four-legged	GermanTeam	1^{st}
2006	RoboCup Dutch Open	Small Size	B-Smart	2^{nd}
	RoboCup Dutch Open	Humanoid	BreDoBrothers	3^{rd}
	RoboCup 2006, Technical Challenge	Small Size	B-Smart	2^{nd}
2007	RoboCup German Open	Small Size	B-Smart	2^{nd}
2008	RoboCup German Open	Small Size	B-Smart	1^{st}
2009	RoboCup German Open	Standard Platform	B-Human	1^{st}
	RoboCup 2009	Standard Platform	B-Human	1^{st}
	RoboCup 2009, Technical Challenge	Standard Platform	B-Human	1^{st}
	RoboCup 2009, Technical Challenge	Small Size	B-Smart	2^{nd}
2010	RoboCup German Open	Standard Platform	B-Human	1^{st}
	RoboCup 2010	Standard Platform	B-Human	1^{st}

Table 2.1: Relevant achievements in RoboCup competitions. This table lists only teams of which I was an active member. These results show that – as claimed in the introduction – the developed methods actually perform well in the difficult situation of a competition.

physical parameter optimization described in Sect. 4.5. The migration of concepts and implementations to a different robot league described in Sect. 2.2 was based on the GermanTeam's overall system.

The team's detailed progress during that time is described in a series of annual team description papers [29, 33, 42, 40, 41] that are a requirement for the participation in the RoboCup world championship. In addition, the GermanTeam has published multiple code releases as described in detail in Sect. A.1.1. These releases are accompanied by detailed team reports [31, 30, 48, 52].

2.1.2 Humanoid League

In parallel to the activities in the Four-legged league, a humanoid team was started on a trial basis in 2005. Again in cooperation with the Universität Dortmund, the *BreDoBrothers* were founded and participated in two competitions in 2006. As described in Sect. 2.2, the software was mostly based on previous works in the Four-legged League, running on robots based on inexpensive construction kits. After RoboCup 2006, the team was split and the local team *B-Human* was founded. After Sony's announcement of the AIBO's discontinuation, educational and research activities were shifted from the Four-legged to the Humanoid League.

I have been co-leading both teams. My research activities involved self-localization (cf. Sect. 3.3), ball tracking (cf. Sect. 3.7), the simulation of image distortions (cf. Sect. 4.3), and bipedal walking (cf. Sect. 4.4). The results of the former two have been used in a number of competitions.

Both teams are described in detail in a series of team description papers [44, 43, 46]. In respect to the upcoming migration of the *B-Human* team to the Standard Platform League (cf. Sect. 2.1.3), the developed software was released (cf. Sect. A.1.2) and described in a team report [47].

2.1.3 Standard Platform League – Two-legged

As a consequence of the low performance – mostly caused by uncompetitive hardware – in the Humanoid League, in 2009, the team B-Human returned to the Standard Platform League which had in the meantime established the usage of the two-legged *Nao* platform (a comparison of the different humanoid platforms is given in Sect. 3.1.1). This was preluded by a revival of the BreDoBrothers in 2008, allowing the researchers and students on both sides to evaluate the new platform and to establish in this new league. Since 2009, B-Human is by far the most successful team in the SPL, having won all 28 matches in four tournaments (cf. Tab. 2.1), having reached a total goal ratio of 210:6.

I am currently the co-leader of the B-Human team, which provided the base for my most recent research results. This includes further works on self-localization (cf. Sect. 3.3), in particular the development of a new approach for pose extraction from sample sets (cf. Sect. 3.4), a performant active vision approach (cf. Sect. 3.5.1), as well as works regarding walking and kicking motions (cf. Sect. 4.4). The results of the motion and localization approaches are currently used in competitions. The team also served as a platform for a detailed evaluation of the overall self-localization performance as described in Sect. 3.6.

The teams are described in detail in a series of team description papers [32, 45, 49] as well as in comprehensive reports [51, 50] accompanying B-Human's code releases (cf. Sect. A.1.2).

2.1.4 Small Size Robot League

The team *B-Smart* was a project for students in the advanced study period at the Universität Bremen. The team has continuously participated in RoboCup Small Size League competitions from 2003 until 2009, including all world championships as well as all major European competitions.

I was the team leader of B-Smart. As the team had a primarily educational focus, no research results directly originated from this RoboCup activity. Nevertheless, the potential field-based navigation (cf. Sect. 3.5.2) was applied to the SSL scenario. Additionally, I co-developed the league's standard vision system (cf. Sect. 3.6.1). Within the SSL community, I held different organizational roles during the past years. Inter alia, I was the Local Organizing Chair of the RoboCup 2006 SSL competition, the Chair of Technical Committee 2008, and I am currently a member of the league's Executive Committee which steers the league's mid-term progress.

B-Smart's progress is documented in a series of team description papers [34, 35, 39, 37, 36, 38]. The software release of the league's standard vision system is described in Sect. A.3.

2.2 Inter-League Transfer

As already described in Sect. 1.2.2, RoboCup is divided into different domains which again might consist of a set of leagues. Within RoboCup Soccer, the leagues have been specified in a way that all important subproblems of the 2050 goal are covered, from complex group behaviors in the Simulation League down to the actual design of humanoid robots in the Humanoid League's classes (cf. Sect. 1.2.3). As the research goals of different leagues partially overlap, it is a reasonable approach to reuse the solutions provided by other leagues.

When entering the Humanoid League, which has a strong focus on robot construction, such a transfer has been done by reusing almost the complete software of the Four-legged GermanTeam. This was possible due to strong similarities between both leagues: the usage of legged robots that have to rely on directed vision as primary sensor (cf. Sect. 3.1.1) and the basically equal environment – a color-coded soccer field (cf. Sect. 3.1.2). As such transfers are not very common – many research groups focus on a single league –, the details and insights have been published in [23].

Having decided to enter the Two-legged competition of the Standard Platform League, a similar transfer was realized vice versa: Using B-Human's Humanoid League code on the Nao robot. The result is described in a team report (cf. [47]) and provided the base for the later success of the team. The software has also been published and thereby enabled other teams an easier entry into the league (cf. Sect. A.1.2). Furthermore, the subsequent code release (cf. [51]) applied technology originating from the Small Size Robot League: the league's standard vision system (cf. Sect. 3.6.1) has been integrated to provide ground truth data for evaluating the precision of the state estimation approaches as presented in Sect. 3.6.3.

2.3 (A) Vision for 2050

Currently, RoboCup competitions take place every year, making incremental steps towards the RoboCup federation's 2050 goal. Although a rapid and remarkable progress has been observed during the first decade of these robot competitions, it is not predictable, if and how the final goal will be reached. There exist rough roadmaps, e.g. by [KITANO and ASADA, 1998] and [BURKHARD et al., 2002], but in many research areas, huge gaps must be bridged within the next 40 years. While this is obvious for several areas such as actuator design (cf. Sect. 2.4) and control, we claim that the situation is surprisingly positive for vision:

> "Within the next decade, it will be possible to develop a vision system that is able to provide all environmental information necessary to play soccer on a human level."
>
> *[5]*

This section describes the base for this claim, accompanied by the idea of context exploitation and first experiments.

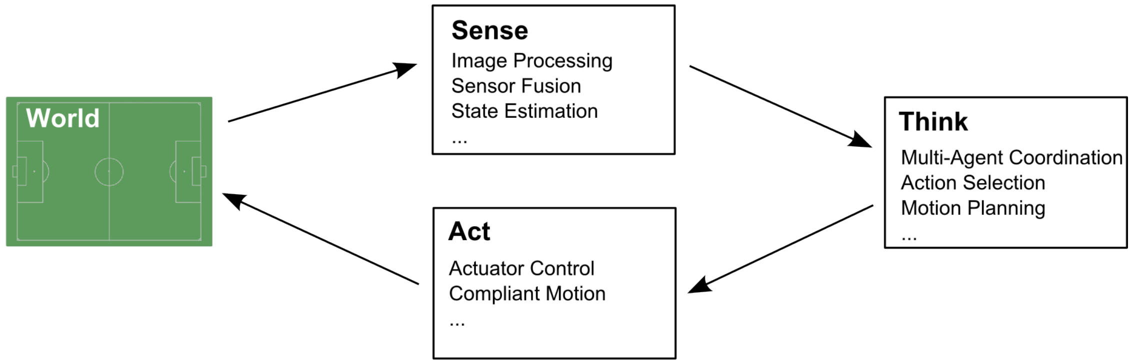


Figure 2.1: The Sense-Think-Act cycle roughly depicting major tasks for playing soccer with a humanoid robot.

2.3.1 Challenges and State of the Art

The global task of playing soccer consists of several different, interdepending challenges. These can be roughly categorized and illustrated according to a Sense-Think-Act cycle (cf. Fig. 2.1).

The most obvious gap may be observed in the field of actuation. Besides fundamental problems such as energy efficiency and power supply, the walking as well as the kicking velocities of current robots are multiple times slower than the top speed reached by human soccer players. Furthermore, soccer is a contact sport leading to physical human-robot interaction that requires approaches that guarantee safety. These issues are discussed in detail in Sect. 2.4.

Regarding the "thinking" part, two different levels may be distinguished: motion planning and high-level multi-agent coordination. The latter has been a research topic in the RoboCup Soccer Simulation League for a while and has reached a remarkable level (cf. Sect. 1.2.3) in tasks such as dealing with the offside rule or playing one-two passes. On the other hand, when playing with real humanoid robots, sophisticated methods for full-body motion planning are needed. Algorithms for this problem already exist (e. g. by [KUFFNER et al., 2002]), but are subject to restrictions that make them inapplicable to tasks such as playing soccer on a human level, e. g. to allow volley-kicking.

According to [KITANO and ASADA, 1998], it is evident that the robots' sensorial capabilities should resemble those of humans. Thus, we could assume to deal with data from cameras and inertial sensors emulating the human eyes and vestibular system. The objects relevant in a soccer match are the ball, the goals, the line markings, and, of course, the players (including their feet and limbs) as well as the referees. Ball, goal, and line markings are geometrical features for which a large number of detection algorithms is known [DAVIES, 2004]. Recognizing other players in detail is more challenging but fortunately, people tracking is an important topic in computer vision with a large body of literature, including works about tracking sportsmen, e. g. by [RAMANAN and FORSYTH, 2003]. In addition, there are already half-automatic systems in the related area of TV soccer scene analysis, for example the ASPOGAMO system by [BEETZ et al., 2006, BEETZ et al., 2007], proving that soccer scene understanding in general is on the edge of being functional.

These works differ strongly from the current state of the art in RoboCup competitions. The participating robots often still rely heavily on color segmentation as described in Sect. 3.1.3. Of course, this approach will not be an option for real-world soccer, mainly due to changing lighting conditions. Therefore, developing a vision system for the RoboCup 2050 challenge is well beyond the current state of the art but surprisingly realistic as we do not need a new level of functionality but a new level of robustness, beyond lab demonstrators requiring nicely setup scenes and lighting conditions.

2.3.2 Robustness through Context

In [5] and [1], we propose to address the question of robustness by utilizing probabilistically modeled context information, formulating the overall scene understanding and prediction problem as a global likelihood optimization task.

Most current vision systems use a data-driven bottom-up approach, e. g. [BEETZ et al., 2007] as well as the systems described in Sect. 3.1.3, extracting low level features from the image and aggregating them through several stages to high level information. We believe that much of the brittleness of current vision systems originates from this way of committing to hard decisions on an early level of processing. Our proposed solution is to understand an image sequence, i. e. to estimate over time. Indeed, successive images are linked by a motion model and this provides most of the context we want to build upon. However, we propose not to use incremental filters, but to look back into the raw images of the last few seconds at least. This approach has surprising advantages. Imagine the ball is kicked, but during the first 100ms there is too little contrast to the background so it is not detected. Now when it is detected, there is new information on where the ball has been before from the ball's motion model. The old images are still in memory and tracking the ball back in time is much less ambiguous than finding the ball without context.

In the data-driven approach the ball finder determines where the ball is and the camera model computes where the ball should be according to the state. Then both are compared with the difference indicating the likelihood of the state, as e. g. described in Sect. 3.7. In the global likelihood optimization approach, in contrast, the camera model computes where the ball should be and then the ball appearance model computes how "ball-like" the image there looks. The difference is that now the ball appearance model, i. e. the computer vision, is inside the optimization loop. Now the lower layer does not have to commit early to a single image circle, but instead it gradually assesses different circles in the image as requested by the minimization algorithm.

Doubtless, if the goal is to track a flying ball, the ball's motion model provides the most valuable context. However, there is other, more semantic information that could be incorporated to increase robustness, e. g. the field geometry or the game semantics providing information about the current game state. However, it is still rather unclear how such background knowledge can be modeled in a way that can effectively help a computer vision system. A classical AI approach would be to use ontologies to describe a soccer match. First steps towards such an ontology for soccer robots have been made by [DYLLA et al., 2008].

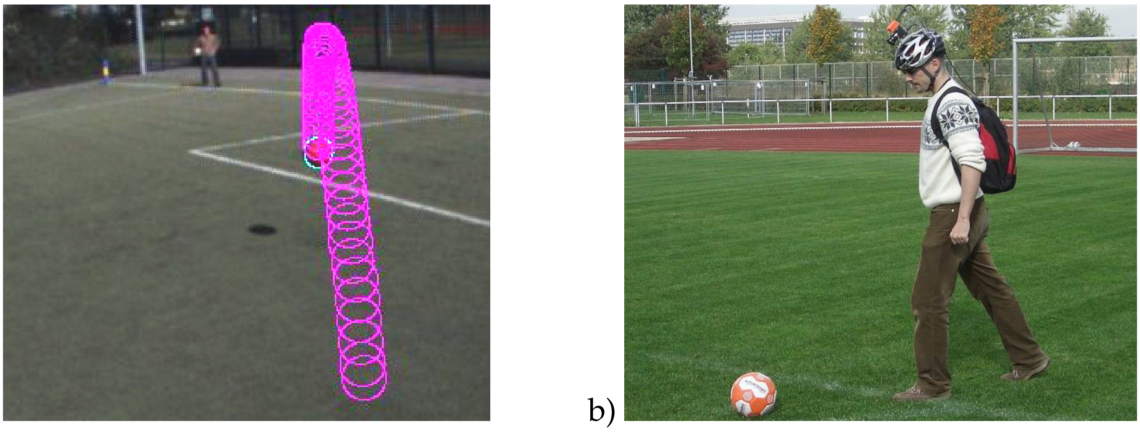
a)  b)

Figure 2.2: a) Predicting the trajectory of a flying ball from a moving camera-inertial system. As an initial study, the ball, the lines, and the goal corners have been manually extracted from the images. b) Our proposed experiment: Mount a camera and an inertial sensor on the head of a human soccer player and use them to extract all the information a humanoid soccer robot would need to take the human's place.

2.3.3 Experimental Evaluation

Since a *real* human level soccer robot will not be available for a long time, our vision is accompanied by a (partially already conducted) set of experiments that verify our claim without needing a robot. The basic idea is to let a human soccer player wear a helmet (cf. Fig. 2.2b) with a camera and an inertial sensor and to verify that the information extracted by the vision system would allow a humanoid robot to take the human's place.

As a first experiment, we propose to record data from a soccer match and run the vision system on that data offline. Since it is hard to obtain ground-truth data, we would use our expert judgment to assess, whether the result would be enough for a humanoid robot to play soccer. This approach allows to concentrate on functionality and robustness first instead of computation time and integration. A very first experiment has already been conducted and published [3]. For this experiment, the ball and the field lines were manually extracted from the recorded images and the ball's trajectory was predicted by least-square estimation (cf. Fig. 2.2a). The results indicate that if the ball can be detected in the image with about one pixel precision, the prediction would be precise enough. Further research in this area – including automatic ball recognition and the ability to track multiple ball hypotheses in real-time – has already been done by [Birbach and Frese, 2009].

Further experiments would include the usage of a tracker-less motion capture suit – measuring joint angles and thereby providing odometry and height information – as well as providing a human with the vision system's output via a head mounted display to see whether s/he can play – the most direct proof that the extracted information is all you need for playing soccer.

a) 

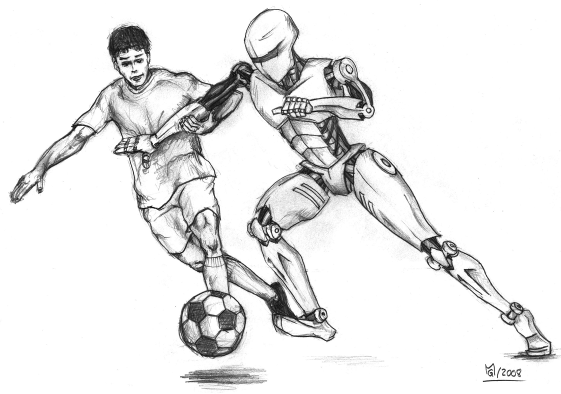
b) 

Figure 2.3: a) Annual match between humans and the current Middle Size Robot League world champion team, in this case the 1. RFC Stuttgart in Graz 2009. b) Artist's depiction of the RoboCup 2050 vision [2].

2.4 Physical Interaction in Human-Robot Soccer

As described in Sect. 1.2, the RoboCup community has the definite goal of winning against the human world soccer champion team by the year 2050 (cf. Fig. 2.3b). Such a soccer match between humans and robots implies physical human-robot interaction (pHRI) including tackles and fouls since soccer is a contact sport and injuries of players are frequent [LEES and NOLAN, 1998]. Even more, the FIFA rules state explicitly that

> "Football is a competitive sport and physical contact between players is a normal and acceptable part of the game. [. . .]"
>
> *Laws of the game* [FÉDÉRATION INTERNATIONALE DE FOOTBALL ASSOCIATION, 2006]

So far, this has not been a subject of discussion within the RoboCup community. The current state of the art is limited to an annual match between RoboCup officials against the Middle Size Robot League world champion (cf. Fig. 2.3a) in which the robots have always been easily defeated. However, there have already been philosophical considerations whether robots would be allowed to participate in an official soccer match at all [STONE et al., 2010]. Nevertheless, in the context of industrial and service robotics applications, similar issues regarding pHRI are currently a matter of research. In this section, works bringing together pHRI and RoboCup are presented.

2.4.1 Safety for Humans and Robots

To determine the possible threats caused by physical interaction in soccer matches, in [6] an analysis of real soccer matches has been conducted, extracting the most common types of interactions, e. g. tripping, body impacts, or getting hit by the ball. These different types have been analyzed regarding their injury potential and the already available approaches from pHRI that allow to prevent any severe injuries. In some cases, even simple coatings turned out to be sufficient (head impacts). However, the introduction of passive and variable joint compliance turned out to be a crucial feature to ensure both, safety to the human and robot soccer player. As described in the following section, compliance can also facilitate the kicking performance of a robot leg.

In addition, by evaluating sports and biomechanics literature, the claim that "The robots should have heights and weights comparable to the human ones (at least for safety reasons) [...]" [BURKHARD et al., 2002] has been confirmed.

2.4.2 Increased Kicking Performance Through Joint Elasticity

Compared to humans, the kicking performance of current robots is low. One exception are the robots in the Middle Size Robot League that are able to kick a real soccer ball over large distances. However, this capability is due to specialized kicking actuators, a temporarily tolerated aid that will be forbidden in a match against humans. The currently most powerful (regarding the strength of their actuators) humanoid soccer robots as well as the mighty KUKA KR500 robot arm are not even able to outperform a 5 year old child. In [2], these differences and the according reasons are analyzed in detail.

One possibility to overcome current limitations appears to be the application of elastic joints. Using a variable stiffness joint developed by the German Aerospace Center's (DLR) Institute of Robotics and Mechatronics [ALBU-SCHÄFFER et al., 2008], a series of kicking experiments – using different kicking techniques as well as different balls including current RoboCup balls – has been conducted, yielding promising results.

2.5 Contributions of the Corresponding Publications

As annual RoboCup competitions are always bound to strict rule sets (defined for the state of the art of the competing robots) and demand competitive robot teams, only incremental progress adapting to actual rule changes is fostered. Thus, the publication *(A) Vision for 2050 - The Road Towards Image Understanding for a Human-Robot Soccer Match* [5] contributes the definition of a feasible mid-term goal for the RoboCup project in order to set a new landmark that could guide the incremental development. An updated and extended journal article will appear under the title *(A) Vision for 2050 – Context-Based Image Understanding for a Human-Robot Soccer Match* [1].

The related publication *Tracking of Ball Trajectories with a Free Moving Camera-Inertial Sensor* [3] contributes an accuracy analysis of ball trajectory estimates of a free moving camera in a soccer scenario. Similar works already exist in the RoboCup context, e. g. by [ROJAS et al., 2007] for the Small Size Robot League or by [VOIGTLÄNDER et al., 2007] for the Middle Size Robot League. However, these approaches rely on camera perspectives that are either static or at least partially fixed.

Although anticipating a goal for the far future, the two publications *Foul 2050: Thoughts on Physical Interaction in Human-Robot Soccer* [6] and *Kick it with Elasticity: Safety and Performance in Human-Robot Soccer* [2] are the first contributions that link the RoboCup with the research area of physical Human Robot Interaction. By evaluating sports medicine and biomechanical literature and conducting various experiments, some claims regarding the RoboCup's 2050 goal [BURKHARD et al., 2002] have successfully been verified.

The team description papers and team reports describing the progress of the different RoboCup teams I was a member of (cf. pp. 70) do not contain any significant scientific contributions but can be considered as a collection of basic contributions to the RoboCup project as a whole.

The documentation of inter-league knowledge transfer described in *Getting Upright: Migrating Concepts and Software from Four-Legged to Humanoid Soccer Robots* [23] as well as in two later team reports [47, 51] contributes concrete examples of successful technology exchange between different RoboCup leagues.

Chapter 3

State Estimation

When executing complex tasks in dynamic environments such as the RoboCup, robots are required to make decisions in real-time. These decisions depend on the robot's estimates of its own position within a predefined frame of reference and of other objects' or other players' positions. Hard limits in computing time and sensor equipment make these tasks challenging. This chapter presents works about efficient state estimation – i. e. self-localization and object tracking – in RoboCup Soccer scenarios.

3.1 Self-Localization in RoboCup – Prerequisites

Before presenting any self-localization approaches, this section briefly describes the necessary prerequisites, i. e. the robots, the environment, and the capabilities of the used vision systems for feature extraction.

3.1.1 Robot Platforms

The major works for this thesis have been realized on four different robot platforms: the Sony AIBO ERS-7 (see Fig. 1.1c), the Aldebaran Nao (see Fig. 3.2), and two humanoid robots based on self-assembly kits (see Fig. 3.1). Appearing quite different at a first glance, these robots have major features in common, which enabled the transfer described in Sect. 2.2: they use vision as a primary sensor; each robot has a single active camera that has a limited field of view and is mounted on a pan-tilt head; the robots move on legs; the processing power is low. A detailed overview is given in Tab. 3.1. These properties differ from the other RoboCup soccer leagues. Apart from using state-of-the-art computers and professional machine vision cameras, the robots in these leagues either have a global static

	AIBO ERS-7	Nao	Bioloid-based	Kondo-based
Robot Height	27.8 cm	58 cm	48 cm	38 cm
Degrees of Freedom	12	21	20	18
Degrees of Freedom (Head)	3	2	2	2
Camera Resolution	208×160	320×240	320×240	320×240
Camera Opening Angle	$56.9° \times 45.2°$	$45.08° \times 34.58°$	$45.1° \times 34.8°$	$45.1° \times 34.8°$
Camera Frame Rate	30/s	30/s	15/s	15/s
Processor	MIPS 576 MHz	Geode 500 MHz	XScale 520 MHz	XScale 520 MHz

Table 3.1: Comparison of relevant properties of the robots used in the context of this thesis.

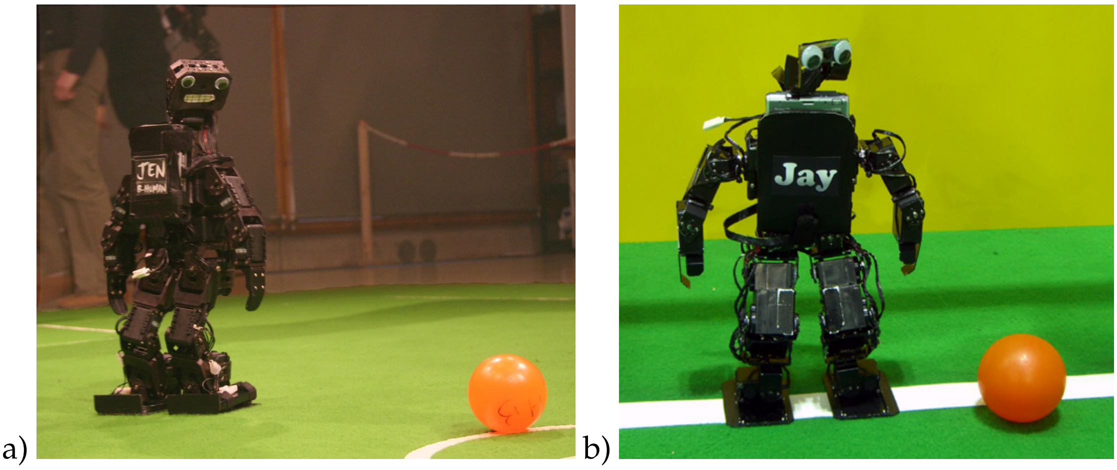


Figure 3.1: Self-assembled humanoid soccer robots: a) Bioloid-based robot *Jen* of the team B-Human at the GermanOpen 2007 and b) Kondo-based robot *Jay* of the team BreDo-Brothers at the RoboCup 2006.

camera perspective (SSL) or use mirrors to obtain an omnidirectional view of the whole environment (MSL).

As already mentioned in Sect. 1.2.3 the AIBO and the Nao are standard robots built by companies. As these robots are already complete regarding their hardware equipment, the teams do not need to – and are not allowed to – modify them. In addition to the vision systems, both platforms feature simple distance sensors that can be used for detecting and avoiding close obstacles (cf. Sect. 3.6.3).

Both self-constructed robots are based on self-assembly kits that provide motors, frames, and control electronics. The Bioloid kit by [ROBOTIS, 2010], which was used by B-Human, has been particularly popular and was successfully used by a number of RoboCup teams such as [MOBALLEGH et al., 2007]. To perform image processing and state estimation on such a robot, a camera and a computing unit need to be added. For both humanoid types, a PDA with integrated camera was used.

3.1.2 The Environment

In RoboCup, all soccer fields are specified up to a precision of one millimeter, including all elements on the field, such as the goals and the field lines. To simplify object recognition, every element is color-coded. In the Standard Platform as well as in the Humanoid League, the field is a green carpet with white lines, the goals are either blue or yellow, the ball is orange, and the robots wear uniforms that are either magenta or cyan, cf. Fig. 3.2. A detailed specification of the current Standard Platform League field is given in [ROBOCUP TECHNICAL COMMITTEE, 2010].

Such a setup might appear quite simplistic and highly specialized. Of course, it supports image processing based on color classification (cf. Sect. 3.1.3). However, in recent years, the field layout has changed towards that of a normal soccer field without any special cues for aiding the robots. As shown in Fig. 3.2, the surroundings of a field are not specified and might contain random content. In the past, a border prevented the robots from confusing the audience with field elements. This protection has been removed some

Figure 3.2: A match in the Standard Platform League: B-Human (in blue) is playing the semi-final against UT Austin Villa at RoboCup 2009 in Graz, Austria.

years ago so that today, the image processing software has to differentiate e. g. between a blue goalpost and a pair of blue jeans. Furthermore, the goals have changed from solid boxes to realistic goals with nets. The Middle Size Robot League even uses goals that are not color-coded. Additional beacons, which facilitate self-localization, have been removed completely in most leagues including the SPL.

One frequent source causing uncertainties in perception are changes in lighting conditions. In the past, this problem was solved by intensively illuminating the field. Currently, the venue's standard lighting conditions (which still have to guarantee a minimum illumination) have to be accepted, including possible changes caused by the sun shining in through the windows.

In addition to changing lighting conditions and random surroundings, major uncertainty is caused by the course of play: other robots cover the line of sight to expected objects, collisions or even falls occur and thus heavily disturb measurements and the estimate, or the referee might replace robots (a classical case of *robot kidnapping*, cf. Sect. 3.2.3). Finally, all robots and the ball remain in constant motion.

3.1.3 Vision Systems

For computer vision in this domain, two main approaches are popular: blob-based and grid-based systems. To extract blobs from an image, full color segmentation is necessary. After segmentation, connected regions of the same color class are determined. A common solution for this task is *CMVision* by [BRUCE et al., 2000]. This approach provides robust results but is quite time-consuming (on robots such as the Nao) since every pixel of the image needs to be examined. Grid-based approaches can be significantly faster,

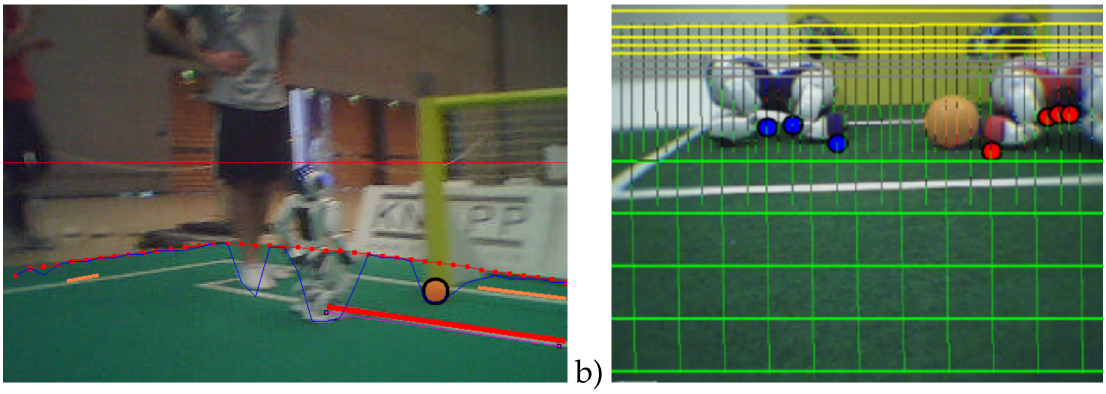


Figure 3.3: Images perceived by robot cameras: a) Extract of an image made by a Nao robot. The thin dashed red line depicts the robot's horizon, the blue line and the red dotted line denote the perceived field and its convex hull respectively. b) Image made by an AIBO robot. The gray, green, and yellow lines show the grid used for object detection. The red and blue dots depict recognized elements of other robots.

since only a fraction of all pixels is interpreted, namely those on a (often horizon-aligned) grid (cf. Fig. 3.3b). This technique was introduced in the Standard Platform League by [Bach and Jüngel, 2002], a similar approach was also used in the Middle Size Robot League before [Jamzad et al., 2002]. Since the grid lines are only one-dimensional, only a small context of a pixel can be considered, which makes this approach more sensitive to outliers. To overcome this problem, so-called *specialists* can be applied, examining a feature's surroundings in more detail, e. g. to detect certain elements of a robot's uniform, as shown in Fig. 3.3b.

A combination of both techniques has been presented in [22]: In a first step, significant segments are searched on a grid. This grid is bounded by the robot's spatial context, i. e. the border of the field and its own body contour (cf. Fig. 3.3a). In a second step, the segments are merged to regions according to a set of constraints. These regions are the base for a final feature detection.

However, all these approaches still rely on color segmentation, requiring a manual calibration.

3.2 State of the Art

In recent years, probabilistic approaches for state estimation turned out to be exceedingly successful in robotics as they are based on mathematical models compromising the uncertainties that occur in a robot's environment [Thrun et al., 2005]. This section briefly describes the most common methods in robotics in general as well as the approaches used in the RoboCup domain in particular. As the works of this thesis are based upon the *Monte-Carlo Localization (MCL)* approach, this method is described in more detail.

3.2.1 Probabilistic Filters

For probabilistic state estimation, recursive filters are applied that compute a belief distribution $bel(x_t)$ of a state x_t by incorporating measurement data z_t as well as control data u_t. The most general approach and thus the basis for all following techniques is the *Bayes filter*. Its name is derived from Bayes' theorem about conditional probabilities which is applied to the update of $bel(x_t)$. A comprehensive introduction and derivation of this filter is given by [THRUN et al., 2005]. In practice, the Bayes filter is only applicable to very simple problems. That is why a number of derivatives have been published in the past.

Probably the most widely used and approved approach is the *Kalman filter* [KÁLMÁN, 1960]. Its benefits are simplicity, optimality, tractability, and robustness [JULIER and UHLMANN, 1997]. Nevertheless, the original approach is not able to cope with non-linear models. Hence the *Extended Kalman Filter (EKF)* might have to be applied, providing a linearization of all nonlinear models. To overcome the commonly arising problems regarding the proper implementation and parametrization of the EKF, the *Unscented Kalman Filter (UKF)* was proposed by [JULIER and UHLMANN, 1997]. Additionally, this filter has also been shown to provide more accurate results than the EKF.

All Kalman filter-based approaches have in common that they are not able to represent multi-modal probability distributions. Nevertheless, some scenarios – such as self-localization in RoboCup (cf. Sect. 3.1.2) – require the consideration and representation of multiple different states, e. g. to overcome kidnapping situations through sensor resetting (cf. Sect. 3.4). Therefore, additional approaches for handling sets of different modes need to be added to these filters, as for instance done by [QUINLAN and MIDDLETON, 2010].

In recent years, the so-called *Monte-Carlo Localization (MCL)* approach [FOX et al., 1999] has gained immense popularity in the robotics community. It approximates the probability distribution by a set of samples (also referred to as particles, coining the alternative name *particle filter*) and is hence able to deal with multi-modal distributions. This filter is described in detail in Sect. 3.2.3.

A combination of the previously described techniques is the *Rao-Blackwellised particle filter* [DOUCET et al., 2000], which allows to split the variables of x_t into a part that is sampled and a part that can be estimated by an optimal filter such as the Kalman filter.

3.2.2 Self-Localization in RoboCup

Due to its aforementioned capability of efficiently coping with non-linear dynamics and multi-modal probability distributions but probably also due to the straightforwardness of its implementation [THRUN et al., 2001], Monte-Carlo Localization is a popular choice for self-localization in RoboCup scenarios.

In soccer leagues with local directed vision systems such as the Standard Platform League (SPL) and the Humanoid KidSize League, MCL can be considered as being a de facto standard. At RoboCup 2009, it was used by all top three teams of the KidSize League – *Darmstadt Dribblers* [FRIEDMANN et al., 2009], *FUmanoid* [MOBALLEGH et al., 2009], and *CIT Brains* [HAYASHIBARA et al., 2009] – as well as by two of the top three teams of the SPL – *B-Human* [51] and *Nao Devils Dortmund* [CZARNETZKI and KERNER, 2009].

Algorithm 1 The Augmented-MCL approach, derived from [THRUN et al., 2005, p.258].

function Augmented_MCL(X_{t-1}, t_y, z_t**):**

static $w_{\text{slow}}, w_{\text{fast}}$
$\bar{X}_t = \emptyset, X_t = \emptyset, w_{\text{avg}} = 0$
for $m = 1$ to M **do**
  $x_t^{[m]} = \textit{sample\_motion\_model}(u_t, x_{t-1}^{[m]})$
  $w_t^{[m]} = \textit{measurement\_model}(z_t, x_t^{[m]})$
  $\bar{X}_t = \bar{X}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$
  $w_{\text{avg}} = w_{\text{avg}} + \frac{1}{M} w_t^{[m]}$
end for
$w_{\text{slow}} = w_{\text{slow}} + \alpha_{\text{slow}}(w_{\text{avg}} - w_{\text{slow}})$
$w_{\text{fast}} = w_{\text{fast}} + \alpha_{\text{fast}}(w_{\text{avg}} - w_{\text{fast}})$
for $m = 1$ to M **do**
  with probability $\max\{0, 1 - w_{\text{fast}}/w_{\text{slow}}\}$
    add new pose to X_t
  else
    draw i with probability $\propto w_t^{[i]}$
    add $x_t^{[i]}$ to X_t
  endwith
end for
return X_t

Actual alternatives are the usage of a standard Extended Kalman Filter, e. g. by the *Northern Bites* [HERMANS et al., 2009], or a multiple model Kalman filter as presented by [QUINLAN and MIDDLETON, 2010]. A combination of a Kalman and a particle filter – sampling the rotational pose component and continuously estimating the translational part – is used by the *UPennalizers* [BRINDZA et al., 2009] in the form of a Rao-Blackwellized particle filter. This technique has also been used by [KWOK and FOX, 2005] for ball tracking in the SPL context. A novel approach based on computationally inexpensive constraint propagation techniques has recently been presented by [GÖHRING et al., 2009]. A localization technique that relies only on memorized bearings to unique objects and odometry information has been introduced by [JÜNGEL, 2007].

In other RoboCup Soccer leagues, self-localization is either not a topic of research – the Small Size League provides a global view on the whole relevant environment (cf. Sect. 3.6.1) – or can be solved in a different way – the Middle Size League allows omnidirectional vision systems, which make approaches based on numeric error minimization as that of [LAUER et al., 2006] an efficient and precise solution.

3.2.3 Monte-Carlo Localization

The MCL approach (depicted as used for this thesis in Alg. 1) describes the probability distribution of a state X_t by a set of M samples $x_t^{[m]}$. Thereby, each sample (also referred to as particle) represents one possible hypothesis. The state of each sample – which is a two-dimensional pose in the context of the works presented in this chapter – is constantly updated according to the robot's intended or measured motion (cf. Alg. 1, line 4). In addition to all state variables, each sample contains a weighting $w_t^{[m]}$ that indicates the likelihood of the hypothesis given the current sensor measurements (cf. Alg. 1, line 5). Based on these weightings, the whole sample set is regularly *resampled*, i. e. an empty new sample set is instantiated and samples from the old sample set are added randomly in proportion to their weighting (cf. Alg. 1, from line 11 onwards). This process increases the probability density in relevant areas of the state space. A comprehensive introduction to MCL and some derivatives is given in [THRUN et al., 2005].

In real-time applications, the number of samples used is often quite low as computing time grows linearly with the number of samples. This causes a very sparse coverage of the state space. In case of a robot kidnapping or a sequence of false measurements (leading to a concentration of samples at a wrong place), the algorithm might take a long time to recover a proper position estimate. A solution to overcome this problem is the regular replacement of samples. These can be drawn from a uniform distribution [FOX et al., 1999] or be sampled from recent observations (cf. Fig. 3.5a). The latter approach is far more efficient and known as *sensor resetting* [LENSER and VELOSO, 2000]; a quite similar, mathematically motivated solution is *Mixture-MCL* [THRUN et al., 2001]. Replacing a constant number of samples in each execution cycle might appear as a straightforward solution, expressing the likelihood of a delocalization. Nevertheless, such a strategy leads to unnecessary sample replacements in case of a precise estimate and might not provide enough new samples in case of a robot kidnapping. A flexible solution is provided by the *Augmented-MCL* approach [GUTMANN and FOX, 2002], which controls the sensor resetting by the rate of change of the sample set's overall weighting. This technique has been used for most self-localization works presented in this thesis.

Overall, the MCL approach is basically simple to implement. In general, the algorithm itself does not need to be adapted to any new scenario as it is a proved and quite generic solution. The main scientific challenge is to create appropriate models accompanied by a proper parametrization describing the robot's motion and sensor characteristics. In the course of this thesis, this has been done for the robot soccer domain (cf. Sect. 3.3) as well as for an outdoor navigation scenario (cf. Sect. 5.3).

Furthermore, the MCL approach itself provides a probability distribution within the state space but not a concrete result state. Especially in scenarios in which multi-modal probability distributions occur (such as RoboCup Soccer), a non-trivial postprocessing step is necessary. For this purpose, a new approach, which is closely coupled with the sensor resetting mechanisms, has been developed (cf. Sect. 3.4).

3.3 Sensor Models

For proper self-localization, the integration of measurements of different cues is often necessary. In the RoboCup scenario, this could be, for instance, goalposts, which provide valuable information about the absolute position of the robot but are seen rarely and mostly at some distance, or lines, which are often seen close to the robot but only provide ambiguous information (cf. Sect. 3.3.1). In some cases, the robot's vision system might be too limited to provide reliable, directly usable perceptions. For certain instances, this

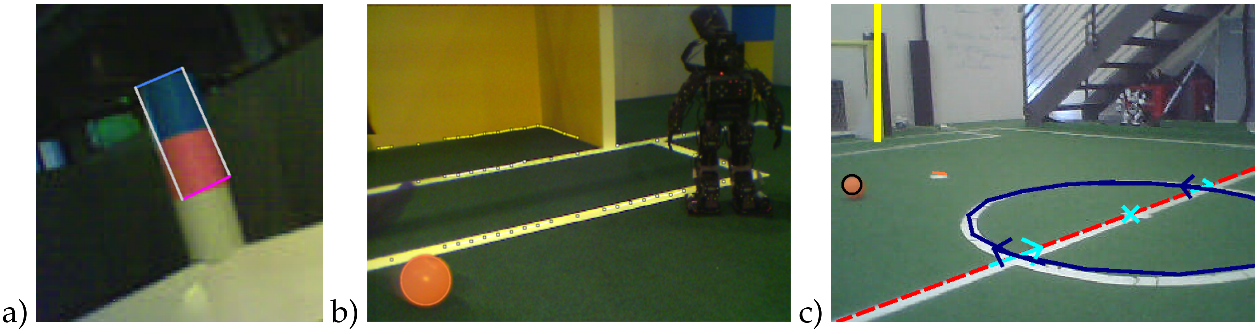


Figure 3.4: Different cues for self-localization (overlayed by drawings of the extracted features): a) A unique beacon that was used in the Four-legged League in 2004, b) a goal and some lines in the Humanoid League in 2007, c) the center circle and one goalpost of a current Standard Platform League field.

can be compensated by an interpretation that takes the current sample set's state into account as described in Sect. 3.3.2. When performance limitations require to deal with quite small sample sets and to apply sensor resetting mechanisms, the sensor models need to be adapted to allow an efficient and stable state estimation (cf. Sect. 3.3.3). Overall, these interdependencies lead to a complex, non-trivial parametrization. An optimization process for this problem is presented in Sect. 3.3.4.

3.3.1 Multi-Cue Localization

When performing the Monte Carlo sensor update step (named as *measurement model* in line 5 of Alg. 1), each sample's weighting $w_t^{[m]}$ is computed based on the deviation of the model $x_t^{[m]}$ from the current perceptions. Let $p_1, \ldots, p_n$ be the perceptions made within one execution cycle and $w(x, p)$ a function computing the likelihood (in the range of $[0 \ldots 1]$) of an observation given a sample's state, a weighting can be computed as follows:

$$w_t^{[m]} = w(x_t^{[m]}, p_1)w(x_t^{[m]}, p_2) \ldots w(x_t^{[m]}, p_n) \tag{3.1}$$

To determine $w(x_t^{[m]}, p)$ for a perception p, two deviations from the expected model need to be taken into account: the deviation of the direction on the ground as well as the difference of the distances. A detailed description of the used representation for distances and the modeling of deviation uncertainty is given in [22].

In the course of this thesis, models for different cues – which are in some cases quite basic due to the limitations of the vision system – have been developed. In [17], a self-localization approach for the AIBO robot based on unique beacons (cf. Fig. 3.4a) and goalposts as well as on directed line points has been presented. Whereas the weighting computation for unique elements is straightforward, an efficient association between a perceived point and a real line requires the implementation of a number of look-up tables.

The latter concept was revisited in [27] for localization in the Humanoid League scenario. This environment contained very large unique elements – Fig. 3.4b shows one of the massive goals – that are hard to perceive to a reasonable extent given the robot's limited field of view. This is why the according sensor models did not contain any concept of unique elements and were completely reduced to the concept of point mapping by integrating the perceptions of the bottom borders of the goal and beacon borders.

For self-localization in the Standard Platform League [22], this has been changed back as in this environment a different goal construction is used and beacons have been disestablished (cf. Fig. 3.2). Nevertheless, a single perceived goalpost only provides ambiguous information. In addition to the line point information, crossings of lines as well as the center circle (cf. Fig. 3.4c), which is the environment's only unique element, are taken into account for weighting computation.

Similar work has been done for the Humanoid League scenario by [STRASDAT et al., 2007]. However, their results are based on superior hardware, i. e. a wide-angle lens providing a broad overview of the environment as well as a compass directly providing orientation information.

3.3.2 State-based Sensor Models

The models described in the previous section represent the standard approach for state estimation: all measurements are considered to be uncertain but still correct, i. e. the perceived objects are assumed to actually exist in the current field of view. However, in reality false positives occur. This kind of measurement results from limitations of the perceptual components. A standard approach for compensating this problem would be to increase the assumed uncertainty in the sensor and – in the MCL context – to increase the number of particles to establish a more flexible probability distribution. Nevertheless, given a reoccurring sequence of false positives, the estimation process will still converge to a wrong state.

That is why, in the course of this thesis, an approach for validating a measurement by using a sample's state $x_t^{[m]}$ has been developed [22] and applied to eliminate false line positives resulting from the goal nets – a problem that turned out to be unsolvable for the current image processing approach. In general, false line positives are not a significant problem as they can be compensated by the standard sensor model's uncertainty. However, close to a goal, the nets cause reoccurring false positives that lead to a reproducibly wrong position estimate. Briefly, the approach is realized as follows: whenever a sample's weighting $w_t^{[m]}$ is updated by line information, it is possible to check – by a precomputed look-up table – whether it might be a goal net segment given the context of $x_t^{[m]}$. In that case, $w_t^{[m]}$ will not be updated. To avoid any inconsistencies within the sample set, all omitted samples are updated afterwards using the average weighting update of all other samples.

This usage of context can be seen as a first step towards achieving more robustness as required for the long-term goal described in Sect. 2.3.2. A different usage of context, which has already been realized for the RoboCup scenario by [HOFFMANN et al., 2005b], is the inclusion of negative information within the sensor model, i. e. the absence of expected measurements given $x_t^{[m]}$.

3.3.3 Particle Depletion and Sensor Resetting

Since the computational resources are limited for real-time applications, a particle filter can often only operate on a small particle set to remain efficient, e. g. the implementation for the B-Human SPL team is currently configured to use only 100 samples. But even when using, for instance, 500 samples, only a sparse coverage of the state space can be realized. Thus, the problem of particle depletion might occur, i. e. the probability distribution is reduced to a small set of particles during the MCL resampling step. Especially in this scenario, using robots with a strongly limited field of view, it might happen that certain local observations, which are not necessarily compatible to the global hypothesis, fit a small subset of samples almost perfectly. Hence, these samples, which often result from a previous sensor resetting step, rule out the majority of other samples approximating the current probability distribution.

One solution to avoid such fluctuations is the keeping of particle weightings over multiple cycles as, for instance, proposed by [RÖFER and JÜNGEL, 2003]. However, this approach bears certain drawbacks as pointed out by [NISTICÒ and HEBBEL, 2009] who propose a modified resampling process.

In [22], a different approach has been applied. Instead of keeping weighting information from previous cycles, a balanced distribution of weightings was achieved by the usage of base probabilities and the assumption of inordinate uncertainty. In addition, the same fixed number of measurements is integrated in every cycle. Particularly the latter significantly stabilizes the Augmented MCL approach as its sensor resetting mechanism depends on the sample set's weighting change over time (cf. lines 7, 9, 10, and 12 of Alg. 1 in combination with Eq. 3.1).

3.3.4 Parameter Optimization

Successfully applying particle filters demands a proper configuration of a variety of different parameters that model the assumed uncertainty or control – in case of the Augmented MCL algorithm – the process of sensor resetting. The implementation used by B-Human has a total of 18 relevant parameters (a detailed description is given in [4]) that are partially interdependent.

This is why we developed an optimization approach that is able to optimize all relevant parameters in order to obtain more precise position estimates [4]. In general, adequate sensor models can be learned as described in [THRUN et al., 2005] but in this case, they have to be optimized together with the algorithm's parameters due to the strong interdependencies between sensor models and sensor resetting (cf. Sect. 3.3.3). The work is based on the Particle Swarm Optimization (PSO) approach by [KENNEDY and EBERHART, 1995] that is able to provide results faster than evolutionary algorithms [ANGELINE, 1998]. PSO has already been applied successfully to gait optimization in the RoboCup context (cf. Sect. 4.4). Since the proposed optimization procedure is computationally intensive, an additional contribution of this work was a modification of the original PSO algorithm that leads to a faster convergence near an optimum in case of noisy problems.

The starting point – and thus also the reference – for the optimization process was the hand-tuned parameterization successfully used by the B-Human team in RoboCup 2009. The approach was able to achieve considerable improvements of an already quite precise

a) b) c)

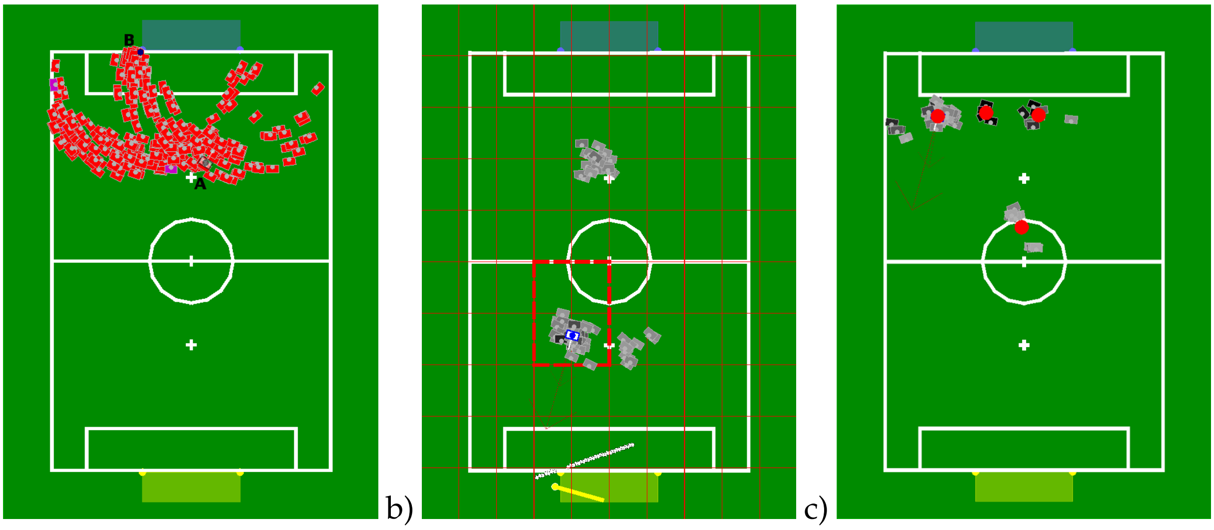


Figure 3.5: Sensor resetting and pose extraction in three different situations: a) The red boxes denote a sample set that has been fully generated using the observation of the non-unique goalpost at position B by a robot standing at position A. b) Pose extraction by binning. The 2×2 sub-square containing the most samples is highlighted. c) Four clusters – each marked by a filled circle – have been found through k-means clustering.

system, as presented in detail in Sect. 3.6.2. In addition, it provided insights into the relevance of some parameters.

3.4 Pose Extraction

In many applications, robot behavior and robot control components that rely on an estimate of the robot's pose within a given frame of reference do not deal with probability distributions. Instead, they usually require a single definite pose (often in 2D) as a base for further computations. This single pose needs to be extracted from the probability distribution and should be the position (in state space) having the highest probability density.

Having a unimodal probability distribution, this is an almost trivial task. However, in many self-localization scenarios, particularly when recovering from kidnapping or determining an initial pose estimate (i. e. starting from an unknown position), the probability distributions representing the current pose estimate are multi-modal. For the extraction of a single, most likely pose from a sample set, different approaches are known [Thrun et al., 2005], but they each have different drawbacks. A computationally inexpensive and popular approach, which was already used in the context of this thesis (cf. [17] and [27]), is *binning* [Thrun et al., 2005]. This technique discretizes the state spaces into so-called *bins*, as depicted in Fig. 3.5b. Although being a straightforward, efficient approach with the capability of identifying multiple modes, binning suffers from certain discretization side-effects, e. g. state instabilities at the edges of the bins. An obvious continuous solution for extracting multiple robot pose hypotheses from a sample set would be the application of a clustering algorithm. However, general clustering is NP-hard in general and even the more efficient *k-means clustering* approach (cf. Fig. 3.5c) is computationally too expensive and requires a known number of clusters.

Due to these shortcomings, a novel approach for computing a pose given a sample-based probability distribution has been presented in [10]. The algorithm – labeled *particle history clustering* – is precise, very efficient, and straightforward. This is achieved through keeping the resampling history of each particle from the moment of sensor resetting. Thereby, a "natural" coherence of particles can be defined as the possession of a common ancestor. The algorithm is accompanied by a set of optional merging heuristics that can be applied to increase its level of reactivity. Real-robot experiments conducted in the RoboCup scenario demonstrated that the approach is able to provide results of a similar quality as approved – but computationally or configurationally more complex – algorithms. Hence, it appears to be an elegant and efficient alternative for some scenarios, such as the outdoor localization approach presented in Sect. 5.3.

3.5 Applications for Multi-Modal Robot Pose Estimates

For reasons of simplification, many robot systems rely on one single pose estimate and thereby ignore potential multi-modalities. However, some approaches allow a reasonable consideration of multi-modal estimates. In the course of this thesis, two such components have been developed: an entropy-based active vision system for the Nao robot (cf. Sect. 3.5.1) as well as a behavior architecture based on potential fields (cf. Sect. 3.5.2).

3.5.1 Entropy-based Active Vision

The estimation of a robot's world model can be improved by actively sensing the environment through considering the current world state estimate, i. e., in the Standard Platform League scenario, by directing the camera to the positions of expected features. A common approach for this task is the minimization of the entropy of future belief distributions [Fox et al., 1998].

In [19], this has been realized for a combination of self-localization and ball tracking as these two tasks are both of similar importance and thus typically compete for camera control. Both models are based on particle distributions (the ball state estimation is described in Sect. 3.7.1) and thereby allow a straightforward integration into the entropy minimization process.

However, the original approach is computationally quite expensive and thus not directly applicable to the SPL scenario. A major contribution of [19] is the presentation of a number of optimizations – inter alia fixed-point arithmetics and the application of different look-up tables – to allow real-time operation on the Nao robot. One major performance improvement was to use clusters – computed by the pose extraction approach presented in Sect. 3.4 – instead of the whole sample set for entropy computation. In practice, samples rarely distribute over the state space but cluster at a few positions, as, for instance, shown in Fig. 3.5b where the probability distribution has only three major modes. Through extracting the distribution's n_m most significant modes from n_s samples and using them as robot hypotheses for entropy calculation, computing time can be reduced by a factor of $\frac{n_s}{n_m}$. Of course, this assumes that the mode extraction is computationally comparatively inexpensive. This is the case for the particle history clustering approach, which runs in linear time.

Quite similar research – however concentrating on self-localization and not considering the task of ball tracking – has been conducted in parallel by [CZARNETZKI et al., 2011].

3.5.2 Potential Field-based Behavior Specification

Artificial potential fields, as developed by [KHATIB, 1986], are a quite popular approach in robot motion planning because of their capability to act in continuous domains in real-time. By assigning an attractive force field to the desired destination and repulsive force fields to obstacles, a robot can navigate along a collision-free path by computing a motion vector from the superimposed force fields. In addition, there also exist applications of potential functions for the purposes of situation evaluation and action selection, e. g. the *Electric Field Approach* by [JOHANSSON and SAFFIOTTI, 2002].

In [8], a new approach was presented that combines the existing techniques in a behavior-based architecture [ARKIN, 1998] by realizing single competing behaviors as potential fields. The architecture has generic interfaces allowing its application on different platforms for a variety of tasks. Its evaluation was conducted by specifying behaviors for the GermanTeam as well as for the B-Smart team in RoboCup.

One specific benefit of this architecture is its ability to fluently integrate not only single, definite state estimates – for the robot as well as for the objects in its environment – but also estimates consisting of multiple, weighted modes, such as those provided by the approaches presented in Sect. 3.4. This is made possible by the internal representation of the robot's environment as continuous functions, which is a major difference to many other approaches, such as the approved state machine approach, inter alia implemented by [LOETZSCH et al., 2006].

3.6 Evaluation of Self-Localization

The most recent versions of B-Human's Standard Platform League localization implementation can be considered as the sum of most works that have been described in this chapter so far. This software has already undergone a thorough evaluation, quantitatively (cf. Sect. 3.6.2) as well as qualitatively (cf. Sect. 3.6.3). The former has been realized using a tracking system that has also been created in the course of this thesis (cf. Sect. 3.6.1).

3.6.1 Ground Truth

A careful evaluation of software components is an important issue. For the precision of state estimation approaches, this can be achieved by an external tracking system that provides reliable information – the so-called *ground truth* – with a precision that is higher than the precision of the system that is evaluated. Previous works in the RoboCup context have e. g. been published by [STULP et al., 2004] who tracked Middle Size League robots with external cameras, [RÖFER and JÜNGEL, 2004] who used a laser range finder

No.	Description of experiment	average error	std. deviation
1	Walking eight figure	116.7*mm*	6.2%
2	Walking eight figure; head mostly focused on goals	146.9*mm*	7.3%
3	Walking circles	116.5*mm*	9.4%
4	Walking circles; one goal displaced by 0.5m	187.2*mm*	23.2%
5	Walking circles; field contains additional fake lines	125.2*mm*	18.7%
6	Robot *Leonard* plays 1 vs. 1 soccer	204.9*mm*	5.2%
7	Robot *Penny* plays 1 vs. 1 soccer	201.5*mm*	6.7%
8	Robot *Leonard* plays 2 vs. 2 soccer (original)	292.3*mm*	6.1%
9	Robot *Leonard* plays 2 vs. 2 soccer (optimized)	188.9*mm*	8.4%
10	Robot *Leonard* plays 2 vs. 2 soccer (original)	307.7*mm*	6.0%
11	Robot *Leonard* plays 2 vs. 2 soccer (optimized)	198.5*mm*	8.9%

Table 3.2: Precision of self-localization in different scenarios.

to determine the exact position of an AIBO robot, and [NIEMÜLLER et al., 2011] who used a professional tracking system to even track body parts of a moving Nao robot.

For evaluating the precision of the previously presented self-localization approach, the *SSL-Vision* application [20] – which was partially developed in the course of this thesis – was used. It is the standard vision application that all teams participating in the Small Size Robot League (cf. Sect. 1.2.3) have to use. SSL-Vision was presented in 2009 and made mandatory in 2010 as the previous solution – every team set up their own global vision system – implied several organizational limitations and thus impaired the league's progress. In addition, most teams have converged on very similar solutions anyway, and have produced only few significant research results to this global vision problem in recent years.

Technically, SSL-Vision is mostly based on approved approaches that have already been used in the SSL for several years. The color segmentation and blob detection is based on CMVision [BRUCE et al., 2000] and the robot pattern detection is based on the work of [BRUCE and VELOSO, 2003]. For camera calibration, an approach has been developed that – in contrast to previous solutions in the SSL – allows a fast and precise calibration without the need of special calibration patterns. These techniques are embedded in a new framework that has been developed from scratch. SSL-Vision has been released as open source, as described in Sect. A.3.

To track anything different from a standard SSL robot – in this case a Nao –, it is in most cases sufficient to put a compatible pattern on top of the robot and adjust a few parameters. For humanoid robots, an additional compensation of the robot's head motion is necessary.

3.6.2 Quantitative Evaluation

In the course of the optimization process described in Sect. 3.3.4 , an extensive evaluation of the self-localization precision – before and after the optimization – was carried out. The results have been published in [4] (optimized) and [22] (unoptimized).

An overview of all experiments and the self-localization precision achieved is given in Tab. 3.2. In all experiments in which the robot was walking around alone and scanning the environment for features (No. 1 – 3), the average localization error was significantly

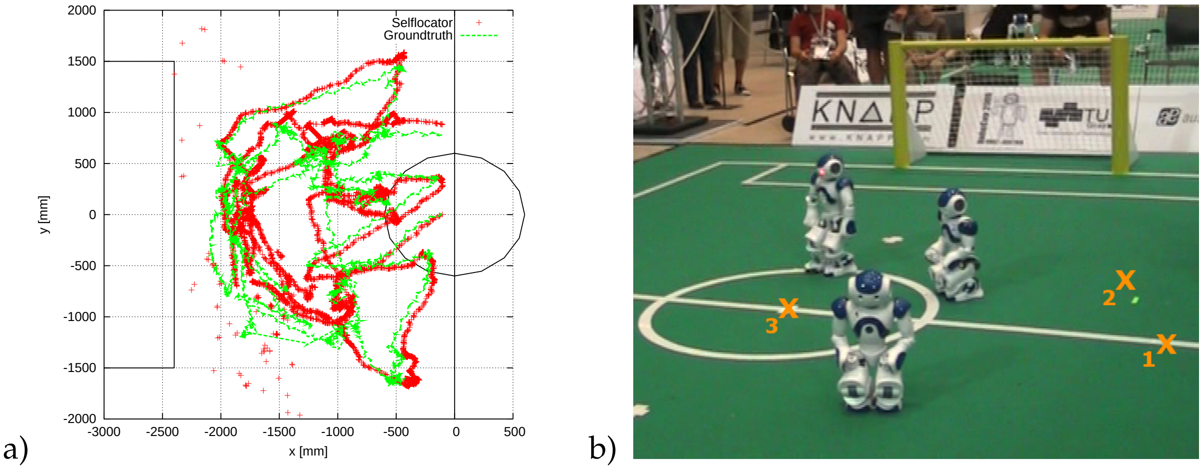


Figure 3.6: Evaluation of self-localization: a) Trajectory of a soccer-playing robot (experiment No. 7 in Tab. 3.2). The red crosses denote the estimate computed by the self-localization. The dashed green line depicts the ground truth. b) B-Human accomplishing the *Localization with Obstacle Avoidance Challenge* at RoboCup 2009, the numbered crosses denote the target positions in visiting order.

below 150 mm. Surprisingly, even when walking on a changed field (which was not modeled in software) – having a displaced goal (No. 4) or additional field lines (No. 5) – the error remained significantly below 200 mm.

The remaining experiments (No. 6 – 11) measured the precision during normal soccer play. They included the presence of other, adversarial robots, periods of time focusing on the ball instead of features for self-localization, as well as several tumbles. Due to these difficulties, the average error increased but still stayed below 250 mm for one opponent and around 300 mm for two opponents. The result of experiment 7 is illustrated in Fig. 3.6a. The experiments 9 and 11 result from the optimization process described in Sect. 3.3.4. In both cases, data from a log file not used for optimization was reprocessed.

The overall computing time of the system's cognitive part (i. e. image processing, state estimation, and action selection) has always to be significantly below 33 ms to keep up with the camera's frame rate of 30 Hz. This is why the self-localization implementation has been optimized to require only 6 ms on average.

3.6.3 Qualitative Evaluation

The approaches described in this chapter have not only been tested under laboratory conditions but have also been applied in actual RoboCup competitions. For this participation, no ground truth data for computing an average error exists. Hence, the overall team performance has to be considered as an indicator of quality.

The team B-Human has successfully participated in the RoboCup German Open 2009 and 2010 as well as in RoboCup 2009 and 2010, as described in detail in Sect. 2.1.3. During the soccer competitions, the precise and robust self-localization provided a huge benefit, not only for reliably kicking in the right direction but also for approaching the ball in a configuration allowing an immediate shoot, whether forward, sidewards, or even backwards. Additionally, the localization enabled an automatic robot placement before kickoff, which provides a clear advantage, as well as a proper goalie position adjustment.

In 2009, the team also participated in the *Localization with Obstacle Avoidance Challenge* that required a robot to walk as close as possible to three specified (but previously unknown) positions within a given amount of time without colliding with any obstacle [ROBOCUP SPL TECHNICAL COMMITTEE, 2009]. The setup of this challenge is shown in Fig. 3.6b. B-Human has been the only team to successfully finish this challenge. Further details about this specific challenge are given in B-Human's 2009 team report [51].

3.7 Object Tracking

State estimation in the context of this thesis mostly refers to self-localization. Nevertheless, keeping track of other objects in the robot's environment is also an important task. Corresponding works include tracking a ball (cf. Sect. 3.7.1) and other robots (cf. Sect. 3.7.2) as well as general obstacle modeling by ultrasound (cf. Sect. 3.7.3).

3.7.1 Ball Tracking

For playing soccer, knowledge about the ball is of major importance. As the ball is a unique object, which even has a unique color in some RoboCup leagues, its position can be computed straightforwardly. However, a state estimation process is necessary to compensate the measurement uncertainty, to determine the ball's velocity, and to keep track of its position when it is not within the current field of view.

In [27], the Augmented-MCL was transferred from self-localization to ball tracking. Representing the probability distribution as particle set (cf. Fig. 3.7a) and using the sensor resetting mechanisms for replacing samples provides a straightforward solution for reestablishing a proper state estimate in case of immediate changes, i. e. the retrieval of the ball at a position significantly different from the last seen position or the change in velocity after a kick. The ball's actual mode, which is either rolling or lying, is also part of the estimated state, in addition to the ball's position and velocity. This allows a more stable estimation over time in case of a small sample set as in many situations a ball's motion cannot reliably be distinguished from noise. The most recent development is an efficient model for ball collisions with the robot, allowing a reasonable estimation in case of dribbling situations [51].

3.7.2 Robot Tracking

Apart from the ball, other robots are the second kind of moving objects in the RoboCup scenario. Knowledge about their positions is necessary to play soccer at a reasonable level, for collaboration with teammates – e. g. passing and positioning – as well as for avoiding opponents – e. g. dodging or successful shots on goal. Whereas, in general, the positions and intentions of the teammates are known by communication, the opponent robots need to be perceived and tracked.

For the robot perception task, [WILKING and RÖFER, 2005] presented a solution based on decision tree learning, a less precise but more efficient approach – which is a further

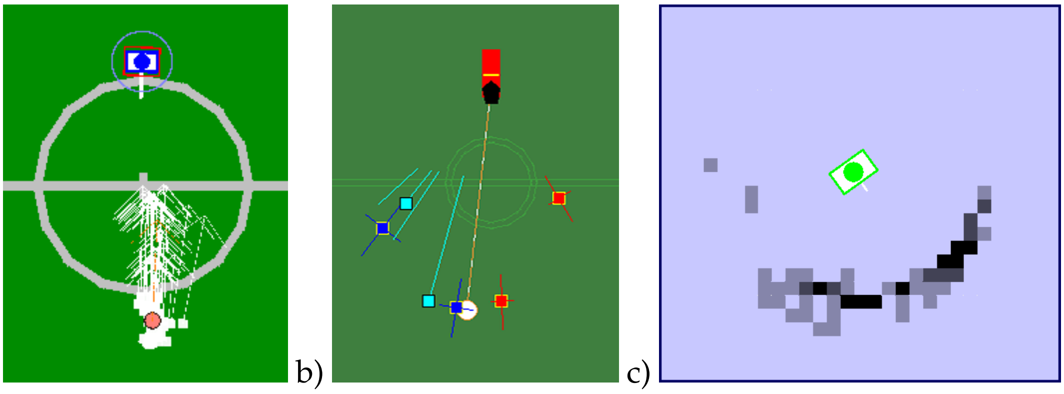


Figure 3.7: Object tracking: a) Ball samples moving towards a robot. The orange circle and arrow show the ball state estimate computed from the probability distribution. b) An AIBO robot estimates positions of other robots. The large dots indicate the positions of the hypotheses. The lines through the dots illustrate the uncertainty of the estimates. c) Occupancy grid for ultrasound-based general obstacle modeling. Darker cells indicate a higher likelihood for the presence of an obstacle.

development of the one depicted in Fig. 3.3b – based on a scan grid has been used by the GermanTeam [40]. An approach for perceiving and modeling general obstacles by considering them as a complement to the green floor has been presented by [HOFFMANN et al., 2005a].

In [9], the perceptions computed by the two latter approaches – which are both efficient but provide quite noisy results – have been used for modeling other robots. It was shown that it is possible to compute accurate position estimates of robots in the Four-Legged League (cf. Fig. 3.7b). The low quality of information that is caused by the low perceptual capabilities of the AIBO robot has been compensated by applying Kalman filter-based state estimation techniques.

3.7.3 Ultrasound Obstacle Modeling

To detect obstacles – which are other robots in most cases – in a robot's local environment, the Nao provides an attractive alternative to the camera: ultrasound sensors. This kind of equipment is an approved solution for measuring distances in robotics. Nevertheless, it has been replaced by laser range finders in many current systems as ultrasound has some significant weaknesses: depending on the surfaces in the robot's environment, false positives as well as false negatives might occur frequently, the sensor's resolution is very low, and in general, ultrasound has a comparably low measurement frequency. These drawbacks require the application of a modeling approach instead of a direct usage of the measurements.

In [51], such a model has been realized by using an occupancy grid, an approved approach in robotics that has been presented by [MORAVEC and ELFES, 1985] and that is also used for driving assistance [18] in the context of the smart wheelchair presented in Sect. 5.1. A visualization of the occupancy grid that models obstacles in a robot's local environment is shown in Fig. 3.7c. In combination with the self-localization, this component significantly contributed to the success in the 2009 RoboCup SPL *Localization with Obstacle Avoidance Challenge* (cf. Sect. 3.6.3).

3.8 Contributions of the Corresponding Publications

For successfully applying approved state estimation algorithms such as the Monte-Carlo Localization approach to a domain, the development of efficient and reliable sensor models is of major importance. Different contributions to this field have been made. In *Particle-Filter-Based Self-Localization Using Landmarks and Directed Lines* [17], a localization based on unique landmarks in combination with an efficient model for non-unique field lines has been presented. The field line model has been extended in *Particle Filter-based State Estimation in a Competitive and Uncertain Environment* [27], allowing a robust integration of only partially visible field elements. An additional contribution of this work was the transfer of the Augmented-MCL approach from self-localization to object tracking. In *Efficient and Reliable Sensor Models for Humanoid Soccer Robot Self-Localization* [22], an overview of the extraordinarily well performing self-localization of the successful B-Human team was given, including new contributions regarding the integration of more cues, the usage of a state-based sensor model and solutions to overcome the particle depletion problem in the robot soccer scenario.

Particle filter-based approaches always require a proper parameterization – for sensor models and dynamic models as well as for the configuration of the algorithm – to operate reliably. In *Optimizing Particle Filter Parameters for Self-Localization* [4], an approach for optimizing all relevant parameters by using the Particle Swarm Optimization algorithm was presented. The approach has been shown to be capable of finding a parameter set that leads to more precise position estimates than the previously used hand-tuned parameterization.

When applying MCL in the RoboCup context, multi-modal probability distributions occur frequently as samples might cluster at different positions within the state space. In *Pose Extraction from Sample Sets in Robot Self-Localization – A Comparison and a Novel Approach* [10], a robust and computationally efficient algorithm for extracting clusters from sample sets was presented.

Two further contributions of this thesis allow a direct usage of multiple hypotheses about a robot's pose: In *A Behavior Architecture for Autonomous Mobile Robots Based on Potential Fields* [8], a new modeling scheme for robot behaviors was presented. It combines already existing approaches for continuous navigation and action selection within one architecture. In *Entropy-based Active Vision for a Humanoid Soccer Robot* [19], an efficient approach for an active vision system supporting self-localization as well as ball tracking was presented. The originally computationally too expensive approach of entropy minimization was optimized – inter alia by facilitating the multiple robot pose hypotheses – to be executable in real-time even on a robot with limited resources such as the Nao.

In *SSL-Vision: The Shared Vision System for the RoboCup Small Size League* [20], a vision-based tracking system was presented. It is now the standard vision system for the RoboCup SSL. As in that league robot detection is based on specific patterns, the application can be used to track a variety of different objects if these are equipped with the correct patterns. Examples are given in [4] and [22], where the system was used to acquire ground truth data for robot state estimation evaluation.

In addition to [27], a second contribution to the area of object tracking was presented in *Integrating Simple Unreliable Perceptions for Accurate Robot Modeling in the Four-Legged League* [9]. By applying a Kalman filter-based state estimator, vision information of two different kinds – both imprecise but in different dimensions – was successfully fused to obtain precise position estimates in real-time.

An overall contribution of these publications is their successful application in real RoboCup tournaments, showing robust and precise state estimation in a competitive real-time scenario.

Chapter 4

Simulation

When working with robots, a simulation is often of significant importance as it supports the process of software development by providing a substitute for robots that are currently not at hand (e. g. broken or used by another person) or not able to endure extensive experiments. Furthermore, the execution of robot programs inside a simulator offers the possibility of directly debugging and testing them more conveniently. Moreover, a simulation might lead to better design decisions and cost savings by allowing the evaluation of different alternatives before the construction phase of a robot system.

In this chapter, works regarding the simulator *SimRobot* are presented, with a focus on the simulation of uncertainties that occur in the real world, in particular when operating a system close to its limits as it is often the case for soccer robots. This includes artifacts in images of fast moving cameras as well as the world's physical behavior during the execution of highly optimized robot motions.

4.1 State of the Art

In the past, several robot simulators have been developed with different focus on complexity, accuracy, and flexibility. This section provides a short overview of current related works on robot simulators for autonomous mobile robots, omitting simulators focussing on industrial environments, such as *RoboLogix* [LOGIC DESIGN INC., 2010].

Webots by [MICHEL, 2004] is a popular commercial robotic simulator for research and education developed by Cyberbotics Ltd. It provides a 3D physics simulation and already features a number of different robot models such as Nao, Sony AIBO, Khepera, or Pioneer2. For the Nao robot, Cyberbotics even organizes a Webots-based online soccer simulation league called *Robotstadium*. A similar simulation project was launched by Microsoft under the name *Microsoft Robotics Developer Studio* [JACKSON, 2007]. This product, originally also commercial but now free, also features components for robot programming and control.

A free simulation environment is *USARSim* (Unified System for Automation and Robot Simulation) by [CARPIN et al., 2007], which is based on – and can thus can only be used with – the commercial *Unreal Tournament* game engine. It has originally been developed in the context of the RoboCup Rescue Virtual Robot Competition – and thus derived its acronym from Urban Search and Rescue Simulation – but can actually be used for general purpose simulation. Another robot simulation project derived from an existing, different application is *Blender for Robotics*. Blender is an advanced open-source 3D graphics application that can be used for a variety of complex modeling and animation tasks. In [BRUYNINCKX, 2004], its usage for robot modeling and simulation as well as necessary extensions are discussed.

An open-source multi-robot simulator also featuring a graphical interface and dynamics simulation is *Gazebo* [KOENIG and HOWARD, 2004]. It has been developed in cooperation with the *Player* and *Stage* projects [GERKEY et al., 2003]; Player is a networked device server, and Stage is a simplified 2D simulator for large populations of mobile robots. In this context, Gazebo serves as a replacement for Stage for scenarios that require a higher fidelity.

Two simulators developed in the context of RoboCup are *UCHILSIM* [ZAGAL and DEL SOLAR, 2005] and *SimSpark* [KÖGLER and OBST, 2004]. The former is an application designed specifically for the Four-legged League. It is limited to the simulation of an AIBO robot but features an integration with the developer's *Back to Reality* [ZAGAL et al., 2004] approach that allows co-learning simulation as well as robot control parameters. This approach is also closely related to the works presented in Sect. 4.4 and Sect. 4.5. The *Spark Generic Physical Multiagent Simulator* (SimSpark) is the official simulator of the RoboCup 3D Simulation League (cf. Sect. 1.2.3). It is able to physically simulate arbitrary robot models – the league's current standard robot is the Nao – but only has abstract sensors, providing high-level information about the soccer environment. Although this project currently focuses primarily on multi-agent research and not on robotics, it is intended to close the gap between the 3D simulation and real robot soccer matches in the future [MAYER et al., 2007].

Most of these simulators share a set of common features: accelerated 3D graphics, a physics engine, support for user-defined robot models, and support for a variety of different sensors and actuators. A standard approach for physics simulation is the integration of an already existing physics engine. One popular choice is the *Open Dynamics Engine* (ODE) by [SMITH, 2007] – among others, it is used by SimRobot as described in Sect. 4.2.2 – as it is an open source project. In this context, the *Multi-Robot-Simulation-Framework* (MuRoSimF) by [FRIEDMANN et al., 2008] is an explicit exception, as this project's unique characteristic is a simulation engine that allows adaptable levels of abstraction.

As described in the following section, SimRobot also shares all these features of current robot simulations. Significant contributions distinguishing it from other applications are the possibility of simulating image distortions (cf. Sect. 4.3), an interface for the optimization of the world's physical parameters (cf. Sect. 4.5), as well as an XML-based modeling language shared with other simulators (cf. Sect. 4.2.3).

A more detailed overview of the current state of the art – including SimRobot – has recently been published by [FRIEDMANN, 2009].

4.2 Development of SimRobot

SimRobot is able to simulate arbitrary user-defined robots in three-dimensional space. To allow an extensive flexibility in building accurate models, a variety of different generic bodies, sensors, and actuators has been implemented. Its XML-based modeling language is shared with other simulators, allowing an exchange of models. Subsequent to different versions supporting only a kinematic simulation and basic graphics capabilities, the

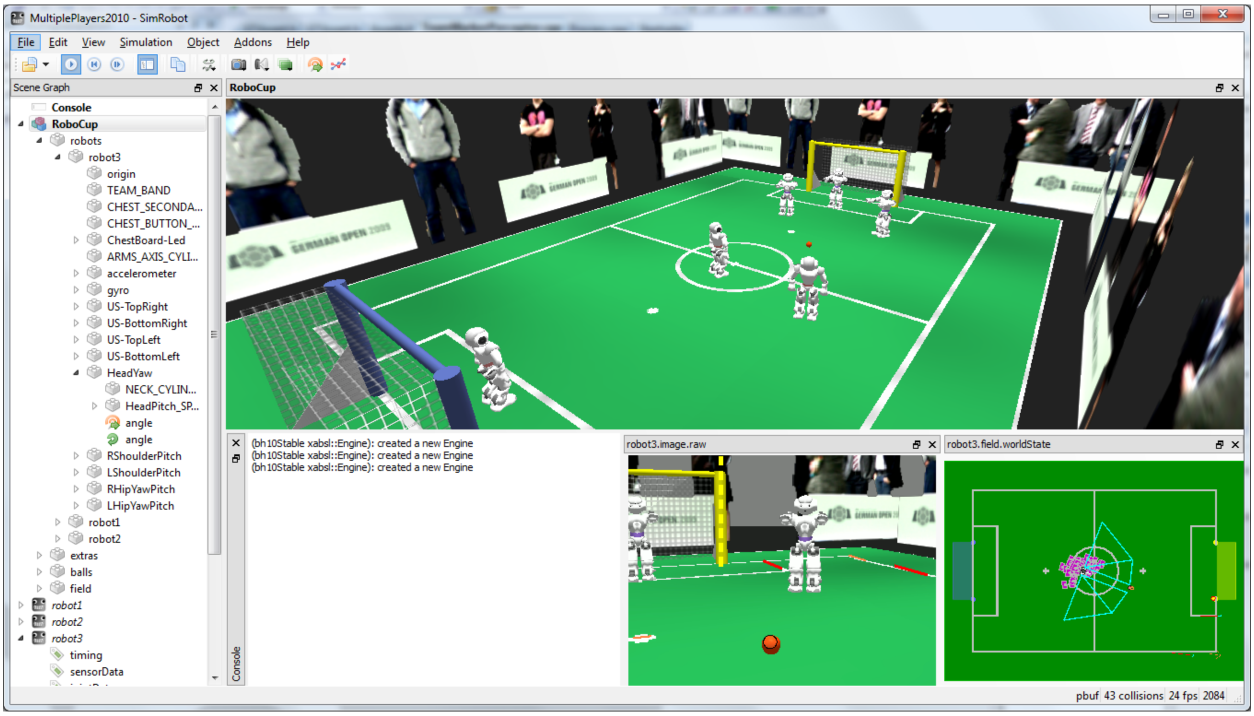


Figure 4.1: SimRobot simulating the robots of the B-Human team playing soccer in a RoboCup environment. The tree on the left provides direct access to all elements of the simulated world that is shown in the main sub-window. The sub-windows below are a console for user interaction, the image made by a simulated robot's camera (additionally containing the visualization of some perceptions), and a view showing a visualization of the robot's current world state estimate.

current SimRobot includes modern 3D graphics acceleration as well as a physical model that is based on rigid body dynamics. During the past years, SimRobot has been used to simulate several different robots, ranging from smart wheelchairs to humanoid soccer robots (cf. Fig. 4.1).

4.2.1 History

The first version of SimRobot was developed by [SIEMS et al., 1994]. Although it did not support any dynamics or accelerated graphics, SimRobot was already a feature-rich application regarding the support of a variety of sensors and user interaction, running on different operating systems such as Windows, OS/2 and UNIX. The primary application was the simulation of an autonomous wheelchair [RÖFER, 1998]. The GermanTeam's first simulation of an AIBO robot was also based on this version of SimRobot [30].

As the base of this application was no longer contemporary and lead to severe performance problems in case of simulated environments containing multiple robots equipped with cameras (e. g. a team of AIBO robots), SimRobot's kernel was reimplemented and based on accelerated OpenGL graphics for visualization as well as for sensor data generation. The first version was released as part of the GermanTeam software [48]. In the meantime, an intermediately developed Java version of SimRobot [MEYER and DUNEKAMP, 1999] was abandoned.

The last major simulation kernel change was the integration of rigid body dynamics [11] through the usage of the Open Dynamics Engine (cf. Sect. 4.2.2 as well as Sect. 4.4), providing a simulation fidelity that is high enough for walking robots. After several years in which SimRobot had been developed further for the Microsoft Windows platform only, a migration to a platform independent GUI toolkit was made, allowing the current version to run at least on Windows, Linux, and MacOS.

The set of robots that has been simulated by SimRobot during the past years includes different models of autonomous wheelchairs, an intelligent walker, an omni-directionally driving SSL robot, the AIBO versions ERS-210 and ERS-7, self-assembled humanoid soccer robots, the Nao, and the *Bremen Ambient Assisted Living Lab* (BAALL) (cf. Sect. 5.4.1).

4.2.2 Current State

SimRobot consists of several modules that are dynamically linked to a single application. This approach, which is different from many other client/server-based simulation concepts, has been chosen as it offers the possibility of halting or stepwise executing the whole simulation without any concurrencies. It also allows a more comprehensive debugging of the executed robot software. The main components of SimRobot are the *SimRobot core*, the *simulation scene*, and the *controller*.

The SimRobot core is the most important part of the application. It models the robots and the environment, simulates sensor readings, and executes commands given by the controller or the user. Even most parts of the visualization and the user interface have recently been integrated into the core. The specification of the robots and the environment, i. e. the *simulation scene*, is modeled via an external XML file (cf. Sect. 4.2.3) and loaded at runtime. When specifying a scene, the user may choose from a variety of sensor classes (i. e. camera, two kinds of distance sensors, touch and force sensors, accelerometer, gyroscope, and actuator status), actuators (i. e. hinge, ball and socket joint, wheel suspension, and slider), and rigid bodies (box, sphere, cylinder, and complex meshes). The two latter classes are more or less directly mapped to the according ODE representations. Most of these elements as well as some extensions to overcome limitations of the physics engine are described in detail in [11].

The user interface of SimRobot – shown in Fig. 4.1 – has been designed to allow as much visualization and interaction as possible as well as to be flexible enough to handle simulations of different kinds of environments. Therefore, a tree of all objects of the scene is the starting point for all user operations. Each node of that tree may be selected to open a view for that kind of object. In case of actuators (e. g. a hinge joint), a control for direct manipulation is opened. For sensors, several different visualization modes are implemented. Through this concept, it is also possible to open several views of arbitrary subsets of the scene graph. Furthermore, it is possible to interactively drag and drop and rotate objects inside the scene or to apply a momentum to an object (e. g. to let a ball roll). This is quite useful to arrange different settings while testing, for instance, a robot's behavior. To add own views to a scene (e. g. the field view in Fig. 4.1), an interface for user-defined views has been implemented, which allows the definition of drawings from inside the controller.

The controller implements the sense-think-act cycle. In each simulation step, it is called by the simulation, reads the available sensors, plans the next action, and sets the actuators to the desired states. A controller that is suitable for the currently loaded scene has to be provided by the user, i. e. it has to contain or to interface the control software of the simulated robots, which usually is as similar as possible to the software running on the real robots.

Comprehensive descriptions of SimRobot have been published in [11] and [24]. SimRobot is an open source project, its releases are described in Sect. A.2.

4.2.3 The RoSiML Language

As aforementioned, the specification of the robots and the environment, in the context of SimRobot named as *scene*, is modeled via an external XML file and loaded at runtime. The use of an external specification language allows a simpler modeling of scenes – even for people without programming skills – without any modifications or extensions of the source code of the simulator.

Together with researchers from the Fraunhofer Institute for Autonomous Intelligent Systems, the specification language *RoSiML* (Robot Simulation Markup Language) [GHAZI-ZAHEDI et al., 2005] has been developed. It is part of a joint effort to establish common interfaces for robot simulations. The aim is to exchange components between different simulators and to allow the migration of robot models among simulators without complicated adaptions. The language itself has been specified in XML Schema.

In addition to SimRobot, the *Yet Another Robot Simulator* (YARS) application [ZAHEDI et al., 2008], which is focussed on requirements originating from the field of evolutionary robotics, uses RoSiML as description language. The SimSpark project also realized an import function for RoSiML models [MAYER et al., 2007]. As described in detail in Sect. 5.4.2, *YAMAMOTO* (*Y*et *A*nother *MA*p *MO*deling *TO*olkit) by [STAHL and HAUPERT, 2006] allows the input as well as the export of RoSiML.

4.3 Simulating Image Distortions

One characteristic trait of all simulations is that they can only approximate the real world, an inherent deficit called *Reality Gap*. This affects all aspects of simulations: the level of detail as well as the characteristics of sensors and actuators (cf. Sect. 4.5). However, for many applications, the gap is of minor relevance as long as the simulated robot system still performs in a reasonable way.

As already described in Sect. 1.1.3, the characteristics of some sensors matter to an extent that leads to a significant mismatch between simulation and reality. Such a mismatch complicates the development of robot software as faults are likely to occur whether on the robot or in the simulation, possibly leading to a suboptimal overall solution. Hence, it is important that a simulation generates all the sensor distortions that are actively handled by the robot control software.

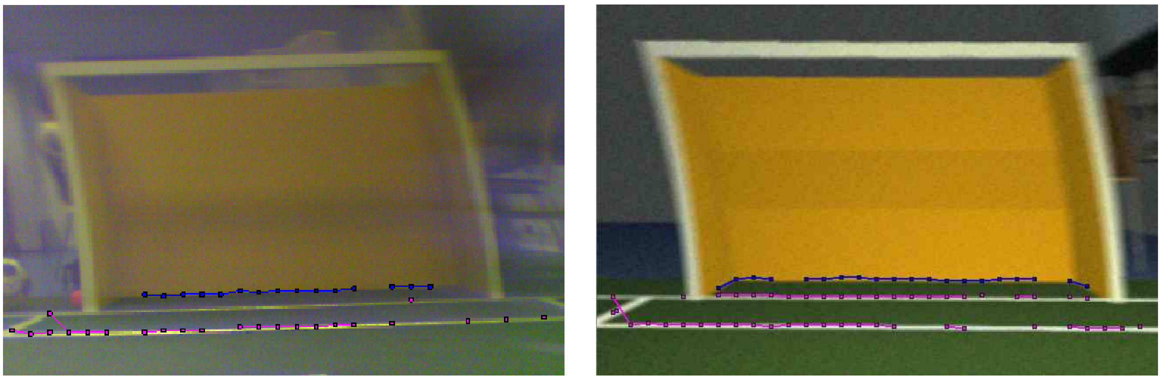


Figure 4.2: Comparison of a real image (on the left) and an equivalent simulated image. The dots and lines represent percepts of the robot's vision system.

Figure 4.2 shows an example of a typical distortion, the image is taken by a CMOS camera that is part of the head of a humanoid robot. While the blurry impression is caused by motion blur, the distortion that bends and squeezes the yellow goal is produced by the so-called rolling shutter. Instead of taking complete images at a certain point in time, a rolling shutter takes an image pixel by pixel, row by row. Thus the last pixel of an image is taken significantly later than the first one. If the camera is moving, this results in image distortions. Compensating this effect becomes significantly important when working with robots that move their cameras fast or operate in a rapidly changing environment. In the RoboCup domain, this particularly affects robots with pan-tilt heads, e. g. in the Standard Platform and the Humanoid League. To overcome problems that arise from these disturbances, different compensation methods have been developed, e. g. by [NICKLIN et al., 2007] or by [RÖFER, 2008]. The necessity of simulating these disturbances can be derived from this explicit handling by the software of different teams.

Within the robotics community, a variety of robot simulators is currently used, as described in Sect. 4.1. Most of these applications are able to simulate camera images at a high level of detail, for instance by simulating lights and shadows. But so far, none of them addresses the problem of image disturbances. In [16], an according SimRobot extension for simulating common image disturbances, i. e. the rolling shutter effect and motion blur, has been presented, a resulting image is shown in Fig. 4.2 . By exploiting the features of modern graphics hardware, these disturbances can be simulated in real-time.

4.4 Excursion: Motion Design for Walking Robots

Creating and learning motions for robots is closely connected with their simulation in two ways: on the one hand, a simulator is a necessary tool for preliminary checking new, possibly unstable motions to save the original hardware; on the other hand, several motions – particularly learned walking gaits – raise severe problems regarding a stable parameterization of the simulated world's physics. The latter problem has been addressed by works presented in Sect. 4.5. The application of a simulation for motion design as well as for learning has been a part of multiple works implemented in the course of this thesis.

In the RoboCup context, parameter optimization for walking gaits is a common practice to achieve fast, flexible, and robust motions. The probably fastest gait for the AIBO ERS-210 has been learned by [RÖFER, 2005] using an evolutionary algorithm. This work also presented a swift walk for the AIBO ERS-7 that was clearly outperformed by [HEBBEL et al., 2006] afterwards. The latter gait as well as the applied evolutionary approach have been the base for the research presented in Sect. 4.5. The Humanoid KidSize domain's currently fastest robot has been trained by [HEMKER et al., 2009].

In [25], we presented a gait optimization approach based on Particle Swarm Optimization [KENNEDY and EBERHART, 1995] (which has also been used in the context of self-localization as described in Sect. 3.3.4) that was able to reach probably the highest known walking speed of a fully-equipped Kondo-based RoboCup robot on a standard field. As such a process usually requires a large number of experiments that take time and wear out the robot, a first optimization stage was carried out in SimRobot. This allowed us to find a proper parameterization for the PSO algorithm that subsequently lead to a faster convergence of the optimization process on the real robot.

For the Nao robot, research regarding walking as well as kicking has been conducted. In [21], a robust closed-loop gait as well as an analytical solution for the inverse kinematics of this platform are presented. Several approaches previously applied to the modeling of gaits have been transferred to the task of robust and adaptive kicking in [15]. The according motion design application is embedded in SimRobot and relies on the physical model of the Nao robot.

4.5 Optimizing Simulation Parameters

As aforementioned, actuator performance is a crucial aspect in the RoboCup domain. In many cases, high robot velocities are achieved through the application of optimization algorithms. One common attribute of these algorithms is the strong exploitation of the environment's features, i. e. certain characteristics of the motors or the properties of the ground structure. This leads to control trajectories that strongly differ from the resulting trajectories of the real robot joints, as shown on the left side of Fig. 4.3. For robot simulations, particularly when working with legged robots that have a high number of degrees of freedom, this requires a proper parametrization to simulate actuators that behave close to real ones. Otherwise, the simulated robot might not only behave unrealistic but could fail completely, i. e. the robot tumbles over.

All previously described simulation applications (cf. Sect. 4.1) allow a detailed specification of the environment. But all of them demand the user to do this manually, which might become an exhausting task given the high number of environmental parameters. Additionally, a once working parameter set is not guaranteed to be compatible with a different walking gait learned at a later point in time. However, from a user's perspective, a detailed and accurate physics model is not important but the realistic behavior of the simulated robot is.

In [7], a general multi-staged process that minimizes the reality gap between real and simulated robots regarding the behavior of actuators and their interaction with the environment has been presented. As an example, an AIBO ERS-7 and its according SimRobot model – whose physics parameters have previously been hand-tuned – have been used

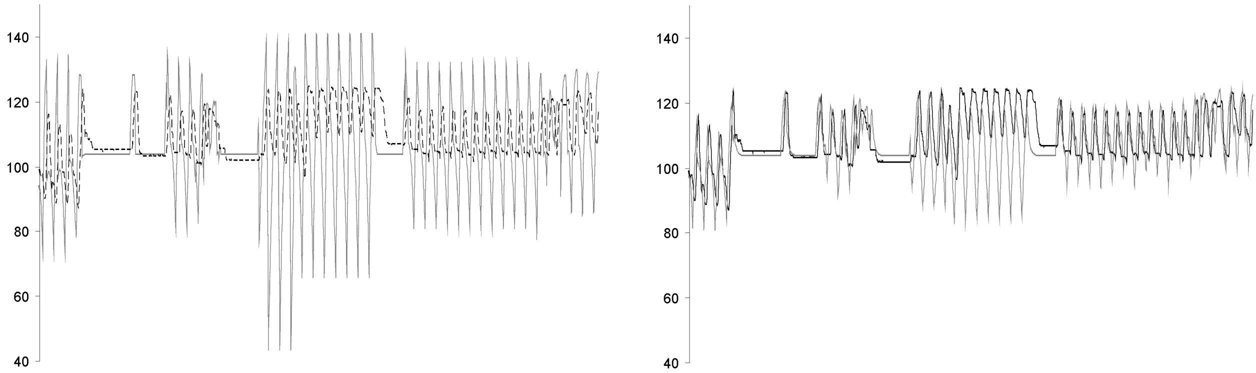


Figure 4.3: Left: an extract from the controlled (solid) and the measured joint angle (dashed) of an AIBO's front right knee joint while playing soccer. Right: an extract from the real (dark) and the simulated joint movement (light) of the same joint after learning the (preliminary) maximum joint forces.

because the simulated robot was not able to reproduce the walking behaviors executed by a real robot during a match. The optimization has been carried out by using an evolution strategy [BEYER and SCHWEFEL, 2002] with self-adaptation [BEYER, 1996], the same algorithm that has already been used to learn the robot's gait.

The final result of this optimization – which was a simulated world that was able to reproduce a number of different walking gaits of the real robot – has been measured to be more accurate than the model delivered with the commercial Webots simulator, which uses – similar to many other simulation applications – the same physics engine as SimRobot. An intermediate result of the optimization process is depicted on the right side of Fig. 4.3.

This approach is similar to the *Back to Reality* concept by [ZAGAL et al., 2004]. Nevertheless, their focus is the co-evolution of robot control and simulation parameters, rather than the detailed optimization of a general purpose model. Although keeping the reality gap small, their approach did not yield any high performance walking gait and thus remained quite far away from any physical limits that particularly stress a simulation's behavior.

4.6 Contributions of the Corresponding Publications

SimRobot is an open source robot simulation that has capabilities similar to most other current applications – including commercial ones – in this category. In addition to research regarding the realism of the simulation, significant features are SimRobot's generic approach, comprehensive support for user interaction, as well as a description language supported by other applications. Detailed overviews have been published in *SimRobot - A General Physical Robot Simulator and Its Application in RoboCup* [11] as well as in *SimRobot - Development and Applications* [24].

To achieve a higher level of realism, reflecting uncertainties occurring in a dynamic environment, research regarding camera sensors as well as actuator behavior has been conducted. The contribution of *Real-time Simulation of Motion-based Camera Disturbances*

[16] is a hardware-accelerated simulation of distortions, i. e. the rolling shutter effect and motion blur, which particularly occur in robot systems using cameras on a pan-tilt head, such as humanoid soccer robots. In *Automatic Parameter Optimization for a Dynamic Robot Simulation* [7], it was shown that it is possible to learn a parameterization for a common physics engine that is able to cope with the extreme walking gaits of an AIBO robot playing soccer. The result of this work was even able to outperform the model of a common commercial simulator.

The availability of a robust simulation was also the base for different contributions regarding motion modeling and learning. In *Gait Optimization on a Humanoid Robot using Particle Swarm Optimization* [25] the probably fastest gait for a humanoid Kondo robot with RoboCup equipment was learned using Particle Swarm Optimization. For the Nao robot, *A Robust Closed-Loop Gait for the Standard Platform League Humanoid* [21] presented the first analytical inverse kinematics solution as well as an actively balanced gait. Extending the research regarding robot walking, *Kicking a Ball – Modeling Complex Dynamic Motions for Humanoid Robots* [15] presented a robust and adaptive approach for designing motions for the Nao robot.

Chapter 5

Beyond RoboCup

As described in Sect. 1.2, the goal of the RoboCup initiative is to foster research in artificial intelligence and robotics. It is not intended to be a closed system but to evaluate technology from other domains as well as to transfer research results to applications outside the RoboCup context.

In this chapter, such transfers of localization and simulation approaches and experiences from RoboCup Soccer to the areas of rehabilitation robotics and Ambient Assisted Living are described. A similar transfer – outside the context of this thesis – has already been realized in the past by migrating the soccer robot's control architecture [RÖFER, 2003] to the rehabilitation platforms.

5.1 Rehabilitation Robots

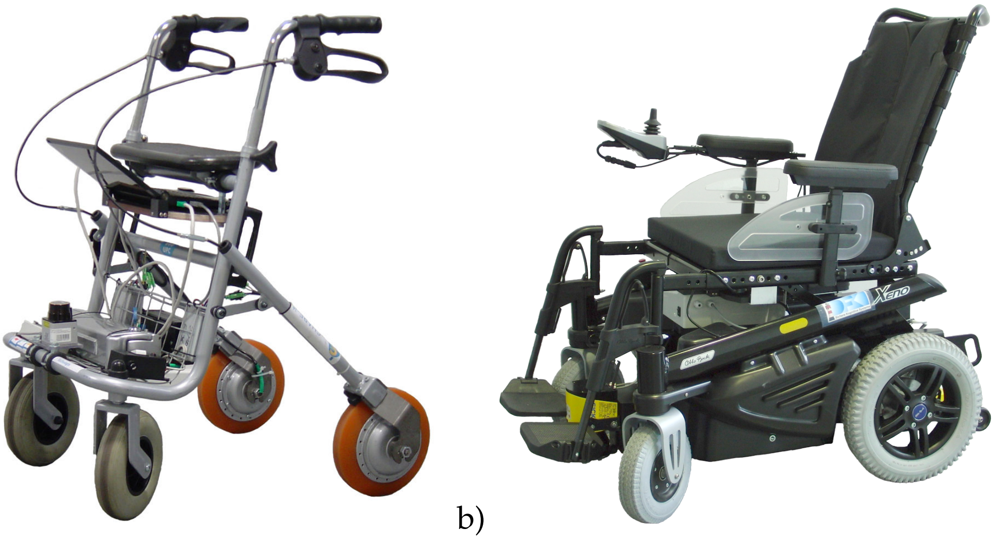
One particular class of service robots are rehabilitation robots. These are mostly automated wheelchairs equipped with intelligent services, such as autonomous navigation, obstacle avoidance, or novel user interfaces for control. Recently, also walkers have been equipped with similar techniques and capabilities.

5.1.1 Rolland – A Smart Wheelchair

Automated wheelchairs that are equipped with sensors and a data processing unit constitute a special class of wheeled mobile robots, termed *smart wheelchairs* in general literature overviews, e. g. as provided by [SIMPSON, 2005] and [LANKENAU and RÖFER, 2000].

The particular platform that has been used in the context of this thesis is the wheelchair *Rolland*, which has been a base for research for over 15 years [RÖFER et al., 2009] at the Universität Bremen and the DFKI Safe and Secure Cognitive Systems group. Different versions of Rolland were built over the years, as well as special input devices that are used with Rolland. Contributions were made to the state of the art in the areas of self-localization, mapping, navigation, safety, and shared control.

The current Rolland is based on the commercial power-wheelchair *Xeno*, manufactured by the German company Otto Bock Mobility Solutions, shown in Fig. 5.1b. The original wheelchair is additionally equipped with two laser range sensors, wheel encoders, and an on-board computer. Various driving assistant modes for the wheelchair Rolland are being developed and evaluated to compensate for diminishing physical and cognitive



a) b)

Figure 5.1: Rehabiliation platforms used in the context of this thesis: a) the *iWalker* and b) the smart wheelchair *Rolland*.

faculties: the *safety assistant* brakes in time; the *driving assistant* avoids obstacles and facilitates passing through a door (cf. Sect. 5.2.2); the *navigation assistant* guides along a route or drives autonomously (cf. Sect. 5.2.3); a head joystick (cf. Sect. 5.2.2) and speech recognition [MANDEL and FRESE, 2007] ease interaction. An overview of all assistants is given in [KRIEG-BRÜCKNER et al., 2010].

5.1.2 iWalker – An Intelligent Walker

The *iWalker* is the prototype of an intelligent walker equipped with electric brakes, wheel encoders, a laser range sensor, and a small control PC (cf. Fig. 5.1a). A first version of the iWalker was developed by [CORTÉS et al., 2008] in the context of the EU project *SHARE-it* (cf. Sect. A.4). A detailed descriptions of enhancements – which have partially been realized in the context of this thesis – is given in [26].

The system's setup is quite analogous to Rolland. Thus the iWalker provides a similar variety of services to users with walking impairments, also with diminished sight. By controlling the electric brakes, the *walking assistant* is able to help the user keeping clear of obstacles; the *navigation assistant* guides the user along a route, either by displaying an arrow on the screen, or by softly braking the wheels, indicating the direction (cf. Sect. 5.2.3). In the variant by [CORTÉS et al., 2008], the iWalker also provides additional safety regulating the speed when going down a slope, and actively helps going up.

Similar research regarding an intelligent walker has also been conducted by [GLOVER et al., 2003], providing additional services such as parking itself and returning to the user.

5.2 Indoor Navigation

One of the common capabilities of a mobile service robot is to autonomously navigate to a given position within the known environment. This demands the accomplishment of the

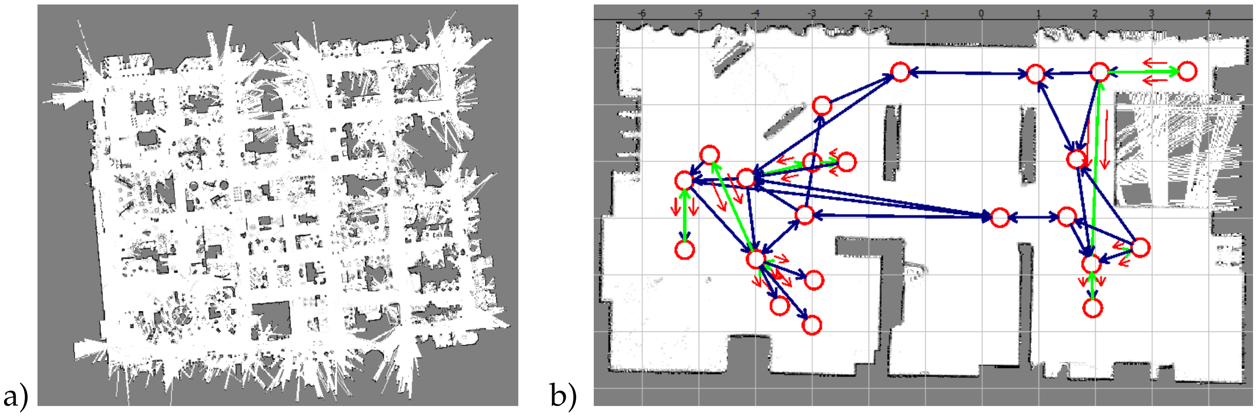


Figure 5.2: a) Large-scale mapping. A map created from data recorded by Rolland exploring a hall of the REHACARE, an international trade fair for rehabilitation, care, prevention, and integration. b) Small-scale navigation. A map of the BAALL with routes for navigation. Circles denote labeled places the system can navigate to, arrows denote paths and driving directions between these places.

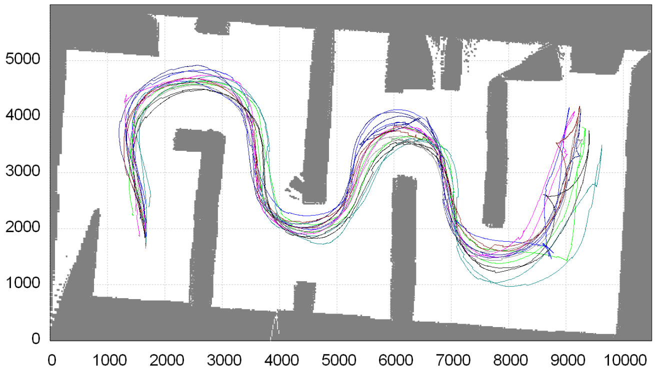
two sub-tasks self-localization and path planning. In the context of this thesis, these two components have been developed for Rolland as well as for the iWalker. In both cases, the primary application domain was indoor navigation, even though, small outdoor scenarios have also been successfully accomplished. A different localization approach for significantly larger domains – i. e. in which the robot travels distances of several kilometers – is described in Sect. 5.3.

5.2.1 Mapping and Localization

The simultaneous localization and mapping (SLAM) problem has been intensively studied by the robotics community in the past, a comprehensive overview from a user's perspective is given by [Frese et al., 2010]. We decided not to reinvent the wheel but to reuse and adapt an existing solution. We have chosen the *GMapping* suite from [Stachniss et al., 2010] since it was originally developed for systems similar to Rolland and the iWalker – wheeled robots equipped with a laser range finder – and has shown the ability to robustly cope with environments up to the size of large complexes of buildings. The GMapping approach uses a Rao-Blackwellized particle filter in which each particle carries an individual grid map of the environment, a detailed description is given in [Grisetti et al., 2007].

Because of GMapping's performance and accuracy improving features, it is a reasonable choice to also use it for self-localization. For this purpose, the original software – which has not been made for this kind of application – has been adapted in the course of this thesis.

The mapping and localization components are currently used for Rolland as well as for the iWalker (cf. Sect. 5.2.3) and have already been applied successfully to a number of different places: the *Bremen Ambient Assisted Living Lab* (BAALL, cf. Sect. 5.4.1 as well as Fig. 5.2b), different levels of an office building, a huge shopping mall, different fairs (cf.

a)


b)


Figure 5.3: a) Evaluation of a driving experiment. The map shows a specific course set up for driving experiments; each colored line depicts the logged trajectory driven by a participating subject using a joystick and being supported by the driving assistant. b) Way-finding assistance on the iWalker. The arrow on the screen depicts the required driving direction towards a previously selected goal.

Fig. 5.2a), as well as to several virtual environments simulated in SimRobot. The changes made to GMapping have been released as open source, cf. Sect. A.4.

5.2.2 Evaluation Support for Shared Control Experiments

For the Rolland platform, a number of different input devices and assistant modes for safe driving with shared control have been developed in recent years. This includes navigation with a brain-computer interface [14] as well as steering by a head-joystick that maps a user's head posture to a driving direction [MANDEL et al., 2007]. The latter has also been combined with an assistance mode for collision-free driving [18], the so-called *driving assistant*.

To compare and evaluate the precision and efficiency of such input and driving modes, experiments involving a detailed tracking of the wheelchair's driving directions are essential. To evaluate the works presented in [14] and [18], specific driving courses have been set up and mapped using the approach presented in the previous section. During each experiment, the according self-localization module recorded the wheelchair's pose within the course. This allowed a comprehensive analysis of situations and configurations that pose problems for users or assistants. In Fig. 5.3a, the result of such an experiment is depicted.

5.2.3 Autonomous Navigation and Way-finding Assistance

As both, Rolland as well as the iWalker, are capable of self-localization in a previously mapped environment, a navigation system supporting both platforms has been developed. It enables the wheelchair to autonomously drive to a selected position out of a set of previously configured positions. Since the iWalker currently cannot drive, it relies on being pushed by a user who receives way-finding assistance by a displayed navigation arrow (as depicted in Fig. 5.3b) or is guided by braking [26].

All valid ways along which the robot can drive are modeled as a route graph [KRIEG-BRÜCKNER et al., 2005] that is completely specified before the start of the system and is not changed during normal operation. The manual creation of the graph – which has been realized as an interactive view within the SimRobot application – ensures that the system stays on routes that are known to be free of hidden (i. e. imperceptible to the current sensorial equipment) obstacles. An example graph describing routes for navigation within the BAALL is shown in Fig. 5.2b.

Given the position of the robot as well as a destination, which can be selected in various ways, e. g. speech or via a touch screen, planning a path is reduced to a graph search problem. This is a trivial task in comparison to efficiently planning a path in continuous space. For this purpose, the A* algorithm [HART et al., 1968] was applied as it is quite fast (in our application by using a metric distance heuristics) and proven to always find the shortest path. To ensure that there is always a path between two random places (and the user does not get stuck in a dead end), the route graph editor already checks the connectivity of the graph during its creation.

For driving (or braking) along the selected path, a component based on the works described in [18] is used, compensating imprecisely modeled routes close to obstacles and ensuring that no accompanying or crossing persons are hit.

The navigation system has already successfully been used on both platforms at a variety of places including the BAALL, which also features additional services depending on the positions and planned routes of the rehabilitation robots (cf. Sect. 5.4.1). One special demonstration has been performed during the CeBIT fair in 2009. Rolland was used as a tour guide, autonomously driving users to a number of fair booths selected by them via a multi-modal dialog system developed by [JIAN, 2008].

5.3 Outdoor Localization in Road Networks

For self-localization in large outdoor scenarios – including travel distances of several kilometers – the previously described approach, which is based on grid maps created by measurements from laser range finders, is not feasible anymore. However, for this kind of scenario, an approved solution already exists in terms of the Global Positioning System (GPS) that can be considered as being the de-facto standard for localization in road networks, delivering a reasonable precision at most places and being available through a huge variety of devices. Nevertheless, if signal reception is disturbed, no position estimates can be computed anymore.

In [13], an approach for localization in road networks has been presented that is based on a particle filter computing estimates relying on basic environmental information: the road network's structure and its height profile. Algorithmically, the approach is almost identical to the one used in the RoboCup context (cf. Sect. 3.2.3). For keeping track of different alternative positions (e. g. at crossings) it also uses the clustering approach presented in Sect. 3.4.

However, the approach has to use a sensor model completely different from those described in Sect. 3.3.1. Elevation is measured by a barometer and compared to a model based on the free *Shuttle Radar Topography Mission* (SRTM) data set [RABUS et al., 2003]. The road network information has been obtained from the *Open Street Map* project [OPEN-

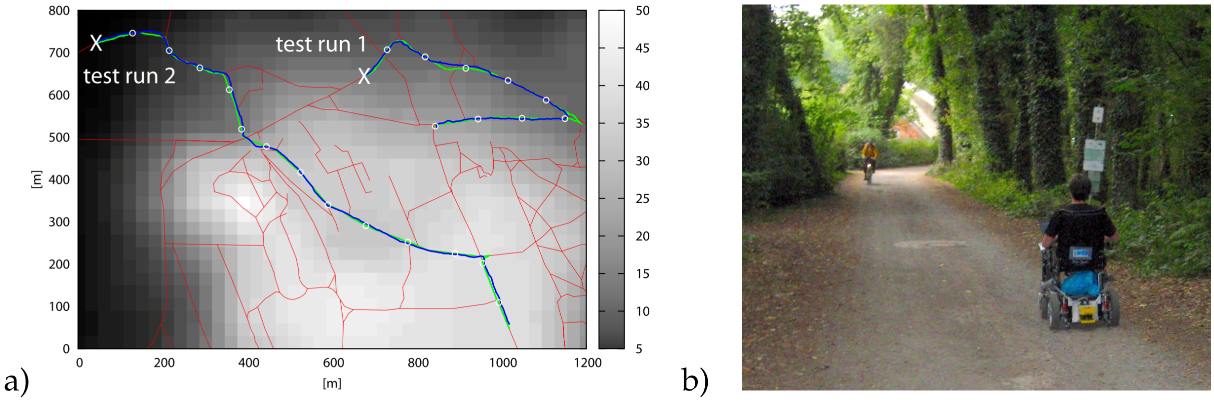


Figure 5.4: Outdoor localization with Rolland: a) Reference trajectories (blue) and estimated trajectories (green) of two experimental test runs conducted in the town of Worpswede. The underlying elevation information is illustrated by shaded $30m \times 30m$ squares. Heights are given in m. b) Driving on a country lane next to a forest.

STREETMAP, 2009]. Similar to the state-based sensor models described in Sect. 3.3.2, the context of the roads – it is assumed that a driver does not leave the road – is used in the sensor model although the roads cannot be measured directly. This partially compensates for the rudimentary odometry model that only provides a traveled distance without any direction information.

The proper functioning of this approach has demonstrated in experiments in which the wheelchair Rolland was driven through a small town along different routes (cf. Fig. 5.4). As these experiments only featured routes of very few kilometers, additional experiments over a total of about 100 km using an accordingly equipped bicycle have been conducted recently.

5.4 Simulating an Ambient Assisted Living Environment

The automated wheelchair Rolland has been simulated by SimRobot since the simulator's earliest versions. Although the application's focus is on mobile robots, mostly static environments with actuated elements, e. g. a room with doors, can be simulated without any difficulty as well. Therefore, SimRobot has been used – in combination with an external modeling toolkit – to simulate an Ambient Assisted Living environment to evaluate the maneuverability of Rolland inside.

5.4.1 The Bremen Ambient Assisted Living Lab

The Bremen Ambient Assisted Living Lab (BAALL) is an apartment suitable for the elderly and people with physical or cognitive impairments [BAALL, 2010]. With a size of 60 m^2, it comprises all necessary conditions for trial living, intended for two persons. It is situated at the Bremen site of the German Research Center for Artificial Intelligence (DFKI). The goal of the project is to investigate how the living environments of seniors to-be can be instrumented with infrastructures that allow incremental upgrades with user assistance systems, as required in their future. A comprehensive description is given by [Krieg-Brückner et al., 2009].

One particular focus of BAALL is to compensate physical impairments of the user through mobility assistance and adaptable furniture. Both rehabilitation robots – Rolland as well as the iWalker (cf. Sect. 5.1) – assist their user to safely navigate within the lab [Krieg-Brückner et al., 2010]. The lab has also been equipped with electronic sliding doors that will open automatically – based on the planned driving route (cf. Sect. 5.2.3) – to let the wheelchair pass through. In order to allow a wheelchair-dependent user to stay together with their non-impaired partner in the same living environment, user-adaptable furniture has been installed and can be adjusted to fit the different physical requirements.

5.4.2 Interactive Evaluation of Configurations

Modeling the layout of an apartment – including different furniture configurations and several moving elements – is a non-trivial task, particularly if accessibility has to be assured e. g. for wheelchairs such as Rolland. In [12], we described a two-staged example of such a task by connecting two different applications: In the first step, the YAMAMOTO toolkit [Stahl and Haupert, 2006] was used to efficiently model the building structure and to plan the furnishing of the environment in 3D. Afterwards, the desired configuration was exported to SimRobot, allowing a realistic evaluation of the physical configuration by interactively driving a simulated wheelchair through the environment.

YAMAMOTO (*Y*et *A*nother *MA*p *MO*deling *TO*olkit) is a graphical editor that was developed for the geometric modeling of co-located buildings and intelligent environments in 3-D. The toolkit has been designed to efficiently model the building's structure from floor plans. Furthermore, YAMAMOTO supports the planning and design of intelligent environments by providing various sensor and actuator elements [Stahl, 2009].

In a first step, the complete BAALL environment – including the furniture and the sliding doors – has been modeled with YAMAMOTO, mainly for visualization and documentation purposes. In order to simulate the dynamic aspect of the sliding doors and to assess the maneuverability of the wheelchair in presence of added shelves, YAMAMOTO was extended to create RoSiML (cf. Sect. 4.2.3) code that specifies a fully functional physical model of the environment. Finally, the physical model of Rolland was added to the SimRobot scene. The model provides a realistic driving behavior and can be interactively controlled by the user. The simulated BAALL environment allows the evaluation of the wheelchair's mobility in bottlenecks, such as the modified kitchen (cf. Fig. 5.5).

5.5 Contributions of the Corresponding Publications

In the two publications *Controlling an Automated Wheelchair via Joystick/Head-Joystick Supported by Smart Driving Assistance* [18] and *Navigating a Smart Wheelchair with a Brain-Computer Interface Interpreting Steady-State Visual Evoked Potentials* [14], new input devices – each combined with driving assistance – for an automated wheelchair have been presented. The localization components developed in the course of this thesis contributed the base for a detailed evaluation of these approaches.

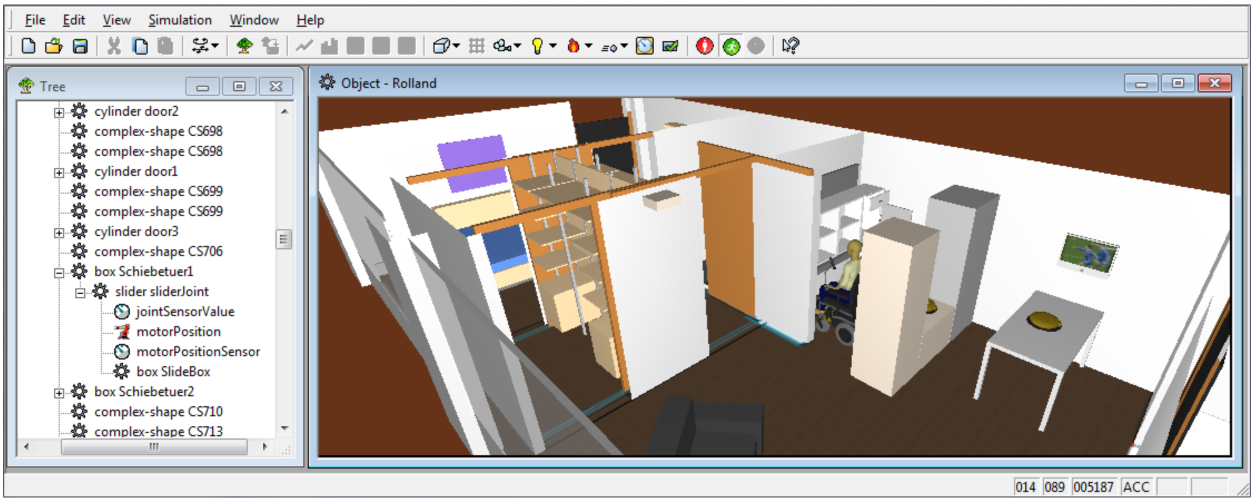


Figure 5.5: Assessing the maneuverability of Rolland within a dynamic simulation of the BAALL.

In *iWalker – An Intelligent Walker providing Services for the Elderly* [26], an overview of this platform including the different assistants is given. The major contribution within the context of this thesis is a robust, efficient, and traceable approach for indoor navigation that has already been applied to a number of different environments.

Complementing the contributions regarding indoor localization, *Particle Filter-based Position Estimation in Road Networks using Digital Elevation Models* [13] presents an approach for outdoor localization that does not rely on GPS but instead utilizes the robot's spatial context as well as measurements from quite basic sensors. Particularly the usage of height information provided by a barometer is a novel contribution in this area.

In *Modeling and Simulating Ambient Assisted Living Environments – A Case Study* [12], the flexibility and interoperability of the SimRobot application has been shown. The main contribution of this publication is a first step towards an efficient process for modeling and evaluating new configurations in an Ambient Assisted Living scenario including mobility assistants.

Chapter 6

Conclusion

In this thesis, I presented a number of works about simulation and state estimation in the domain of autonomous mobile robots. A common feature of these works is the treatment of uncertainties that occur in real, dynamic environments.

Additionally, these works deal with robustness and efficiency. The aim of this is to render an application of the developed approaches in highly dynamic, competitive scenarios such as RoboCup competitions possible. In recent years, I participated in different RoboCup teams, which used different robot platforms. The state estimation approaches presented in this thesis, including self-localization and object tracking, were applied to these platforms. Overall, these teams became world champion and German champion multiple times.

The simulator SimRobot was developed with a focus on RoboCup robots but is not limited to any special kind of mobile land robot. At the moment, SimRobot is the major development tool for different robotics projects in the DFKI Safe and Secure Cognitive Systems department. As simulations are conducted offline, efficiency is not needed to such an extent as on real robots. Nevertheless, the application is currently able to simulate multiple soccer robots as well as different rehabilitation robots on a modern standard computer in real-time.

RoboCup should not be self-contained. That is why the experiences that were made and the algorithms that were developed in the course of this thesis have successfully been transferred to the fields of rehabilitation robotics and Ambient Assisted Living. In addition to their simulation, the target platforms – an automated wheelchair as well as an intelligent walker – have been enabled to localize and navigate in the real world. This was demonstrated in different indoor and outdoor environments.

Most of the presented works are settled in the domain of RoboCup Soccer. Hence, their future development is highly influenced by the incremental changes of the environment and the rules made by RoboCup Federation towards its 2050 goal. In Sect. 2.3, I presented a possible development path alongside the RoboCup competitions, anticipating parts of that final goal.

However, in order to make robots successfully play soccer against humans, humans should be involved as soon as possible. As indicated by the annual matches of humans against the current Middle Size League champion, the performance of current robot systems is far too low (cf. Sect. 2.4) to be somewhat competitive. This is why upcoming works should focus on games that are much simpler but involve humans as opponents or partners. A few of these games already exist. A modified version of soccer that is based on *Segway* vehicles was developed by [ARGALL et al., 2006]. In this game, autonomous

Segways play together and against humans standing on Segways. As a result, the actuation disadvantages of the robots are compensated. The table soccer robot *KiRo* [WEIGEL and NEBEL, 2003] also plays a limited variant of soccer to which humans only have access via rotary handles. This machine plays on a remarkable level and is even commercially available. Two robots that play simpler but more direct games are the *RoboKeeper* [FRAUNHOFER IML and 4ATTENTION GMBH & CO. KG, 2010], a robot goalkeeper that is able to parry penalty shots by professional soccer players, and DLR's *Justin*, a humanoid torso that is able to catch balls thrown by humans [BIRBACH et al., 2011].

Such games provide a challenging environment and thus a new strong benchmark – in which the human's performance is the gold standard – for cognitive systems, beyond the scope of the robot vs. robot games in RoboCup. Subsequent to the works presented in this thesis, I started the development of a 2-DOF game robot, which is intended to play a simple ball game with humans. This robot will be based upon the experiences gained in the RoboCup competitions and use an updated version of the tracking approach presented in Sect. 2.3.3, provided by [BIRBACH and FRESE, 2009]. From a methodical point of view, real-world robustness will be improved by incorporating context (cf. Sect. 2.3.2), similar to the self-localization approach described in Sect. 3.3.2.

Overall, the domain of sport robotics is an emerging field of research. However, it is not expected to reach a commercial relevance comparable to that of service or industrial robotics, except for certain niches such as the entertainment sector. Nevertheless, sport robotics contributes to the general field of robotics by developing and benchmarking approaches that are capable of robustly dealing with highly dynamic environments in real-time. Afterwards, these approaches can be transferred, for instance, to service robots – such as the rehabilitation assistants described in Sect. 5 – that fulfill practically relevant tasks in the real world.

List of Publications by the Author

Reviewed Publications

In Scientific Journals

[1] FRESE, UDO, TIM LAUE, OLIVER BIRBACH and THOMAS RÖFER: *(A) Vision for 2050 – Context-Based Image Understanding for a Human-Robot Soccer Match*. Electronic Communications of the EASST. To appear.

My share is 40%
This paper is an updated and extended version of [5].

[2] HADDADIN, SAMI, TIM LAUE, UDO FRESE, SEBASTIAN WOLF, ALIN ALBU-SCHÄFFER and GERD HIRZINGER: *Kick it with Elasticity: Safety and Performance in Human-Robot Soccer*. Robotics and Autonomous Systems, Special Issue on Humanoid Soccer Robots, 57(8):761–775, 2009.

My share is 25%
This paper is partially based on [6]. Additional contributions are the co-design of experiments and the development of a common thread given various initially independent works.

At Conferences

[3] BIRBACH, OLIVER, JÖRG KURLBAUM, TIM LAUE and UDO FRESE: *Tracking of Ball Trajectories with a Free Moving Camera-Inertial Sensor*. In IOCCHI, LUCA, HITOSHI MATSUBARA, ALFREDO WEITZENFELD and CHANGJIU ZHOU (editors): *RoboCup 2008: Robot Soccer World Cup XII*, volume 5399 of *Lecture Notes in Artificial Intelligence*, pages 49–60. Springer, 2009.

My share is 5%
I co-initiated this work and established its RoboCup context. I revised the paper and presented the work at the RoboCup Symposium in Suzhou, China.

[4] BURCHARDT, ARMIN, TIM LAUE and THOMAS RÖFER: *Optimizing Particle Filter Parameters for Self-Localization*. In SOLAR, JAVIER RUIZ DEL, ERIC CHOWN and PAUL G. PLOEGER (editors): *RoboCup 2010: Robot Soccer World Cup XIV*, volume 6556 of *Lecture Notes in Artificial Intelligence*, pages 145–156. Springer, 2011.

My share is 30%
This publication resulted from a diploma thesis that I initiated and co-advised. The optimization is based on my particle filter implementation. I presented this work at the RoboCup Symposium in Singapore.

[5] FRESE, UDO and TIM LAUE: *(A) Vision for 2050 - The Road Towards Image Understanding for a Human-Robot Soccer Match*. In FILIPE, JOAQUIM, JUAN ANDRADE-CETTO and JEAN-LOUIS FERRIER (editors): *ICINCO 2008, Proceedings of the Fifth International Conference on Informatics in Control, Automation and Robotics, Robotics and Automation 1*, pages 317–322, Funchal, Madeira, Portugal, 2008. INSTICC Press.

My share is 50%
The ideas presented in this position paper are a result of joint discussions. I presented the paper at the ICINCO conference in Funchal, Madeira, Portugal.

[6] HADDADIN, SAMI, TIM LAUE, UDO FRESE and GERD HIRZINGER: *Foul 2050: Thoughts on Physical Interaction in Human-Robot Soccer*. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007)*, pages 3243 – 3250, San Diego, CA, USA, 2007.

My share is 35%
I co-analyzed and categorized physical interactions in real soccer matches. I established the paper's RoboCup context.

[7] LAUE, TIM and MATTHIAS HEBBEL: *Automatic Parameter Optimization for a Dynamic Robot Simulation*. In IOCCHI, LUCA, HITOSHI MATSUBARA, ALFREDO WEITZENFELD and CHANGJIU ZHOU (editors): *RoboCup 2008: Robot Soccer World Cup XII*, volume 5399 of *Lecture Notes in Artificial Intelligence*, pages 121–132. Springer, 2009.

My share is 50%
The optimization process is the result of joint discussions. I adapted SimRobot accordingly and identified all relevant parameters for optimization. I presented the work at the RoboCup Symposium in Suzhou, China.

[8] LAUE, TIM and THOMAS RÖFER: *A Behavior Architecture for Autonomous Mobile Robots Based on Potential Fields*. In NARDI, DANIELE, MARTIN RIEDMILLER, CLAUDE SAMMUT and JOSÉ SANTOS-VICTOR (editors): *RoboCup 2004: Robot Soccer World Cup VIII*, volume 3276 of *Lecture Notes in Artificial Intelligence*, pages 122–133. Springer, 2005.

My share is 95%
I developed the architecture that is described in this paper and presented the work at the RoboCup Symposium in Lisbon, Portugal.

[9] LAUE, TIM and THOMAS RÖFER: *Integrating Simple Unreliable Perceptions for Accurate Robot Modeling in the Four-Legged League*. In LAKEMEYER, GERHARD, ELIZABETH SKLAR, DOMENICO G. SORRENTI and TOMOICHI TAKAHASHI (editors): *RoboCup 2006: Robot Soccer World Cup X*, volume 4434 of *Lecture Notes in Artificial Intelligence*, pages 474–482. Springer, 2007.

My share is 95%
I developed the approach that is described in this paper and presented the work at the RoboCup Symposium in Bremen, Germany.

[10] LAUE, TIM and THOMAS RÖFER: *Pose Extraction from Sample Sets in Robot Self-Localization - A Comparison and a Novel Approach*. In PETROVIĆ, IVAN and ACHIM J.

Lilienthal (editors): *Proceedings of the 4th European Conference on Mobile Robots - ECMR'09*, pages 283–288, Mlini/Dubrovnik, Croatia, 2009.

My share is 90%
I developed the approach presented in this paper, wrote the major share of this paper, and presented the approach at the ECMR conference in Mlini, Croatia.

[11] Laue, Tim, Kai Spiess and Thomas Röfer: *SimRobot - A General Physical Robot Simulator and Its Application in RoboCup*. In Bredenfeld, Ansgar, Adam Jacoff, Itsuki Noda and Yasutake Takahashi (editors): *RoboCup 2005: Robot Soccer World Cup IX*, volume 4020 of *Lecture Notes in Artificial Intelligence*, pages 173–183. Springer, 2006.

My share is 50%
I have been the maintainer of the SimRobot application. I implemented a major share of the application and wrote a major share of the paper. I presented the work at the RoboCup Symposium in Osaka, Japan.

[12] Laue, Tim and Christoph Stahl: *Modeling and Simulating Ambient Assisted Living Environments – A Case Study*. In Augusto, Juan Carlos, Juan M. Corchado, Paulo Novais and Cesar Analide (editors): *Ambient Intelligence and Future Trends – International Symposium on Ambient Intelligence (ISAmI 2010)*, volume 72 of *Advances in Intelligent and Soft Computing*, pages 217–220. Springer, 2010.

My share is 50%
I wrote the major share of this paper. I am one of the major authors of the RoSiML language (and its parser) which enables the process presented in this paper.

[13] Mandel, Christian and Tim Laue: *Particle Filter-based Position Estimation in Road Networks using Digital Elevation Models*. In *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5744–5749, Taipei, Taiwan, 2010.

My share is 45%
The approach is a result of joint discussions. I developed major parts of the particle filter implementation and co-conducted the experiments.

[14] Mandel, Christian, Thorsten Lüth, Tim Laue, Thomas Röfer, Axel Gräser and Bernd Krieg-Brückner: *Navigating a Smart Wheelchair with a Brain-Computer Interface Interpreting Steady-State Visual Evoked Potentials*. In *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1118 – 1125, St. Louis, MO, USA, 2009.

My share is 10%
I provided the ground truth for the evaluation of the presented approaches.

[15] Müller, Judith, Tim Laue and Thomas Röfer: *Kicking a Ball – Modeling Complex Dynamic Motions for Humanoid Robots*. In Solar, Javier Ruiz del, Eric Chown and Paul G. Ploeger (editors): *RoboCup 2010: Robot Soccer World Cup XIV*, volume 6556 of *Lecture Notes in Artificial Intelligence*, pages 109–120. Springer, 2011.

My share is 10%
This publication resulted from a diploma thesis that I co-advised. I initiated the publication and performed major editing.

[16] PACHUR, DENNIS, TIM LAUE and THOMAS RÖFER: *Real-time Simulation of Motion-based Camera Disturbances*. In IOCCHI, LUCA, HITOSHI MATSUBARA, ALFREDO WEITZENFELD and CHANGJIU ZHOU (editors): *RoboCup 2008: Robot Soccer World Cup XII*, volume 5399 of *Lecture Notes in Artificial Intelligence*, pages 591–601. Springer, 2009.

My share is 25%
This publication resulted from a diploma thesis that I initiated and co-advised. I wrote parts of the paper and presented the work at the RoboCup Symposium in Suzhou, China.

[17] RÖFER, THOMAS, TIM LAUE and DIRK THOMAS: *Particle-filter-based Self-Localization using Landmarks and Directed Lines*. In BREDENFELD, ANSGAR, ADAM JACOFF, ITSUKI NODA and YASUTAKE TAKAHASHI (editors): *RoboCup 2005: Robot Soccer World Cup IX*, volume 4020 of *Lecture Notes in Artificial Intelligence*, pages 608–615. Springer, 2006.

My share is 15%
I implemented and described the perception of the beacons that are used for the presented localization approach.

[18] RÖFER, THOMAS, CHRISTIAN MANDEL and TIM LAUE: *Controlling an Automated Wheelchair via Joystick/Head-Joystick Supported by Smart Driving Assistance*. In *Proceedings of the 2009 IEEE 11th International Conference on Rehabilitation Robotics*, pages 743–748, Kyoto, Japan, 2009.

My share is 15%
I provided the ground truth for the evaluation of the presented approaches. In addition, I co-designed the experimental environment and conducted part of the experiments.

[19] SEEKIRCHER, ANDREAS, TIM LAUE and THOMAS RÖFER: *Entropy-based Active Vision for a Humanoid Soccer Robot*. In SOLAR, JAVIER RUIZ DEL, ERIC CHOWN and PAUL G. PLOEGER (editors): *RoboCup 2010: Robot Soccer World Cup XIV*, volume 6556 of *Lecture Notes in Artificial Intelligence*, pages 1–12. Springer, 2011. Best Paper Award Winner.

My share is 25%
This publication resulted from a diploma thesis that I initiated and co-advised. The approach facilitates my particle filter implementation, its optimization partly relies on my cluster extraction approach.

[20] ZICKLER, STEFAN, TIM LAUE, OLIVER BIRBACH, MAHISORN WONGPHATI and MANUELA VELOSO: *SSL-Vision: The Shared Vision System for the RoboCup Small Size League*. In BALTES, JACKY, MICHAIL G. LAGOUDAKIS, TADASHI NARUSE and SAEED SHIRY (editors): *RoboCup 2009: Robot Soccer World Cup XIII*, volume 5949 of *Lecture Notes in Artificial Intelligence*, pages 425–436. Springer, 2010.

My share is 30%
I co-developed the new calibration approach that is presented in this paper. I co-initiated the SSL-Vision project and was – as member of the league's executive committee – responsible for the introduction as official standard system.

At Workshops

[21] GRAF, COLIN, ALEXANDER HÄRTL, THOMAS RÖFER and TIM LAUE: *A Robust Closed-Loop Gait for the Standard Platform League Humanoid*. In ZHOU, CHANGJIU, ENRICO PAGELLO, EMANUELE MENEGATTI, SVEN BEHNKE and THOMAS RÖFER (editors): *Proceedings of the Fourth Workshop on Humanoid Soccer Robots in conjunction with the 2009 IEEE-RAS International Conference on Humanoid Robots*, pages 30 – 37, Paris, France, 2009.

My share is 5%
This work emerged from development activities of the B-Human team which I co-lead.

[22] LAUE, TIM, THIJS JEFFRY DE HAAS, ARMIN BURCHARDT, COLIN GRAF, THOMAS RÖFER, ALEXANDER HÄRTL and ANDRIK RIESKAMP: *Efficient and Reliable Sensor Models for Humanoid Soccer Robot Self-Localization*. In ZHOU, CHANGJIU, ENRICO PAGELLO, EMANUELE MENEGATTI, SVEN BEHNKE and THOMAS RÖFER (editors): *Proceedings of the Fourth Workshop on Humanoid Soccer Robots in conjunction with the 2009 IEEE-RAS International Conference on Humanoid Robots*, pages 22 – 29, Paris, France, 2009.

My share is 35%
This work emerged from development activities of the B-Human team which I co-lead. I am the main developer of the self-localization approach presented in this paper. I initiated this publication and performed major editing.

[23] LAUE, TIM and THOMAS RÖFER: *Getting Upright: Migrating Concepts and Software from Four-Legged to Humanoid Soccer Robots*. In PAGELLO, ENRICO, CHANGJIU ZHOU and EMANUELE MENEGATTI (editors): *Proceedings of the Workshop on Humanoid Soccer Robots in conjunction with the 2006 IEEE International Conference on Humanoid Robots*, Genoa, Italy, 2006.

My share is 50%
I was responsible for the migration of a number of software components to the humanoid platform. I initiated this publication and wrote half of the text. I presented the work at the Workshop on Humanoid Soccer Robots in Genoa, Italy.

[24] LAUE, TIM and THOMAS RÖFER: *SimRobot - Development and Applications*. In AMOR, HENI BEN, JOSCHKA BOEDECKER and OLIVER OBST (editors): *The Universe of RoboCup Simulators - Implementations, Challenges and Strategies for Collaboration. Workshop Proceedings of the International Conference on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2008)*, Venice, Italy, 2008.

My share is 60%
This publication is an overview of SimRobot's status to which both authors contributed to a similar extent. I presented the publication at the workshop in Venice, Italy.

[25] NIEHAUS, CORD, THOMAS RÖFER and TIM LAUE: *Gait Optimization on a Humanoid Robot using Particle Swarm Optimization*. In PAGELLO, ENRICO, CHANGJIU ZHOU, EMANUELE MENEGATTI and SVEN BEHNKE (editors): *Proceedings of the Second Workshop on Humanoid Soccer Robots in conjunction with the 2007 IEEE-RAS International Conference on Humanoid Robots*, Pittsburgh, PA, USA, 2007.

My share is 10%
This publication resulted from a diploma thesis that I initiated and co-advised. I created the simulated robot model used in the first optimization stage. I presented the work at the Workshop on Humanoid Soccer Robots in Pittsburgh, PA, USA.

Unreviewed Publications

At Conferences

[26] RÖFER, THOMAS, TIM LAUE and BERND GERSDORF: *iWalker - An Intelligent Walker providing Services for the Elderly*. In *Technically Assisted Rehabilitation 2009*, Berlin, Germany, 2009.

My share is 40%
I developed and described the iWalker's localization and navigation capabilities.

At Workshops

[27] LAUE, TIM and THOMAS RÖFER: *Particle Filter-based State Estimation in a Competitive and Uncertain Environment*. In *Proceedings of the 6th International Workshop on Embedded Systems*. Vaasa, Finland, 2007.

My share is 65%
I developed major parts of the described self-localization approach as well as the ball tracking solution. I wrote a major share of this paper. I presented the work at the workshop in Vaasa, Finland.

Workshop Proceedings

[28] VISSER, UBBO, SAHAR ASADI, TIM LAUE and N. MICHAEL MAYER (editors): *Proceedings of the AAMAS 2010 Workshop on Agents in Real-time and Dynamic Environments*, Toronto, Canada, 2010.

My share is 25%
As one of four co-editors, I was involved in all aspects of the workshop organization, including the compilation of the proceedings.

RoboCup Team Descriptions

[29] BRUNN, RONNIE, UWE DÜFFERT, MATTHIAS JÜNGEL, TIM LAUE, MARTIN LÖTZSCH, SEBASTIAN PETTERS, MAX RISLER, THOMAS RÖFER, KAI SPIESS and ANDREAS SZTYBRYC: *GermanTeam 2001*. In BIRK, ANDREAS, SILVIA CORADESCHI and

SATOSHI TADOKORO (editors): *RoboCup 2001: Robot Soccer World Cup V*, volume 2377 of *Lecture Notes in Artificial Intelligence*, pages 705–708. Springer, 2002.

[30] BURKHARD, HANS-DIETER, UWE DÜFFERT, JAN HOFFMANN, MATTHIAS JÜNGEL, MARTIN LÖTZSCH, RONNIE BRUNN, MARTIN KALLNIK, NICOLAI KUNTZE, MICHAEL KUNZ, SEBASTIAN PETTERS, MAX RISLER, OSKAR VON STRYK, NILS KOSCHMIEDER, TIM LAUE, THOMAS RÖFER, KAI SPIESS, ARTHUR CESARZ, INGO DAHM, MATTHIAS HEBBEL, WALTER NOWAK and JENS ZIEGLER: *GermanTeam RoboCup 2002*, 2002. Only available online: http://www.informatik.uni-bremen.de/kogrob/papers/GermanTeam2002.pdf.

[31] BURKHARD, HANS-DIETER, UWE DÜFFERT, MATTHIAS JÜNGEL, MARTIN LÖTZSCH, NILS KOSCHMIEDER, TIM LAUE, THOMAS RÖFER, KAI SPIESS, ANDREAS SZTYBRYC, RONNIE BRUNN, MAX RISLER and OSKAR VON STRYK: *GermanTeam RoboCup 2001*, 2001. Only available online: http://www.informatik.uni-bremen.de/kogrob/papers/GermanTeam2001report.pdf.

[32] CZARNETZKI, STEFAN, MATTHIAS HEBBEL, SÖREN KERNER, TIM LAUE, WALTER NISTICÓ and THOMAS RÖFER: *BreDoBrothers - Team Description for RoboCup 2008*. In IOCCHI, LUCA, HITOSHI MATSUBARA, ALFREDO WEITZENFELD and CHANGJIU ZHOU (editors): *RoboCup 2008: Robot Soccer World Cup XII Preproceedings*, Suzhou, China, 2008. RoboCup Federation.

[33] DÜFFERT, UWE, MATTHIAS JÜNGEL, TIM LAUE, MARTIN LÖTZSCH, MAX RISLER and THOMAS RÖFER: *GermanTeam 2002*. In KAMINKA, GAL A., PEDRO U. LIMA and RAUL ROJAS (editors): *RoboCup 2002: Robot Soccer World Cup VI Preproceedings*, Fukuoka, Japan, 2002. RoboCup Federation.

[34] KURLBAUM, JÖRG, TIM LAUE, BJÖRN LÜCK, BJÖRN MOHRMANN, MARTIN POLOCZEK, DSCHEN REINECKE, TIM RIEMENSCHNEIDER, THOMAS RÖFER, SIMON HENDRIK and UBBO VISSER: *Bremen Small Multi-Agent Robot Team (B-Smart) Team Description for RoboCup 2004*. In NARDI, DANIELE, MARTIN RIEDMILLER, CLAUDE SAMMUT and JOSÉ SANTOS-VICTOR (editors): *RoboCup 2004: Robot Soccer World Cup VIII Preproceedings*, Lisbon, Portugal, 2004. RoboCup Federation.

[35] KURLBAUM, JÖRG, TIM LAUE, FLORIAN PENQUITT and MARIAN WEIRICH: *B-Smart - Team Description for RoboCup 2005*. In BREDENFELD, ANSGAR, ADAM JACOFF, ITSUKI NODA and YASUTAKE TAKAHASHI (editors): *RoboCup 2005: Robot Soccer World Cup IX Preproceedings*, Osaka, Japan, 2005. RoboCup Federation.

[36] LAUE, TIM, ARMIN BURCHARDT, KAI CIERPKA, SEBASTIAN FRITSCH, NILS GÖDE, KAMIL HUHN, TEODOSIY KIRILOV, BIANCA LASSEN, EYVAZ LYATIF, MARKUS MIEZAL, MARKUS MODZELEWSKI, ULFERT NEHMIZ, MALTE SCHWARTING, ANDREAS SEEKIRCHER and RUBEN STEIN: *B-Smart - Team Description for RoboCup 2008*. In IOCCHI, LUCA, HITOSHI MATSUBARA, ALFREDO WEITZENFELD and CHANGJIU ZHOU (editors): *RoboCup 2008: Robot Soccer World Cup XII Preproceedings*, Suzhou, China, 2008. RoboCup Federation.

[37] LAUE, TIM, ARMIN BURCHARDT, KAI CIERPKA, SEBASTIAN FRITSCH, NILS GÖDE, KAMIL HUHN, TEODOSIY KIRILOV, BIANCA LASSEN, EYVAZ LYATIF, MARKUS MIEZAL, MALTE SCHWARTING, ANDREAS SEEKIRCHER and RUBEN STEIN: *B-Smart - Team Description for RoboCup 2007*. In VISSER, UBBO, FERNANDO RIBEIRO, TAKESHI OHASHI and FRANK DELLAERT (editors): *RoboCup 2007: Robot Soccer World Cup XI Preproceedings*, Atlanta, GA, USA, 2007. RoboCup Federation.

[38] LAUE, TIM, ARMIN BURCHARDT, SEBASTIAN FRITSCH, SVEN HINZ, KAMIL HUHN, TEODOSIY KIRILOV, ALEXANDER MARTENS, MARKUS MIEZAL, ULFERT NEHMIZ, MALTE SCHWARTING and ANDREAS SEEKIRCHER: *B-Smart - Extended Team Description for RoboCup 2009*. In BALTES, JACKY, MICHAIL G. LAGOUDAKIS, TADASHI NARUSE and SAEED SHIRY (editors): *RoboCup 2009: Robot Soccer World Cup XIII Preproceedings*, Graz, Austria, 2009. RoboCup Federation.

[39] LAUE, TIM, TORBEN SCHINDLER, FLORIAN PENQUITT, ARMIN BURCHARDT, OLIVER BIRBACH, CARSTEN ELFERS and KAI STOYE: *B-Smart - Team Description for RoboCup 2006*. In LAKEMEYER, GERHARD, ELIZABETH SKLAR, DOMENICO G. SORRENTI and TOMOICHI TAKAHASHI (editors): *RoboCup 2006: Robot Soccer World Cup X Preproceedings*, Bremen, Germany, 2006. RoboCup Federation.

[40] RÖFER, THOMAS, JÖRG BROSE, EIKE CARLS, JAN CARSTENS, DANIEL GÖHRING, MATTHIAS JÜNGEL, TIM LAUE, TOBIAS OBERLIES, SVEN OESAU, MAX RISLER, MICHAEL SPRANGER, CHRISTIAN WERNER and JÖRG ZIMMER: *GermanTeam 2006*. In LAKEMEYER, GERHARD, ELIZABETH SKLAR, DOMENICO G. SORRENTI and TOMOICHI TAKAHASHI (editors): *RoboCup 2006: Robot Soccer World Cup X Preproceedings*, Bremen, Germany, 2006. RoboCup Federation.

[41] RÖFER, THOMAS, JÖRG BROSE, DANIEL GÖHRING, MATTHIAS JÜNGEL, TIM LAUE and MAX RISLER: *GermanTeam 2007*. In VISSER, UBBO, FERNANDO RIBEIRO, TAKESHI OHASHI and FRANK DELLAERT (editors): *RoboCup 2007: Robot Soccer World Cup XI Preproceedings*, Atlanta, GA, USA, 2007. RoboCup Federation.

[42] RÖFER, THOMAS, RONNIE BRUNN, INGO DAHM, MATTHIAS HEBBEL, JAN HOFFMANN, MATTHIAS JÜNGEL, TIM LAUE, MARTIN LÖTZSCH, WALTER NISTICÓ and MICHAEL SPRANGER: *GermanTeam 2004*. In NARDI, DANIELE, MARTIN RIEDMILLER, CLAUDE SAMMUT and JOSÉ SANTOS-VICTOR (editors): *RoboCup 2004: Robot World Cup VIII Preproceedings*, Lisbon, Portugal, 2004. RoboCup Federation.

[43] RÖFER, THOMAS, CHRISTOPH BUDELMANN, MARTIN FRITSCHE, TIM LAUE, JUDITH MÜLLER, CORD NIEHAUS and FLORIAN PENQUITT: *B-Human Team Description for RoboCup 2007*. In VISSER, UBBO, FERNANDO RIBEIRO, TAKESHI OHASHI and FRANK DELLAERT (editors): *RoboCup 2007: Robot Soccer World Cup XI Preproceedings*, Atlanta, GA, USA, 2007. RoboCup Federation.

[44] RÖFER, THOMAS, MARTIN FRITSCHE, MATTHIAS HEBBEL, THOMAS KINDLER, TIM LAUE, CORD NIEHAUS, WALTER NISTICÓ and PHILIPPE SCHOBER: *BreDoBrothers Team Description for RoboCup 2006*. In LAKEMEYER, GERHARD, ELIZABETH SKLAR, DOMENICO G. SORRENTI and TOMOICHI TAKAHASHI (editors): *RoboCup 2006:*

Robot Soccer World Cup X Preproceedings, Bremen, Germany, 2006. RoboCup Federation.

[45] RÖFER, THOMAS, TIM LAUE, OLIVER BÖSCHE, INGO SIEVERDINGBECK, THIEMO WIEDEMEYER and JAN-HENDRIK WORCH: *B-Human Team Description for RoboCup 2009*. In BALTES, JACKY, MICHAIL G. LAGOUDAKIS, TADASHI NARUSE and SAEED SHIRY (editors): *RoboCup 2009: Robot Soccer World Cup XIII Preproceedings*, Graz, Austria, 2009. RoboCup Federation.

[46] RÖFER, THOMAS, TIM LAUE, ARMIN BURCHARDT, ERIK DAMROSE, MARTIN FRITSCHE, JUDITH MÜLLER and ANDRIK RIESKAMP: *B-Human Team Description for RoboCup 2008*. In IOCCHI, LUCA, HITOSHI MATSUBARA, ALFREDO WEITZENFELD and CHANGJIU ZHOU (editors): *RoboCup 2008: Robot Soccer World Cup XII Preproceedings*, Suzhou, China, 2008. RoboCup Federation.

[47] RÖFER, THOMAS, TIM LAUE, ARMIN BURCHARDT, ERIK DAMROSE, KATHARINA GILLMANN, COLIN GRAF, THIJS JEFFRY DE HAAS, ALEXANDER HÄRTL, ANDRIK RIESKAMP, ANDRÉ SCHRECK and JAN-HENDRIK WORCH: *B-Human Team Report and Code Release 2008*, 2008. Only available online: `http://www.b-human.de/file_download/16/bhuman08_coderelease.pdf`.

[48] RÖFER, THOMAS, TIM LAUE, HANS-DIETER BURKHARD, JAN HOFFMANN, MATTHIAS JÜNGEL, DANIEL GÖHRING, MARTIN LÖTZSCH, UWE DÜFFERT, MICHAEL SPRANGER, BENJAMIN ALTMEYER, VIVIANA GOETZKE, OSKAR VON STRYK, RONNIE BRUNN, MARC DASSLER, MICHAEL KUNZ, MAX RISLER, MAXIMILIAN STELZER, DIRK THOMAS, STEFAN UHRIG, UWE SCHWIEGELSHOHN, INGO DAHM, MATTHIAS HEBBEL, WALTER NISTICÓ, CARSTEN SCHUMANN and MICHAEL WACHTER: *GermanTeam RoboCup 2004*, 2004. Only available online: `http://www.germanteam.org/GT2004.pdf`.

[49] RÖFER, THOMAS, TIM LAUE, COLIN GRAF, TOBIAS KASTNER, ALEXANDER FABISCH and CHRISTIAN THEDIECK: *B-Human Team Description for RoboCup 2010*. In CHOWN, ERIC, AKIHIRO MATSUMOTO, PAUL PLOEGER and JAVIER RUIZ DEL SOLAR (editors): *RoboCup 2010: Robot Soccer World Cup XIV Preproceedings*, Singapore, 2010. RoboCup Federation.

[50] RÖFER, THOMAS, TIM LAUE, JUDITH MÜLLER, ARMIN BURCHARDT, ERIK DAMROSE, ALEXANDER FABISCH, FYNN FELDPAUSCH, KATHARINA GILLMANN, COLIN GRAF, THIJS JEFFRY DE HAAS, ALEXANDER HÄRTL, DANIEL HONSEL, PHILIPP KASTNER, TOBIAS KASTNER, BENJAMIN MARKOWSKY, MICHAEL MESTER, JONAS PETER, OLE JAN LARS RIEMANN, MARTIN RING, WIEBKE SAUERLAND, ANDRÉ SCHRECK, INGO SIEVERDINGBECK, FELIX WENK and JAN-HENDRIK WORCH: *B-Human Team Report and Code Release 2010*, 2010. Only available online: `http://www.b-human.de/file_download/33/bhuman10_coderelease.pdf`.

[51] RÖFER, THOMAS, TIM LAUE, JUDITH MÜLLER, OLIVER BÖSCHE, ARMIN BURCHARDT, ERIK DAMROSE, KATHARINA GILLMANN, COLIN GRAF, THIJS JEFFRY DE HAAS, ALEXANDER HÄRTL, ANDRIK RIESKAMP, ANDRÉ SCHRECK, INGO

Sieverdingbeck and Jan-Hendrik Worch: *B-Human Team Report and Code Release 2009*, 2009. Only available online: http://www.b-human.de/file_download/26/bhuman09_coderelease.pdf.

[52] Röfer, Thomas, Tim Laue, Michael Weber, Hans-Dieter Burkhard, Matthias Jüngel, Daniel Göhring, Jan Hoffmann, Benjamin Altmeyer, Thomas Krause, Michael Spranger, Oskar von Stryk, Ronnie Brunn, Marc Dassler, Michael Kunz, Tobias Oberlies, Max Risler, Uwe Schwiegelshohn, Matthias Hebbel, Walter Nisticó, Stefan Czarnetzki, Thorsten Kerkhof, Matthias Meyer, Carsten Rohde, Bastian Schmitz, Michael Wachter, Tobias Wegner and Christine Zarges: *GermanTeam RoboCup 2005*, 2005. Only available online: http://www.germanteam.org/GT2005.pdf.

References

[ALBU-SCHÄFFER et al., 2008] ALBU-SCHÄFFER, ALIN, O. EIBERGER, M. GREBENSTEIN, S. HADDADIN, C. OTT, T. WIMBÖCK, S. WOLF and G. HIRZINGER (2008). *Soft Robotics: From Torque Feedback-Controlled Lightweight Robots to Intrinsically Compliant Systems*. Robotics & Automation Magazine: Special Issue on Adaptable Compliance / Variable Stiffness for Robotic Applications, 15(3):20–30.

[ALLAN, 1979] ALLAN, ROGER (1979). *The amazing micromice: see how they won*. IEEE Spectrum, 16(9):62–65.

[ANGELINE, 1998] ANGELINE, PETER J. (1998). *Evolutionary Optimization versus Particle Swarm Optimization: Philosophy and Performance Differences*. In PORTO, V. W., N. SARAVANAN, D. WAAGEN and A. E. EIBEN, eds.: *Evolutionary Programming VII: Proceedings of the 7th International Conference*, vol. 1447 of *Lecture Notes in Computer Science*, pp. 601–610. Springer.

[ARGALL et al., 2006] ARGALL, BRENNA, B. BROWNING, Y. GU and M. VELOSO (2006). *The First Segway Soccer Experience: Towards Peer-to-Peer Human-Robot Teams*. In *Proceedings of the 2006 Conference on Human-Robot Interaction*, Salt Lake City, UT, USA.

[ARKIN, 1998] ARKIN, RONALD C. (1998). *Behavior-Based Robotics*. The MIT Press.

[ASADA et al., 2003] ASADA, MINORU, O. OBST, D. POLANI, B. BROWNING, A. BONARINI, M. FUJITA, T. CHRISTALLER, T. TAKAHASHI, S. TADOKORO, E. SKLAR and G. A. KAMINKA (2003). *An Overview of RoboCup-2002 Fukuoka/Busan*. AI Magazine, 24(2):21–40.

[BAALL, 2010] BAALL (2010). *BAALL – Bremen Ambient Assisted Living Lab – Official Website*. `http://www.baall.net`.

[BACH and JÜNGEL, 2002] BACH, JOSCHA and M. JÜNGEL (2002). *Using pattern matching on a flexible horizon-aligned grid for robotic vision*. In *Concurrency Specification and Programming - CSP'2002*, vol. 1, pp. 11–19.

[BALCH and YANCO, 2002] BALCH, TUCKER and H. A. YANCO (2002). *Ten Years of the AAAI Mobile Robot Competition and Exhibition - Looking Back and to the Future*. AI Magazine, 23(1):13–22.

[BALTES et al., 2010] BALTES, JACKY, M. G. LAGOUDAKIS, T. NARUSE and S. SHIRY, eds. (2010). *RoboCup 2009: Robot Soccer World Cup XIII*, vol. 5949 of *Lecture Notes in Artificial Intelligence*. Springer.

[BEETZ et al., 2006] BEETZ, MICHAEL, J. BANDOUCH, S. GEDIKLI, N. VON HOYNINGEN-HUENE, B. KIRCHLECHNER and A. MALDONADO (2006). *Camera-based Observation of Football Games for Analyzing Multi-agent Activities*. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, Hakodate, Japan.

[BEETZ et al., 2007] BEETZ, MICHAEL, S. GEDIKLI, J. BANDOUCH, B. KIRCHLECHNER, N. v. HOYNINGEN-HUENE and A. PERZYLO (2007). *Visually Tracking Football Games Based on TV Broadcasts*. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI)*, Hyderabad, India.

[BEHNKE, 2006] BEHNKE, SVEN (2006). *Robot Competitions – Ideal Benchmarks for Robotics Research*. In DEL PÔBIL, ANGEL P., ed.: *Proceedings of the IROS 2006 Workshop on Benchmarks in Robotics Research*, pp. 13–17, Beijing, China.

[BEYER, 1996] BEYER, HANS-GEORG (1996). *Toward a Theory of Evolution Strategies: Self-Adaptation*. Evolutionary Computation, 3(3):311–347.

[BEYER and SCHWEFEL, 2002] BEYER, HANS-GEORG and H.-P. SCHWEFEL (2002). *Evolution strategies – A comprehensive introduction*. Natural Computing, 1(1):3–52.

[BIRBACH and FRESE, 2009] BIRBACH, OLIVER and U. FRESE (2009). *A Multiple Hypothesis Approach for a Ball Tracking System*. In FRITZ, MARIO, B. SCHIELE and J. H. PIATER, eds.: *Proceedings of the 7th International Conference on Computer Vision Systems, ICVS 2009*, vol. 5815, pp. 435–444.

[BIRBACH et al., 2011] BIRBACH, OLIVER, U. FRESE and B. BÄUML (2011). *Realtime Perception for Catching a Flying Ball with a Mobile Humanoid*. In *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA 2011)*, Shanghai, China. Submitted.

[BONARINI et al., 2006] BONARINI, ANDREA, W. BURGARD, G. FONTANA, M. MATTEUCCI, D. G. SORRENTI and J. D. TARDOS (2006). *RAWSEEDS: Robotics Advancement through Web-publishing of Sensorial and Elaborated Extensive Data Sets*. In DEL PÔBIL, ANGEL P., ed.: *Proceedings of the IROS 2006 Workshop on Benchmarks in Robotics Research*, pp. 19–23, Beijing, China.

[BRINDZA et al., 2009] BRINDZA, JORDAN, A. LEE, A. MAJUMDAR, B. SCHARFMAN, A. SCHNEIDER, R. SHOR and D. LEE (2009). *UPennalizers RoboCup Standard Platform League Team Report 2009*. Only available online: `http://www.seas.upenn.edu/~robocup/files/UPennalizers_team_report_spl_2009.pdf`.

[BRUCE et al., 2000] BRUCE, JAMES, T. BALCH and M. VELOSO (2000). *Fast and inexpensive color image segmentation for interactive robots*. In *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '00)*, vol. 3, pp. 2061–2066, Takamatsu, Japan.

[BRUCE and VELOSO, 2003] BRUCE, JAMES and M. VELOSO (2003). *Fast and accurate vision-based pattern detection and identification*. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA '03)*, vol. 1, pp. 1277–1282, Taipei, Taiwan.

[BRÄUNL, 1999] BRÄUNL, THOMAS (1999). *Research Relevance of Mobile Robot Competitions*. IEEE Robotics and Automation Magazine, 66(4):32–37.

[BRUYNINCKX, 2004] BRUYNINCKX, HERMAN (2004). *Blender for robotics and robotics for Blender*. In *Proceedings of the Blender Conference 2004*, Amsterdam, The Netherlands.

[BUEHLER et al., 2007] BUEHLER, MARTIN, K. IAGNEMMA and S. SINGH (2007). *The 2005 DARPA Grand Challenge: The Great Robot Race*, vol. 36 of *Springer Tracts in Advanced Robotics*. Springer.

[BUEHLER et al., 2010] BUEHLER, MARTIN, K. IAGNEMMA and S. SINGH (2010). *The DARPA Urban Challenge: Autonomous Vehicles in City Traffic*, vol. 56 of *Springer Tracts in Advanced Robotics*. Springer.

[BURKHARD et al., 2002] BURKHARD, HANS-DIETER, D. DUHAUT, M. FUJITA, P. LIMA, R. MURPHY and R. ROJAS (2002). *The Road to RoboCup 2050*. IEEE Robotics and Automation Magazine, 9(2):31–38.

[CAMPBELL et al., 2002] CAMPBELL, MURRAY, A. J. HOANE and F.-H. HSU (2002). *Deep Blue*. Artificial Intelligence, 134(1-2):57 – 83.

[CARPIN et al., 2007] CARPIN, STEFANO, M. LEWIS, J. WANG, S. BALAKIRSKY and C. SCRAPPER (2007). *USARSim: a robot simulator for research and education*. In *Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA 2007)*, pp. 1400–1405, Rome, Italy.

[CORADESCHI et al., 2000] CORADESCHI, SILVIA, L. KARLSSON, P. STONE, T. BALCH, G. KRAETZSCHMAR and M. ASADA (2000). *Overview of RoboCup-99*. AI Magazine, 21(3):11–18.

[CORTÉS et al., 2008] CORTÉS, ULISES, A. MARTÍNEZ-VELASCO, C. BARRUÉ, T. BENEDICO, F. CAMPANA, C. CALTAGIRONE and R. ANNICCHIARICO (2008). *A SHARE-it service to elders' mobility using the i-walker*. Gerontechnology, 7(2):95.

[CZARNETZKI and KERNER, 2009] CZARNETZKI, STEFAN and S. KERNER (2009). *Nao Devils Dortmund - Team Description for RoboCup 2009*. In BALTES, JACKY, M. G. LAGOUDAKIS, T. NARUSE and S. SHIRY, eds.: *RoboCup 2009: Robot Soccer World Cup XII Preproceedings*. RoboCup Federation.

[CZARNETZKI et al., 2011] CZARNETZKI, STEFAN, S. KERNER and M. KRUSE (2011). *Real-time Active Vision by Entropy Minimization Applied to Localization*. In SOLAR, JAVIER RUIZ DEL, E. CHOWN and P. G. PLOEGER, eds.: *RoboCup 2010: Robot Soccer World Cup XIV*, vol. 6556 of *Lecture Notes in Artificial Intelligence*, pp. 266–277. Springer.

[DAVIES, 2004] DAVIES, E. R. (2004). *Machine Vision. Theory, Algorithms, Practicalities*. Morgan Kaufmann.

[DEAN and BONASSO, 1993] DEAN, THOMAS and R. P. BONASSO (1993). *1992 AAAI Robot Exhibition and Competition*. AI Magazine, 14(1):35–48.

[DEL PÔBIL, 2006] DEL PÔBIL, ANGEL P., ed. (2006). *Proceedings of the IROS 2006 Workshop on Benchmarks in Robotics Research*, Beijing, China.

[DOUCET et al., 2000] DOUCET, ARNAUD, N. DE FREITAS, K. MURPHY and S. RUSSELL (2000). *Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks*. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence (UAI 2000)*, pp. 176–183, Stanford, CA, USA.

[DYLLA et al., 2008] DYLLA, FRANK, A. FERREIN, G. LAKEMEYER, J. MURRAY, O. OBST, T. RÖFER, S. SCHIFFER, F. STOLZENBURG, U. VISSER and T. WAGNER (2008). *Approaching a Formal Soccer Theory from Behaviour Specifications in Robotic Soccer*. In DABNICKI, PETER and A. BACA, eds.: *Computers in Sports*, pp. 161–185. WIT Press; London / Southampton, UK.

[FÉDÉRATION INTERNATIONALE DE FOOTBALL ASSOCIATION, 2006] FÉDÉRATION INTERNATIONALE DE FOOTBALL ASSOCIATION (2006). *Laws of the Game 2006*. http://www.fifa.com/documents/fifa/laws/LOTG2006_e.pdf.

[FIRA, 2010] FIRA (2010). *Federation of International Robot-soccer Association – Official Site*. http://www.fira.net.

[FOX et al., 1999] FOX, DIETER, W. BURGARD, F. DELLAERT and S. THRUN (1999). *Monte-Carlo Localization: Efficient Position Estimation for Mobile Robots*. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pp. 343 – 349, Orlando, FL, USA.

[FOX et al., 1998] FOX, DIETER, W. BURGARD and S. THRUN (1998). *Active Markov localization for mobile robots*. Robotics and Autonomous Systems, 25(3-4):195–207.

[FRAUNHOFER IML and 4ATTENTION GMBH & CO. KG, 2010] FRAUNHOFER IML and 4ATTENTION GMBH & CO. KG (2010). *RoboKeeper Web Site*. http://www.robokeeper.com.

[FRESE et al., 2010] FRESE, UDO, R. WAGNER and T. RÖFER (2010). *A SLAM Overview from a User's Perspective*. KI – Künstliche Intelligenz, 24(3):191–198.

[FRIEDMANN, 2009] FRIEDMANN, MARTIN (2009). *Simulation of Autonomous Robot Teams With Adaptable Levels of Abstraction*. PhD thesis, Technische Universität Darmstadt.

[FRIEDMANN et al., 2009] FRIEDMANN, MARTIN, K. PETERSEN, S. PETTERS, K. RADKHAH, D. SCHOLZ, D. THOMAS and O. VON STRYK (2009). *Darmstadt Dribblers - Team Description for Humanoid KidSize League of RoboCup 2009*. In BALTES, JACKY, M. G. LAGOUDAKIS, T. NARUSE and S. SHIRY, eds.: *RoboCup 2009: Robot Soccer World Cup XII Preproceedings*. RoboCup Federation.

[FRIEDMANN et al., 2008] FRIEDMANN, MARTIN, K. PETERSEN and O. VON STRYK (2008). *Tailored Real-Time Simulation for Teams of Humanoid Robots*. In VISSER, UBBU, F. RIBEIRO, T. OHASHI and F. DELLAERT, eds.: *RoboCup 2007: Robot Soccer World Cup XI*, vol. 5001 of *Lecture Notes in Artificial Intelligence*, pp. 425–432. Springer.

[FUJITA and KITANO, 1998] FUJITA, MASAHIRO and H. KITANO (1998). *Development of an Autonomous Quadruped Robot for Robot Entertainment*. Autonomous Robots, 5(1):7–18.

[GERKEY et al., 2003] GERKEY, BRIAN, R. T. VAUGHAN and A. HOWARD (2003). *The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems*. In *Proceedings of the 11th International Conference on Advanced Robotics*, pp. 317 – 323, Coimbra, Portugal.

[GERMANTEAM, 2008] GERMANTEAM (2008). *GermanTeam Web Site*. http://www.germanteam.org/.

[GHAZI-ZAHEDI et al., 2005] GHAZI-ZAHEDI, KEYAN, T. LAUE, T. RÖFER, P. SCHÖLL, K. SPIESS, A. TWICKEL and S. WISCHMANN (2005). *RoSiML - Robot Simulation Markup Language*. http://www.informatik.uni-bremen.de/spprobocup/RoSiML.html.

[GÖHRING et al., 2009] GÖHRING, DANIEL, H. MELLMANN and H.-D. BURKHARD (2009). *Constraint Based Belief Modeling*. In IOCCHI, LUCA, H. MATSUBARA, A. WEITZENFELD and C. ZHOU, eds.: *RoboCup 2008: Robot Soccer World Cup XII*, vol. 5399 of *Lecture Notes in Artificial Intelligence*, pp. 73–84. Springer.

[GLOVER et al., 2003] GLOVER, JARED, D. HOLSTIUS, M. MANOJLOVICH, K. MONTGOMERY, A. POWERS, J. WU, S. KIESLER, J. MATTHEWS, and S. THRUN (2003). *A Robotically-Augmented Walker for Older Adults*. Technical Report CMU-CS-03-170, Carnegie Mellon University, Computer Science Department, Pittsburgh, PA, USA.

[GOUAILLIER et al., 2008] GOUAILLIER, DAVID, V. HUGEL, P. BLAZEVIC, C. KILNER, J. MONCEAUX, P. LAFOURCADE, B. MARNIER, J. SERRE and B. MAISONNIER (2008). *The NAO humanoid: a combination of performance and affordability*. CoRR, abs/0807.3223.

[GRISETTI et al., 2007] GRISETTI, GIORGIO, C. STACHNISS and W. BURGARD (2007). *Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters*. IEEE Transactions on Robotics, 23:34–46.

[GUTMANN and FOX, 2002] GUTMANN, JENS-STEFFEN and D. FOX (2002). *An Experimental Comparison of Localization Methods Continued*. In *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2002)*, vol. 1, pp. 454–459, Lausanne, Switzerland.

[HART et al., 1968] HART, PETER, N. NILSSON and B. RAPHAEL (1968). *A Formal Basis for the Heuristic Determination of Minimum Cost Paths*. IEEE Transactions on Systems Science and Cybernetics, 4(2):100–107.

[HASHEMI et al., 2010] HASHEMI, EHSAN, O. A. GHIASVAND, M. G. JADIDI, A. KARIMI, R. HASHEMIFARD, M. LASHGARIAN, M. SHAFIEI, S. M. FARAHANI, K. ZAREI, F. FARAJI, M. A. Z. HARANDI and E. MOUSAVI (2010). *MRL Team Description 2010 Standard Platform League*. In CHOWN, ERIC, A. MATSUMOTO, P. PLOEGER and J. R. DEL SOLAR, eds.: *RoboCup 2010: Robot Soccer World Cup XIV Preproceedings*, Singapore. RoboCup Federation.

[HAYASHIBARA et al., 2009] HAYASHIBARA, YASUO, H. MINAKATA, Y. SEIKE, S. OGURA, K. ICHIZAWA, K. MACHI, K. TAKAMATSU, S. YOSHIZAWA, Y. YAMADA, T. HORIUCHI, M. FUKUTA, S. FUJITA, K. IRIE, H. KAMINAGA, T. KAWAKAMI, M. S. Y. NISHIZAKI and H. SAKAMOTO (2009). *CIT BRains (Kid Size League)*. In BALTES, JACKY, M. G. LAGOUDAKIS, T. NARUSE and S. SHIRY, eds.: *RoboCup 2009: Robot Soccer World Cup XII Preproceedings*. RoboCup Federation.

[HEBBEL et al., 2006] HEBBEL, MATTHIAS, W. NISTICO and D. FISSELER (2006). *Learning in a high dimensional space: Fast omnidirectional quadrupedal locomotion*. In LAKEMEYER, GERHARD, E. SKLAR, D. G. SORRENTI and T. TAKAHASHI, eds.: *RoboCup 2006: Robot Soccer World Cup X*, vol. 4434 of *Lecture Notes in Artificial Intelligence*, pp. 314–321. Springer.

[HEMKER et al., 2009] HEMKER, THOMAS, H. SAKAMOTO, M. STELZER and O. VON STRYK (2009). *Efficient walking speed optimization of a humanoid robot*. International Journal of Robotics Research, 28(2):303 – 314.

[HERMANS et al., 2009] HERMANS, TUCKER, J. STROM, G. SLAVOV, J. MORRISON, A. LAWRENCE, E. KROB and E. CHOWN (2009). *Northern Bites 2009 Team Report*. Only available online: `http://robocup.bowdoin.edu/public/NorthernBitesReport2009.pdf`.

[HIROSE and OGAWA, 2007] HIROSE, MASATO and K. OGAWA (2007). *Honda humanoid robots development*. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 365(1850):11–19.

[HOFFMANN et al., 2005a] HOFFMANN, JAN, M. JÜNGEL and M. LÖTZSCH (2005a). *A Vision Based System for Goal-Directed Obstacle Avoidance*. In NARDI, DANIELE, M. RIEDMILLER, C. SAMMUT and J. SANTOS-VICTOR, eds.: *RoboCup 2004: Robot Soccer World Cup VIII*, vol. 3276 of *Lecture Notes in Artificial Intelligence*, pp. 418–425. Springer.

[HOFFMANN et al., 2005b] HOFFMANN, JAN, M. SPRANGER, D. GÖHRING and M. JÜNGEL (2005b). *Making Use Of What You Don't See: Negative Information In Markov Localization*. In *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2947 – 2952, Edmonton, Alberta, Canada.

[HOWARD and ROY, 2010] HOWARD, ANDREW and N. ROY (2010). *The Robotics Data Set Repository (Radish)*. `http://radish.sourceforge.net/`.

[ISHIDA et al., 2001] ISHIDA, TATSUZO, Y. KUROKI, J. YAMAGUCHI, M. FUJITA and T. T. DOI (2001). *Motion Entertainment by a Small Humanoid Robot Based on OPEN-R*. In *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001)*, vol. 2, pp. 1079–1086, Maui, HI, USA.

[JACKSON, 2007] JACKSON, JARED (2007). *Microsoft Robotics Studio: A Technical Introduction*. IEEE Robotics and Automation Magazine, 14(4):82–87.

[JACOFF et al., 2002] JACOFF, ADAM, E. MESSINA and J. EVANS (2002). *Performance Evaluation of Autonomous Mobile Robots*. Industrial Robot: An International Journal, 29(3):259–267.

[JAMZAD et al., 2002] JAMZAD, M., B. SADJAD, V. MIRROKNI, M. KAZEMI, H. CHITSAZ, A. HEYDARNOORI, M. HAJIAGHAI and E. CHINIFOROOSHAN (2002). *A Fast Vision System for Middle Size Robots in RoboCup*. In BIRK, ANDREAS, S. CORADESCHI and S. TADOKORO, eds.: *RoboCup 2001: Robot Soccer World Cup V*, vol. 2377 of *Lecture Notes in Artificial Intelligence*, pp. 71–80. Springer.

[JIAN, 2008] JIAN, CUI (2008). *Interpretation der Routeninstruktion mit qualitativen und quantitativen Informationen für mobile Roboter*. Master's thesis, Universität Bremen.

[JÜNGEL, 2007] JÜNGEL, MATTHIAS (2007). *Self-Localization Based on a Short-Term Memory of Bearings and Odometry*. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2007)*, pp. 2494 – 2499, San Diego, CA, USA.

[JOHANSSON and SAFFIOTTI, 2002] JOHANSSON, STEFAN. J. and A. SAFFIOTTI (2002). *Using the Electric Field Approach in the RoboCup Domain*. In BIRK, ANDREAS, S. CORADESCHI and S. TADOKORO, eds.: *RoboCup 2001: Robot Soccer World Cup V*, vol. 2377 of *Lecture Notes in Artificial Intelligence*, pp. 117–138. Springer.

[JONES, 2006] JONES, JOSEPH L. (2006). *Robots at the Tipping Point: The Road to iRobot Roomba*. IEEE Robotics & Automation Magazine, 13(1):76–78.

[JULIER and UHLMANN, 1997] JULIER, SIMON J. and J. K. UHLMANN (1997). *A New Extension of the Kalman Filter to Nonlinear Systems*. In *Proceedings of AeroSense: The 11th International Symposium on Aerospace/Defense Sensing, Simulation and Controls*, pp. 182–193, Orlando, FL, USA.

[KÁLMÁN, 1960] KÁLMÁN, RUDOLF EMIL (1960). *A New Approach to Linear Filtering and Prediction Problems*. Transactions of the ASME – Journal of Basic Engineering, 82 (Series D):35–45.

[KEIDAR et al., 2010] KEIDAR, MATAN, I. AHARON, D. BARDA, O. KAMARA, A. LEVY, E. POLOSETSKI, D. RAMATI, L. SHLOMOV, J. SHOSHAN, A. YAKIR, A. ZILKA, G. A. KAMINKA and E. KOLBERG (2010). *RoboCup 2010 Standard Platform League Team BURST Description*. In CHOWN, ERIC, A. MATSUMOTO, P. PLOEGER and J. R. DEL SOLAR, eds.: *RoboCup 2010: Robot Soccer World Cup XIV Preproceedings*, Singapore. RoboCup Federation.

[KENNEDY and EBERHART, 1995] KENNEDY, JAMES and R. EBERHART (1995). *Particle swarm optimization*. In *Proceedings of the 1995 IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Piscataway, NJ, USA.

[KHATIB, 1986] KHATIB, OUSSAMA (1986). *Real-time Obstacle Avoidance for Manipulators and Mobile Robots*. The International Journal of Robotics Research, 5(1):90–98.

[KITANO, 1998] KITANO, HIROAKI, ed. (1998). *RoboCup-97: Robot Soccer World Cup I*, vol. 1395 of *Lecture Notes in Artificial Intelligence*. Springer.

[KITANO and ASADA, 1998] KITANO, HIROAKI and M. ASADA (1998). *RoboCup Humanoid Challenge: That's One Small Step for A Robot, One Giant Leap for Mankind*. In *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 1998)*, pp. 419–424, Victoria, B.C., Canada.

[KITANO et al., 1995] KITANO, HIROAKI, M. ASADA, Y. KUNIYOSHI, I. NODA and E. OSAWA (1995). *RoboCup: The Robot World Cup Initiative*. In *Working Notes of the IJCAI-95 Workshop on Entertainment and AI/Alife*, pp. 19–24, Montreal, Quebec, Canada.

[KITANO et al., 1997] KITANO, HIROAKI, M. ASADA, Y. KUNIYOSHI, I. NODA, E. OSAWA and H. MATSUBARA (1997). *RoboCup: A Challenge Problem for AI*. AI Magazine, 18(1):73–85.

[KITANO et al., 1999] KITANO, HIROAKI, S. TADOKORO, I. NODA, H. MATSUBARA, T. TAKAHASHI, A. SHINJOU and S. SHIMADA (1999). *RoboCup Rescue: Search and Rescue in Large-Scale Disasters as a Domain for Autonomous Agents Research*. In *Proceedings of the 1999 IEEE International Conference on Systems, Man, and Cybernetics (SMC-99)*, vol. 6, pp. 739–743.

[KOENIG and HOWARD, 2004] KOENIG, NATHAN and A. HOWARD (2004). *Design and Use Paradigms of Gazebo, an Open-Source Multi-Robot Simulator*. In *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)*, vol. 3, pp. 2149 – 2154, Sendai, Japan.

[KÖGLER and OBST, 2004] KÖGLER, MARCO and O. OBST (2004). *Simulation League: The Next Generation*. In POLANI, DANIEL, A. BONARINI, B. BROWNING and K. YOSHIDA, eds.: *RoboCup 2003: Robot Soccer World Cup VII*, vol. 3020 of *Lecture Notes in Artificial Intelligence*, pp. 458 – 469. Springer.

[KRIEG-BRÜCKNER et al., 2010] KRIEG-BRÜCKNER, BERND, T. RÖFER, H. SHI and B. GERSDORF (2010). *Mobility Assistance in the Bremen Ambient Assisted Living Lab*. GeroPsych: The Journal of Gerontopsychology and Geriatric Psychiatry, 23(23):121–130.

[KRIEG-BRÜCKNER et al., 2005] KRIEG-BRÜCKNER, BERND, U. FRESE, K. LÜTTICH, C. MANDEL, T. MOSSAKOWSKI and R. ROSS (2005). *Specification of an Ontology for Route Graphs*. In FREKSA, CHRISTIAN, M. KNAUFF, B. KRIEG-BRÜCKNER, B. NEBEL and T. BARKOWSKY, eds.: *Spatial Cognition IV. Reasoning, Action, and Interaction*, vol. 3343 of *Lecture Notes in Artificial Intelligence*, pp. 390–412. Springer.

[KRIEG-BRÜCKNER et al., 2009] KRIEG-BRÜCKNER, BERND, B. GERSDORF, M. DÖHLE and K. SCHILL (2009). *Technik für Senioren in spe im Bremen Ambient Assisted Living Lab*. In *2. Deutscher AAL-Kongress 2009*, Berlin-Offenbach, Germany. VDE-Verlag. In German.

[KUFFNER et al., 2002] KUFFNER, JAMES J., S. KAGAMI, K. NISHIWAKI, M. INABA and H. INOUE (2002). *Dynamically-stable Motion Planning for Humanoid Robots*. Autonomous Robots, 12(1):105–118.

[KWOK and FOX, 2005] KWOK, CODY and D. FOX (2005). *Map-based Multiple Model Tracking of a Moving Object*. In NARDI, DANIELE, M. RIEDMILLER, C. SAMMUT and J. SANTOS-VICTOR, eds.: *RoboCup 2004: Robot Soccer World Cup VIII*, vol. 3276 of *Lecture Notes in Artificial Intelligence*, pp. 18–33. Springer.

[LANKENAU and RÖFER, 2000] LANKENAU, AXEL and T. RÖFER (2000). *Smart Wheelchairs - State of the Art in an Emerging Market*. Künstliche Intelligenz. Schwerpunkt Autonome Mobile Systeme, (4):37–39.

[LAUER et al., 2006] LAUER, MARTIN, S. LANGE and M. RIEDMILLER (2006). *Calculating the PerfectMatch: An Efficient and Accurate Approach for Robot Self-localization*. In BREDENFELD, ANSGAR, A. JACOFF, I. NODA and Y. TAKAHASHI, eds.: *RoboCup 2005: Robot Soccer World Cup IX*, vol. 4020 of *Lecture Notes in Artificial Intelligence*, pp. 142–153. Springer.

[LEES and NOLAN, 1998] LEES, ADRIAN and L. NOLAN (1998). *The Biomechanics of Soccer: A Review*. Journal of Sport Sciences, 16(3):211–234.

[LENSER and VELOSO, 2000] LENSER, SCOTT and M. VELOSO (2000). *Sensor Resetting Localization for Poorly Modelled Mobile Robots*. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA 2000)*, vol. 2, pp. 1225–1232, San Francisco, CA, USA.

[LIMA et al., 2005] LIMA, PEDRO, L. CUSTÓDIO, L. AKIN, A. JACOFF, G. KRAEZSCHMAR, B. K. NG, O. OBST, T. RÖFER, Y. TAKAHASHI and C. ZHOU (2005). *RoboCup 2004 Competitions and Symposium*. AI Magazine, 26(2):36–61.

[LOETZSCH et al., 2006] LOETZSCH, MARTIN, M. RISLER and M. JÜNGEL (2006). *XABSL - A Pragmatic Approach to Behavior Engineering*. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2006)*, pp. 5124–5129, Beijing, China.

[LOGIC DESIGN INC., 2010] LOGIC DESIGN INC. (2010). *RoboLogix Web Site*. `http://www.robologix.com`.

[LUND and PAGLIARINI, 1999] LUND, HENRIK HAUTOP and L. PAGLIARINI (1999). *Robot Soccer with LEGO Mindstorms*. In ASADA, MINORU and H. KITANO, eds.: *RoboCup-98: Robot Soccer World Cup II*, vol. 1604 of *Lecture Notes in Artificial Intelligence*, pp. 141–151. Springer.

[MACKWORTH, 1993] MACKWORTH, ALAN K. (1993). *On Seeing Robots*. In BASU, A. and X. LI, eds.: *Computer Vision: Systems, Theory and Applications*, pp. 1–13. World Scientific Press, Singapore.

[MANDEL and FRESE, 2007] MANDEL, CHRISTIAN and U. FRESE (2007). *Comparison of Wheelchair User Interfaces for the Paralysed: Head-Joystick vs. Verbal Path Selection from an offered Route-Set*. In BURGARD, WOLFRAM and H.-M. GROSS, eds.: *Proceedings of the 3rd European Conference on Mobile Robots (ECMR 2007)*, pp. 217–222, Freiburg, Germany.

[MANDEL et al., 2007] MANDEL, CHRISTIAN, T. RÖFER and U. FRESE (2007). *Applying a 3DOF Orientation Tracker as a Human-Robot Interface for Autonomous Wheelchairs*. In *Proceedings of the IEEE 10th International Conference on Rehabilitation Robotics*, pp. 52 – 59, Noordwijk, The Netherlands.

[MAYER et al., 2007] MAYER, NORBERT MICHAEL, J. BOEDECKER, R. DA SILVA GUERRA, O. OBST and M. ASADA (2007). *3D2Real: Simulation League Finals in Real Robots*. In LAKEMEYER, GERHARD, E. SKLAR, D. G. SORRENTI and T. TAKAHASHI, eds.: *RoboCup 2006: Robot Soccer World Cup X*, vol. 4434 of *Lecture Notes in Artificial Intelligence*, pp. 25–34. Springer.

[MEYER and DUNEKAMP, 1999] MEYER, NORBERT and D. DUNEKAMP (1999). *SimRobot 2 - Ein multiagentenfähiger 3D-Robotiksimulator in Java*. Master's thesis, Universität Bremen.

[MICHEL, 2004] MICHEL, OLIVIER (2004). *Cyberbotics Ltd. - WebotsTM: Professional Mobile Robot Simulation*. International Journal of Advanced Robotic Systems, 1(1):39–42.

[MOBALLEGH et al., 2007] MOBALLEGH, HAMID REZA, R. GUILBOURD, G. HOHL, A. REUSCHENBACH, J. YU and R. ROJAS (2007). *FUmanoid Team Description 2007*. In VISSER, UBBO, F. RIBEIRO, T. OHASHI and F. DELLAERT, eds.: *RoboCup 2007: Robot Soccer World Cup XI Preproceedings*, Atlanta, GA, USA. RoboCup Federation.

[MOBALLEGH et al., 2009] MOBALLEGH, HAMID REZA, G. HOHL, T. LANDGRAF, B. FISCHER, T. LANGNER, T. FASSBENDER, S. OTTE, K. STOLL, A. TUCHSCHERER, D. STEIG, S. HEINRICH, S. MIELKE, D. SEIFERT, M. KUKULSKI, B. KAHLERT and R. ROJAS (2009). *FUmanoid Team Description 2009*. In BALTES, JACKY, M. G. LAGOUDAKIS, T. NARUSE and S. SHIRY, eds.: *RoboCup 2009: Robot Soccer World Cup XII Preproceedings*. RoboCup Federation.

[MOBILEROBOTS INC, 2010] MOBILEROBOTS INC (2010). *Intelligent Robots for Commercial & Research Applications*. `http://www.mobilerobots.com`.

[MORAVEC and ELFES, 1985] MORAVEC, HANS P. and A. ELFES (1985). *High Resolution from Wide Angle Sonar*. In *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, vol. 2, pp. 116–121, St. Louis, MO, USA.

[NICKLIN et al., 2007] NICKLIN, STEVEN P., R. D. FISHER and R. H. MIDDLETON (2007). *Rolling Shutter Image Compensation*. In LAKEMEYER, GERHARD, E. SKLAR, D. G. SORRENTI and T. TAKAHASHI, eds.: *RoboCup 2006: Robot Soccer World Cup X*, vol. 4434 of *Lecture Notes in Artificial Intelligence*, pp. 402–409. Springer.

[NIEMÜLLER et al., 2011] NIEMÜLLER, TIM, A. FERREIN, G. ECKEL, D. PIRRO, P. PODBREGAR, T. KELLNER, C. RATH and G. STEINBAUER (2011). *Providing Ground-truth Data for the Nao Robot Platform*. In SOLAR, JAVIER RUIZ DEL, E. CHOWN and P. G. PLOEGER, eds.: *RoboCup 2010: Robot Soccer World Cup XIV*, vol. 6556 of *Lecture Notes in Artificial Intelligence*, pp. 133–144. Springer.

[NISTICÒ and HEBBEL, 2009] NISTICÒ, WALTER and M. HEBBEL (2009). *Particle Filter with Temporal Smoothing for Mobile Robot Vision-Based Localization*. In ANDRADE-CETTO, JUAN, J.-L. FERRIER and J. FILIPE, eds.: *Informatics in Control, Automation and Robotics*, vol. 37 of *Lecture Notes in Electrical Engineering*, pp. 167–180. Springer.

[NODA et al., 1998] NODA, ITSUKI, S. SUZUKI, H. MATSUBARA, M. ASADA and H. KITANO (1998). *RoboCup-97: The First Robot World Cup Soccer Games and Conferences*. AI Magazine, 19(3):49–59.

[OPENSTREETMAP, 2009] OPENSTREETMAP (2009). *OpenStreetMap Web Site*. `http://www.openstreetmap.org`.

[QUINLAN and MIDDLETON, 2010] QUINLAN, MICHAEL J. and R. H. MIDDLETON (2010). *Multiple Model Kalman Filters: A Localization Technique for RoboCup Soccer*. In BALTES, JACKY, M. G. LAGOUDAKIS, T. NARUSE and S. SHIRY, eds.: *RoboCup 2009: Robot Soccer World Cup XIII*, vol. 5949 of *Lecture Notes in Artificial Intelligence*, pp. 276–287. Springer.

[RABUS et al., 2003] RABUS, BERNHARD, M. EINEDER, A. ROTH and R. BAMLER (2003). *The Shuttle Radar Topography Mission (SRTM) – A New Class of Digital Elevation Models Acquired by Spaceborne Radar*. ISPRS Journal of Photogrammetry & Remote Sensing, 57(4):241–262.

[RAIBERT et al., 2008] RAIBERT, MARC, K. BLANKESPOOR, G. NELSON, R. PLAYTER and THE BIG-DOG TEAM (2008). *BigDog, the Rough-Terrain Quadruped Robot*. In *Proceedings of the 17th World Congress of the International Federation of Automatic Control*, pp. 10822–10825, Seoul, Korea.

[RAINWATER et al., 2010] RAINWATER, R. STEVEN, R. ARRICK and J. BROWN (2010). *robots.net – Robot Competitions*. `http://robots.net/rcfaq.html`.

[RAMANAN and FORSYTH, 2003] RAMANAN, DEVA and D. FORSYTH (2003). *Finding and Tracking People from the Bottom Up*. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Madison, WI, USA.

[ROBOCUP FEDERATION, 2010] ROBOCUP FEDERATION (2010). *RoboCup Official Site*. `http://www.robocup.org`.

[ROBOCUP SMALL SIZE LEAGUE TECHNICAL COMMITTEE, 2010] ROBOCUP SMALL SIZE LEAGUE TECHNICAL COMMITTEE (2010). *Laws of the RoboCup Small Size League 2010*. `http://small-size.informatik.uni-bremen.de/_media/rules:ssl-rules-2010.pdf`.

[ROBOCUP SPL TECHNICAL COMMITTEE, 2009] ROBOCUP SPL TECHNICAL COMMITTEE (2009). *Technical Challenges for the RoboCup 2009 Standard Platform League Competition*. `http://www.tzi.de/spl/pub/Website/Downloads/Challenges2009.pdf`.

[ROBOCUP TECHNICAL COMMITTEE, 2010] ROBOCUP TECHNICAL COMMITTEE (2010). *RoboCup Standard Platform League (Nao) Rule Book*. `http://www.tzi.de/spl/pub/Website/Downloads/Rules2010.pdf`.

[ROBOTIS, 2010] ROBOTIS (2010). *ROBOTIS Web Site*. `http://www.robotis.com/zbxe/main`.

[RÖFER, 1998] RÖFER, THOMAS (1998). *Strategies for Using a Simulation in the Development of the Bremen Autonomous Wheelchair*. In ZOBEL, R. and D. MOELLER, eds.: *Simulation-Past, Present and Future*, pp. 460–464. Society for Computer Simulation International.

[RÖFER, 2003] RÖFER, THOMAS (2003). *An Architecture for a National RoboCup Team*. In KAMINKA, GAL A., P. LIMA and R. ROJAS, eds.: *RoboCup 2002: Robot Soccer World Cup VI*, vol. 2752 of *Lecture Notes in Artificial Intelligence*, pp. 417–425. Springer.

[RÖFER, 2005] RÖFER, THOMAS (2005). *Evolutionary Gait-Optimization Using a Fitness Function Based on Proprioception*. In NARDI, DANIELE, M. RIEDMILLER, C. SAMMUT and J. SANTOS-VICTOR, eds.: *RoboCup 2004: Robot Soccer World Cup VIII*, vol. 3276 of *Lecture Notes in Artificial Intelligence*, pp. 310–322. Springer.

[RÖFER, 2008] RÖFER, THOMAS (2008). *Region-Based Segmentation with Ambiguous Color Classes and 2-D Motion Compensation*. In VISSER, UBBO, F. RIBEIRO, T. OHASHI and F. DELLAERT, eds.: *RoboCup 2007: Robot Soccer World Cup XI*, vol. 5001 of *Lecture Notes in Artificial Intelligence*, pp. 369–376. Springer.

[RÖFER and JÜNGEL, 2003] RÖFER, THOMAS and M. JÜNGEL (2003). *Vision-Based Fast and Reactive Monte-Carlo Localization*. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA-2003)*, pp. 856–861, Taipei, Taiwan.

[RÖFER and JÜNGEL, 2004] RÖFER, THOMAS and M. JÜNGEL (2004). *Fast and Robust Edge-Based Localization in the Sony Four-Legged Robot League*. In POLANI, DANIEL, A. BONARINI, B. BROWNING and K. YOSHIDA, eds.: *RoboCup 2003: Robot World Cup VII*, vol. 3020 of *Lecture Notes in Artificial Intelligence*, pp. 262–273. Springer.

[RÖFER et al., 2009] RÖFER, THOMAS, C. MANDEL, A. LANKENAU, B. GERSDORF and U. FRESE (2009). *15 Years of Rolland*. In HOFFMANN, BERTHOLD, C. LÜTH, T. MOSSAKOWSKI and L. SCHRÖDER, eds.: *Festschrift Dedicated to Bernd Krieg-Brückner on the Occasion of his 60th Birthday*, pp. 255–272. Sichere Kognitive Systeme, DFKI Bremen.

[ROJAS et al., 2007] ROJAS, RAÚL, M. SIMON and O. TENCHIO (2007). *Parabolic Flight Reconstruction from Multiple Images from a Single Camera in General Position*. In LAKEMEYER, GERHARD, E. SKLAR, D. G. SORRENTI and T. TAKAHASHI, eds.: *RoboCup 2006: Robot Soccer World Cup X*, vol. 4434 of *Lecture Notes in Artificial Intelligence*, pp. 183–193. Springer.

[SABE et al., 2004] SABE, KOHTARO, M. FUKUCHI, J.-S. GUTMANN, T. OHASHI, K. KAWAMOTO and T. YOSHIGAHARA (2004). *Obstacle Avoidance and Path Planning for Humanoid Robots Using Stereo Vision*. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA 2004)*, pp. 592–597, New Orleans, LA, USA.

[SIEMS et al., 1994] SIEMS, UWE, C. HERWIG and T. RÖFER (1994). *SimRobot, ein System zur Simulation sensorbestückter Agenten in einer dreidimensionalen Umwelt*. No. 1/94 in *ZKW Bericht*. Zentrum für Kognitionswissenschaften. Universität Bremen.

[SIMPSON, 2005] SIMPSON, RICHARD C. (2005). *Smart wheelchairs: A literature review*. Journal of Rehabilitation Research & Development, 42(4):423–436.

[SKLAR et al., 2003] SKLAR, ELIZABETH, A. EGUCHI and J. JOHNSON (2003). *RoboCupJunior: Learning with Educational Robotics*. In KAMINKA, GAL A., P. LIMA and R. ROJAS, eds.: *RoboCup 2002: Robot Soccer World Cup VI*, vol. 2752 of *Lecture Notes in Artificial Intelligence*, pp. 238–253. Springer.

[SMITH, 2007] SMITH, RUSSELL (2007). *Open Dynamics Engine*. `http://www.ode.org`.

[STACHNISS et al., 2010] STACHNISS, CYRILL, U. FRESE and G. GRISETTI (2010). *OpenSLAM web page*. `http://OpenSLAM.org/`.

[STAHL, 2009] STAHL, CHRISTOPH (2009). *Spatial Modeling of Activity and User Assistance in Instrumented Environments*. PhD thesis, Department of Computer Science, Saarland University.

[STAHL and HAUPERT, 2006] STAHL, CHRISTOPH and J. HAUPERT (2006). *Taking Location Modelling to new Levels: A Map Modelling Toolkit for Intelligent Environments*. In HAZAS, MIKE, T. STRANG and J. KRUMM, eds.: *Proceedings of the International Workshop on Location- and Context-Awareness (LoCA)*, vol. 3987 of *Lecture Notes in Computer Science*, pp. 74–85. Springer.

[STONE et al., 2010] STONE, PETER, M. QUINLAN and T. HESTER (2010). *Can Robots Play Soccer?*. In RICHARDS, TED, ed.: *Soccer and Philosophy: Beautiful Thoughts on the Beautiful Game*, vol. 51 of *Popular Culture and Philosophy*, pp. 75–88. Open Court Publishing Company.

[STONE et al., 2005] STONE, PETER, R. S. SUTTON and G. KUHLMANN (2005). *Reinforcement Learning for RoboCup-Soccer Keepaway*. Adaptive Behavior, 13(3):165–188.

[STRASDAT et al., 2007] STRASDAT, HAUKE, M. BENNEWITZ and S. BEHNKE (2007). *Multi-cue Localization for Soccer Playing Humanoid Robots*. In LAKEMEYER, GERHARD, E. SKLAR, D. G. SORRENTI and T. TAKAHASHI, eds.: *RoboCup 2006: Robot Soccer World Cup X*, vol. 4434 of *Lecture Notes in Artificial Intelligence*, pp. 245–257. Springer.

[STULP et al., 2004] STULP, FREEK, S. GEDIKLI and M. BEETZ (2004). *Evaluating Multi-Agent Robotic Systems Using Ground Truth*. In *Proceedings of the Workshop on Methods and Technology for Empirical Evaluation of Multi-agent Systems and Multi-robot Teams (MTEE)*, Ulm, Germany.

[THRUN et al., 2005] THRUN, SEBASTIAN, W. BURGARD and D. FOX (2005). *Probabilistic Robotics*. MIT Press.

[THRUN et al., 2001] THRUN, SEBASTIAN, D. FOX, W. BURGARD and F. DELLAERT (2001). *Robust Monte Carlo Localization for Mobile Robots*. Artificial Intelligence, 128(1-2):99–141.

[TILDEN, 2004] TILDEN, MARK W. (2004). *Neuromorphic robot humanoid to step into the market*. The Neuromorphic Engineer, 1(1):12.

[VISSER and BURKHARD, 2007] VISSER, UBBO and H.-D. BURKHARD (2007). *RoboCup: 10 Years of Achievements and Future Challenges*. AI Magazine, 28(2):115–130.

[VOIGTLÄNDER et al., 2007] VOIGTLÄNDER, ARNE, S. LANGE, M. LAUER and M. RIEDMILLER (2007). *Real-time 3D Ball Recognition using Perspective and Catadioptric Cameras*. In *Proceedings of the 3rd European Conference on Mobile Robots (ECMR)*, Freiburg, Germany.

[WANG et al., 2009] WANG, CHIEH-CHIH, S.-C. WANG, H.-C. YEN and C.-H. CHANG (2009). *NTU Robot PAL 2009 Team Report*. Only available online: `http://www.csie.ntu.edu.tw/~bobwang/RoboCupSPL/NTU_Robot_PAL_09Report.pdf`.

[WEIGEL and NEBEL, 2003] WEIGEL, THILO and B. NEBEL (2003). *KiRo - An Autonomous Table Soccer Player*. In KAMINKA, GAL A., P. LIMA and R. ROJAS, eds.: *RoboCup 2002: Robot Soccer World Cup VI*, vol. 2752 of *Lecture Notes in Artificial Intelligence*, pp. 384–392. Springer.

[WILKING and RÖFER, 2005] WILKING, DIRK and T. RÖFER (2005). *Real-time Object Recognition Using Decision Tree Learning*. In NARDI, DANIELE, M. RIEDMILLER, C. SAMMUT and J. SANTOS-VICTOR, eds.: *RoboCup 2004: Robot Soccer World Cup VIII*, vol. 3276 of *Lecture Notes in Artificial Intelligence*, pp. 556–563. Springer.

[WISSPEINTNER et al., 2010] WISSPEINTNER, THOMAS, T. VAN DER ZANT, L. IOCCHI and S. SCHIFFER (2010). *RoboCup@Home: Results in Benchmarking Domestic Service Robots*. In BALTES, JACKY, M. G. LAGOUDAKIS, T. NARUSE and S. SHIRY, eds.: *RoboCup 2009: Robot Soccer World Cup XIII*, vol. 5949 of *Lecture Notes in Artificial Intelligence*, pp. 390–401. Springer.

[WURMAN et al., 2008] WURMAN, PETER R., R. D'ANDREA and M. MOUNTZ (2008). *Coordinating Hundreds of Cooperative, Autonomous Vehicles in Warehouses*. AI Magazine, 29(1):9–19.

[ZAGAL and DEL SOLAR, 2005] ZAGAL, JUAN CHRISTÓBAL and J. R. DEL SOLAR (2005). *UCHILSIM: A Dynamically and Visually Realistic Simulator for the RoboCup Four Legged League*. In NARDI, DANIELE, M. RIEDMILLER, C. SAMMUT and J. SANTOS-VICTOR, eds.: *RoboCup 2004: Robot Soccer World Cup VIII*, vol. 3276 of *Lecture Notes in Artificial Intelligence*, pp. 34–45. Springer.

[ZAGAL et al., 2004] ZAGAL, JUAN CHRISTÓBAL, J. R. DEL SOLAR and P. VALLEJOS (2004). *Back to Reality: Crossing the Reality Gap in Evolutionary Robotics*. In *Proceedings of the 5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles - IAV2004*, Lisbon, Portugal.

[ZAHEDI et al., 2008] ZAHEDI, KEYAN, A. VON TWICKEL and F. PASEMANN (2008). *YARS: A Physical 3D Simulator for Evolving Controllers for Real Robots*. In CARPIN, STEFANO, I. NODA, E. PAGELLO, M. REGGIANI and O. VON STRYK, eds.: *Simulation, Modeling and Programming for*

Autonomous Robots (SIMPAR 2008), vol. 5325 of *Lecture Notes in Artificial Intelligence*, pp. 75–86. Springer.

[VAN DER ZANT and WISSPEINTNER, 2006] ZANT, TIJN VAN DER and T. WISSPEINTNER (2006). *RoboCup X: A Proposal for a New League Where RoboCup Goes Real World*. In BREDENFELD, ANSGAR, A. JACOFF, I. NODA and Y. TAKAHASHI, eds.: *RoboCup 2005: Robot Soccer World Cup IX*, vol. 4020 of *Lecture Notes in Artificial Intelligence*, pp. 166–172. Springer.

[ZHOU et al., 2009] ZHOU, CHANGJIU, E. PAGELLO, E. MENEGATTI, S. BEHNKE and T. RÖFER, eds. (2009). *Proceedings of the Fourth Workshop on Humanoid Soccer Robots in conjunction with the 2009 IEEE-RAS International Conference on Humanoid Robots*, Paris, France.

Appendix A

Released Software

In addition to publishing contributions at conferences or in journals, releasing software or data is an important part of scientific work that allows other researchers the reproduction or even the reuse of your own work. In this chapter, the software releases of five different projects, which – at least partially – originated from the works described in this dissertation, are described. This includes the code releases of the two RoboCup teams *GermanTeam* (cf. Sect. A.1.1) and *B-Human* (cf. Sect. A.1.2), the robot simulator *SimRobot* (cf. Sect. A.2), the robot tracking suite *SSL-Vision* (cf. Sect. A.3), and the *GMapping Localization Library* (cf. Sect. A.4). For each software release its relations to the previously described scientific work are pointed out.

A.1 Released RoboCup Code

In the RoboCup community, it is a common practice – and sometimes even an obligation – of participating teams to make their code available for others after a competition. This policy aims to foster the overall progress within the RoboCup community. In the context of this thesis, a number of research results have been released to the public as part of the code releases of the GermanTeam and B-Human.

A.1.1 The GermanTeam

Until 2005, the GermanTeam (cf. Sect. 2.1.1) provided a code release together with a comprehensive report after each RoboCup competition. The most influential publications have been those after the team's RoboCup victories in 2004 [48] and 2005 [52]. Many teams all over the world have used the published source code as a starting basis for their development [GERMANTEAM, 2008]. The most recent code and documentation is available at:

`http://www.germanteam.org`

The potential field-based behavior architecture [8] as well as the self-localization approach presented in [17] have been released within this context. In addition, intermediate versions of SimRobot are part of these releases (cf. Sect. A.2).

A.1.2 B-Human

Similar to the GermanTeam, B-Human (cf. Sect. 2.1.3) also provided code releases and reports for the community since 2008 [47, 51, 50]. Currently, several other SPL teams base their development on these releases, e. g. *NTU Robot PAL* from Taiwan [WANG et al., 2009], BURST from Israel [KEIDAR et al., 2010], and MRL from Iran [HASHEMI et al., 2010]. The software and the documentation are available at:

`http://www.b-human.de/en/publications/`

These releases are closely related to the scientific contributions regarding state estimation that are presented in Chap. 3. They include recent implementations of most humanoid robot-related approaches. In addition, up-to-date SimRobot versions (cf. Sect. A.2), the walking approach presented in [21], and the dynamic kicking engine described in [15] have also been released in this context.

A.2 SimRobot

In the course of this thesis, the further development of the SimRobot application has been done in the context of RoboCup. Therefore, several intermediate releases of SimRobot's at that point of time latest version have been made available to the public in the course of regular code releases of the GermanTeam (cf. Sect. A.1.1) and B-Human (cf. Sect. A.1.2). In addition, a stand-alone version – not including any dependencies to any RoboCup software – has been made available on the SimRobot homepage:

`http://www.informatik.uni-bremen.de/simrobot/`

Research and publications regarding SimRobot have been described in detail in Chap. 4 as well as in Sect. 5.4.2. As the application is an important base for most robotics research presented in this thesis, it also indirectly contributed to the realization of other work.

A.3 SSL-Vision

SSL-Vision is an open source software project maintained by the RoboCup Small Size League community. Although it provides a general framework for image processing tasks, it features particular components that are necessary for object recognition in the SSL scenario. Since 2010, it is the league's official and exclusive vision system [ROBOCUP SMALL SIZE LEAGUE TECHNICAL COMMITTEE, 2010]. The application is available through a public Subversion repository hosted at:

`http://code.google.com/p/ssl-vision/`

A short description of SSL-Vision is given in Sect. 3.6.1. A comprehensive overview is provided in the according publication [20]. The system has also been used to provide ground truth for the research presented in [4, 10, 19].

A.4 GMapping Localization Library

The GMapping localization library is a result of the *SHARE-it* (Supported Human Autonomy for Recovery and Enhancement of cognitive and motor abilities using information technologies) project, which has been co-funded in the EU's Sixth Framework Programme. The library has been published on the project's web site at:

`http://www.ist-shareit.eu/shareit/material/gmapping-library`[1]

In addition, parts of the changes of the GMapping library have been integrated into its original repository that is open to the public [Stachniss et al., 2010].

The library has been created and used in the context of the research regarding indoor navigation as described in Sect. 5.2 and published in [14, 18] and [26].

[1] In case of unavailability, an alternative URL is `http://www.informatik.uni-bremen.de/~timlaue/gfs/GridFastSlamLocalizer.zip`