Lehrstuhl für Realzeit-Computersysteme
Technische Universität München

Motion Compensation in Minimally Invasive Robotic Surgery

Tobias Johannes Ortmaier

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Information-stechnik der Technischen Universität München zu Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr.-Ing., Dr.-Ing. E.h. G. Schmidt
Prüfer der Dissertation:
      1. Univ.-Prof. Dr.-Ing. G. Färber
      2. Hon.-Prof. Dr.-Ing. G. Hirzinger

I

II

Preface

This Ph.D. thesis was written during my work at the *Institute of Robotics and Mechatronics* at the German Aerospace Center (DLR) in Oberpfaffenhofen. First of all I like to thank Prof. G. Hirzinger who gave me the freedom and possibility to work in the field of medical robotics. I also like to thank the supervisor of my Ph.D. thesis Prof. G. Färber, *Institute for Real-Time Computer Systems* at the Technical University of Munich, for support and invaluable advice.

This work took place in the medical robotics group at our institute. Many thanks to my colleagues for good teamwork and interesting discussions. It has been great fun working with you.

Some experiments and implementations were carried out during internships and master thesis' of students. Those contributions were an important part of the *motion compensation* project and played an important role for its success.

Many thanks to the *service team* at our institute, without whom life in the jungle of bureaucracy would be much harder.

My colleagues, reading and correcting this manuscript, proofed to be patient and humorous. I hope, it wasn't too hard for you and – someday – it will be my turn to read your work. My friends encouraged me in finishing this thesis. Your motivation was incredible. Someday, I will pay this back to you. My girlfriend Veronika had to suffer most, especially at the end of my work, and supported me with (re-) drawing numerous figures. Thanks a lot. And, last but not least, I would like to thank my parents, without them, I would not be what I am.

Munich, August 2002   Tobias J. Ortmaier

Contents

Abstract

Minimally invasive beating heart surgery enables the patient to experience a more gentle form of surgery, but increases the requirements for the surgeon: The remaining motion of the mechanically stabilized beating heart makes fast and safe surgery difficult. The goal of an advanced robotic surgery system is to compensate for this motion. This work presents control and vision algorithms necessary for such novel robotic surgery applications.

Cartesian position and velocity control laws in a minimally invasive surgery environment allow correct hand eye coordination which the surgeon got used to in open surgery. In combination with appropriate filtering and scaling of input commands, high accuracy manipulation of fine structures is possible. A self-adapting Cartesian force control law reduces the risk of unintentional damage of delicate tissue structures.

Motion of the mechanically stabilized beating heart is locally captured by tracking natural landmarks with an affine motion model. To circumvent disturbances of the tracking approach, specular reflections on the heart surface have to be handled appropriately. Automatic detection of landmarks allowing reliable determination of the affine model parameters can be achieved by special confidence measures. A new motion prediction framework is introduced to further increase robustness of the motion tracking scheme. This framework is able to compensate for short occlusions and small disturbances. Additional signals correlated with the heart motion (e.g. electrocardiogram) are included in this prediction scheme to reduce the dependency on visual information.

Chapter 1

Introduction

According to medical doctors, the 20th century was the century of surgery. The treatment of diseases, which led to death before, such as appendicitis, have become clinical routine now. The last century was also the period of huge traumata, due to large incisions, which are necessary to gain access to the operation area. The 21st century is expected to be the century of minimal and micro access techniques, reducing the patients' trauma dramatically. Here, computer and robot assisted surgery will lead to a revolution of the operating techniques, similar to the introduction of imaging techniques before.

During the last years several surgery robots developed at research institutes have entered hospitals for experimental or even routine application. Robodoc[TM] from Integrated Surgical Systems Inc. [33] or Caspar[TM] from URS Universal Robot Systems [75] are used for bone surgery, whereas the daVinci[TM] system from Intuitive Surgery Inc. [25] or Zeus[TM] from Computer Motion Inc. [14] have been designed for minimally invasive surgery. Numbers of clinical applications increase due to the expected advantages such as high precision, use of preoperative planning data, and the possibility of new surgery techniques, such as minimally invasive beating heart surgery. The disadvantages of robotic surgery include the increased time for training and surgery, high costs, and the restriction to only few indications. Regarding the requirements of surgical robots, roughly three different medical fields of application can be distinguished: In neuro-surgery workspace is very limited, payload is low, but precision demands are very high. Bone surgery leads to high forces and vibrations, therefore modified industrial robots are used often. Dealing with soft tissue as in abdominal or heart surgery, leads to low endeffector forces, but increases the work space. Other characteristics are tissue deformation in case of contact as well as tissue movement due to blood flow and respiration.

This chapter briefly introduces the peculiarities of minimally invasive surgery (MIS) as well as requirements for an advanced minimally invasive robotic surgery

(MIRS) system. The last section takes a closer look at minimally invasive cardiac surgery.

1.1   Minimally Invasive Surgery

Minimally invasive surgery (MIS) is an operation technique established in the 1980s. It differs from open surgery in that the surgeon works with long instruments through small incisions (typically <10 mm) and that he has no direct access to the operation field as in open surgery. Usually, four small incisions are necessary: two for the surgical instruments, one for the laparoscope (rigid endoscope), and one for insufflating CO_2 (see Fig. 1.1).
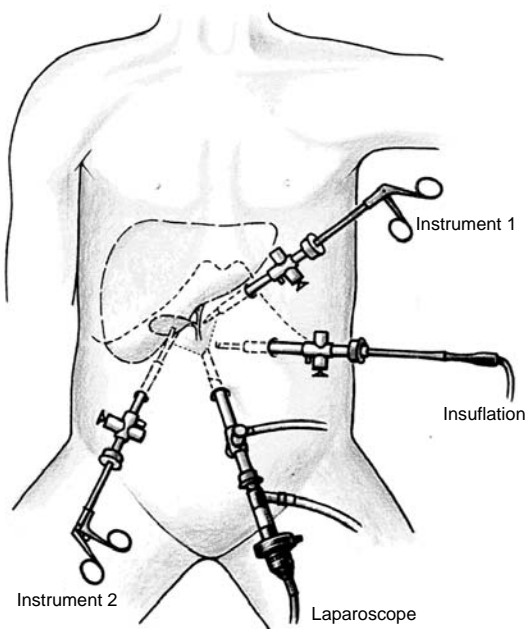
Figure 1.1: MIS setup (source unknown).

Despite the advantages of MIS (as given below) this technique did not evolve as predicted after its introduction in the operating room (OR). The main exception is minimally invasive gallbladder surgery: According to [72] about 70% of all gallbladder surgeries in the United States, Europe and Japan in the year 1992 were performed laparoscopically. This lack of market share occurs because the advantages of MIS are mainly for the patients and the disadvantages remain at the surgeon.

The advantages of MIS compared to open surgery are, among others: small inci-

sions which reduce pain and trauma, shorter hospital stays, shorter rehabilitation time, and cosmetical advantages.

Of course, MIS has disadvantages as well: The use of a laparoscope requires the surgeon to cope with reduced sight, which can lead to severe orientation problems during surgery, e.g. if the surgeon has to find anatomical structures identified previously in preoperative planning. Furthermore, he loses the direct hand eye coordination he has during open surgery [73]. The long instruments (approx. 30 cm) have to be moved around an *invariant point* (trocar point or entry point) on the patients' chest or abdominal wall. As a result reverse hand motion (or chop-stick effect) occurs, as well as configuration dependent scaling of the tool-tip velocity of the instrument and amplification of the surgeon's tremor (see Fig. 1.2(a)). The friction in the trocar reduces haptic feedback. Palpation of tissue is not possible, because the surgeon does not have direct access to operating area. The loss of haptic feedback may be compensated by visual feedback: tissue deformation can be interpreted as a measure of the exerted forces; of course this does not work with stiff materials such as needles. Additionally, the surgeon's dexterity when performing tasks is reduced dramatically. Only four degrees of freedom (DoFs) remain inside the body (see Fig. 1.2(b)), due to the kinematic restrictions at the trocar point. Therefore, the surgeon cannot reach any point in the work space at an arbitrary orientation. This is a main drawback of MIS that makes complex tasks like knot tying very time consuming and requires intensive training [28, 32, 65, 20, 58, 45].

(a) Scaling of velocity   (b) DOFs inside the body

Figure 1.2: Kinematic situation during MIS.

3

1.2 Minimally Invasive Robotic Surgery

To avoid the drawbacks of manual MIS, minimally invasive robotic surgery (MIRS) plays an important role. MIRS systems help the surgeon to overcome barriers such as the patient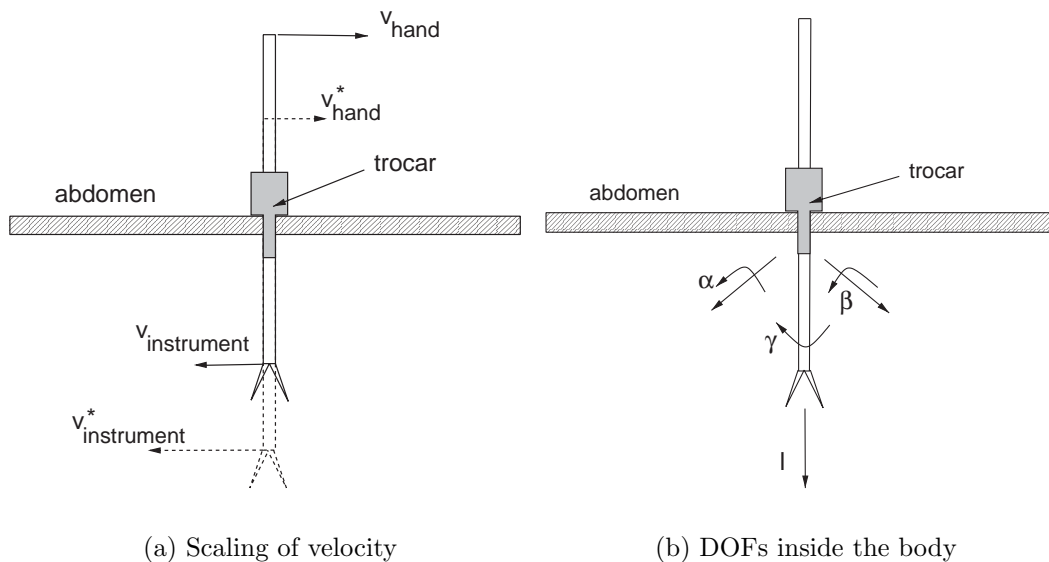's chest or abdomen, which separate him from the operating area. Furthermore, it is possible to overcome distances if surgeon and patient are located in different rooms or even hospitals (so-called virtual hospital) [21].

This section derives requirements for an advanced MIRS system in detail. As in Sec. 2.3, where the DLR MIRS scenario is given, the properties presented here are divided into three parts according to the physical components of a telepresence system:

1. Slave

2. Master

3. Communication between slave and master.

The *slave system* consists of several subsystems: The minimally invasive instruments should be small (diameter less than 10 mm) in order to reduce pain and trauma to a minimum. They should have at least two additional DoFs to guarantee full manipulability inside the body [41] and should allow the measurement of force as well as tactile information. This sensor data can be used for feedback to increase the quality of immersion of the operator and for more intelligent control laws of surgical robots. Control laws should allow the use of a variety of command modes (e.g. position, velocity, and force) to make the use of the entire system more intuitive and secure: Force measurements can be used to limit the maximum manipulation force and to detect collision between instruments. Furthermore, force control laws can be used to compensate the organ movement in case of contact between instrument and organ and thus help to avoid damage of the tissue. Minimally invasive surgery robots are expected to allow a large variety of applications. This will help to reduce the costs of MIRS. Additionally, they have to be very lightweight, so that they can be handled by one person easily: this is very important in case of emergency situations, when the robots have to be removed to get direct access to the patient. Furthermore, this reduces the set-up time before an operation. The change of instruments, still a very time consuming task, has to be enhanced further.

The *master system* has to provide high quality feedback, both tactile and kinesthetic. The first helps the surgeon to use palpation, as in open surgery. This is necessary to find invisible structures (e.g. blood vessels below a fat layer). The latter gives him direct access to the forces at the operating area and therefore increases the quality of the operation [77]. Movements of instruments should be possible without kinematic limitations (e.g. trocar point) in Cartesian space.

This makes the use of a surgical telepresence system more intuitive. Scaling the surgeon's motion and filtering the surgeon's tremor are two additional important features to increase the safety and accuracy of a MIRS system. The application of high quality 3D vision is strongly recommended to gain depth information. Augmented reality will help to close the gap between preoperative planning and intraoperative navigation; for which the registration problem has to be solved. A high-level user interface (e.g. to move the camera) seems to be an appropriate supplement to ease the handling of the MIRS system.

The *communication* between master and slave has to be flexible to allow the connection of different master stations (not necessarily located in the same OR) to get support by an additional expert (which is today limited to video conferencing) or to enhance training of surgeons. Here, an unexperienced surgeon can get immediate support from an experienced colleague. Therefore, the communication network has to be safe (guaranteed bandwidth and communication delay) and secure (no undesired third-party listening). The communication should be independent of the underlying network layer (ATM, ISDN, ADSL,...) by defining and acquiring an appropriate Quality of Service (QoS).

By these efforts MIS will get safer and faster in the future. Faster surgery leads to two important results: costs are reduced (one minute in a cardiac OR costs about $ 15) and so are possible postoperative complications for the patient. Here, autonomy, as a logical result of the above-mentioned topics, plays an important role. Possible autonomous tasks (those that can be executed by a robot autonomously) include: Automatic camera guidance [79, 2], holding of needles, positioning of instruments, grasping of tissue, automatic cutting and suturing (in combination with new surgical techniques and instruments). To realize autonomous functions, special care has to be taken for the organ motion. This motion is induced by the patient's respiration and heart beat and has to be detected and compensated reliably.

1.3 Minimally Invasive Robotic Cardiac Surgery

Of about 100,000 cardiac surgeries performed in Germany every year 75% are coronary artery bypass graft surgeries [7]. Usually the stenosed part of the coronary artery is bridged with an unaffected artery taken from the chest wall or the forearm. The conventional approach contains severe risks and a significant morbidity for the patient. This is due to:

- splitting the sternum and spreading the ribcage to gain access to the heart

- the use of the heart-lung machine to sustain the circulation (on-pump coronary artery bypass)

The heart-lung machine can lead to severe intraoperative (e.g. stroke syndrome) and postoperative (e.g. neurological complications) risks. The splitting of the sternum leads to high trauma and is the main reason for the long convalescent time of 2-3 months [7]. Currently there are two main approaches to reduce these risks: To avoid the heart-lung machine, the surgeon operates on the beating heart. This includes the use of a mechanical stabilizer (e.g. Octopus^{TM} from Medtronic) to reduce motion in the area of interest [37]. MIS avoids splitting of the sternum, because the surgeon works through small incisions (Sec. 1.1). This surgical technique is called TECAB (totally endoscopic coronary artery bypass) and minimally invasive robotic cardiac surgery (MIRCS) systems are widely used. TECAB at the beating heart (OPCAB: off-pump coronary artery bypass) combines the advantages of conventional beating heart surgery and MIS. The drawbacks of this technique are the increased demands to the surgeon: additionally to the drawbacks of commercially available MIRCS systems (e.g. reduced sight, no haptic feedback, reduced manipulability, see Sec. 2.2) the remaining motion of the stabilized heart complicates safe and fast surgery. This disadvantage directly leads to the demand of measuring and compensating the organ movement. If the MIRCS system is able to measure and compensate the motion of the heart reliably, the surgeon can work on a virtually stabilized beating heart and gains back the safety he was used to in on-pump coronary artery bypass surgery.

1.4 Thesis Organisation

Figure 1.3 illustrates the closed visual servoing loop for motion compensation and is described in the following. The laparoscope, which is present anyway, seems attractive to capture the heart motion since introducing additional sensors into the very crowded operating field is problematic. Moreover, the laparoscope can capture the motion of the whole operating field and not only of a few selected points. Using artificial landmarks is problematic, too, due to the limited space. Therefore, prominent image structures on the heart surface are used as natural landmarks. The motion of the landmarks is approximated by an affine motion model. The parameter of the motion model are found at the minimum of a dissimilarity measure (e.g. sum of squared differences). These computations are carried out within the *tracking* module of Fig. 1.3. With this, the actual position of the landmark is found and the motion history can be used to predict the position of the landmark in near future. A promising approach for prediction which provides stable results is time series analysis (*prediction* module of Fig. 1.3). Prediction of the heart motion is useful to detect outliers of the tracking scheme. Furthermore, the obtained near-future positions of the landmarks are used to command the robot such that both heart and instrument move synchronously.

The following paragraphs give the structure as well as the scientific contributions
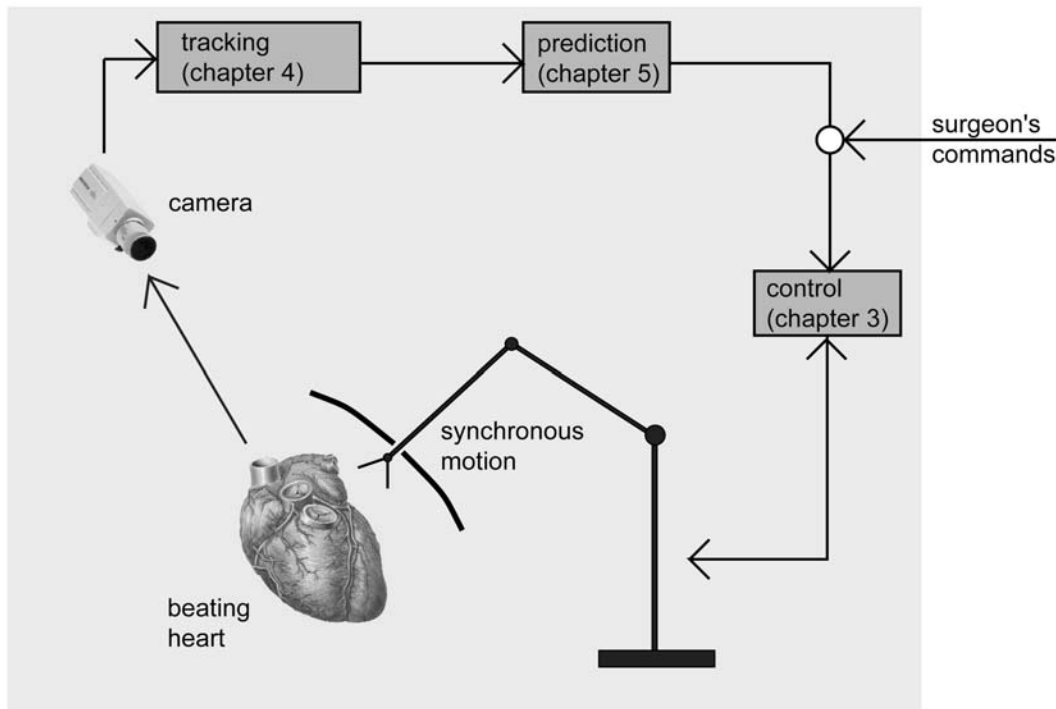
Figure 1.3: Motion Compensation Scheme (heart picture from [50]).

of this Ph.D. thesis:

Chapter 2 presents MIRS systems built at research institutes and companies. Thereafter, the DLR minimally invasive robotic surgery scenario which has been developed during this work is introduced. The slave system, the master system, and the communication between master and slave are discussed in detail.

Chapter 3 presents the forward and inverse kinematics of the surgical robot used. In the subsequent sections a Cartesian velocity and position control law is derived which allows unrestricted motion of the surgical instrument and correct hand-eye coordination. Both have the disadvantage of tissue damage due to unintentional high forces. Therefore, this chapter is concluded by a force control law avoiding this drawback.

Chapter 4 motivates the motion model used to capture the motion of the heart surface. To realize autonomous functions (Sec. 1.2) reliable motion estimation of the area of interest is one of the key issues. Therefore, special care is taken to handle specular reflections which occur on the heart surface. After that, tracking results are evaluated which includes the relevance of the parameters of the motion model, the accuracy of the tracking results, and the amplitude spectrum of

the computed trajectories. An algorithm for automatic detection of prominent structures on the heart surface, which allows reliable tracking, concludes this chapter.

Chapter 5 presents numerous techniques making motion estimation of the heart surface more robust with respect to occlusions and other disturbances. These techniques include the prediction of the heart motion as well as a net of simultaneously tracked landmarks. An overall tracking scheme exploiting additional sensor signals (such as the electrocardiogram, ECG), which are correlated with the heart motion, allows robust motion estimation in beating heart surgery. Prediction is necessary to detect and compensate outliers of the tracking scheme and to overcome the delay-time of the closed visual servoing loop (see Fig. 1.3), thus allowing for a proper compensation of the heart motion.

Chapter 6 concludes this work and presents directions of further research. New applications can be found in tumor biopsy and treatment, where motion compensation may increase accuracy and safety [67].

Chapter 2

Minimally Invasive Robotic Surgery Systems

From the surgeon's point of view, a good MIRS system should fulfill the following requirements (some of them are also given in [66]): Modern MIRS should provide both realistic tactile and kinesthetic feedback, which gives the surgeon direct impression of tissue and underlying structures (e.g. blood vessels). Visual 3D feedback in combination with Cartesian control should allow correct hand-eye coordination as in open surgery. Actuated instruments are expected to provide full manipulability inside the body which is necessary to reduce surgery time as well as training time. Intelligent assistance functions, such as automatic positioning of instruments, automatic cutting, and grasping, as well as safety features are also desired.

Currently several groups at international research institutes and companies are working on MIRS systems. One has to distinguish between assistance systems and complete MIRS telepresence systems, consisting of master and slave subsystems.

Robotic assistance systems are mainly used to hold the laparoscope [47, 64, 3] or as flexible and intelligent tool holders [16]. The interaction with these assistance systems is either via voice [47, 64], via tracking of a pointer [3] or with force sensors mounted on the robot [16]. Not all benefits of a complete MIRS system such as kinesthetic feedback and full manipulability can be achieved by these assistance systems. Therefore, they are not considered any further here.

This chapter presents telepresence systems developed at research institutes and companies. After that, the DLR MIRS system which has been developed during this work is discussed in detail.

2.1 MIRS Systems at Research Institutes

The following paragraphs present several MIRS telepresence systems, developed at research institutes all over the world. Of course the list presented here is far from being complete, but the intention is to give an overview about different research activities.

A research group at UC Berkeley/UC San Francisco is developing a complete telepresence system [11] (see Fig. 2.1(a) for the prototype of the system): The slave is based on the Millirobot^{TM} (with a diameter of 10 mm, see Fig. 2.1(b)) that realizes two actuated degrees of freedom inside the body and thus a total of six DoFs and therefore full manipulability. The master is based on a modified Impulse Engine^{TM} 3000 from Immersion Inc. and is therefore prepared for force feedback in four DoFs. No force feedback is available with the prototype system [11], although this is strongly recommended by the authors. Recently, the second generation of the telepresence system has been presented [10, 12]: The master is based on a PHANToM (see Fig. 2.2(a)), the modified Millirobot can be seen in Fig. 2.2(b). According to [12] the system still does not provide force feedback.
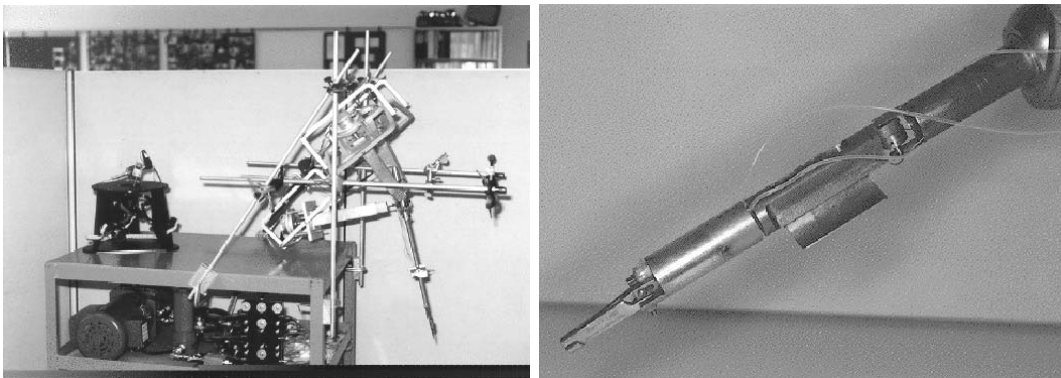
(a) Master (left) and slave (right)   (b) Millirobot

Figure 2.1: The Berkeley telesurgery scenario (prototype) [74].

In Korea a group at KAIST [43, 38] has developed a telepresence system for micro surgical tasks. It allows six DoF force/torque reflection at the master console. The slave consists of an industrial six DoF robot equipped with a modified six DoF Stewart platform for micro manipulation. Besides the fact that industrial robots are not designed for use in the operating room (OR), the system does not allow full manipulability if used in laparoscopic surgery.

ARTEMIS (Advanced Robot and Telemanipulator System for Minimal Invasive Surgery) is one of the first MIRS systems and was developed at the "Forschungs-

(a) Master

(b) Modified Millirobot

Figure 2.2: The Berkeley telesurgery scenario [74].

zentrum Karlsruhe" in Germany. The master (see Fig. 2.3(a)) consists of various input devices: two haptic manipulators for the slave robots, voice recognition for the laparoscope, and foot pedals. The slave consists of three robots: two hold and manipulate surgical instruments (TISKA-robots, see Fig. 2.3(b), the two outer robots) while the other holds the laparoscope (ROBOX-robot, see Fig. 2.3(b), the robot in the center). The relative position between the TISKA-robots and the ROBOX-robot is known, therefore the laparoscope can follow the surgical instruments automatically. As a drawback the ARTEMIS system does not have force feedback at the master side and there are not any instruments with additional degrees of freedom at the slave side.
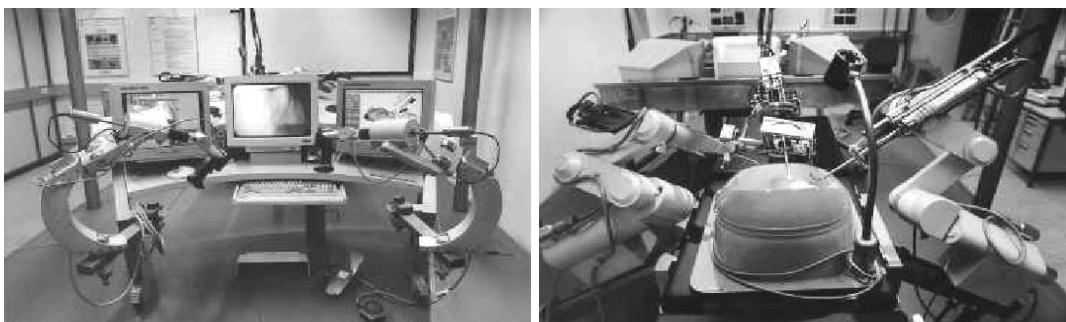
(a) ARTEMIS master

(b) ARTEMIS slave

Figure 2.3: The ARTEMIS system [4].

11

In Switzerland a project of the research program CIMINT [13] (Computer Aided and Image Guided Medical Interventions) deals with robot-assisted minimally invasive bypass surgery. It is planned to develop a new telepresence system, designed for beating heart surgery, which consists of new instruments that allow force and tactile measurement to support the surgeon to find invisible structures. The focus of this project lies on motion compensation algorithms that enable, in combination with new surgical instruments, (semi-) automatic anastomosis.

At the German Aerospace Center (DLR) a complete MIRS system based on a modified AESOP^{TM} 3000 from Computer Motion Inc. has been developed during this work. It is discussed in Sec. 2.3 in detail. Its current main drawbacks are the slow serial communication between robot and controller that limits the bandwidth of the closed (control) loop as well as the lack of minimally invasive surgical instruments with additional DoFs.

2.2 Commercially Available MIRS Systems

At the moment two widely known MIRS systems are commercially available: Computer Motion Inc. (CMI) has developed the Zeus^{TM} System (consisting of modified AESOP robots [64] with a master console) and Intuitive Surgical Inc. (ISI) has developed the daVinci^{TM} [25] system. The Zeus and the daVinci systems have been evaluated mainly in the field of minimally invasive heart surgery, but further applications will be established in future.
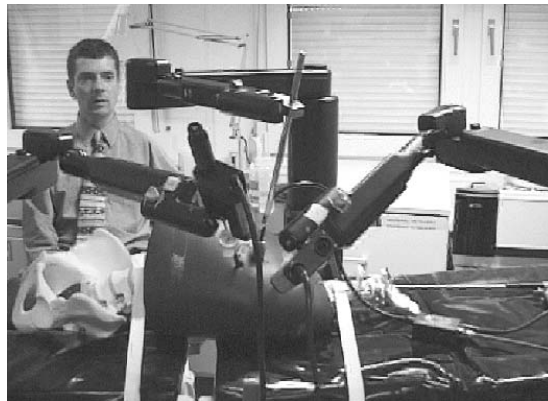
(a) Master [14]   (b) Slave

Figure 2.4: The Zeus^{TM} components.

Both systems offer only position control, there is no force measurement at the slave side as well as no haptic feedback at the master console. None of them

| (a) Master | (b) Slave | (c) Endowrist™ |
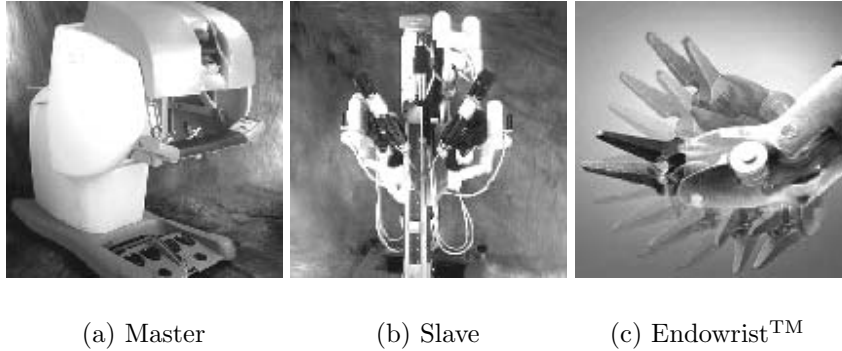
Figure 2.5: The daVinci™ components [34].

offers autonomous execution of tasks, such as monitoring the work space, nor is it possible to integrate preoperative planning data into the system. Scaling of motion commands, filtering of the surgeon's tremor, and optical magnification is available with both systems.

The main difference currently lies in the minimally invasive instruments sold by the two companies: Instruments from CMI have 5 DoFs inside the human body, whereas ISI enables 6 DoFs with its Endowrist™ system (see Fig. 2.5(c)).

Training time as well as setup time before an operation increases with both systems. Additionally, MIRS procedures take longer than pure manual (open) surgery [35] and thus lead to higher costs. One reason for the increased operation duration is the difficult intraoperative navigation to find relevant blood vessels, which is a direct result of the loss of palpation in MIRS. Also, the lack of force feedback leads to the difficult task of interpreting organ deformation as a measure of the forces at the slave side [77].

2.3   The DLR MIRS Scenario

The telesurgery scenario shown in Fig. 2.6 has been developed during this work to overcome the drawbacks of today's MIRS systems and to establish new techniques. It can be divided into three parts: local environment (master system), remote environment (slave system), and communication inbetween. This section is a summary of [56, 57] and briefly presents the current state of the scenario as well as further directions of research.

An AESOP 3000 DS (CMI) equipped with a medical instrument (e.g. scalpel, forceps) is used as the teleoperator (slave). Haptic data can be collected in combination with a sensor which enables force/torque measurement at the tool

13
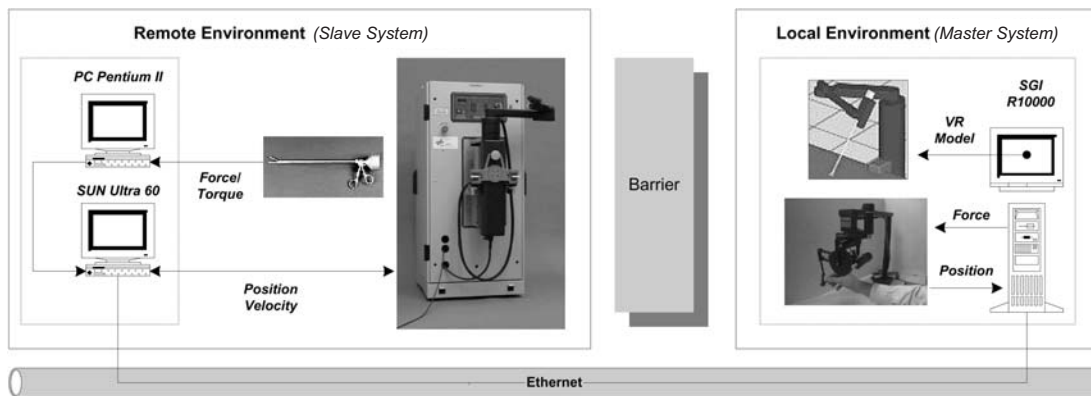
Figure 2.6: Schematic overview of the DLR telesurgery scenario

center point of the instrument. The visualization of the teleoperator environment consists of a stereo video stream generated by a laparoscope which is currently displayed on a video screen with stereo capabilities. Additionally, an Open Inventor based simulation model is displayed, running on a SGI RS 10000 workstation. This simulation requests the actual joint values of the robot to animate the virtual robot representation as well as contact forces which can be rendered using force arrows. This approach provides an intuitive possibility to move the instrument and gives visual feedback of contact forces. The PHANToM T-Model from Sensable Technologies is used as a haptic display for forces measured by the sensors of the remote instrument. The teleoperator can be commanded in Cartesian space and forces can be reflected in three degrees of freedom (without torques). One way CORBA-TCP/IP connections were implemented to transmit data between the operator (master) and the teleoperator.

First the developments in the slave system are presented. After that, the improvements at the master side of the DLR minimally invasive robotics surgery scenario (MIRS) are given. The following section takes a closer look at the communication between master and slave.

2.3.1 Slave System

The focus of this section lies on the development of the slave system, based on an existing surgical robot (AESOP 3000 DS). Included are new surgical instruments, control algorithms, and autonomous functions.

14

Surgical Instruments

Presently, instruments for MIRS differ from those for conventional MIS mainly in the substitution of the handles by electrically driven joints. Additional capabilities making the instruments more similar to the surgeon's hands and reducing the restriction due to the trocar point are realized only seldomly:

- Force and torque sensors at the instrument tip
- Tactile sensors at the tool tip
- Additional degrees of freedom inside the body
- Multi-functional instruments, e.g. a combination of scalpel and forceps

The use of the advantages of medical robotics is limited, due to the technical state mentioned in Sec. 2.1 and Sec. 2.2. Therefore, further development is necessary to realize all benefits of MIRS. Two topics are investigated here: additional degrees of freedom and measurement of forces and tactile information inside the human body. Additional degrees of freedom at the tool tip allow faster and safer surgery because the kinematics of the surgeon's hand is imitated inside the human body (beyond the trocar point). The surgeon can thus work intuitively in a way similar to open surgery. Force measurement at the instrument tip allows a variety of additional functions: First of all, force information can be displayed in the picture from the laparoscope, giving the surgeon visual feedback of actual forces. Furthermore, haptic feedback (with an appropriate input device) is possible: the surgeon receives force feedback at his fingertips. Additionally, autonomous features, limitation to a maximum force (e.g. during suture) or a constant force (e.g. holding tissue while the organ is moving) are possible. One can also think about reliable collision detection between two tooltips, realized by force/torque sensors.

Sensorized Scalpel   As mentioned previously, today's minimally invasive instruments do not allow the measurement of forces. However, by experience, the surgeon is able to interpret tissue deformations as a measure of the exerted forces and to compensate for the lack of haptic information. Unfortunately, tissue elasticity depends on the patient and may also vary over time (e.g. due to blood pressure). Furthermore, this interpretation leads to a reduction of the surgeon's dexterity to perform manual work. A sensorized scalpel for MIS with a diameter of only 10 mm (see Fig. 2.7(a) and Fig. 2.7(b)) was developed to circumvent this drawback and to give the surgeon direct haptic information. Forces can be measured in all dimensions and so can torque about the instrument axis. The sensor is placed directly behind the blade and can measure forces up to 20 N. The software of the scalpel runs on a digital signal processor (DSP) board (PC 32 of Innovative Integration, processor speed 60 MHz). A sample rate of 1.7 ms for all

(a) Model   (b) Realisation
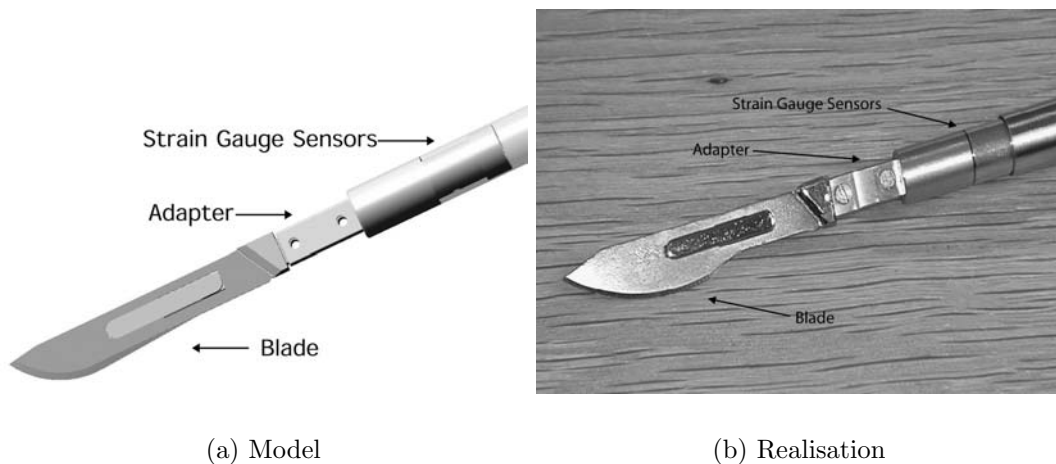
Figure 2.7: Sensorized scalpel.

four channels (three forces and one torque) guarantees high quality force measurement. To reduce signal noise to a minimum, the analog signals are amplified inside the instrument, digitized (10 bit resolution) and then transmitted (fast serial TMS 320 protocol with 2 μs clockperiod) to the DSP board.

Sensorized Forceps A rigid instrument limits the work space behind the trocar. Full manipulability can not be achieved because additional degrees of freedom, similar to the human hand, are missing (see Fig. 1.2(b)). This is the reason why complicated operations such as coronary surgery require long training and are very time consuming. A pair of forceps with two additional actuated degrees of freedom near the tooltip is currently developed. This enables the surgeon to move the instrument's tooltip in six DoF inside the human body. Thus the surgeon gains back full manipulability as in open surgery.

The drives for the joints and the forceps themselves are realized as electro-mechanical actuators and are located outside the body. As the instrument is equipped with sensors close to the tip, real manipulation forces are measured. Figure 2.8 shows a CAD model of the tool tip as well as the first prototype. The forceps have a diameter of 10 mm. Mechanics and sensors can handle forces up to 10-15 N and can be sterilized.

The use of electronics inside the body leads to several additional constraints, including the ability to guarantee the compatibility of the electronics with the human body and vice versa. This has to be done for the electronic parts themselves as well as for the electric signals. Furthermore, one has to guarantee electro-magnetic compatibility: other electrical devices may not be disturbed by the forceps and vice versa. There are still unanswered questions concerning the

16

(a) Model   (b) Realisation
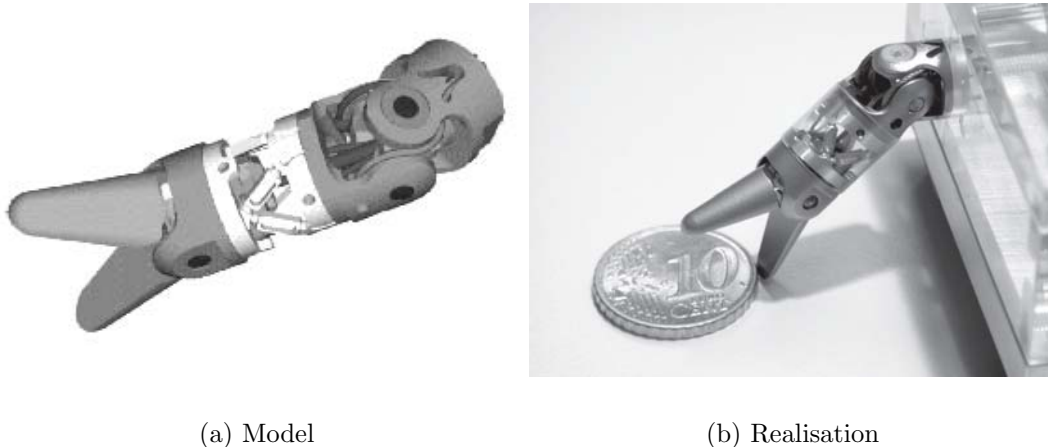
Figure 2.8: Sensorized forceps.

sterilizability of electrical components, so one may have to use alternative methods to measure and transmit information (e.g. optical methods). Due to the high complexity those instruments cannot be built entirely as being disposable. One has to develop sterilizable instruments or a combination of complex, reusable components with simple and easy to change disposable components.

Robot Control

Development in control laws is necessary to make the application of MIS robots more intuitive and to increase the safety of the entire system. Furthermore, autonomous capabilities reduce the requirements of the communication between master and slave system and support the surgeon during his work.

The robot is connected with the controller workstation via a serial port (RS232). This slow communication line (communication delay $T_D \approx 22$ ms) limits the bandwidth of the closed control loop and corrupts the quality of force control laws and haptic feedback. Nevertheless, it is sufficient to demonstrate new techniques and capabilities. A detailed description is given in Chap. 3, therefore, only a brief overview is presented here.

Cartesian Control   As mentioned in Sec. 1.1 one of the drawbacks of MIS is the restriction of motion due to the trocar point (entry point), see Fig. 1.2(b). Control algorithms have been developed to realize free, unrestricted motion inside the human body, avoiding the chop-stick effect. They allow the use of an arbitrary Cartesian input device, such as the PHANToM or the SpaceMouse$^{\text{TM}}$. Additionally this is a standard interface for a wide range of control algorithms.

A detailed description of Cartesian control is given in Sec. 3.1.

Control Modes Today's MIRS systems have only one control mode: position control. Pure position control has several drawbacks: high forces exerted, either to tissue or between two robots, are a potential source of damage and injury. Depending on the chosen scaling between master and slave, indexing is often necessary. Furthermore, no autonomous tasks can be realized safely, because it is not possible to detect contact situations reliably. To overcome these disadvantages, several control algorithms were developed, implemented, and evaluated.

A velocity control law was developed to avoid indexing. The main problem lies in the fact that motion of the robot is very slow and only joint positions of the robot can be obtained, so noise corrupts the calculated velocity values. Therefore, after identifying the dynamics of the robot, an observer was built. Closing the control loop with the observer yielded good performance. The entire control structure is discussed in detail in Sec. 3.2.

One of the main advantages of the developed velocity controller is that it can be extended to force control or mixed force/velocity control easily without switching the control structure. If no forces are measured inside the human body the commands are interpreted as velocity commands and after contact in one or more directions the commands are interpreted as force.

A control law for mixed position/force control is also available. Again, after detecting contact with the environment, manual switching of the control structure is not necessary. To obtain best performance for force control, the stiffness of the environment is estimated at every sample and the parameters of the controller are tuned accordingly (Sec. 3.5). Only the global behavior of the control loop has to be defined (e.g. overshooting, damping).

Force control is necessary to detect collision between instruments and to avoid damage of tissue. Furthermore, this is a step towards autonomous tasks: a robot is able to hold tissue with a certain allowed maximum force and compensates the motion of the organ automatically. Pure position control, on the other hand, would possibly damage tissue.

Autonomy In advanced telesurgery scenarios the surgeon has to cope with (multi-) sensor data. He processes this data and derives commands which are transmitted to the robot. This huge amount of information offered to the surgeon leads to high demands to his sensomotoric skills, which cannot always be met (e.g. in minimally invasive beating heart surgery). Therefore, a reduction of these demands is strongly recommended. This leads to a control structure in which the surgeon and robots share tasks: The robot takes charge of simple tasks requiring high bandwidth, whereas the surgeon is responsible for complex tasks with low bandwidth. One example is the movement of the tool tip along a

predefined path: the surgeon commands the velocity only (one degree of freedom) and the robot moves the tool tip along the six DoF path. This requires online measurement of the position of the target, because it may change due to the patient's respiration and heart beat. In minimally invasive beating heart surgery motion compensation is a highly desired functionality: The robot compensates the heart motion, such that the relative pose between the heart surface and the tool center point of the surgical instrument remains constant. The surgeon can then work on a virtually stabilized heart as he was used to in on-pump surgery (here the heart-lung machine is used to sustain the circulation and the heart does not move). Another autonomous task that has been introduced into the OR recently is automatic camera guidance for MIS using color markers at the instrument, developed by the DLR [79, 2].

Motion estimation Before any autonomous task, as suggested in Sec. 1.2 can be performed, one has to cope with the organ motion arising from the patient's breathing or heart beat. Therefore, reliable algorithms to measure this motion are an essential part of an advanced MIS robotic system. Algorithms have been developed which are able to track the motion of the beating heart surface, exploiting natural landmarks (Chap. 4). Figure 2.9 shows three natural landmarks (size of 30×15 pixels) and the corresponding translational search areas (size of 70×70 pixels). Tracking of landmarks is easily disturbed by specular reflections on the heart surface. Thus several algorithms to detect specularities and reconstruct the underlying surface structure have been developed and evaluated, see Sec. 4.3 and Sec. 4.4. Furthermore, criteria have been derived to automatically
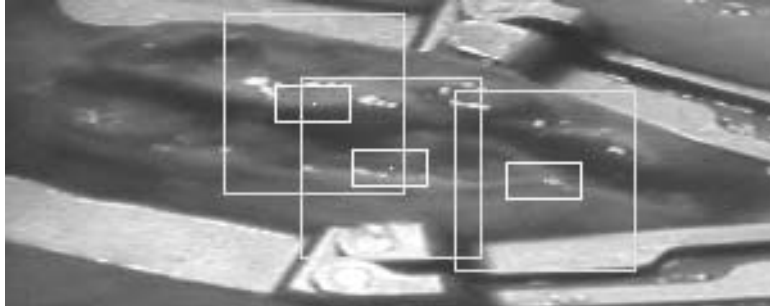
Figure 2.9: Natural landmarks and tracking area.

detect such regions of interest on the heart surface that are appropriate for the tracking algorithm (Sec. 4.6).

Efforts were made to make vision based motion recognition more reliable: a framework to cope with (temporal) occlusion of the considered landmarks and to detect outliers is presented in Chap. 5. Furthermore, several prediction algorithms were evaluated, again to improve the reliability by delivering additional
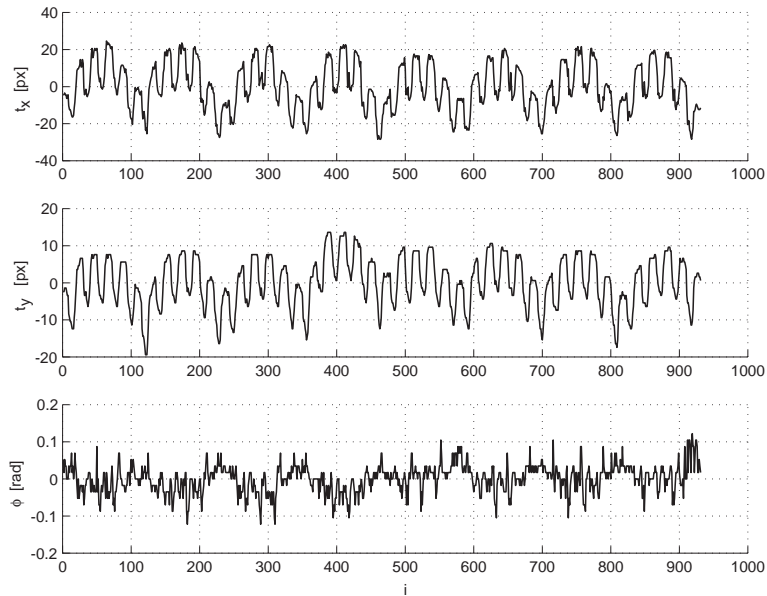
Figure 2.10: Detected trajectories.

information (Sec. 5.2 and Sec. 5.5). Prediction is possible, because the motion of landmarks on the heart surface is made up of two quasi-periodic movements: the heart beat itself and motion induced from the patient's respiration. Figure 2.10 shows three tracking parameters versus the discrete sample time i: translation (t_x and t_y are given in pixels [px]) and rotation (ϕ is given in radiant [rad]). For the translational parameters the quasi-periodic behavior can be seen well, whereas this is not easily possible for rotation.

Additionally, algorithms that allow the use of ECG signals as an additional source of information to make tracking more robust are investigated in Sec. 5.4. A surgical system equipped with reliable information about the motion of the area of interest will be able to compensate this motion automatically and to guide the instrument e.g. to a point moving on the heart surface.

2.3.2   Master System

The master system allows the surgeon to command the surgical robots and to perform the surgery. Therefore, the master has to provide an accurate and realistic impression of the surgical situation at the remote side as well as an intuitive and ergonomic working environment.

Medical master systems available today do not have the possibility of force or tactile feedback. Furthermore, they are restricted to position control as the only command mode and do not allow the use of additional information from preop-

20

erative tools for path-planning etc.

Haptic Input Device

Input devices available from CMI or ISI have no force feedback and differ in their kinematics in that the CMI input device is kinematically restricted while ISI allows free motion in six DoF.

To allow a variety of control modes as described in Sec. 2.3.1, two different six DoF input devices are used: the SpaceMouse for velocity and velocity/force control and the PHANToM for position and position/force control. This gives the surgeon the possibility to evaluate different control modes and decide which one to use in different situations during surgery. Additionally, force feedback with the PHANToM offers the possibility to display the actual forces during MIS. This information helps to avoid damage of tissue, needles or threads.

To compensate the communication delay between master and slave an estimation of the remote environment stiffness (Sec. 3.5) is used to calculate a local update for the force feedback loop [61].

Vision

3D video is essential in the DLR and commercially available MIRS systems and is well developed, whereas the combination with 3D-models and data from preoperative planning seems far away. This is because several problems still need to be solved or further development is necessary: registration of the patient, fast (realtime) elastodynamic simulation of organs, 3D vision-based modeling of organs, matching of multimodal data in realtime, etc.

To give the surgeon a realistic impression of the configuration of the robot at the operation table a simulation system that displays the movement of the robot at the remote station (see Fig. 2.11) was developed.

Additionally the actual forces which occur in contact situations are given. With this simulation the surgeon gets an impression of the remote forces, even if he does not have force feedback input devices.

2.3.3 Communication

As described previously, telepresence technology in combination with a medical robot assistant enables the surgeon to operate as efficiently as possible from a local desktop. The desktop and the robot are simply connected via a local point-to-point communication link using a dedicated protocol to exchange data. This limits the distance between the surgeon and the slave to a few meters. Otherwise,
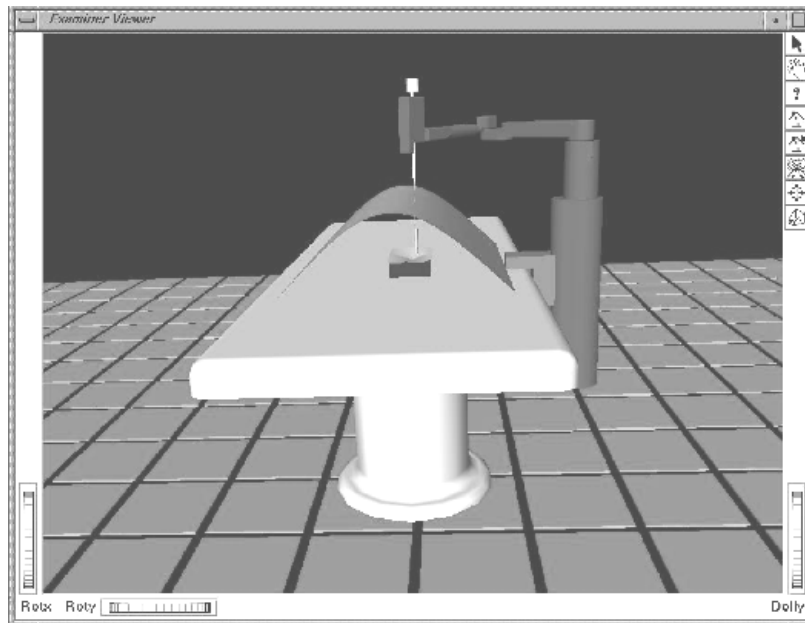
Figure 2.11: Configuration of remote robot.

applications for telecooperation and teleconsultation are already in development which use broadband technology to connect various desktops all over the world and setup a virtual hospital, whereas currently available MIRS systems are meant to act in a single user environment: Each surgeon uses his own strictly separate desktop without interacting with his assistant. This limits the consultation of a remote expert to audio and video information, received from a separate ISDN-based video conferencing system. Furthermore, surgery is a very complex task typically performed by more than one surgeon, but the provision for a third party connectivity does not exist.

The prototype of the telepresence system is shown in Fig. 2.6. Sensorized robotic technology is combined with modern telecommunication and information technologies (object-oriented middleware) to support minimally invasive surgery and to overcome the limitations of currently available MIRS systems using a supplementary robot control computer, which is connected to the local desktop via an Ethernet link [62].

Distributed Objects

The development of distributed telepresence systems is a complex, time consuming task. To simplify this task the Object Management Architecture (OMA), defined by the Object Management Group (OMG) [51], describes an object-oriented infrastructure in their Common Object Request Broker Architecture (CORBA),

which allows objects to communicate platform and implementation independent. This is realized by a collection of objects which separates the service provider (server) from the service requester (client) by using a well-defined encapsulating interface.

Interfaces

The telepresence system represents an object framework divided into several client/server objects which handle haptic and video stream data. The object connectivity is realized by CORBA-TCP/IP adapters (see Fig. 2.12).

To describe the remote environment virtually, Open Inventor and its scene graph are used. Therefore a CORBA-based Open Inventor Viewer was developed. Started as a server on the master side this viewer displays the remote robot configuration. A graphical user interface supports flexible settings like indexing or scaling at run-time (see also [60]).

The shared autonomy module on the slave side provides several tasks scheduled by a separate task controller. Implemented as a periodically timed task the controller triggers both the command stream and the model update stream task using CORBA interfaces. In fact, the command stream task uses a CORBA interface of the communication task to communicate with the PHANToM on the master side. The surgeon interacts with the remote robot by commanding the Cartesian position of the remote instrument simply by moving the local PHANToM.

The actual joint angles of the remote robot as well as the torques and forces are sent to the Inventor Viewer for visualization by the world model stream task. As interprocess communication between local software modules a shared memory approach realized with posix/solaris threads is implemented.

2.4   Conclusions

In this section MIRS systems developed at research institutes or companies were discussed. Due to their technical state not all benefits of MIRS systems can be realized (e.g. full manipulability and force feedback). Therefore, MIRS systems lead to increased operation durations and thus to higher costs. This restrains the fields of application of robotic telesurgery systems.

The DLR MIRS scenario presented in the previous sections is summarized as follows: The focus of the DLR MIRS system regarding miniaturized surgical instruments lies in additional DoFs inside the human body to gain full manipulability and in additional sensors to measure force and tactile information.

Free Cartesian motion of the instrument tool tip is necessary to allow an intuitive control of surgical devices. Depending on the surgical task, position, velocity,

and/or force control is necessary. Force control itself is a step towards autonomy of the robots, because damage of tissue, collision detection, and motion compensation can be realized. Furthermore, algorithms are proposed that allow tracking of natural landmarks on the beating heart and thus make it possible to measure and compensate the organ motion of a beating heart.

Force feedback, either with appropriate haptic devices or with a model based simulation, provides the surgeon with a more realistic impression of the remote situation and helps to increase safety.

Network independent communication between master and slave, realized by COR-BA, gives the possibility to build a flexible telepresence system and is a step towards telesurgery and virtual hospitals.
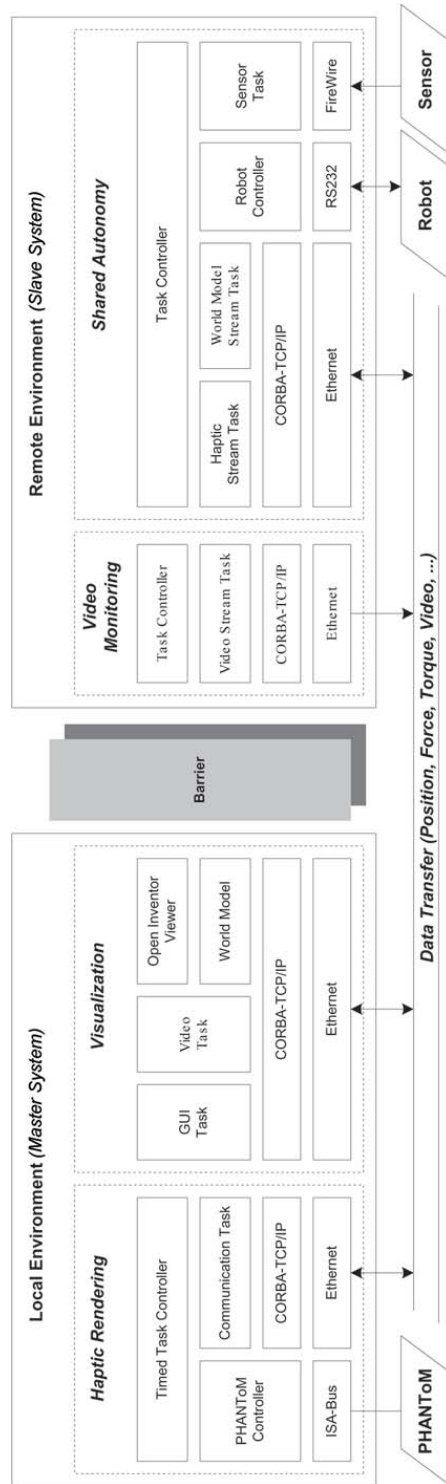
Figure 2.12: Distributed master-slave system architecture.

Chapter 3

Control in Minimally Invasive Surgery

Unrestricted Cartesian motion of surgical instruments is one of the key issues to enhance the quality of a MIRS system (Sec. 1.2): The surgeon can move his instruments as he was used to in open surgery. The spherical movement around the entry point is avoided, as well as the inverse hand motion (see Sec. 1.1). Furthermore, there is no scaling of the velocity of the instrument which depends on the position of the entry point. This means that the velocity of the instrument tool-tip is independent of the actual working geometry. Also a Cartesian interface is needed for further control algorithms, such as force control or autonomous tasks.

In this chapter a Cartesian control algorithm for the AESOP robot used in the DLR MIS telepresence scenario is presented. After introducing the forward and inverse kinematics (Sec. 3.1) an algorithm to calculate the position of the entry point is presented, followed by the calculation of the inverse kinematics. These are the fundamental algorithms that allow Cartesian velocity control (see Sec. 3.2), Cartesian position control (see Sec. 3.4), and Cartesian force control (see Sec. 3.5). Parts of Sec. 3.1 and Sec. 3.2 were published by the author in [54] and parts of Sec. 3.3 can be found in [55].

3.1   Fundamentals of Cartesian Control

The robot consists of four active joints, the linear axis θ_1 and three rotatory joints $\theta_2, \theta_3, \theta_7$ and two passive joints θ_5 and θ_6 (see Fig. 3.1(a)). These two passive joints are not motor driven and have no breaks. They guarantee that no forces are exerted to the entry point (trocar point, see Fig. 1.2(a)). This entry point can be considered as a geometric constraint to the kinematics of the robot

that binds two degrees of freedom (DoF). As result, inside the abdomen only four DoFs are left (see Fig. 1.2(b)). Additionally, the robot consists of another rotatory joint θ_4 that can be used to suit the geometry to the actual situation. The transformation from the tool-center point frame to the world frame is:

$$\boldsymbol{X}_{\text{TCP}}(\boldsymbol{\Theta}) = \boldsymbol{A}_1(\Theta_1)\boldsymbol{A}_2(\Theta_2)\cdots\boldsymbol{A}_7(\Theta_7) \tag{3.1}$$

with

$$\boldsymbol{\Theta} = [\Theta_1, \ldots, \Theta_7]^T \tag{3.2}$$

and $\boldsymbol{A}_k(\Theta_k)$ (with $k \in \{1,..,7\}$) representing homogeneous transformations.

(a) Joints

(b) Frames
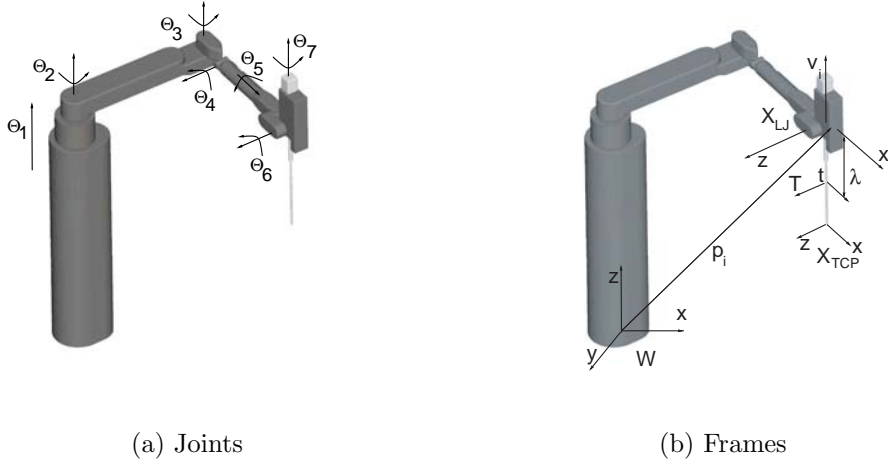
Figure 3.1: Minimally invasive surgery robot.

3.1.1 Entry Point Estimation

As already mentioned the position of the entry point has a strong influence on the kinematics of the robot. To allow Cartesian control of the tool-tip the entry position has to be calculated.

One possibility to acquire the position of the entry point is to place an electromagnetic tracker near the trocar. The disadvantages of this method are that these trackers are very sensitive to steel or aluminum and that additional equipment in the operating room is needed. An other idea might be to move the tool-center point of the robot to the trocar and to compute $\boldsymbol{X}_{\text{TCP}}$. This demands a calibration for every surgical instrument used. In [42] the authors use a force/torque sensor mounted on the robot to calculate the position of the entry point; here, again, additional equipment is necessary.

To avoid these drawbacks another way is chosen. Joint position data during the first few seconds are collected, after the instrument has entered the abdomen. With these data the position of the entry point can be calculated. At each time-step t_i the joint values are:

$$\boldsymbol{\Theta}_i = \boldsymbol{\Theta}(t_i) \qquad \text{with} \qquad i \in \{1, \ldots, N\} \ . \tag{3.3}$$

The homogeneous transformations can be written as:

$$\boldsymbol{A}_{k_i} = \boldsymbol{A}_k \left(\Theta_k \left(t_i \right) \right) \quad \text{with} \quad k \in \{1, .., 7\} \quad \text{and} \quad i \in \{1, .., N\} \ . \tag{3.4}$$

The frame $\boldsymbol{X}_{\text{LJ}}$ located at the last joint (see Fig. 3.1(b)) with respect to the world frame $\boldsymbol{W}$ can be calculated:

$$\boldsymbol{X}_{\text{LJ}}(\boldsymbol{\Theta}_i) = \boldsymbol{A}_{1_i} \ldots \boldsymbol{A}_{6_i} = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}_i \ . \tag{3.5}$$

Where

$$\boldsymbol{p}_i = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}_i \tag{3.6}$$

is the position of the last joint in the world frame $\boldsymbol{W}$ and

$$\boldsymbol{v}_i = \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix}_i \tag{3.7}$$

is the vector of the instrument axis in $\boldsymbol{W}$. A line $\boldsymbol{g}_i$ coincident with the instrument axis can be written as:

$$\boldsymbol{g}_i : \qquad \boldsymbol{x} = \boldsymbol{p}_i + \lambda_i \boldsymbol{v}_i \qquad \text{with} \qquad \lambda_i \in \mathbb{R} \ . \tag{3.8}$$

Under the assumption that the entry point does not move, the intersection of the two lines $\boldsymbol{g}_i$ and $\boldsymbol{g}_{i+1}$ is the entry point. Due to noise and slow movement of the entry point (e.g. due to the patient's respiration) these two lines might not have an intersection but pass each other. As an approximation the minimum distance d between $\boldsymbol{g}_i$ and $\boldsymbol{g}_{i+1}$ is calculated and it is assumed that the entry point lies on $\boldsymbol{g}_i$. Therefore, a plane E perpendicular to $\boldsymbol{g}_i$ is defined:

$$E : \qquad \left[\boldsymbol{x} - \left(\boldsymbol{p}_i + \lambda_i \boldsymbol{v}_i \right) \right]^T \boldsymbol{v}_i = 0 \ . \tag{3.9}$$

The intersection of E with $\boldsymbol{g}_{i+1}$ is calculated. This leads to

$$\lambda_{i+1}^* = \frac{\boldsymbol{p}_i^T \boldsymbol{v}_i + \lambda_i - \boldsymbol{p}_{i+1}^T \boldsymbol{v}_i}{\boldsymbol{v}_{i+1}^T \boldsymbol{v}_i} \ . \tag{3.10}$$

The optimal $\lambda_i = \lambda_i^*$ minimizing the squared distance d^2 is given as:

$$\arg\min_{\lambda_i} d^2 = \arg\min_{\lambda_i} \|\boldsymbol{p}_i + \lambda_i \boldsymbol{v}_i - \boldsymbol{p}_{i+1} - \lambda_{i+1}^* \boldsymbol{v}_{i+1}\|_2^2 \ . \tag{3.11}$$

Solving

$$\frac{\partial}{\partial \lambda_i} d^2 = 0 \tag{3.12}$$

for λ_i leads to

$$\lambda_i = \lambda_i^* \quad = \quad \frac{n}{1 + k^2 - 2k\boldsymbol{v}_i^T \boldsymbol{v}_{i+1}} \tag{3.13}$$

with

$$\begin{aligned} n \quad = \quad & -\boldsymbol{v}_i^T \boldsymbol{p}_i + \boldsymbol{p}_{i+1}^T \boldsymbol{t}_i - k\boldsymbol{p}_{i+1}^T \boldsymbol{v}_{i+1} + \\ & + c\boldsymbol{v}_i^T \boldsymbol{v}_{i+1} + k\boldsymbol{p}_i^T \boldsymbol{v}_{i+1} - ck \ , \end{aligned}$$

and

$$k = \frac{1}{\boldsymbol{v}_i^T \boldsymbol{v}_{i+1}} \quad c = \frac{\boldsymbol{p}_i^T \boldsymbol{v}_i}{\boldsymbol{v}_{i+1}^T \boldsymbol{v}_i} - \frac{\boldsymbol{p}_{i+1}^T \boldsymbol{v}_i}{\boldsymbol{v}_{i+1}^T \boldsymbol{v}_i} \ . \tag{3.14}$$

An estimation for the entry point position can then be written as

$$\boldsymbol{t}_i = \boldsymbol{p}_i + \lambda_i^* \boldsymbol{v}_i \ . \tag{3.15}$$

To reduce the influence of noise an average value is calculated:

$$\boldsymbol{t}_{\text{av}} = \frac{1}{N-1} \sum_{i=1}^{N-1} \boldsymbol{t}_i \ . \tag{3.16}$$

To consider movements of the entry point during surgery an online update $\boldsymbol{t}_i$ for $\boldsymbol{t}_{\text{av}}$ after initialization is computed. The point $\boldsymbol{x}_i = \boldsymbol{t}_i \in \boldsymbol{g}_i$ that has the minimal distance from $\boldsymbol{t}_{\text{av}}$ as an approximation for position of the entry point is chosen:

$$\boldsymbol{t}_i = \boldsymbol{p}_i + \lambda_i \boldsymbol{v}_i \quad \text{with} \quad \arg\min_{\lambda_i} \|\boldsymbol{t}_{\text{av}} - \boldsymbol{t}_i\|_2 \ . \tag{3.17}$$

3.1.2 Inverse Kinematics

This section presents the calculation of the inverse kinematics of the robot which is used in the subsequent sections to close Cartesian position, velocity, and force control loops. As the inverse kinematics is strongly influenced by the position of the entry point, the estimation presented in Sec. 3.1.1 is included in the calculations given here.

The Jacobian matrix $\boldsymbol{J}$ of the position of the tool-center point $\boldsymbol{x}$ can be written as

$$\boldsymbol{J} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\Theta}_t} \quad \text{with} \quad \boldsymbol{\Theta}_t = [\Theta_1, \Theta_2, \Theta_3, \Theta_5, \Theta_6]^T \;, \tag{3.18}$$

because Θ_7 has no effect on the position of the tool-center point and Θ_4 is for initial setup only. The velocity of the instrument tip in $\boldsymbol{W}$ is:

$$\dot{\boldsymbol{x}} = \boldsymbol{J}\dot{\boldsymbol{\Theta}}_t \tag{3.19}$$

with

$$\boldsymbol{J} = \begin{bmatrix} j_{11} & j_{12} & \cdots & j_{15} \\ j_{21} & j_{22} & \cdots & j_{25} \\ j_{31} & j_{32} & \cdots & j_{35} \end{bmatrix} . \tag{3.20}$$

As mentioned in Sec. 3.1, Θ_5 and Θ_6 are passive joints whose values are determined by the position of the trocar. This geometric constraint has to be considered in the calculation of the inverse kinematics. It is obvious that Eq. 3.21 has to be fulfilled at the entry point:

$$\dot{t}_{x_T} = 0 \quad \text{and} \quad \dot{t}_{z_T} = 0 \;, \tag{3.21}$$

where the subscript T denotes the trocar frame $\boldsymbol{T}$. In Sec. 3.1.1 the distance λ between the last joint and the entry point has been calculated, so frame $\boldsymbol{T}$ can be determined as:

$$\boldsymbol{T} = \boldsymbol{A}_1(\Theta_1)\boldsymbol{A}_2(\Theta_2)\cdots\boldsymbol{A}_6(\Theta_6)\boldsymbol{T}_y(\lambda) = \begin{bmatrix} n_x & s_x & a_x & t_x \\ n_y & s_y & a_y & t_y \\ n_z & s_z & a_z & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.22}$$

with

$$\boldsymbol{T}_y(\lambda) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -\lambda \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} .$$

The position of the entry point $\boldsymbol{t}_W$ is:

$$\boldsymbol{t}_W = [t_x, t_y, t_z]^T = \boldsymbol{g}(\Theta_1, \Theta_2, \Theta_3, \Theta_4, \Theta_5, \Theta_6, \lambda) \;. \tag{3.23}$$

The velocity of the instrument in world frame $\boldsymbol{W}$ at the entry point can be calculated by:

$$\dot{\boldsymbol{t}}_W = \boldsymbol{J}_t\dot{\boldsymbol{\Theta}}_t \quad \text{with} \quad \boldsymbol{J}_t = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{\Theta}_t} \;. \tag{3.24}$$

The matrix $\boldsymbol{R}^{-1}$ is used to transform the velocity $\dot{\boldsymbol{t}}_W$ into $\dot{\boldsymbol{t}}_T$:

$$\boldsymbol{R}^{-1} = \boldsymbol{R}^T = \begin{bmatrix} n_x & s_x & a_x \\ n_y & s_y & a_y \\ n_z & s_z & a_z \end{bmatrix}^T . \tag{3.25}$$

Now the velocity in the trocar frame $\boldsymbol{T}$ can be computed:

$$\dot{\boldsymbol{t}}_T = \boldsymbol{R}^{-1}\dot{\boldsymbol{t}}_W = \underbrace{\boldsymbol{R}^{-1}\boldsymbol{J}_t}_{\boldsymbol{C}}\dot{\boldsymbol{\Theta}}_t . \tag{3.26}$$

For abbreviation $\boldsymbol{C}$ is defined:

$$\boldsymbol{C} = \begin{bmatrix} a_1 & b_1 & c_1 & d_1 & e_1 \\ a_2 & b_2 & c_2 & d_2 & e_2 \\ a_3 & b_3 & c_3 & d_3 & e_3 \end{bmatrix} . \tag{3.27}$$

If Eq. 3.21, Eq. 3.26 and Eq. 3.27 are combined and solved for $\dot{\Theta}_5$ and $\dot{\Theta}_6$, the following equations are obtained:

$$\begin{aligned} \dot{\Theta}_5 &= \frac{1}{-d_3 e_1 + d_1 e_3}\Big((a_3 e_1 - a_1 e_3)\dot{\Theta}_1 + \\ &+ (b_3 e_1 - b_1 e_3)\dot{\Theta}_2 + (c_3 e_1 - c_1 e_3)\dot{\Theta}_3 \Big) , \end{aligned} \tag{3.28}$$

$$\begin{aligned} \dot{\Theta}_6 &= \frac{1}{d_3 e_1 - d_1 e_3}\Big((a_3 d_1 - a_1 d_3)\dot{\Theta}_1 + \\ &+ (b_3 d_1 - b_1 d_3)\dot{\Theta}_2 + (c_3 d_1 - c_1 d_3)\dot{\Theta}_3 \Big) . \end{aligned} \tag{3.29}$$

Defining new abbreviations:

$$\begin{aligned} \dot{\Theta}_5 &= k_1\dot{\Theta}_1 + k_2\dot{\Theta}_2 + k_3\dot{\Theta}_3 , & (3.30) \\ \dot{\Theta}_6 &= l_1\dot{\Theta}_1 + l_2\dot{\Theta}_2 + l_3\dot{\Theta}_3 & (3.31) \end{aligned}$$

and combining these with Eq. 3.19 and Eq. 3.20 the elements p_{nm} (n-th row and m-th column) of the resulting Jacobian matrix $\boldsymbol{P}^{3\times3}$ can be written as:

$$\dot{\boldsymbol{x}} = \boldsymbol{P}\begin{bmatrix} \dot{\Theta}_1 \\ \dot{\Theta}_2 \\ \dot{\Theta}_3 \end{bmatrix} \tag{3.32}$$

with

$$p_{nm} = j_{nm} + j_{n4}k_m + j_{n5}l_m \tag{3.33}$$

and

$$\text{n, m} \in \{1, 2, 3\} .$$

Finally,

$$\begin{bmatrix} \dot{\Theta}_1 \\ \dot{\Theta}_2 \\ \dot{\Theta}_3 \end{bmatrix} = \boldsymbol{P}^{-1}\dot{\boldsymbol{x}} \tag{3.35}$$

for the resulting inverse kinematics is obtained. The inverse kinematics takes information about the position of the entry point into account and therefore leads to good results when the Cartesian control laws are evaluated (see the following sections).

3.2 Velocity Control

The main advantage of velocity control in telesurgery applications is that large distances at the slave side (e.g. in liver biopsy) can be bridged without violating work space constraints at the master side. As a drawback position errors occur due to noise and model errors. Additionally, force in case of collision between two instruments or between instruments and tissue is not supervised and therefore is a source of danger.

To close a velocity control loop one needs information about the joint velocities. In the current setup only joint positions can be obtained, because of the interface between joint controller and software. Computing the joint velocity $\dot{\boldsymbol{\Theta}}_i \approx \frac{\boldsymbol{\Theta}_{i-1} - \boldsymbol{\Theta}_i}{\Delta t}$ does not lead to useful values due to noisy sensor signals and slow movement of the joints. To avoid additional velocity sensors an observer was build. Thereafter, a velocity control loop can be closed. Experimental results are given in the subsequent section.

3.2.1 Observer

As the robot moves slowly the dynamics of the joints can be assumed to be decoupled and can be modeled separately. Then, according to [29], the dynamics of each joint Θ_k can be modeled as a PT_1 transfer-function with dead time T_D. This model is appropriate if the robot has a position- or velocity-interface for the joints. Therefore, the velocity transfer function of each joint Θ_k can be written as follows:

$$F_k^* = \frac{1}{1 + T_k s} e^{-T_D s} \ . \tag{3.36}$$

The delay T_D due to communication (between the joint controllers and the software) and signal processing ($T_D = 22$ ms) is equal for all joints and much smaller than the time constant T_k of the joints (see Tab.3.1) and is therefore neglected while the observer is derived:

$$F_k = \frac{1}{1 + T_k s} \ . \tag{3.37}$$

Table 3.1: PT$_1$ parameters.

Joint	Θ_1	Θ_2	Θ_3	Θ_7
T_k in s	0.23	0.20	0.20	0.22

A new state-variable $\tilde{z}$ for the observer that contains all active joints $\tilde{\boldsymbol{\Theta}}_a$ and their velocities $\dot{\tilde{\boldsymbol{\Theta}}}_a$ is defined:

$$\tilde{z} = \left[\dot{\tilde{\boldsymbol{\Theta}}}_a^T, \tilde{\boldsymbol{\Theta}}_a^T\right]^T \tag{3.38}$$

with

$$\tilde{\boldsymbol{\Theta}}_a = \left[\tilde{\Theta}_1, \tilde{\Theta}_2, \tilde{\Theta}_3, \tilde{\Theta}_7\right]^T . \tag{3.39}$$

The equations for the observer are:

$$\begin{aligned}
\dot{\tilde{z}} &= \boldsymbol{A}_o\tilde{z} + \boldsymbol{B}_o\boldsymbol{u} + \boldsymbol{L}\boldsymbol{y} - \boldsymbol{L}\tilde{\boldsymbol{y}} \\
&= (\boldsymbol{A}_o - \boldsymbol{L}\boldsymbol{C}_o)\tilde{z} + \boldsymbol{L}\boldsymbol{y} + \boldsymbol{B}_o\boldsymbol{u} \tag{3.40} \\
\tilde{\boldsymbol{y}} &= \boldsymbol{C}_o\tilde{z} = \tilde{\boldsymbol{\Theta}}_a \tag{3.41} \\
\boldsymbol{y} &= \boldsymbol{\Theta}_a = [\Theta_1, \Theta_2, \Theta_3, \Theta_7]^T \tag{3.42}
\end{aligned}$$

with

$$\boldsymbol{A}_o = \begin{bmatrix}
-\frac{1}{T_1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -\frac{1}{T_2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{1}{T_3} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{1}{T_4} & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0
\end{bmatrix}$$

$$\boldsymbol{B}_o = \begin{bmatrix}
\frac{1}{T_1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{T_2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{T_3} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{T_4} & 0 & 0 & 0 & 0
\end{bmatrix}^T$$

$$\boldsymbol{C}_o = \begin{bmatrix}
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}$$

33

$$\boldsymbol{L} \;=\; \begin{bmatrix} l_{1,1} & 0 & 0 & 0 & l_{5,1} & 0 & 0 & 0 \\ 0 & l_{2,2} & 0 & 0 & 0 & l_{6,2} & 0 & 0 \\ 0 & 0 & l_{3,3} & 0 & 0 & 0 & l_{7,3} & 0 \\ 0 & 0 & 0 & l_{4,4} & 0 & 0 & 0 & l_{8,4} \end{bmatrix}^{T} .$$

The chosen structure of $\boldsymbol{L}$ allows to determinate the solutions of

$$\det\left(s\boldsymbol{E}^{8\times8} - (\boldsymbol{A}_o - \boldsymbol{L}\boldsymbol{C}_o)\right) = 0 \tag{3.43}$$

describing the poles, effectively. The solutions

$$s_{o_{2i-1,2i}} = \frac{-1 - l_{4+i,i}T_i \pm \sqrt{n_i}}{2T_i} \quad \text{for} \quad i = 1,..,4$$

with

$$n_i = -4T_i(l_{4+i,i} + l_{i,i}T_i) + (1 + l_{4+i,i}T_i)^2$$

are obtained. Considering the poles of the original system

$$\det(s\boldsymbol{E}^{8\times8} - \boldsymbol{A}_o) = 0 \tag{3.46}$$

the solutions are:

$$s_{1,2,3,4} = 0 \tag{3.47}$$

and

$$s_i = -\frac{1}{T_{i-4}} \quad \text{for} \quad i = 5,..,8 . \tag{3.48}$$

Solving

$$s_{o_{2i}}K = s_{4+i} \quad \text{and} \quad s_{o_{2i-1}} = s_{o_{2i}} \quad \text{for} \quad i = 1,..,4$$

leads to:

$$l_{i,i} = \frac{1 + \frac{1}{K^3} - \frac{3}{K}}{T_i^3} \qquad l_{4+i,i} = \frac{2 - K}{KT_i} . \tag{3.49}$$

With K it is possible to determine how fast, compared to the original system, the observer converges to the steady state. A small K leads to fast convergence but the observer is more sensitive to noise. A higher value for K filters noise but leads to a greater phase shift that degrades the stability of a closed loop system.

3.2.2 Velocity Control Loop

After building the observer a velocity control loop can be closed. The structure can be seen in Fig. 3.2: The subscript D of the variables denotes desired values, the subscript p passive joints ($\boldsymbol{\Theta}_p = [\Theta_5, \Theta_6]^T$), the subscript a active joints, and the tilde symbol ($\sim$) values computed with the observer. The transfer function $F_D = e^{-\frac{T_D}{2}s}$ represents the communication delay. The input is split into the Cartesian velocity $\dot{\boldsymbol{x}}$ and the angular velocity of the last joint $\dot{\theta}_7$. A simple P-controller with

$$\boldsymbol{F}_R = \mathrm{diag}(C_k) = \begin{bmatrix} C_1 & 0 & 0 & 0 \\ 0 & C_2 & 0 & 0 \\ 0 & 0 & C_3 & 0 \\ 0 & 0 & 0 & C_4 \end{bmatrix} \tag{3.50}$$

is used. Due to slow communication (serial port) between robot and controller one cannot use high error amplifications, because this will cause instability.
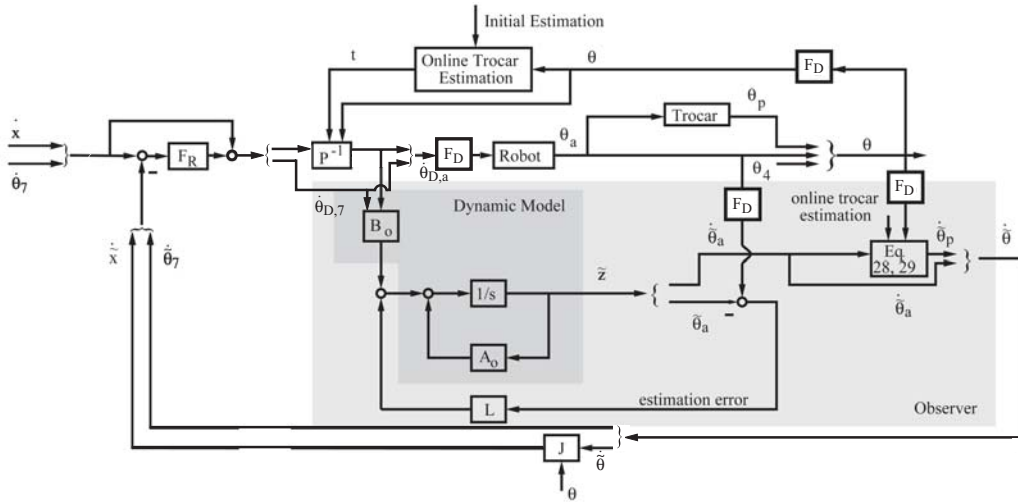

Figure 3.2: Velocity control structure.

3.2.3 Experiments

In the experiments $K = 0.5$ for the observer and $C_i = 0.5$ for the controller parameters were chosen. This is a good compromise between low noise amplification and small phase shift.

Figure 3.3 shows the online computation of the entry point position, as described in Sec. 3.1.1. Figure 3.4 shows the desired velocity (solid line) of the TCP in world frame $\boldsymbol{W}$ and the actual velocity computed by the observer (dashed line). It can

be seen that good tracking behavior has been achieved. Figure 3.5 visualizes the measured velocity of the TCP. Due to the large noise it is not possible to build a control loop with good performance, using the measured $\dot{\boldsymbol{x}}$ for feedback.
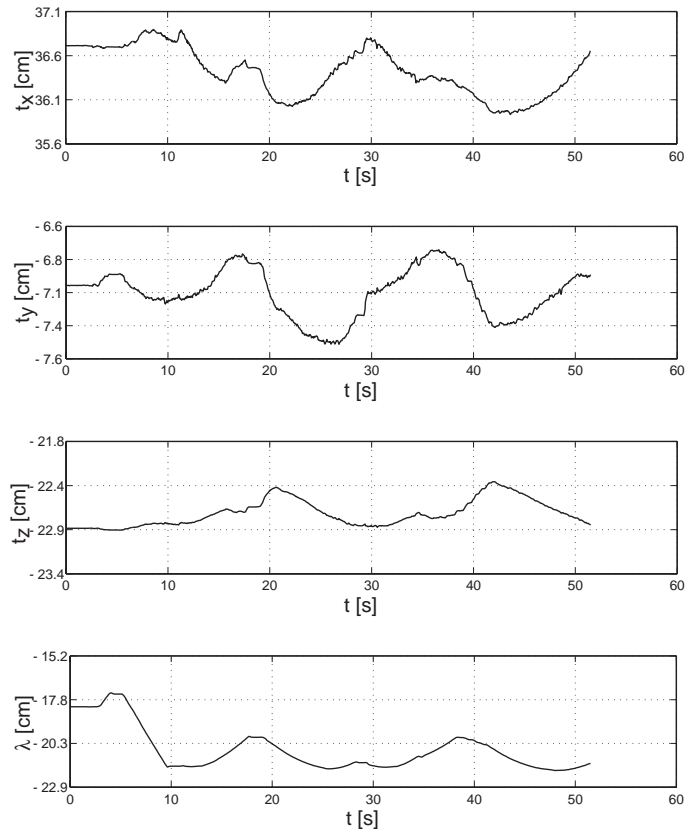

Figure 3.3: Entry point estimation.

Figure 3.6 shows the computed desired velocity for each joint (solid line), including the two passive joints Θ_5 and Θ_6, according to Eq. 3.28 and Eq. 3.29, as well as the joint velocities calculated by the observer (dashed line). Again good tracking has been achieved. These trajectories can be compared with those of Fig. 3.7 which presents the measured joint velocities. The error between the observed active joints and the measured active joints is given in Fig. 3.8; the difference remains small all over the sequence. The filtered trajectory (solid line) can be compared with data provided by the observer (dashed line), both given in Fig. 3.9: A small phase shift between these two trajectories occurs.
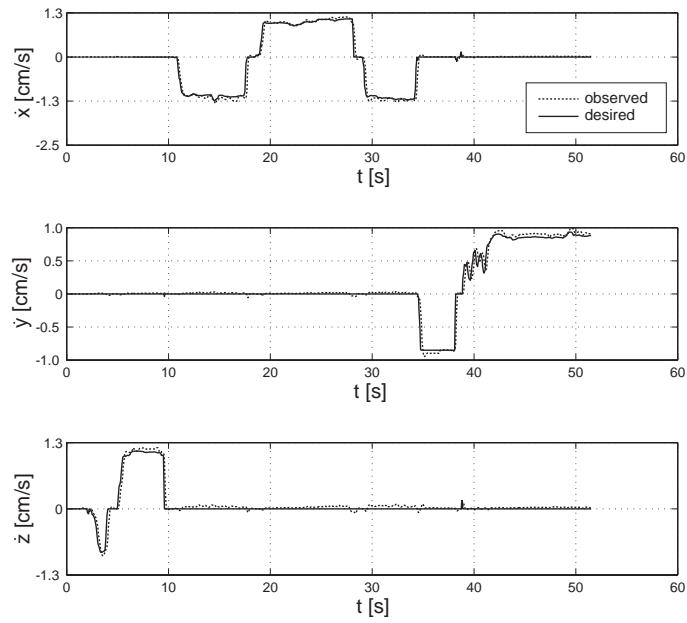
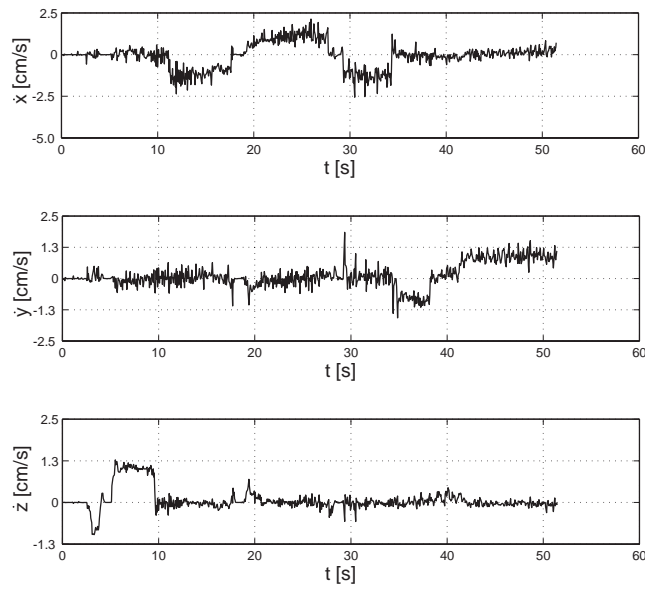Figure 3.4: Velocity of TCP, desired and observed.

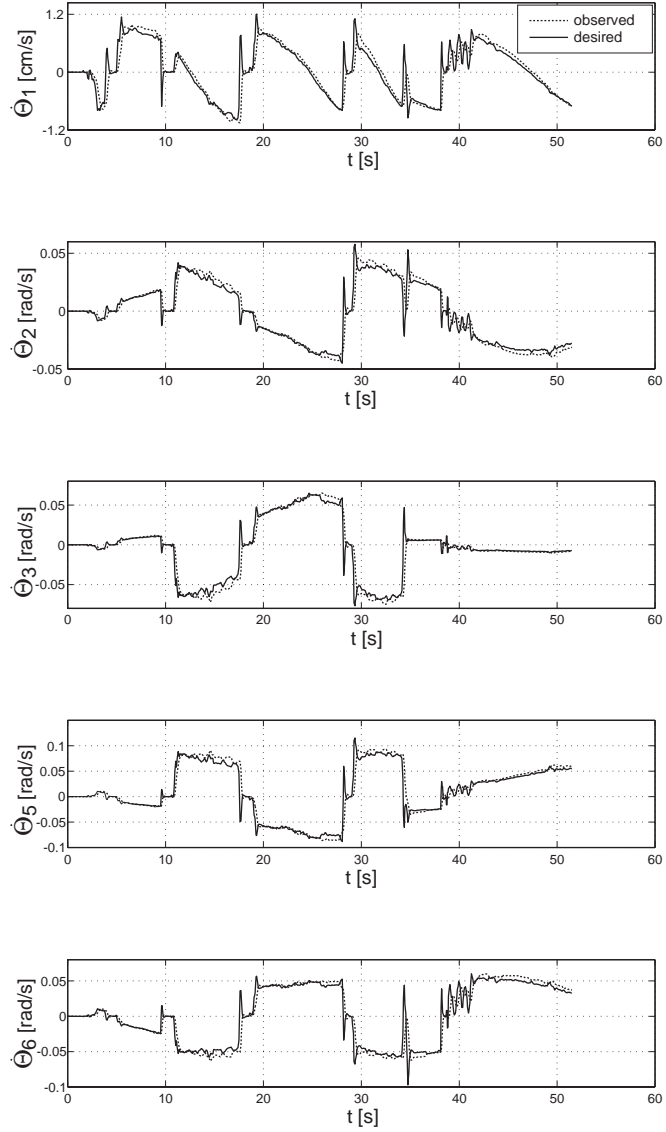Figure 3.5: Measured velocity of TCP.

37

Figure 3.6: Velocities $\dot{\Theta}_1 \,..\, \dot{\Theta}_6$, desired and observed.
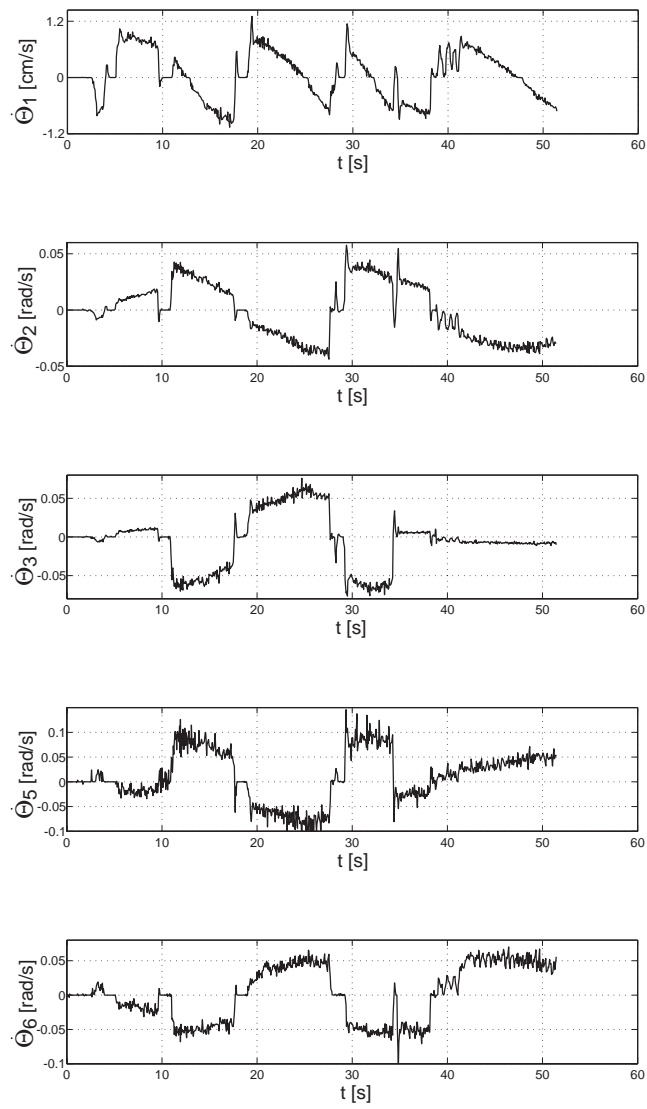
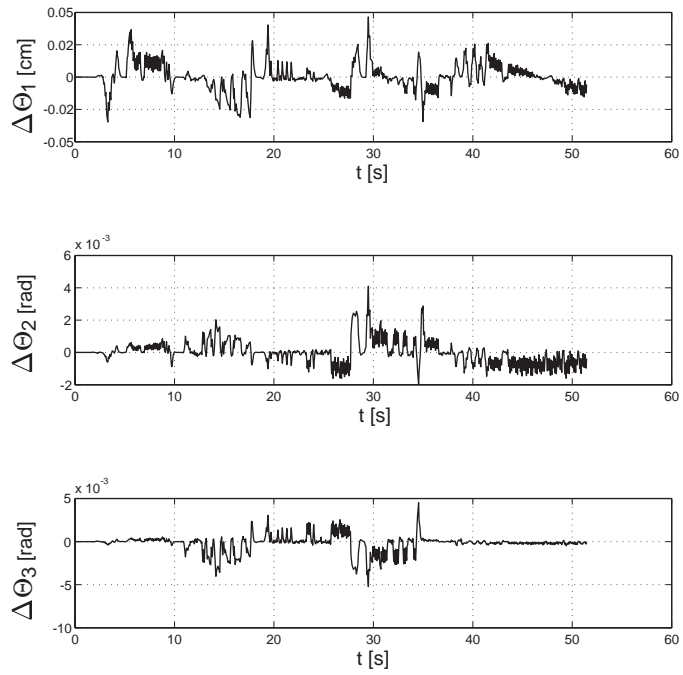Figure 3.7: Measured velocities $\dot{\Theta}_1 \; .. \; \dot{\Theta}_6$.

Figure 3.8: Errors $\Delta\Theta_{1..3}$.
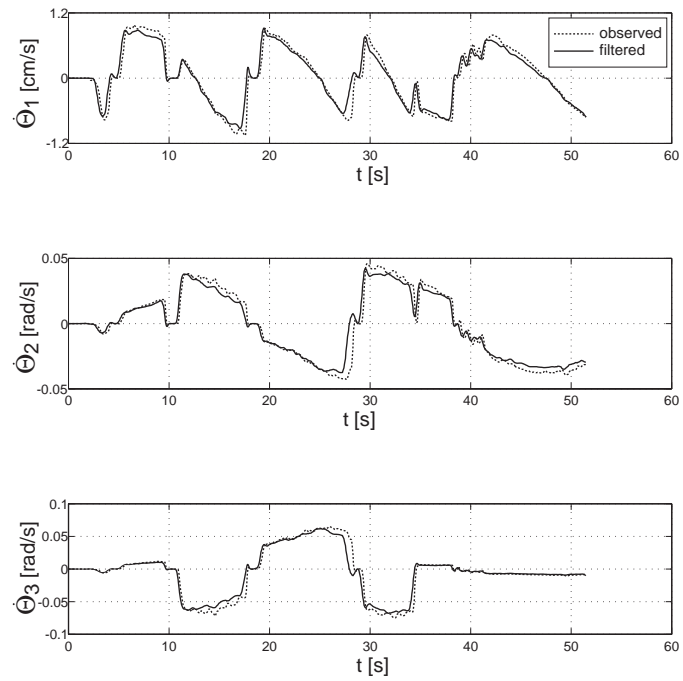

Figure 3.9: Offline filtered and observed $\dot{\Theta}_{1..3}$.

3.2.4 Summary

A control algorithm that allows Cartesian velocity control of the AESOP robot was given. Therefore, an estimation of the entry point position to calculate an approximation of the velocity of the passive joints is necessary. This leads to an algorithm to compute the inverse Jacobian matrix. Due to sensor noise and slow motion of the robot an observer for the joint velocities has to be built. Finally, a velocity control loop can be closed. The experiments given show good results in tracking the desired velocity.

3.3 Fundamentals of Position Control

This section addresses the fundamentals for Cartesian position control performed with the AESOP robot, before the entire position control loop is presented in Sec. 3.4. The main problem is that the joints of the robot have different time constants (see Tab. 3.1). This leads to configuration dependent dynamics and a coupling of the Cartesian degrees of freedom. A self-adjusting controller is presented which is able to deal with the nonlinear dynamics of the robot.

Dealing with nonlinear dynamics of robots various algorithms have been developed. The feedback linearization scheme for example decouples the plant's non-linear dynamics and enables the definition of the desired Cartesian dynamics. However, in common robot controller implementations decentralized cascaded joint controllers are widely used [68]. This section addresses the problem that the AESOP robot offers an interface which allows velocity control for each joint and that the time-responses of the individual joints differ from each other. If these differences are significant, the Cartesian time-responses of the robot become configuration dependent, which has to be considered during the controller design, in order to reach the desired specification. Instead of a design procedure which deals with the worst-case scenario (conservative design) a self-adjusting controller is presented which takes the time-varying behavior into account, in order to achieve a better bandwidth of the control loop. The changes of the Cartesian time responses are not very fast but a large range of values is covered, so quasi-stationary treatment is appropriate.

First the configuration dependent dynamics are analyzed, thereafter the self-adjusting controller is presented. The properties of the closed loop are discussed in in the following section. Experimental results with the chosen self-adjusting controller are shown and compared with the results of a conventionally designed controller. When introducing the self-adjusting controller, only a 2 DoF case is considered to keep complexity low. The entire position control scheme for the AESOP robot is presented and discussed in Sec. 3.4.

3.3.1 Cartesian Time-Response

According to Sec. 3.2.1 the dynamics of each joint Θ_k can be modeled as a PT_1 transfer-function with dead time T_D; see Eq. 3.51. This model is appropriate if the robot has a position- or velocity-interface for the joints and if the robot moves slowly enough to allow separate consideration of each joint. This means no or neglectable dynamic coupling:

$$F_k^* = \frac{1}{1 + T_k s} e^{-T_D s} \, . \tag{3.51}$$

If the dead times T_D are equal for all joints, they do not need to be considered while deriving the nonlinear behavior:

$$F_k = \frac{1}{1 + T_k s} \, . \tag{3.52}$$

The time constants T_k are assumed to be constant and not configuration dependent. If

$$T_k \neq T_l \quad \text{for} \quad k \neq l \, , \tag{3.53}$$

the Cartesian time-constants $T_{\dot{\boldsymbol{x}}}$ are configuration dependent:

$$T_{\dot{\boldsymbol{x}}} = T_{\dot{\boldsymbol{x}}}(\boldsymbol{\Theta}) \, , \tag{3.54}$$

as described below.

Nonlinear Dynamics The following block-scheme of a robot with a velocity interface as shown in Fig. 3.10 is considered. The variable $\dot{\boldsymbol{x}}_D$ denotes the desired Cartesian velocity vector, $\dot{\boldsymbol{\Theta}}_D$ the desired joint velocity vector and $\dot{\boldsymbol{x}}$ the actual velocity vector.
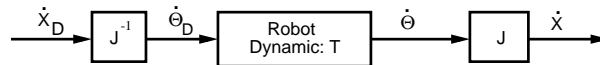

Figure 3.10: Block scheme of robot with velocity interface.

The open loop equation is:

$$\dot{\boldsymbol{x}} = \underbrace{\boldsymbol{J} \boldsymbol{T} \boldsymbol{J}^{-1}}_{\boldsymbol{D}} \dot{\boldsymbol{x}}_D \, , \tag{3.55}$$

with $\boldsymbol{J}$ being the Jacobian matrix of the robot and

$$\boldsymbol{T} = \begin{bmatrix} F_1 & 0 & \dots & 0 \\ 0 & F_2 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \dots & 0 & F_N \end{bmatrix} \tag{3.56}$$

42

describing the joint dynamics. Using the definitions:

$$\boldsymbol{J} = \begin{bmatrix} j_{11} & \cdots & j_{1N} \\ \vdots & \ddots & \vdots \\ j_{N1} & \cdots & j_{NN} \end{bmatrix} \tag{3.57}$$

$$\boldsymbol{J}^{-1} = \begin{bmatrix} g_{11} & \cdots & g_{1N} \\ \vdots & \ddots & \vdots \\ g_{N1} & \cdots & g_{NN} \end{bmatrix} = \boldsymbol{G} \tag{3.58}$$

the elements d_{ij} of $\boldsymbol{D}$ can be calculated:

$$d_{ij} = \sum_{k=1}^{N} F_k g_{ki} j_{jk} . \tag{3.59}$$

If off-diagonal elements of $\boldsymbol{D}$ are unequal to zero, i.e.

$$d_{ij} \neq 0 \qquad \text{for} \qquad i \neq j , \tag{3.60}$$

the Cartesian dynamics of the robot are coupled, which means that a desired motion in one Cartesian direction leads to an undesired motion in other Cartesian directions. This happens, if at least one T_k differs; see the following example for a step-input in x-direction:

$$\dot{\boldsymbol{x}} = \boldsymbol{D}\dot{\boldsymbol{x}}_D = \boldsymbol{D} \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} d_{11} \\ \vdots \\ d_{1N} \end{bmatrix} . \tag{3.61}$$

It can also be seen that the elements d_{ij} are dependent on the actual configuration $\boldsymbol{\Theta}$, because of $\boldsymbol{J} = \boldsymbol{J}(\boldsymbol{\Theta})$. This leads to the conclusion that the Cartesian time-response is also configuration dependent. For further calculations a simplified model is given next.

Simplified Robot Dynamics   The time-derivative of Eq. 3.59 is:

$$\dot{d}_{ij} = \sum_{k=1}^{N} \dot{a}_k g_{ki} j_{jk} + a_k \dot{g}_{ki} j_{jk} + a_k g_{ki} \dot{j}_{jk} , \tag{3.62}$$

with $a_k = 1 - e^{-t/T_k}$ being the time-domain representation of F_k. At $t = 0$ the following equation is satisfied:

$$\dot{d}_{ij}|_{t=0} = \dot{d}^*_{ij}|_{t=0} = \sum_{k=1}^{N} \frac{1}{T_k} g_{ki} j_{jk} = \frac{1}{T^*_{ij}} , \tag{3.63}$$
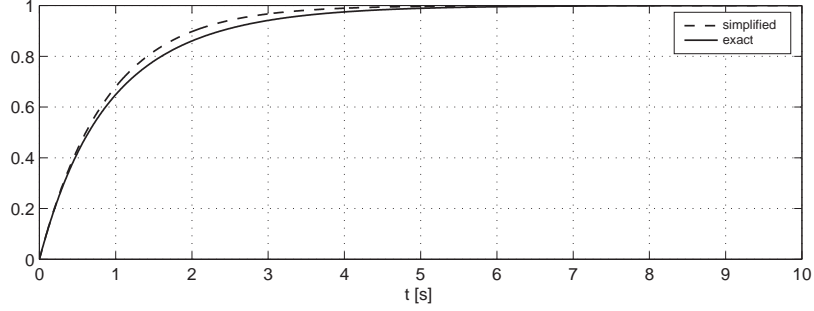
43

Figure 3.11: 2 DoF example, step response.

because $a_k(t = 0) = 0$. For $i = j$ the simplified model

$$d^*_{jj} = 1 - e^{-\frac{t}{T^*_{jj}}} \tag{3.64}$$

fits good enough for further calculations:
At $t = 0$

$$\dot{d}^*_{jj} = \dot{d}_{jj} \tag{3.65}$$

and for $t \to \infty$ (step response)

$$d^*_{jj} \to 1 \quad \text{and} \quad d_{jj} \to 1 \ , \tag{3.66}$$

because

$$\sum_{k=1}^{N} g_{kj} j_{jk} = 1 \qquad \text{and} \qquad \lim_{t \to \infty} F_k = 1 \ . \tag{3.67}$$

For $i \neq j$ T^*_{ij} provides a measure of the coupled dynamics.

Figure 3.11 shows a simulation of the simplified model for a 2 DoF case (d_{11}: solid line and d^*_{11}: dashed line). It can be seen, that a good match is achieved. The values of the time constants are chosen in order to visualize the accuracy of the simplified model:

$$d_{11} = 0.3(1 - e^{-t/0.54s}) + 0.7(1 - e^{-t/1.2s}) \tag{3.68}$$

$$d^*_{11} = 1 - e^{-t/0.878s} \ . \tag{3.69}$$

If this simplification is applied to the AESOP robot used in the MIRS scenario, the time constants are more similar to each other (see Tab. 3.1). Therefore, the differences between the exact model and the simplified model can be neglected.

3.3.2 Self-Adjusting Controller

In this section the chosen controller structure and adaption law, which provide an intuitive way to deal with the nonlinear behavior described in Sec. 3.3.1, is presented. The control loop for the one degree of freedom case, shown in Fig. 3.12, is considered.
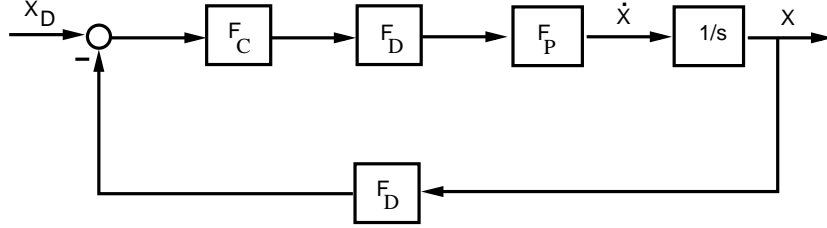

Figure 3.12: Closed control loop.

The controller transfer function is written as

$$F_C = K \frac{1 + T_1 s}{1 + T_2 s} \; , \tag{3.70}$$

the plant transfer function is:

$$F_P = \frac{1}{1 + T_{\dot{x}} s} \quad \text{with} \quad T_{\dot{x}} = T_{jj}^*(\boldsymbol{\Theta}) \; , \tag{3.71}$$

and the communication delay is

$$F_D = e^{\frac{-T_D}{2} s} \; . \tag{3.72}$$

If the delay-times T_D of the joints, e.g. mechanical or signal-processing delay, are significant and cannot be neglected they can be included in Eq. 3.72. Since the Bode plot is used to calculate the parameters of F_C the open loop equations have to be considered:

$$F_{\text{openloop}} = K \frac{1 + T_1 s}{1 + T_2 s} \frac{1}{1 + T_{\dot{x}} s} \frac{1}{s} e^{-T_D s} \; . \tag{3.73}$$

The gain can be computed as follows:

$$A = K \frac{\sqrt{1 + (w T_1)^2}}{\sqrt{1 + (w T_2)^2}} \frac{1}{\sqrt{1 + (w T_{\dot{x}})^2}} \frac{1}{w} = K A^* \; . \tag{3.74}$$

The phase is:

$$\phi = \text{atan2}(w T_1, 1) + \text{atan2}(-w T_2, 1) + \text{atan2}(-w T_{\dot{x}}, 1) - \frac{\pi}{2} - w T_D \; . \tag{3.75}$$

Considering the structure of the controller F_C three parameters have to be determined: K, T_1, and T_2. First $T_2 = kT_s$ is chosen, with T_s being the sample time of the digital implementation, to move the negative part of the phase of $\frac{1}{1+T_2 s}$ as far as possible towards high w. A good value proofed to be $k = 4$ or $k = 5$. One way to compute T_1 is to choose $T_1 = T_{\dot{x}}$ to compensate the pole of the plant; this works well for $T_D \ll T_{\dot{x}}$ only. An adaption law able to handle the general case is presented: For a small w_{g1} a desired phase margin ϕ_{R1} is chosen that provides good damping e.g. $\phi_{R1} = 100°$. The necessary phase shift $\Delta\phi_1$ that has to be provided by the controller F_C at w_{g1} can be computed as follows [48]:

$$\Delta\phi_1 = -\pi + \frac{\phi_{R1}}{180°}\pi + \frac{\pi}{2} - (-w_{g1}T_D + \text{atan2}(-w_{g1}T_{\dot{x}}, 1)) . \tag{3.76}$$

The phase of F_C at w_{g1} is:

$$\angle F_C = \angle \frac{1 + jw_{g1}T_1}{1 + jw_{g1}T_2} . \tag{3.77}$$

Solving Eq. 3.77 for T_1 leads to:

$$T_1 = \frac{\tan(\Delta\phi_1) + T_2 w_{g1}}{w_{g1}(1 - T_2 w_{g1}\tan(\Delta\phi_1))} . \tag{3.78}$$

The last parameter to be calculated is K. A phase-margin ϕ_{R2} for F_{openloop} at the gain crossover-frequency w_{g2} is chosen and Eq. 3.79 is solved for $w = w_{g2}$:

$$\begin{aligned} \Delta\phi_2 &= 0 \\ &= -\pi + \frac{\phi_{R2}}{180°}\pi + \frac{\pi}{2} - (-w_{g2}T_D + \text{atan2}(-w_{g2}T_{\dot{x}}, 1) + \\ &\quad + \text{atan2}(w_{g2}(T_1 - T_2), 1 + T_1 T_2 w_{g2}^2)) . \end{aligned} \tag{3.79}$$

Finally, with Eq. 3.74:

$$K = \frac{1}{A^*(w_{g2})} , \tag{3.80}$$

because w_{g2} is the gain crossover-frequency.

According to Eq. 3.78, $T_1 \to \infty$ only if

$$\Delta\phi_1 \to \arctan\left(\frac{1}{T_2 w_{g1}}\right) . \tag{3.81}$$

In this case the desired phase shift cannot be provided by the chosen controller. It has to be guaranteed that for all possible $\boldsymbol{\Theta}$ this case will never occur. According to Eq. 3.80 $K \to \infty$ only if $w_{g2} \to \infty$. For the control law chosen here this is impossible to satisfy. As the T_{jj} differ by a large range, but change slowly, the system can be considered as quasi-linear and the control law is stable.

The algorithm presented here provides a conservative trend of the phase margin as illustrated in the example shown in Fig. 3.13, with $T_D = 0.04$ s, $T_{\dot{x}} = 0.5$ s, $w_{g1} = 1$ rad/s, $T_1 = 1.1988$ s, $K = 1.365$, $\phi_{R1} = 100°$, $T_2 = 5T_s = 0.2$ s, $\phi_{R2} = 80°$.
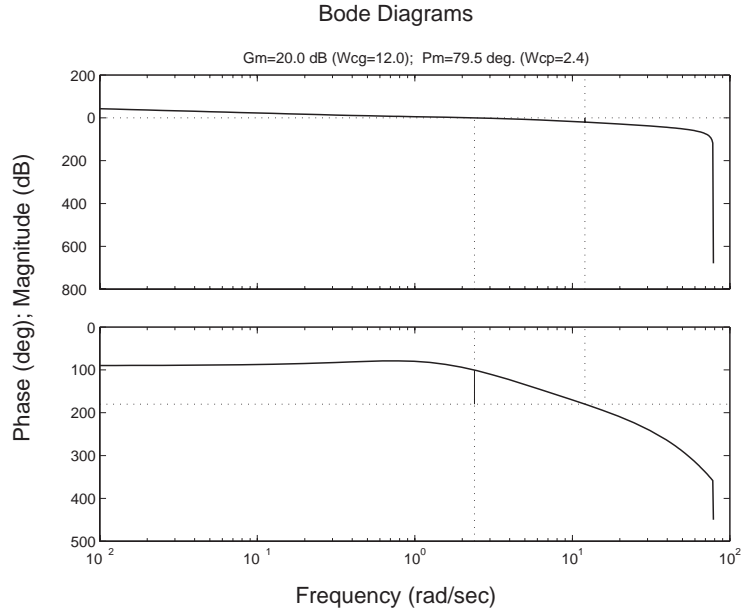
Figure 3.13: Bode plot.

3.3.3 Experimental Results

In this section the results of an experiment with a 2 DoF Scara like robot with the following characteristics, are presented:

- time constant of joint one: $T_1 = 1.2$ s

- time constant of joint two: $T_2 = 0.54$ s

- length of segment one and two: $l_1 = l_2 = 38.4$ cm

During the experiment the following values for the setup were chosen:

- delay time: $T_D = 0.04$ s

- sampling time: $T_s = 0.04$ s

- frequency $w_{g1} = 1$ rad/s

- phase-margin $\phi_{R1} = 100°$

- phase-margin $\phi_{R2} = 80°$

Two different controllers are analyzed: The first one is according to the self-adjusting controller described in Sec. 3.3.2 (case 1), the second considers the worst case scenario, without any adjustment during the experiment (case 2).

In Fig. 3.14 the ramp input for x_d (dashed dotted line) and the constant input for y_d (also dashed dotted line), and the robot trajectories for case 1 (solid line) and case 2 (dashed line) are shown. It can be seen that for both cases an error
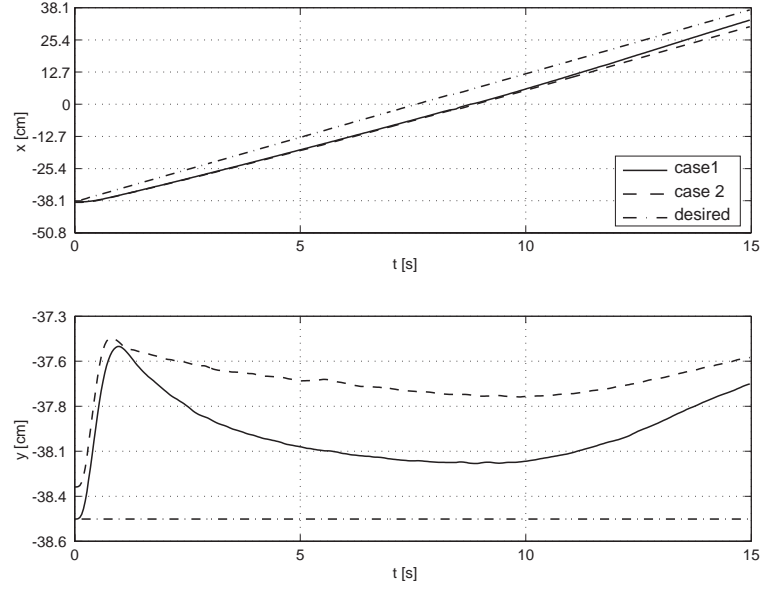

Figure 3.14: The x- and y-trajectories.

occurs. The error in y-direction is a result of the nonlinear coupling effect, which is smaller in case one compared to case two. The increasing error after about $t = 10$ s is a result of the changing Cartesian time constants (see Eq. 3.63 with $i = j$) shown in Fig. 3.15.

To compare the performance, the sum of absolute errors, according to Eq. 3.82 is computed:

$$e_x = \frac{\sum_{k=1}^{N} |x_{d_k} - x_k|}{N} \quad \text{and} \quad e_y = \frac{\sum_{k=1}^{N} |y_{d_k} - y_k|}{N} \ . \tag{3.82}$$

For case 1:

$$e_x = 4.7 \text{ cm} \quad \text{and} \quad e_y = 0.5 \text{ cm} \ .$$

For case 2:

$$e_x = 5.4 \text{ cm} \quad \text{and} \quad e_y = 0.8 \text{ cm} \ .$$

The improvement by the self-adjusting controller compared to the conservative design is significant:

$$\Delta_x = 14.0 \ \% \quad \text{and} \quad \Delta_y = 61.4 \ \% \ .$$

The improvement enlarges if the workspace enlarges, because the difference between the conservative controller design and the self- adjusting controller design becomes much more significant.

Figure 3.15 shows the online computation of the Cartesian time constant in x- and y-directions (solid line) and the values chosen for case two (dashed line). Figure 3.16 shows the computed parameters T_1 and K for the controller in x-
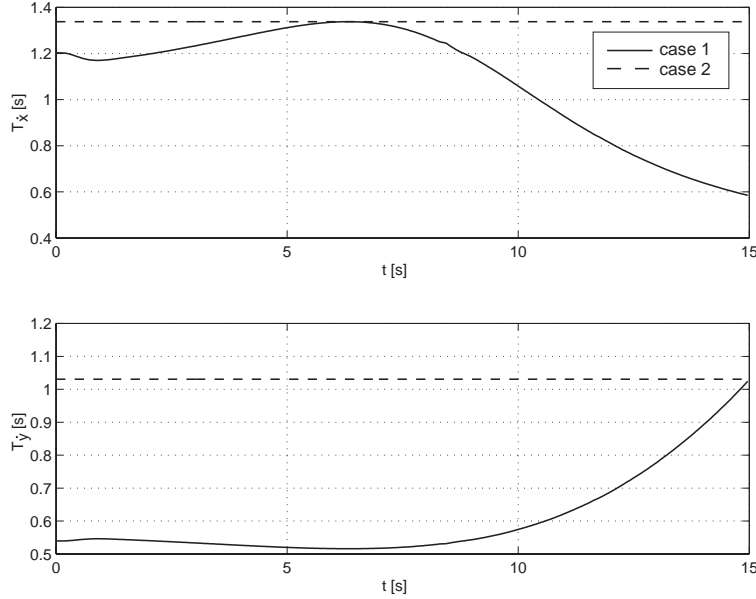

Figure 3.15: Cartesian time constants.

direction (normal solid line) and y-direction (bold solid line), as well as the values for case two (normal dashed line and bold dashed line, respectively).

3.3.4   Summary

In it has been shown that joints with different time constants lead to configuration dependent Cartesian transfer functions. Furthermore, a dynamic coupling between the Cartesian degrees of freedom occurs. A self-adjusting control law that is able to deal with the configuration dependent dynamics of robots was presented. (Only the diagonal elements of $\boldsymbol{D}$ are considered, as they are the main reason for the undesired nonlinear behavior. The off-diagonal elements of $\boldsymbol{D}$ which are responsible for the coupling of the Cartesian DoFs are neglected.) The proposed control law allows an intuitive way to tune the behavior of the closed control loop. Experiments show the advantages of the self-adjusting control law. Section 3.4 introduces the position control law for the robot used in the DLR MIRS scenario.
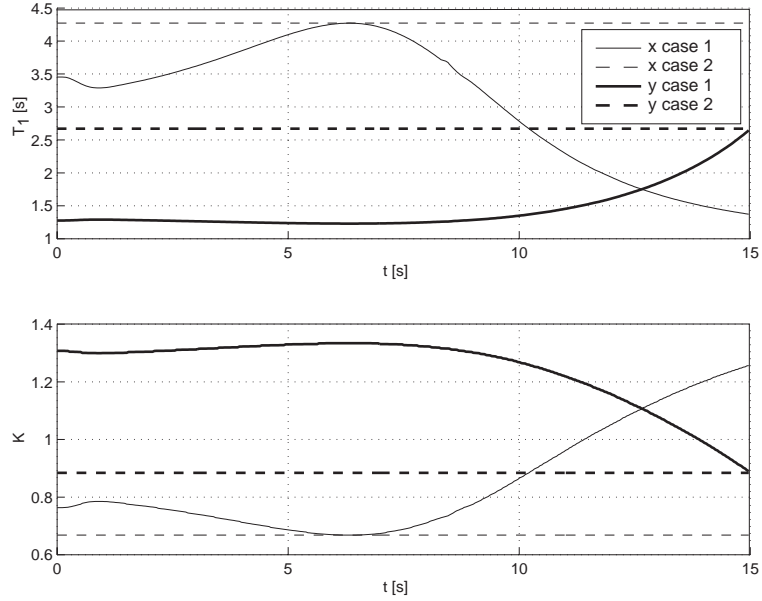
Figure 3.16: Computed controller parameters.

3.4   Position Control

Cartesian position commands are the most common way to command a telesurgery system. Therefore, a Cartesian position control law is needed. It allows correct hand-eye coordination and therefore increases the dexterity of the entire system. In addition to appropriate scaling of the input trajectories, manipulation of small structures with high accuracy is possible. First the position control law is described in detail, after that experimental results are given. A brief summary concludes the position control section.

3.4.1   Position Control Loop

The position control law is sketched in Fig. 3.17. As with the velocity control law the entry point position estimation is used to calculate the inverse kinematics P^{-1}. Furthermore, the inverse kinematics is used to calculate the simplified dynamics (Sec. 3.3.1) which is a prerequisite for the self-adjusting controller (Sec. 3.3.2). The controller is tuned for the x-, y-, and z-directions separately, the transfer functions are $F_{\mathrm{C,x}}$, $F_{\mathrm{C,y}}$, and $F_{\mathrm{C,z}}$, respectively. They are chosen according to Sec. 3.3.2. As before, the subscript D of the variables denotes desired values, the subscript p passive joints, and the subscript a active joints. The variable t is the entry point and F_D represents the communication delay transfer function with $F_D = e^{-\frac{T_D}{2}s}$.
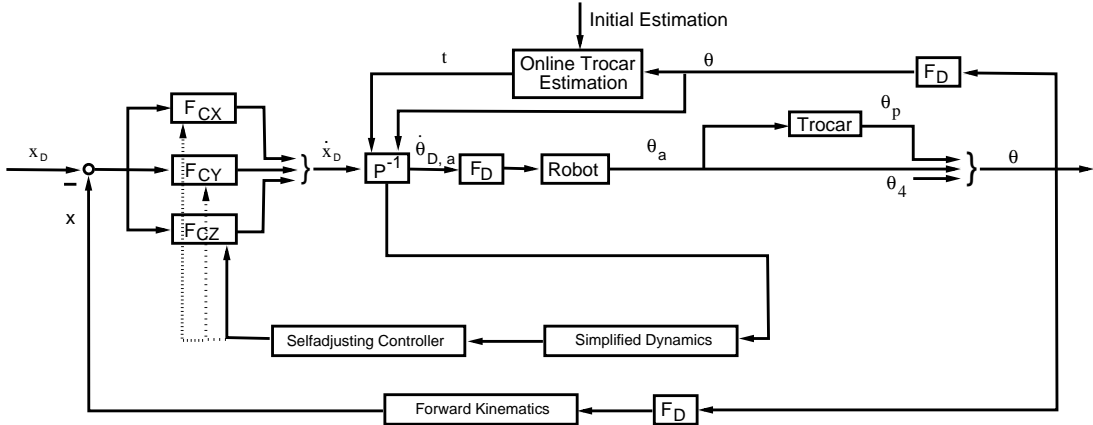
50

Figure 3.17: Position control scheme.

3.4.2 Experiments

A Cartesian input device (PHANToM) was coupled with the robot via the computer network of the institute and was used for online-generating a desired input trajectory. For the controller design the following parameters have been chosen and applied to each Cartesian degree of freedom separately:

- sampling time: $T_s = 22$ ms

- delay time: $T_D = 22$ ms

- phase-margin: $\phi_{R1} = 80°$

- frequency: $w_{g1} = 1$ rad/s

- phase-margin: $\phi_{R2} = 60°$

- controller parameter: $T_2 = 5T_s$

The input trajectory as well as the robot position are given in Fig. 3.18: A phase shift between the input trajectory and the robot position occurs. This is due to the very conservative controller design (huge damping) as well as the communication delay between controller and robot ($T_D = 22$ ms).

The approximation of the Cartesian time-response parameter in x-, y-, and z-direction as well as the computed controller parameters are shown in Fig. 3.19. The parameter adaption, corresponding to the current configuration, can be seen well. The adaption of T_1 and K is less evident than in Sec. 3.3.3. This is due to the minor differences of the parameters of the joint transfer functions (Tab. 3.1). The parameters change slowly enough to allow quasi-stationary consideration.

The adaption guarantees a well damped behavior and good dynamics over the entire work space.
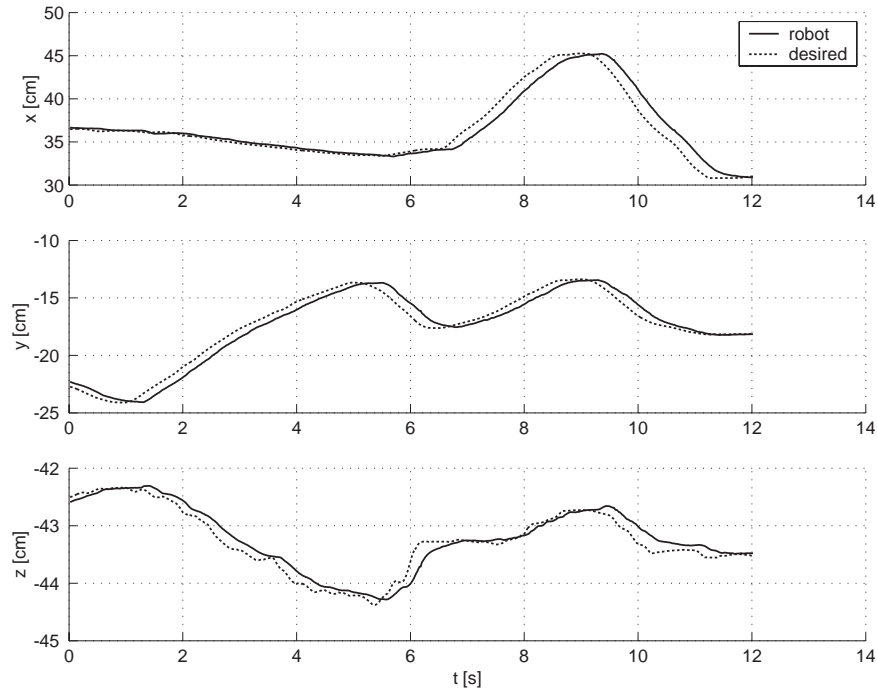

Figure 3.18: Position control experiment.

3.4.3 Summary

Cartesian position control guarantees correct hand-eye coordination The surgeon's movements become similar to his movements in open surgery. The position control scheme proposed here provides good results. The control law is stable and a well damped behavior is guaranteed. In all configurations of the robot the self-adjusting controller provides sufficiently good behavior (according to the design parameters). The tracking error (deviation between desired position and position of the tool-tip of the robot) occurs due to the conservative controller design and the communication delay (between controller and robot). As a drawback, position control can lead to unintentional damage of tissue, because manipulation forces are not supervised. This problem is addressed in Sec. 3.5.
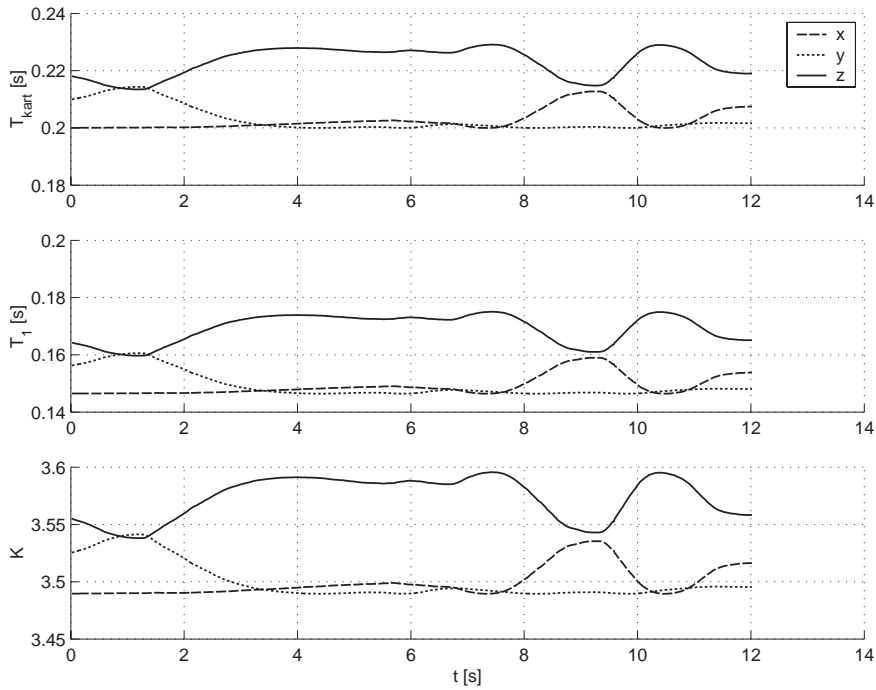
Figure 3.19: Computed controller parameters.

3.5 Force Control

Force control is an inportant control strategy for minimally invasive surgery robots. As the force at the tool-tip of the instrument is observed, force control helps to avoid unintentional damage of tissue. Therefore, the safety of telesurgery systems is dramatically increased. In beating heart surgery a force controlled robot having contact with the heart surface is able to exert a constant force to the organ and thus to compensate the heart motion. Another application of force control is the preparation of the mamaria artery, where a constant force has to be applied to the artery.

The performance of a force control loop strongly depends on the stiffness of the environment. As tissue stiffness is not known a priori and may even vary between patients or during an operation, an online stiffness estimator is used to adapt the controller parameters. The force control law is presented first. Thereafter experimental results are given. The subsequent summary concludes this section.

3.5.1 Force Control Loop

The force control scheme is depicted in Fig. 3.20. It is similar to the position control scheme presented in Sec. 3.4.1: The subscript D of the variables denotes

desired values, the subscript p passive joints, and the subscript a active joints. The variable t is the entry point, F_D represents the communication delay transfer function with $F_D = e^{-\frac{T_D}{2}s}$, and x_{env} is the position of the environment. The sample time $T_s = 1.7$ ms of the sensorized scalpel which is used to measure the contact forces is not shown in Fig. 3.20, because it is far less than the communication delay (between the joint controllers and the software) $T_D = 22$ ms and can be neglected.

As the performance of a force control law strongly depends on the environment stiffness, a stiffness estimator is used to tune the controller parameters. The stiffness in x-direction is calculated as:

$$K_{\text{x,env}} = \frac{\Delta F_x}{\Delta x} \ . \tag{3.86}$$

In y- and z-direction, $K_{\text{y,env}}$ and $K_{\text{z,env}}$ are calculated accordingly. The environment stiffness is used to adapt the controller parameters in x-, y-, and z-direction:

$$H_{\text{C,x}} = \frac{1}{K_{\text{x,env}}}F_{\text{C,x}} \ , \quad H_{\text{C,y}} = \frac{1}{K_{\text{y,env}}}F_{\text{C,y}} \ , \quad \text{and} \quad H_{\text{C,z}} = \frac{1}{K_{\text{z,env}}}F_{\text{C,z}} \ . \tag{3.87}$$

The transfer functions $F_{\text{C,x}}$, $F_{\text{C,y}}$, and $F_{\text{C,z}}$ are chosen according to Sec. 3.3.2. The parameter adaption ensures that the specifications for the closed loop behavior (Sec. 3.3.2) are met, independent of the environment stiffness. As motion in MIS is slow, a more detailed identification of the environment (e.g. damping) is not necessary (see also the experimental results in Sec. 3.5.2).
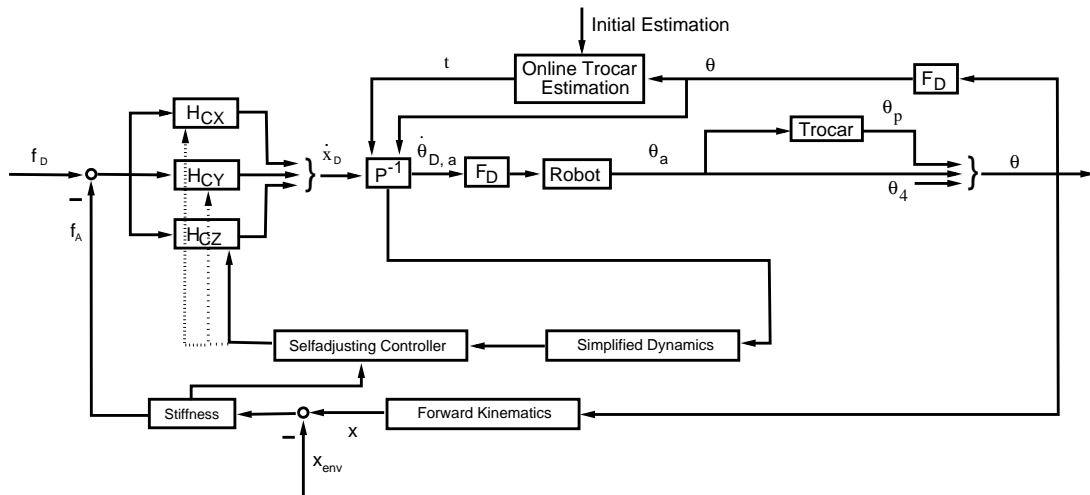

Figure 3.20: Force control scheme.

3.5.2 Experiments

The sensorized scalpel (see Sec. 2.3.1) was attached to the AESOP robot. The measured contact forces and the robot position were used to calculate an estimation of the environment stiffness which is used to tune the controller parameters (see Sec. 3.5.1). The robot was coupled with a Cartesian input device (PHANToM) via the computer network of the institute. The PHANToM has no force sensor attached to the stylus, so the desired force trajectory cannot be measured directly. The static situation can be considered as an equilibrium between the force $\boldsymbol{f}_{\mathrm{PH}}$ commanded at the PHANToM (the force reflected to the user) and the force $\boldsymbol{f}_{\mathrm{A}}$ measured at the remote side. Even slow motion can be interpreted as quasi-static. The position offset between master and slave is scaled with $\boldsymbol{K}_P$ and used to increase or decrease the desired force $\boldsymbol{f}_d$. Therefore, the desired force is calculated as follows:

$$\boldsymbol{f}_{\mathrm{PH}} = \boldsymbol{f}_d = \boldsymbol{f}_{\mathrm{desired}} = \boldsymbol{K}_P(\boldsymbol{x}_{\mathrm{A}} - \boldsymbol{x}_{\mathrm{P}}) + \boldsymbol{f}_A \qquad (3.88)$$

with

$$\boldsymbol{K}_P = \mathrm{diag}(30, 30, 30) \ , \qquad (3.89)$$

and $\boldsymbol{x}_{\mathrm{A}}$ denoting the position of the AESOP robot and $\boldsymbol{x}_{\mathrm{P}}$ denoting the position of the PHANToM. A more detailed description of Eq. 3.88 including some master-slave experiments is given in [61].

Figure 3.21 shows experimental results. The soft organs of the patient were simulated by a piece of rubber foam. The results are given in one direction (z-axis) only. The other directions provide similar results. The upper part of the figure shows that the proposed control law provides good tracking of the desired force trajectory. The major error occurs between $t \in [12s, 15s]$. Here, the remote forces are very small and are therefore set to zero. This is necessary, because low forces cannot be distinguished from sensor noise. The middle part of Fig. 3.21 gives the calculated estimation of the environment stiffness $K_{\mathrm{z,env}}$ as well as the computed controller parameter K_z. A lower limit for the environment stiffness $K_{\mathrm{z,env,min}} = 0.4$ N/cm is provided to guarantee reasonable values for $K_{\mathrm{z,env}}$ in case of measurement errors. If enough measurement values to allow robust determination of $K_{\mathrm{z,env}}$ are available the controller parameters are tuned. The lower part of the figure shows the tracking error of the force trajectory. The error remains remarkable small during the experiment.

Figure 3.22 gives experimental results if no adaption law is implemented. The upper part of the figure shows the tracking of a force trajectory if the environment stiffness $K_{\mathrm{z,env}}$ is chosen too large: $K_{\mathrm{z,env}} = 3.9$ N/cm. The gain of the controller is therefore low and a large tracking error occurs. If $K_{\mathrm{z,env}}$ is too low (here $K_{\mathrm{z,env}} = 0.2$ N/cm) the force controller becomes instable (lower part of Fig. 3.22), as the gain of $H_{\mathrm{C,z}}$ is too large.
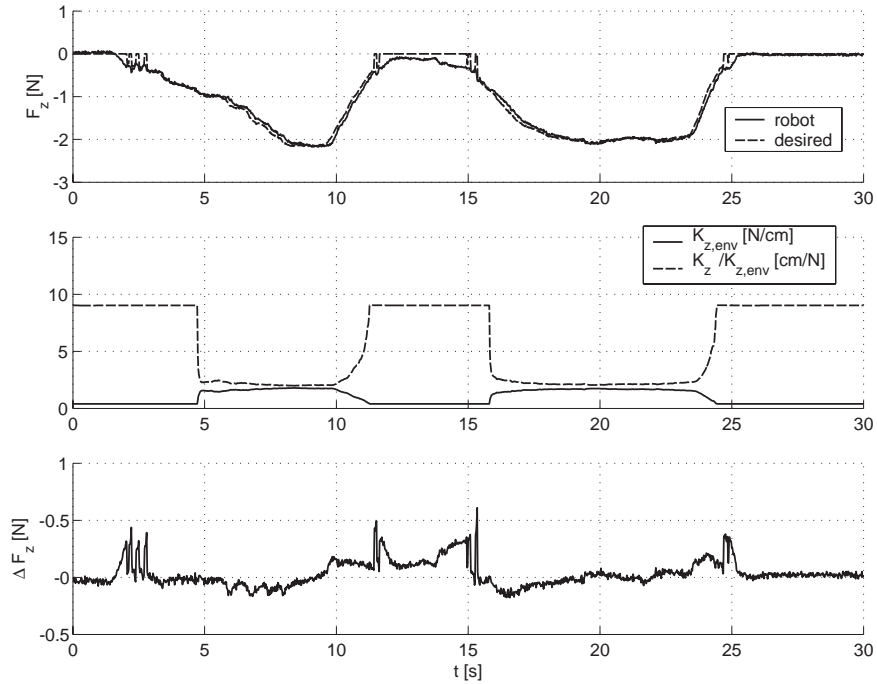
Figure 3.21: Stiffness estimation.

3.5.3   Summary

In this section a force control law which takes into account the actual stiffness of the remote environment was presented. This makes an online estimation of the environment stiffness necessary. In combination with the self-adjusting controller this leads to good results.

Even more, without the stiffness estimation, the performance of the closed loop degrades. This is shown in the experiments section: A too large estimation of the remote stiffness leads to a large tracking error, whereas a too small estimation leads to an unstable control loop.

Further experiments in a medical master-slave scenario including force-feedback are given in [61]. In this publication also different types of coupling (e.g. position – position or force – force) of master and slave are discussed.

3.6   Conclusions

The focus of this chapter lies on Cartesian control laws, allowing unrestricted Cartesian motion of surgical instruments. This leads to the reestablishment of the correct hand-eye coordination and makes MIS more similar to open surgery.
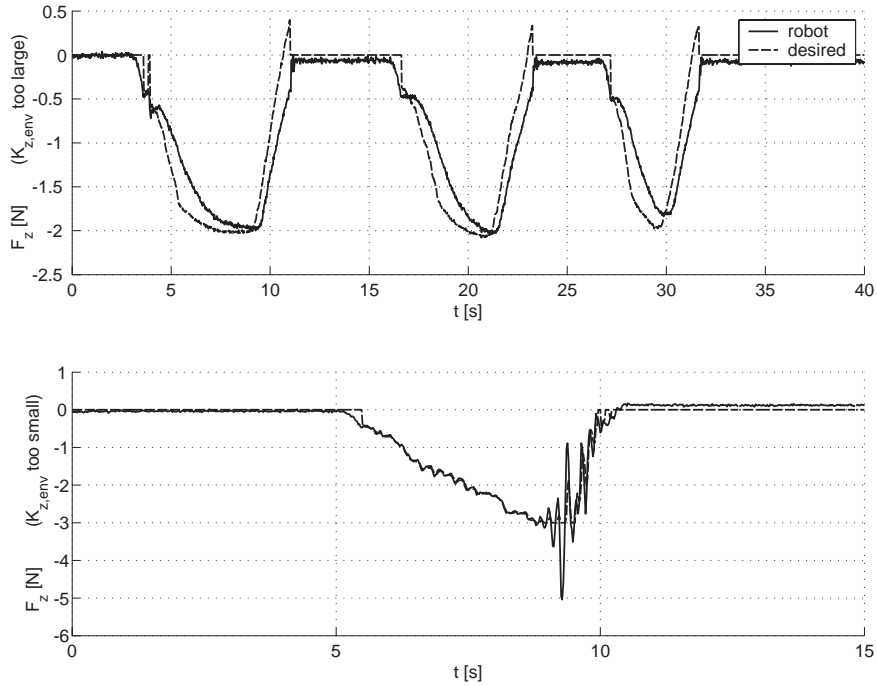
Figure 3.22: No stiffness estimation.

A prerequisite for Cartesian control laws in MIRS is the knowledge of the entry point position. If this position is known the inverse kinematics of the robot can be calculated.

Section 3.2 presents a velocity control loop. As the motion of the robot is slow and the communication delay is large ($T_D = 22$ ms) the measured joint velocities $\dot{\Theta}$ are very noisy and an observer is built to close the control loop. The experiments presented show the good performance of the controller, indicated by small tracking errors.

The joints of the robot have different time constants, which leads to configuration dependent dynamics of the Cartesian degrees of freedom of the robot. A self-adjusting controller is used to circumvent this undesired behavior and to close the position control loop as described in Sec. 3.4.

Force control helps to avoid unintentional tissue damage and is a first step towards motion compensation in MIRS (Sec. 3.5). The force control law presented is based on a stiffness estimation enhancing the performance of the closed loop as it allows an adaption of the controller parameters.

The control laws presented here can be merged to mixed velocity/force or position/force control, depending on the contact situation. An evaluation by surgeons is necessary to identify the optimal control laws for different situations during

surgery. It is expected that the variety of the proposed control laws helps to adapt the robotic surgery system best to the operation. This will increase the quality of the therapy itself and enhance the ergonomics of the entire robotic surgery system. Furthermore, if the robot is able to compensate organ motion (e.g. with the methods proposed in Chapter 4 and Chapter 5) and allows the surgeon to work on a virtually stabilized organ completely new high quality operation techniques will be possible.

Chapter 4

Tracking in MIRCS

Heart surgery currently evolves from open surgery to minimally invasive robotic surgery (MIRS), as described in Sec. 1.3. First steps towards MIRS at the beating heart are made to avoid the drawbacks of a heart-lung-machine. To reduce the motion of the beating heart mechanical stabilizers (e.g. Octopus from Medtronic) are widely used. However, the remaining motion (about $1.5 - 2$ mm [37]) inside the stabilized area makes a safe operation difficult and time-consuming [19, 41].

For these reasons, compensation of the remaining motion is highly recommended. In robotic surgery this can be performed by the robot arm itself. Ideally, the relative pose between heart surface and tool center point of the robot remains constant: the heart is virtually stabilized. This leads to the problem of measuring the remaining motion of the heart surface, either by additional equipment (e.g. accelerator sensors on the heart surface as proposed by Computer Motion [78] or small color markers placed near the anastomosis) or by surgical equipment already used during surgery. An appropriate sensor seems to be the stereo-laparoscope, a rigid endoscope that is widely used in MIRS. Using the laparoscope as the sensor and exploiting natural landmarks has some important advantages: The already limited workspace is not restricted further by additional sensors, no contact between sensor and heart surface is necessary, and the motion of the entire area of interest can be captured. The main disadvantage of the method proposed here, is that vision algorithms can be disturbed easily. This problem is handled by the framework presented in Chap. 5.

The vision algorithms presented here are applied to natural landmarks inside the stabilized area, see Fig. 4.1. Of course, parts of the mechanical stabilizer are prominent landmarks (e.g. edges or additional color markers on the stabilizer) too which can be tracked very well, but to achieve high accuracy tracking on the heart surface itself is inevitable.

The distortion of the heart surface in the image-plane is strongly nonlinear. The tracking model used and its degrees of freedom are presented in Sec. 4.2. Due to

the wet heart surface specularities occur very frequently and disturb the tracking algorithm. Strategies to detect these specularities and to eliminate them are given in Sec. 4.3 and Sec. 4.4. The tracking results are evaluated in Sec. 4.5 and the computed trajectories are discussed in the subsequent section. Section 4.6 presents a confidence measure, evaluating the reliability of landmarks with respect to the tracking framework. 3D investigations to gain depth information are discussed in Sec. 4.7.
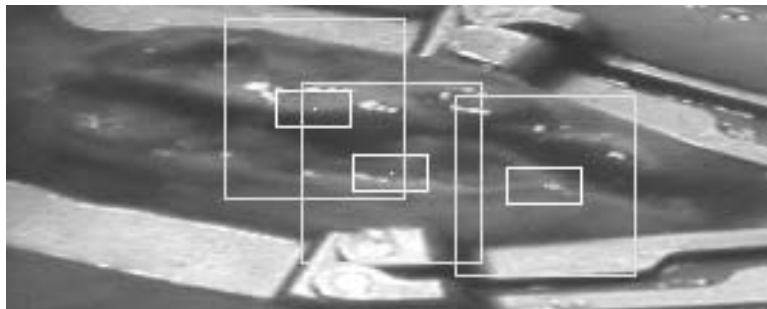

Figure 4.1: Stabilized heart with tracking areas.

4.1 Related Work

Tracking is very important in robotic surgery to capture the motion of the aiming points to update preoperative planning data. During the last decade it has been investigated by many research groups dealing with bonesurgery (e.g. spinesurgery [82] and hipsurgery [17]) or neurosurgery [18]. Often implanted fiducial markers or stereotactic frames [84] have been used to track the motion. New tracking techniques try to exploit natural anatomical landmarks to avoid artificial markers [6, 9]. However, these techniques can not be applied easily to deformable soft tissue as it appears in cardiac surgery.

Recent works in the context of soft tissue deal with motion detection and compensation in X-Ray cancer treatment to increase accuracy [67, 69]. Here, motion is induced by the patient's respiration. Although the remaining motion is identified as one of the most important drawbacks of minimally invasive beating heart surgery [19], only few research groups which work in this field are known: Motion compensation in beating heart surgery is currently investigated by another group located in Switzerland [13]. The company VISTA Development has published on its web-sites a short description of their beating heart motion tracking system which exploits artificial landmarks by the stereo laparoscopic cameras [76]. Unfortunately, no quantitave nor qualitative results can be found.

4.2 Tracking Model

According to [40] two different tracking approaches can be distinguished: tracking based on image features and tracking of complete regions or patterns in an image.

Feature-based tracking requires the extraction of image features, allowing fast subsequent matching due to the reduced amount of information. Region-based tracking saves the cost of feature extraction but finding the best matching pattern in the subsequent image requires a lot of computation time. Direct operation on image intensities makes special care for preprocessing tasks (e.g. illumination changes, occlusions etc.) necessary.

A region-based approach seems good for tracking landmarks on the heart surface, since the extraction of reliable features in realtime (e.g. edges) is hardly possible due to the nonlinear deformation of the heart surface.

4.2.1 Affine Motion Model

Despite the fact that the distortion of the tissue is nonlinear, an affine motion model as described in [26] is applied. This is possible if the pattern size is small enough to allow linear approximation of the nonlinear deformation. An affine motion model can be defined by the following equation:

$$\boldsymbol{v}' = \boldsymbol{T}(\boldsymbol{v}) = \boldsymbol{A} \cdot \boldsymbol{v} + \boldsymbol{t} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} , \qquad (4.1)$$

where $\boldsymbol{t}$ is the translation vector, $\boldsymbol{A}$ is called warping matrix and $\boldsymbol{v}$ is the position of the pixel to be transformed. The affine transformation has six DoFs, two for translation $\boldsymbol{t} = [t_x, t_y]^T$ and four warping parameters a_{ij}. They can be decomposed as follows:

$$\boldsymbol{A} = s \begin{bmatrix} \cos(\phi) & -\sin(\phi) \\ \sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \tau \end{bmatrix} \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} ,$$

where s is the scaling parameter, ϕ the rotation angle, and τ and α are the shear parameters value and direction. The parameter vector $\boldsymbol{\mu}$ can be written as:

$$\boldsymbol{\mu} = [t_x, t_y, s, \phi, \tau, \alpha]^T . \qquad (4.3)$$

If a reference pattern r is to be tracked over an image sequence, the task is to compute the parameters $\boldsymbol{\mu}_{\text{opt}}$ of the transformation $\boldsymbol{T} = \boldsymbol{T}(\boldsymbol{v})$ that map the current pattern p best to the original pattern r. Therefore, all pixels $\boldsymbol{v}$ of the actual and distorted pattern p are projected to the undistorted pattern p'. Using the sum of squared differences (SSD) as a similarity measure, this landmark is

then compared with the reference landmark r:

$$J(r, p') = \sqrt{\frac{1}{|\text{dom}(r)|} \sum_{i \in \text{dom}(r)} (r(i) - p'(i))^2} \ , \tag{4.4}$$

with dom(r) denoting the domain of pattern r and $|\text{dom}(r)|$ denoting the cardinality of dom(r). The optimal parameter vector $\boldsymbol{\mu}_{\text{opt}}$ is calculated as:

$$\boldsymbol{\mu}_{\text{opt}} = \arg \min_{\boldsymbol{\mu} \in \text{M}_\mu} \text{J}(\text{r}, \text{p}') \tag{4.5}$$

where M_μ is the domain of $\boldsymbol{\mu}$.

4.2.2   Further Aspects

Tracking on Halfimages   Tracking is performed on halfimages only, to avoid motion artifacts which arise from the interlaced nature of the camera images.

Search Strategy   A global search strategy taking every (discretized) $\boldsymbol{\mu} \in M_\mu$ into account is applied. This has the drawback that the computational effort increases exponentially with the DoFs of the search space and is suitable for realtime applications only if the dimension for the search space can be limited (Sec. 4.5.6). The main advantage of a global search strategy compared to a gradient based, as e.g. proposed by Hager [26], is that tracking is more robust with respect to local minima.

Spatial interpolation   Since the transformed coordinates usually do not lie on the discrete image grid of the CCD chip of the camera, bilinear interpolation as presented in [22] is used for mapping. Bilinear interpolation involves more computing time than simple nearest neighbor interpolation, but is a reasonable compromise between computational effort and precision.

Illumination Compensation   Illumination compensation is a prerequisite to deal with changes of the pattern intensity. Hager combines the affine motion model with an illumination model in one search space [26]. The current illumination is then calculated by linear combination of a set of basis images, recorded under different lighting conditions in advance [5]. In MIRCS the light source moves only seldomly and the environment is well defined, so a simple illumination model can be applied [15]:

$$f_{\text{comp}} = \alpha f + \beta \tag{4.6}$$

for an image f, and $\alpha, \beta \in \mathbb{R}$.

Equation 4.6 includes mean compensation as well as a normalization of the standard deviation (contrast) of the pattern intensity. Mean compensation can be achieved by:

$$\beta = -\bar{g}(f) = -\frac{1}{|\mathrm{dom(f)}|} \sum_{x \in \mathrm{dom(f)}} f(x) \quad \text{and} \quad \alpha = 1 \ . \tag{4.7}$$

Once an image pattern has been mean compensated, the contrast can be normalized with:

$$\alpha = \frac{1}{\sqrt{\frac{1}{|\mathrm{dom(f)}|} \sum_{x \in \mathrm{dom(f)}} f(x)^2}} \ . \tag{4.8}$$

As presented in Sec. 4.5.3 contrast normalization does not improve tracking behavior significantly, whereas mean compensation is necessary.

4.3 Detection of Specular Reflections

Specular reflections (see Fig. 4.2) that appear on the wet heart surface strongly disturb tracking of natural landmarks because of their sudden motion.

Figure 4.2: Image of the beating heart with specularities.

On mirror-like surfaces specular reflections occur, which means that all parallel rays hitting the surface are reflected as parallel rays [36]. The following properties of specularities on the beating heart surface are important for the subsequent handling:

- Specularities have very high intensity, which is independent of the surrounding pixel intensity.
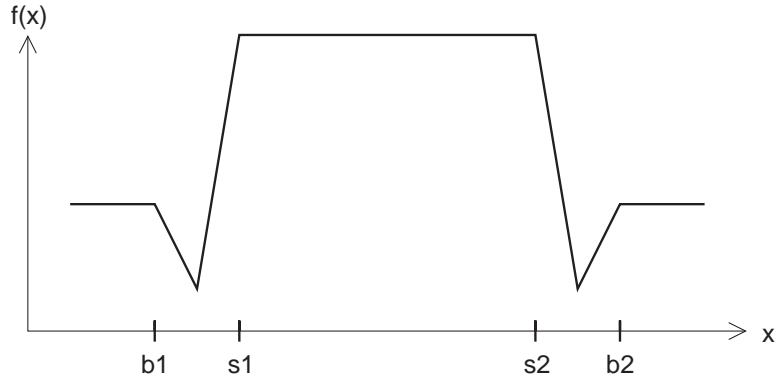
Figure 4.3: Specular reflection: one-dimensional intensity function f (schematic).

- Around the specularities a usually two pixels wide black boundary artifact with heavily decreased pixel intensity occurs (see areas $[b_1, s_1]$ and $[s_2, b_2]$ in Fig. 4.3).

- Rapid movement of the specularities occurs because the motion of the heart changes the orientation between heart surface and light source.

Specularities should not be tracked over an image sequence due to their sudden motion and appearance/disappearance. Therefore, reliable detection and subsequent treatment is necessary to avoid disturbance of the tracking. Keeping in mind that these steps have to be performed in realtime, powerful methods as proposed by Nayar et al [49] exploiting color and polarization information or the reflectance model by Wolf [83] are not suitable. Therefore, a simple intensity based technique is proposed, consisting of two major steps:

1. thresholding (Sec. 4.3.1) and

2. dilation (Sec. 4.3.2).

4.3.1   Thresholding

Thresholding is a pixel oriented operation that transforms a grey value image $f : M \times N \to G = \{0, .., 255\}$ into a binary image $f_\theta : M \times N \to \{0, 1\}$:

$$f_\theta(m, n) = \begin{cases} 1, & \text{if} \quad f(m, n) > \theta \\ 0, & \text{otherwise} \end{cases}$$

with $m \in M$ and $n \in N$.

Since specular reflections have a high intensity within the upper third of the value range, a global threshold to segment specularities in the whole image can be found:

$$\theta = \theta(f) = \bar{g}(f) + \theta_{\text{offset}} \tag{4.9}$$

with $\theta_{\text{offset}} \in \mathbb{N}$ and $\bar{g}(f)$ being the image mean. Consideration of the image mean leads to an adaption of $\theta(f)$ to different lighting conditions during the image sequence.

As shown in Fig. 4.4, the image mean $\bar{g}(f)$ varies between grey values of 85 and 95, here $\theta_{\text{offset}} = 60$ leads to robust segmentation of specular reflections over the given image sequence. Under different lighting or setup conditions, θ_{offset} may have to be adapted. See Fig. 4.5 for an example of a thresholded image.
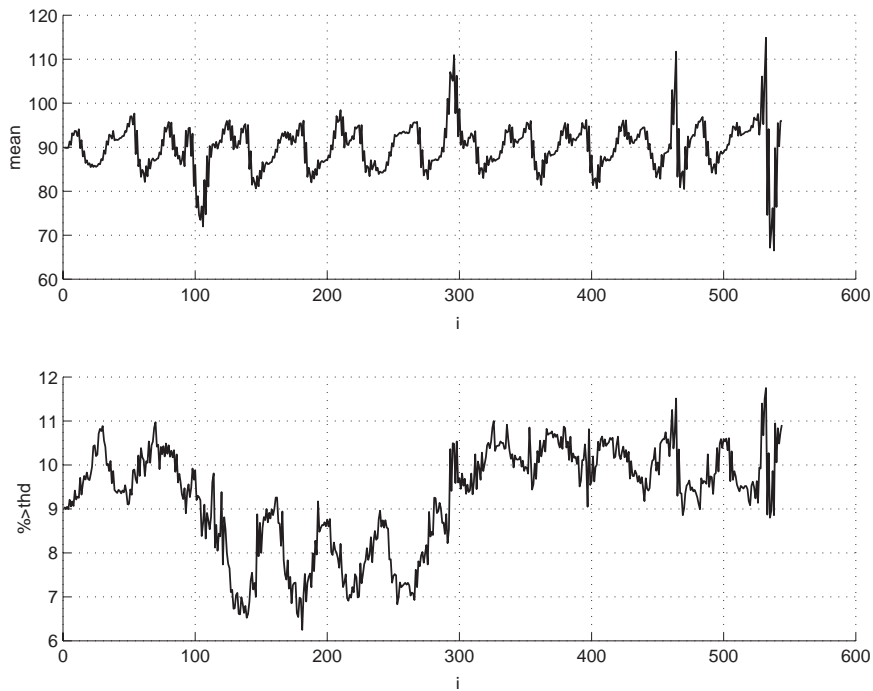

Figure 4.4: Global mean value over the image sequence and proportion of segmented pixels.

4.3.2 Dilation

The thresholding algorithm detects the area between $[s_1, s_2]$ as shown in Fig. 4.3 reliably. However, it is necessary to handle the black boundary artifacts ($[b_1, s_1]$ and $[s_2, b_2]$) as well. Therefore, the detected area needs to be extended by an
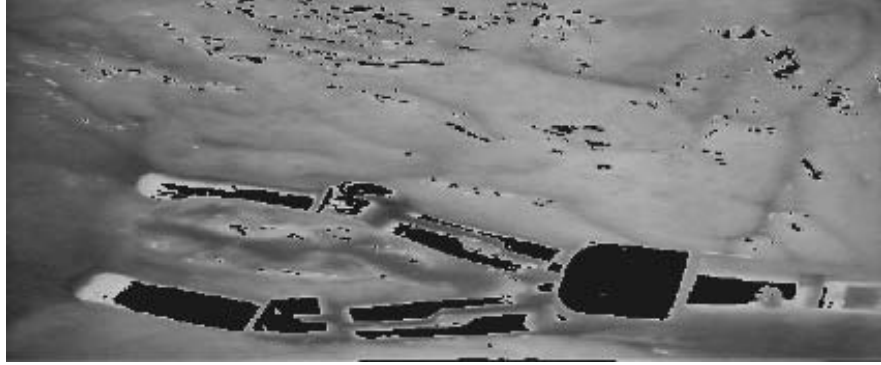
65

Figure 4.5: Image of the beating heart after thresholding (specular areas masked out).

appropriate number of pixels (here, the artifact is usually two pixels wide). This can be done by applying a dilation operation as described in [44] to the detected artifacts. The structuring element $\boldsymbol{S}$ with

$$\boldsymbol{S} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

is applied to each detected pixel, such that the thresholded area is enlarged. To cover the entire black boundary artifact, this is done twice (see Fig. 4.6).
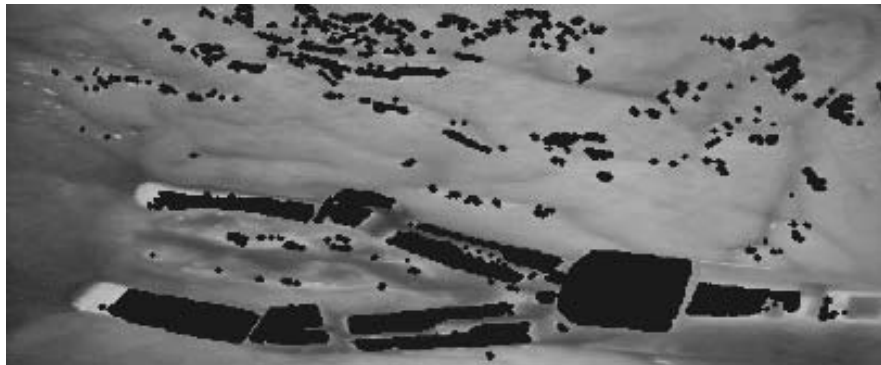

Figure 4.6: Image of the beating heart after dilation (specular areas masked out).

4.4 Elimination of Specular Reflections

After detecting the specular areas, their handling in the context of the tracking framework is presented. In the following sections two different methods are

considered: Detection of local structure information is a prerequisite for the first method dealing with the reconstruction of underlying structure information. The second method masks out the detected specularities.

4.4.1 Detection of Structure Orientation

In [24] two reconstruction methods, suitable for small disturbances, such as specular reflections are presented and discussed in detail: The first method is based on an anisotropic confidence-based filling in scheme. This iterative scheme provides good results but is not suitable for realtime applications. The second method, linear interpolation based on the structure tensor, provides almost the same results and is far less time consuming.

The following paragraphs introduce the structure tensor according to [80] briefly:

Building the tensor product with the derivative $\nabla f_\sigma \stackrel{\text{def}}{=} \left[\frac{\partial f_\sigma}{\partial x}, \frac{\partial f_\sigma}{\partial y} \right]^T$ of the Gaussian smoothed image f_σ, leads to:

$$\boldsymbol{J}_0(\nabla f_\sigma) \stackrel{\text{def}}{=} \nabla f_\sigma \otimes \nabla f_\sigma = \nabla f_\sigma \nabla f_\sigma^T = \left[\begin{array}{cc} (\frac{\partial f_\sigma}{\partial x})^2 & \frac{\partial f_\sigma}{\partial x} \frac{\partial f_\sigma}{\partial y} \\ \frac{\partial f_\sigma}{\partial x} \frac{\partial f_\sigma}{\partial y} & (\frac{\partial f_\sigma}{\partial y})^2 \end{array} \right] . \tag{4.10}$$

Since $\boldsymbol{J}_0$ is symmetric, the eigenvectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ of $\boldsymbol{J}_0$ form an orthonormal basis and are oriented according to the image gradient, i.e. $\boldsymbol{v}_1 \parallel \nabla f_\sigma$ and $\boldsymbol{v}_2 \perp \nabla f_\sigma$. The contrast (squared gradient) is given by the eigenvalues $|\nabla f_\sigma|^2$ and 0. To average orientation with neighbored pixels, the components of $\boldsymbol{J}_0$ are convolved with a Gaussian kernel g_ρ (standard deviation $\rho \geq 0$), which leads to the structure tensor:

$$\boldsymbol{J}_\rho(\nabla f_\sigma) \stackrel{\text{def}}{=} g_\rho * [\nabla f_\sigma \otimes \nabla f_\sigma] . \tag{4.11}$$

The *noise scale* σ used for the Gaussian kernel g_σ reduces the image noise before applying the gradient operator. The *integration scale* ρ, allows to adjust $\boldsymbol{J}_\rho$ to the size of structures which are to be detected.

The eigenvalues λ_1 and λ_2 of the structure tensor $\boldsymbol{J}_\rho$ specify the contrast in direction of the eigenvectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$. The eigenvector $\boldsymbol{v}_1$ corresponding to the eigenvalue λ_1 with $|\lambda_1| \geq |\lambda_2|$ gives the orientation of the highest contrast. The second eigenvector $\boldsymbol{v}_2$, with $\boldsymbol{v}_1 \perp \boldsymbol{v}_2$, gives the orientation of the local structure.

4.4.2 Structure Tensor Driven Reconstruction

The approach presented here uses local structure information around the specularity to reconstruct the underlying structure of the specular area. This leads to smooth transitions at the boundaries of the specularity, which are necessary for robust tracking.

Considering a specular point p_s, the two boundary points p_1 and p_2 along the structure orientation given by the eigenvector v_2 have to be searched. Linear interpolation between the intensities at p_1 and p_2 leads to the new intensity value at p_s (Fig. 4.7). To obtain reasonable structure information for points lying inside the specularity, the integration scale ρ has to be large enough to take the surroundings of the specularity into account. In other words, the structure to be detected has to be larger than the size of the specularities considered.
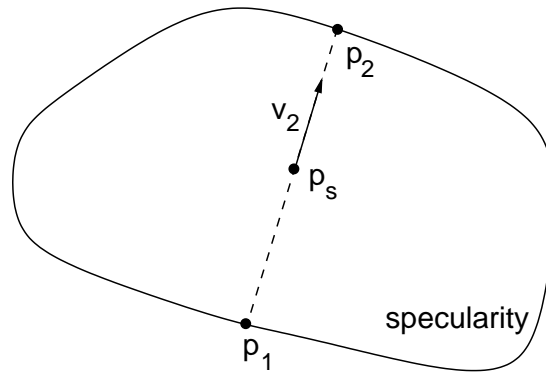

Figure 4.7: Illustration of reconstruction scheme.

With this interpolation method smoothness inside the specular area is not guaranteed, so Gaussian low-pass filtering is applied after the reconstruction process.

Figure 4.8 shows details of a reconstructed area. The noise scale was set to $\sigma = 1$ and the integration scale was set to $\rho = 2.8$.

Figure 4.8: Image reconstructed by structure tensor interpolation (detail).

4.4.3 Masking Specular Reflections

Besides the attempt to reconstruct structure information as presented in Sec. 4.4.2 another possibility is to exclude areas detected as specular form the similarity

measure (Eq. 4.4). This avoids the time-consuming reconstruction method, but has the drawback of not using all the information available, especially structure information from the local neighborhood of the specularities.

If a pattern of size $N = |\text{dom(r)}|$ contains N_r pixels belonging to reflections, the SSD measure has to be modified:

$$J = \sqrt{\frac{1}{N - N_r} \sum_{i \in \text{dom(r)} \backslash \text{dom(r}_\text{r})} (r(i) - p'(i))^2} \, , \tag{4.12}$$

with $\text{dom(r}_\text{r})$ being the domain of the N_r specular pixels. Additionally, the constraint

$$N_r \leq N_{\max} \tag{4.13}$$

has to be regarded to assure that enough structure information is still contained in the considered pattern. Otherwise, the tracking algorithm tends to areas filled with lots of specularities to minimize the similarity measure. This effect is similar to the aperture problem, if periodic structure is considered. Unfortunately, it is not possible to provide an upper bound for $N_{\max}$ for all considered landmarks, because it strongly depends on the structure of the pattern and the position of the specularities.

4.5   Tracking Results

In this section the proposed algorithms are evaluated. Therefore, two measures to quantify the tracking performance are introduced. The tracking environment itself is described in the subsequent section. The following section evaluates the illumination model proposed in Sec. 4.2.2. The treatment of specularities is judged for the reconstruction method as well as for the masking method, respectively. The evaluation of the tracking model shows the relevance of the warping parameters of the affine tracking model given in Sec. 4.2.1. Visual inspection as well as an analysis of the periodicity of the trajectories allows to discuss the accuracy of the proposed tracking scheme. Some details of this analysis have been published in [23].

4.5.1   Measures

Two measures to judge the performance of the tracking algorithms are introduced. With these measures it is not possible to quantify the accuracy of the tracking scheme, because the real pattern position in the image sequence is unknown, so visual inspection is inevitable (Sec. 4.5.7). Nevertheless, as shown in the subsequent sections, these measures allow a first insight into the performance of the algorithms.

Sum of squared differences The sum of squared differences (SSD) (Sec. 4.2) measures the similarity between two patterns. Finding $\boldsymbol{\mu}_{\text{opt}}$ does not guarantee that the correct affine parameters were found, but allows to compare the properness of different transformations $\boldsymbol{T}_i$.

Outlier measure The outlier measure o quantifies the deviation of the trajectory t from its smoothed version t_s:

$$o(t) = \sqrt{\frac{1}{|\text{dom}(t)|} \sum_{i \in \text{dom}(t)} (t(i) - t_s(i))^2} \ . \tag{4.14}$$

Therefore, a zero-phase forward and reverse filter to calculate t_s, is used. Figure 4.9 shows the original trajectory t as well as its smoothed version t_s. A zero-phase forward and reverse filter can be applied in off-line data analysis only due to causality.
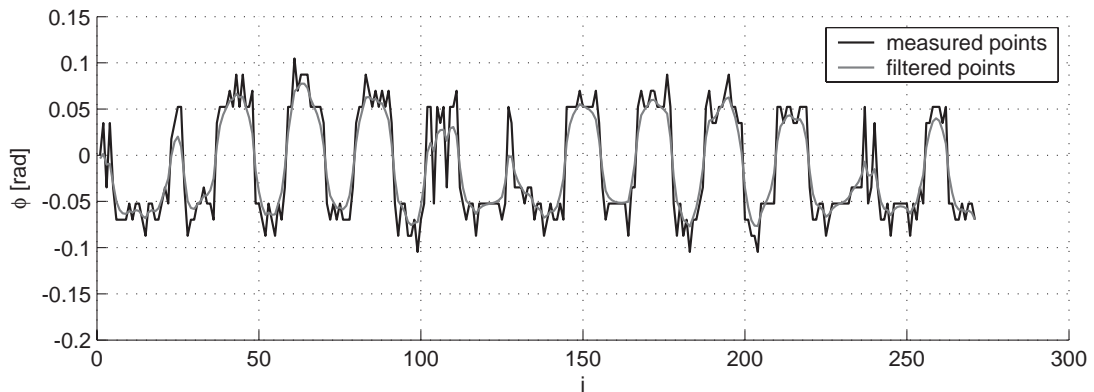

Figure 4.9: Rotation parameter unfiltered and filtered.

A high value for o indicates numerous outliers or a very noisy trajectory. As with the SSD measure no evaluation of the correctness of the optimal parameters $\boldsymbol{\mu}_{\text{opt}}$ is possible.

4.5.2 Tracking Environment

The following paragraphs present the tracking environment which is used for the subsequent analysis of $\boldsymbol{\mu}_{\text{opt}}$.

Image data A video sequence sampled during beating heart surgeries at the University Hospital Grosshadern (Munich, Prof. Reichenspurner, Dr. Boehm)

is investigated. To avoid motion artifacts tracking is performed on halfimages only, which are sampled at a frame rate of 25 Hz. All landmarks are chosen from inside the stabilized area (between the two arms of the mechanical stabilizer), in which a linear approximation of the nonlinear deformation with an affine model (cf. Sec. 4.2.1) is possible. Video sequences of different beating heart surgeries lead to similar results and are considered in Chap. 5.

Selected landmarks   Three landmarks are chosen to represent different cases: Figure 4.10 shows their location: From left to right landmarks LM2, LM8 and LM1 together with their translational search areas are presented. The size of the landmarks given in pixel ([px]) is 30 px × 15 px and the translational search space is 70 px × 70 px respectively.
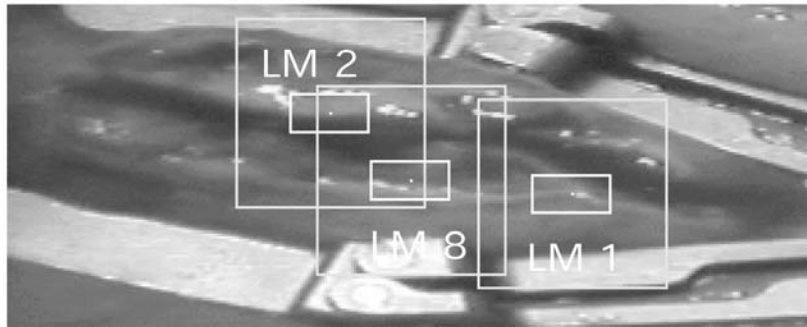

Figure 4.10: Landmarks within the tracking area.

Landmark one (LM1) is located at a y-crossing of two structures. This provides structure information in all major directions and thus allows robust tracking. Moreover, this landmark is not occluded by large specularities of the image sequence. So LM1 is a good landmark, and would not even require reconstruction in this case.

Landmark two (LM2) is often exposed to major reflections which disturb tracking, so this landmark gives a good example for disruptions caused by reflections. It is a good position to evaluate the treatment of specular reflections (see Sec. 4.4.2 and Sec. 4.4.3).

Landmark eight (LM8) contains horizontal structure only which makes robust tracking difficult, since not enough structure information is available to determine the parameter vector $\boldsymbol{\mu}_{\mathrm{opt}}$ reliably.

As mentioned in Sec. 4.2.2 a global search strategy is applied to calculate $\boldsymbol{\mu}_{\mathrm{opt}}$.

4.5.3 Evaluation of Illumination Models

In this section the illumination models as presented in Sec. 4.2.2 are evaluated by the outlier measure $o(t)$. Mean compensation is indispensable, since the trajectories have a lot of outliers otherwise (Tab. 4.1). Table 4.1 shows that refining mean compensation is not necessary, since the outlier measure for contrast normalized patterns does not improve significantly for the translational parameters. The outlier measure for the rotational parameter improves significantly, but as shown in Fig. 4.11 the translational trajectory itself is not improved (the same holds for other affine warping parameters, such as shear or scaling). Therefore, the time consuming contrast normalization can be avoided.
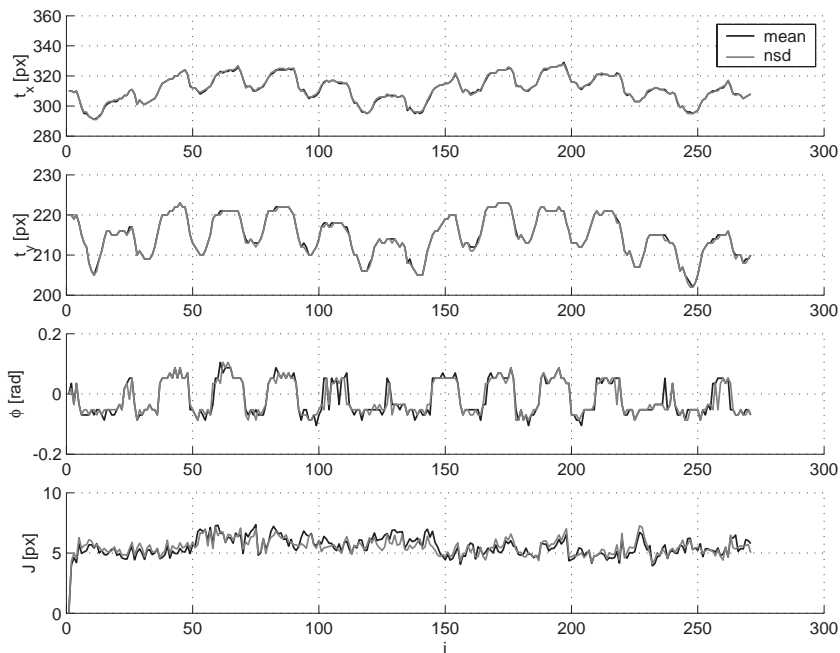

Figure 4.11: Illumination: standard deviation normalized (light) for translation and rotation at LM1.

4.5.4 Evaluation of Reconstruction

An evaluation of the structure driven reconstruction method proposed in Sec. 4.4.2 is presented. After applying mean compensation as recommended in Sec. 4.5.3 $\boldsymbol{\mu}_{\mathrm{opt}}$ is calculated. The outlier measure is used to judge the reconstruction scheme. Here, only LM2 and LM8 are considered, because they proved to be most sensitive to specular reflections and thus allow a good evaluation.

In Tab. 4.2 the outlier measures for tracking of patterns burdened with reflections

72

Table 4.1: Outlier measures for no mean compensation (no mean), mean compensation without normalization of standard deviation (mean), and mean compensation with normalization of standard deviation (nsd).

landmark /	outlier measure		
parameter	no mean	mean	nsd
LM1 t_x	6.98 px	1.30 px	1.30 px
LM1 t_y	4.13 px	0.86 px	0.88 px
LM1 ϕ	0.05 rad	0.02 rad	0.02 rad

as well as patterns reconstructed with the structure tensor method are given (translational parameters only). Generally, the outlier measures for t_x are worse than those for t_y. This is due to the horizontal structure of the patterns, which leads to a more robust detection of t_y.

Table 4.2: Outlier measures for tracking original and reconstructed specularities.

landmark /	method	
parameter	original	reconstructed
LM2 t_x	4.89 px	1.96 px
LM2 t_y	4.24 px	1.07 px
LM8 t_x	2.33 px	1.49 px
LM8 t_y	0.84 px	0.85 px

This survey is completed by Fig. 4.12 which shows frequent outliers for the non-reconstructed pattern. Additionally, the significantly lower SSD measure in Fig. 4.12 shows the superior performance of structure tensor interpolation based tracking.

4.5.5 Masking of Specular Reflections

After showing the performance of the reconstruction method, masking of specularities as proposed in Sec. 4.4.3 is evaluated. Therefore, global search is performed on reconstructed images (see Sec. 4.4.2) and on landmarks with masked specular reflections (see Sec. 4.4.3), respectively. In Fig. 4.13 the difference between these two methods is given; it can be seen that tracking behavior is almost the same.

Calculating the outlier measure of trajectories gained from reconstructed landmarks and from landmarks with masked specularities shows that only in case of LM2 the reconstruction method leads to superior behavior in t_x and t_y. For
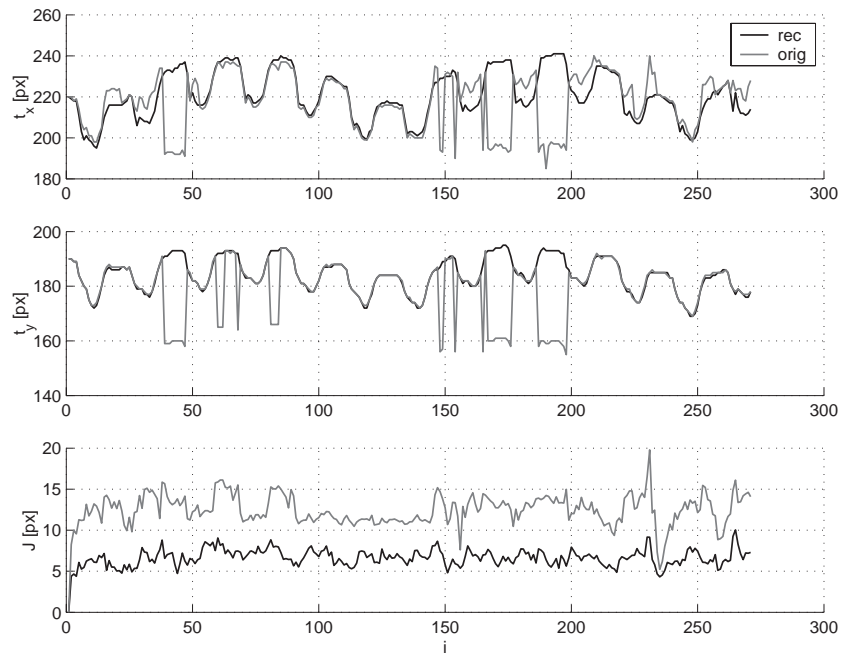
Figure 4.12: Structure tensor reconstruction (dark) versus original image at LM2.
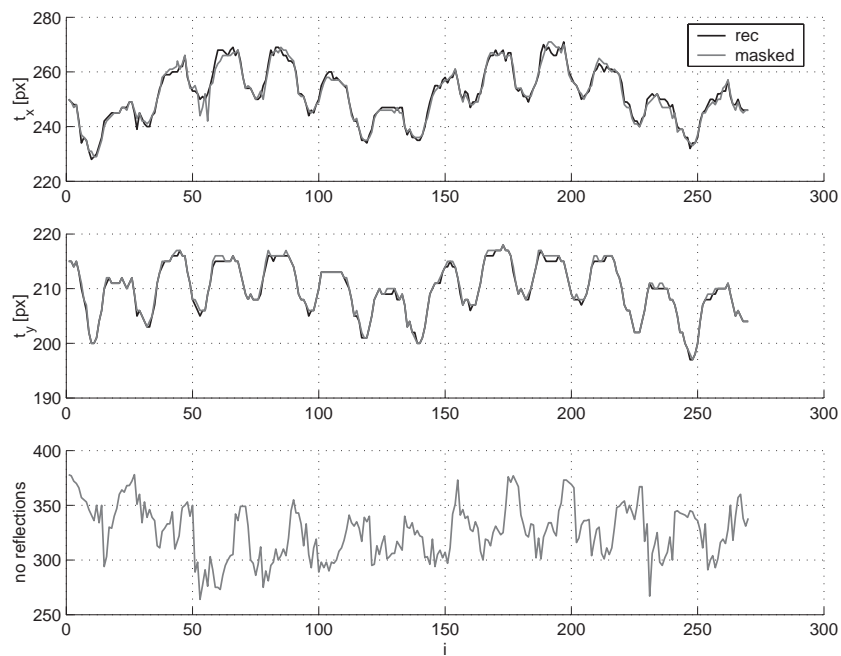

Figure 4.13: Trajectories for LM8: reconstructed (dark) and masked.

LM1 and LM8 masking of specular reflections and reconstructing the underlying structure show similar performance (Tab. 4.3). The superior behavior of the reconstruction method in case of LM2 can be explained as follows: Landmark LM2 is exposed to major reflections. Reconstruction of the underlying structure adds more information from the local neighborhood to the similarity measure than the masking method. Therefore, determination of $\boldsymbol{\mu}_{\mathrm{opt}}$ is more robust with first method.

Table 4.3: Outlier measures for masked and reconstructed specularities.

method	LM1		LM2		LM8	
	t_x	t_y	t_x	t_y	t_x	t_y
masked	1.35 px	0.83 px	2.19 px	1.07 px	1.48 px	0.85 px
reconstructed	1.30 px	0.86 px	1.96 px	1.06 px	1.49 px	0.85 px

Despite the fact that masking specular reflections is faster than structure driven reconstruction, subsequent analysis is done with trajectories based on the reconstruction method. This is because this method is better suited for specular reflections that move on the underlying pattern structure (e.g. LM2).

4.5.6 Evaluation of Affine Tracking Model

Section 4.5.4 shows that after appropriate handling of specular reflections, a reduction of the search space towards only two DoFs ($\boldsymbol{\mu} = [t_x, t_y]^T$) is possible. Nevertheless, the question arises, whether adding additional DoFs to the search space (here: the warping parameters of the affine motion model) allows to derive additional information. The drawback of an increased search space is the additional amount of computation time necessary to determine $\boldsymbol{\mu}_{\mathrm{opt}}$. Further investigations of the affine tracking model are published in [23], too.

The rotational parameter presented in Fig. 4.14 shows quasi-periodic behavior but contains a lot of noise. Furthermore, increasing the dimension of the search space degrades the trajectories of the translational parameters slightly, as it can be seen in Tab. 4.4. The considered landmark does not contain enough structure to allow a robust determination of all considered affine parameters and is therefore easily disturbed.

Adding more DoFs (here shear about any axis: $\boldsymbol{\mu} = [t_x, t_y, \tau, \alpha]^T$) degrades the translational parameters significantly, whereas the shear parameters themselves do not contain any significant information (Fig. 4.15). The same holds for scaling and rotation (summarized in Tab. 4.4).

Warping parameters contain a lot of noise and prevent robust determination of the translational parameters. Only the rotation parameter seems to provide
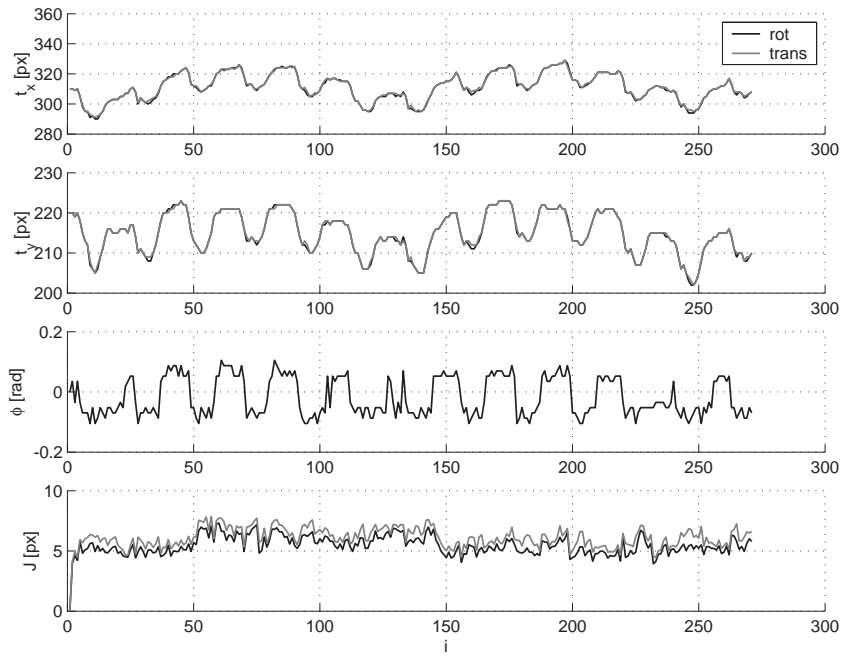
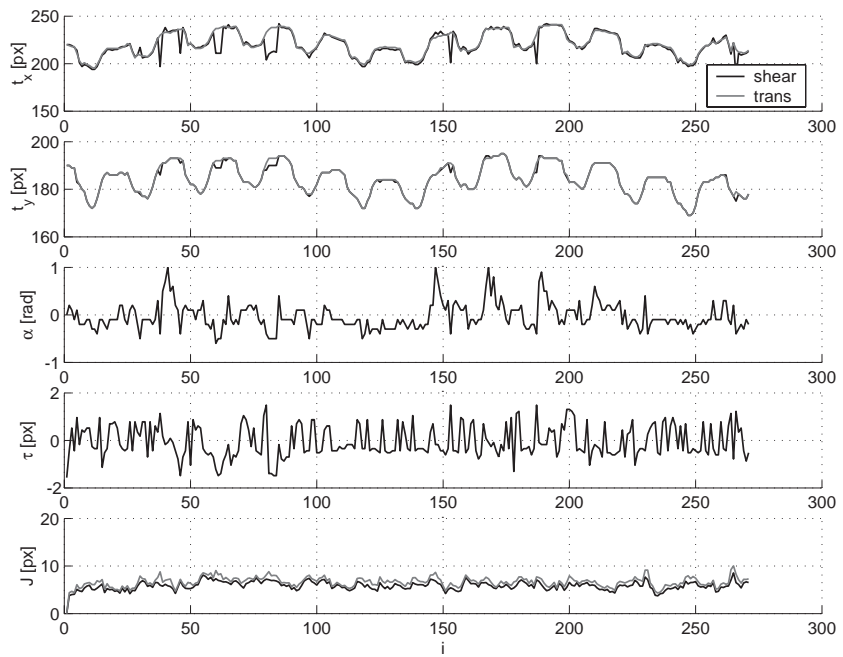Figure 4.14: Evaluation of rotation parameter at LM1.

Figure 4.15: Evaluation of shear parameters at LM1.

Table 4.4: Outlier measures for different DoFs of the search space.

parameters	LM1		LM2		LM8	
	t_x	t_y	t_x	t_y	t_x	t_y
translation only	1.30 px	0.86 px	1.96 px	1.06 px	1.49 px	0.85 px
rotation ϕ	1.38 px	0.88 px	2.15 px	1.06 px	1.48 px	0.84 px
shear (τ, α)	1.31 px	0.87 px	4.01 px	1.11 px	1.99 px	0.83 px
rot. + scale (ϕ, s)	1.29 px	0.89 px	2.08 px	1.07 px	1.40 px	0.84 px

reasonable additional information (without degrading t_x and t_y too much) which might be useful in subsequent analysis. It can also be seen that determining the translational parameters only is sufficient for robust tracking. These results are valid for other beating heart surgery video sequences, too (see Chap. 5).

4.5.7 Visual Inspection

Analysis of trajectories and corresponding measures (SSD and outlier measure) allow the evaluation and comparison of different models (e.g. illumination) and parameters (e.g. warping parameters). The true parameters of the affine motion model are unknown, so final avluation of correctness of $\boldsymbol{\mu}_{\mathrm{opt}}$ and accuracy can be achieved by visual inspection only.
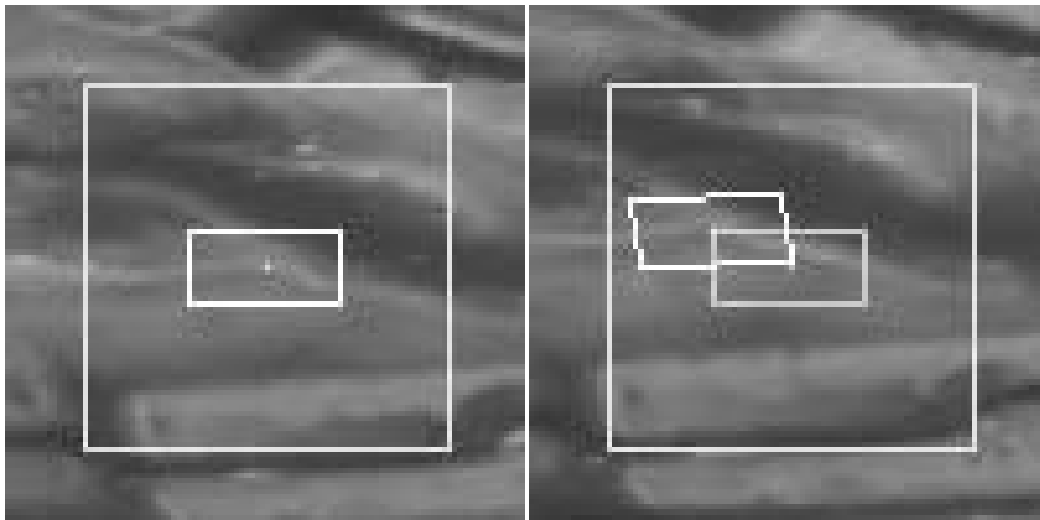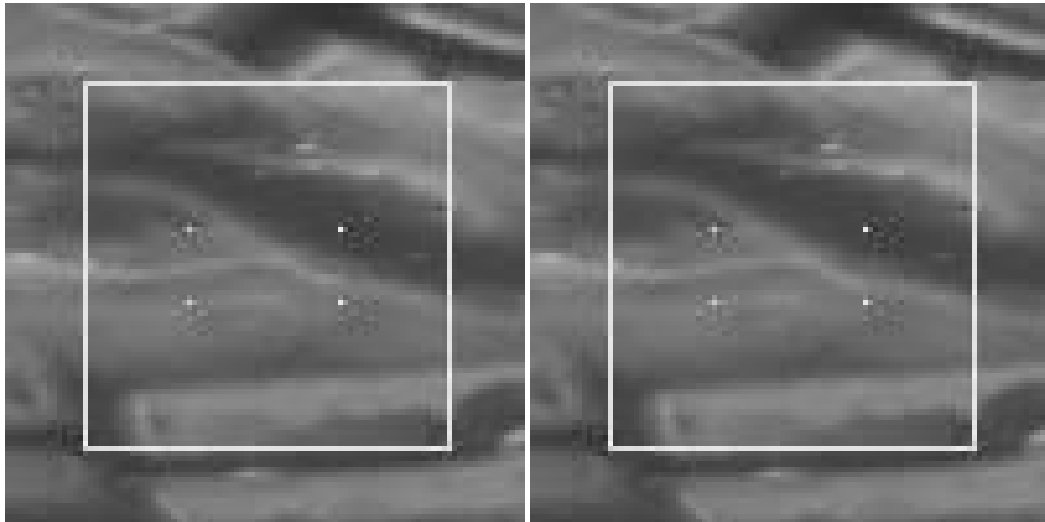

(a) Reference pattern   (b) Tracked pattern

Figure 4.16: Tracking of LM1.

As an example, tracking is investigated at landmark LM1 for a temporal displacement of 7 frames between the two images. Two different cases are considered: Translation only and translation in combination with rotation.

Figure 4.16(a) shows LM1 in detail, surrounded by a bright rectangle. Figure 4.16(b) shows the tracked target after 7 frames. It occurs displaced and rotated from the pose of the reference pattern.

(a) Pure translation   (b) Translation and rotation

Figure 4.17: Tracked pattern pasted into reference image.

The tracked pattern is then pasted into the reference image (see Fig. 4.17(a) for pure translation and Fig. 4.17(b) for translation with rotation). The surrounding structure is well continued through the pattern, the transitions for translation and rotation are slightly smoother than for pure translation.

Visual inspection shows that the correct position and the warping parameters of the affine motion model are calculated with high accuracy. The given example also shows that rotation can be neglected.

4.5.8   Detection of Periodicity

As presented in the preceding sections, periodicity is apparent for the translational parameters as well as for rotation. The discrete Fourier transformation (DFT) is applied to the trajectories of these parameters (calculated over a sequence of 931 images), to gain detailed information about the dominant frequencies. Figure 4.18 presents the tracking results for LM2 only, but LM1 and LM8
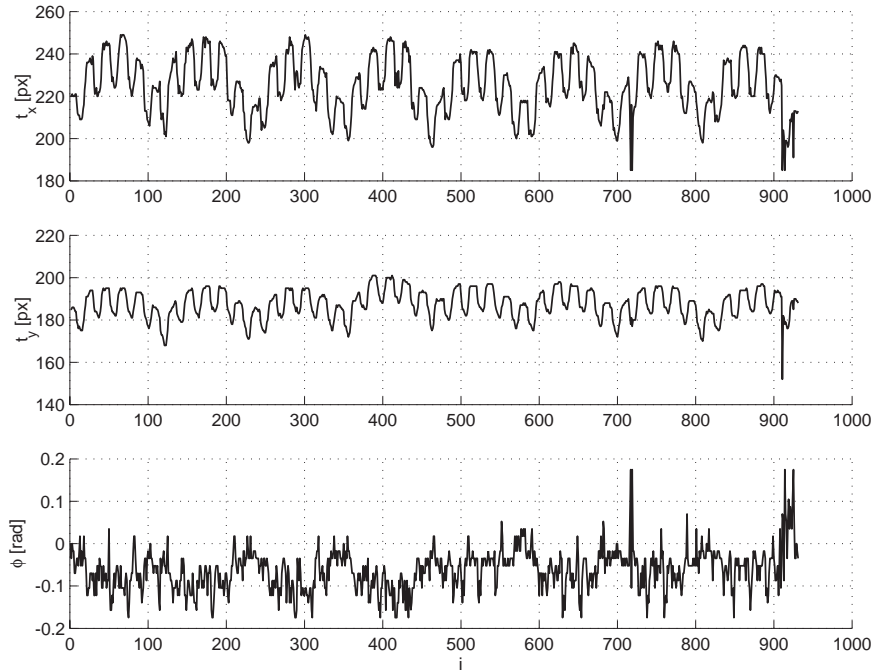
show similar behavior.

Figure 4.18: Trajectories of selected affine parameters at LM2.

For the translational parameters two dominant peaks, at $f_1 = 14.5$ min^{-1} and $f_2 = 70.9$ min^{-1}, can be detected (cf. Fig. 4.19). Frequency f_1 corresponds to the respiration rate and frequency f_2 to the heart beat rate. Additionally, the first and second harmonic of the heart beat (that is not sinusoidal) can be seen, with $f_3 = 141.8$ min^{-1} and $f_4 = 212.7$ min^{-1}. This demonstrates the accuracy of the tracking scheme presented here. The same holds for rotation, but the spectrum contains more noise. The assumption that f_1 and f_2 correspond to physiological parameters is confirmed in Sec. 5.4, where in addition to the video sequences also the ECG (electrocardiogram) and the RPS (respiration pressure signal) are considered.

4.6 Quality of Landmarks

As the total amount of computation time in realtime applications is limited, the quality of the considered landmarks with respect to robust tracking is important. It is better to track few but reliable landmarks than plenty of unreliable landmarks. A confidence measure able to distinguish between landmarks which lead to robust and non robust tracking seems to be useful. This measure has to fulfill the following requirements:
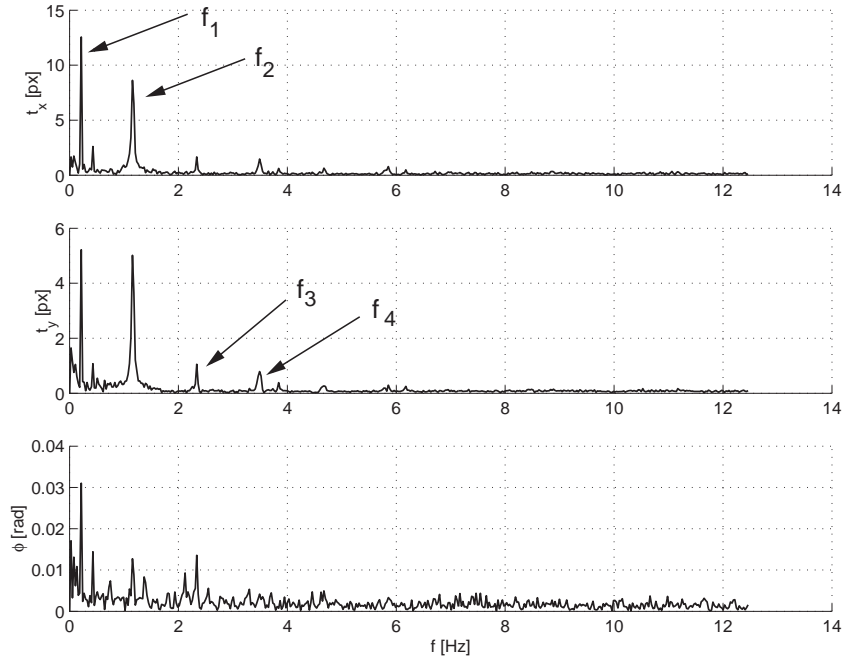
Figure 4.19: Amplitude spectrum of selected affine parameters at LM2.

- low computational cost

- high significance, when applied to real data

- local and global properties of the pattern and the corresponding search area
  are taken into account (see below)

First an appropriate confidence measure for judging the tracking quality of a
landmark is introduced. Thereafter, experimental results for the proposed confidence measure are presented, when it is applied to video sequences derived during
heart surgeries.

4.6.1 Confidence Measures

In [70] the following error sources that might influence successful matching of the
reference pattern have been identified:

Significance Considering the shape of the error function J, the curvature at
the minimum has to be very high. This eases reliable detection of the correct
minimum, even in case of noise. Figure 4.20 shows the plot of the error function
for the translational search space $\boldsymbol{\mu} = [t_x, t_y]^T$ for LM1. Here, matching of the

80

reference pattern is applied to the reference image itself. The high curvature at the global minimum $\boldsymbol{\mu}_{\mathrm{opt}} = [0,0]^T$ (marked with vertical line) indicates, that LM1 is a very significant pattern. This is confirmed by investigations presented in Sec. 4.5.
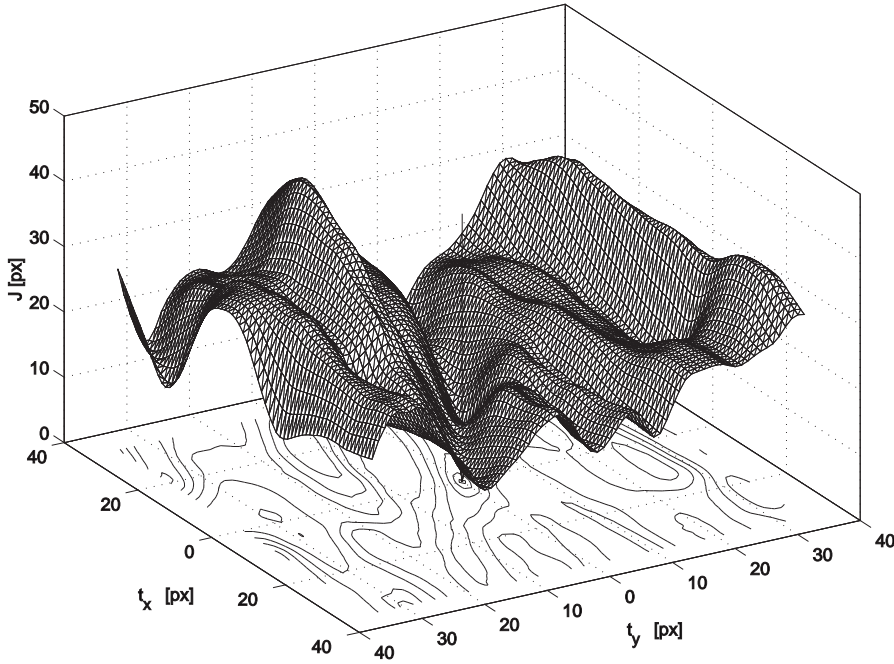

Figure 4.20: SSD measure versus t_x, t_y at LM1 (applied to same image).

White Wall Problem A pattern with almost no underlying structure information (e.g. taken from a white wall) allows no reliable matching in subsequent images. The minimum of the error function is strongly influenced by noise and no reliable motion detection is possible. In the application investigated here, the white wall problem can be neglected, because some structure is always available.

Aperture Problem A pattern that does not allow to determine $\boldsymbol{\mu} = [t_x, t_y]^T$ due to the limited field of view (e.g. the pattern contains part a horizontal line only: this does not allow the determination of the horizontal position of the pattern). The minimum in this direction is flat, the global minimum found is strongly influenced by noise. Figure 4.21 shows the error plot for LM2. It can be seen that the curvature is flat in x-direction, which is due to the structure captured by LM2. The determination of t_x is therefore unreliable (see Sec. 4.5.4).
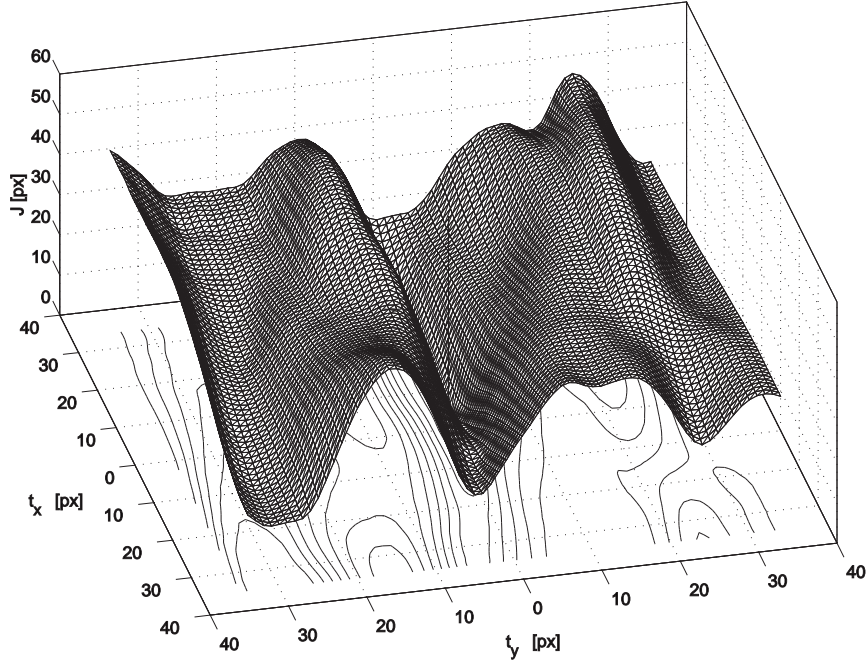
Figure 4.21: SSD measure versus t_x, t_y at LM2 (applied to same image).

Periodic Structure If the pattern contains parts of a periodic structure that can be found elsewhere in the search area several minima with almost the same value of J occur. The determination of the global minimum is therefore strongly dependend on noise and correct calculation of $\boldsymbol{\mu}_{\mathrm{opt}} = [t_x, t_y]^T$ cannot be guaranteed. Unfortunately, periodic structure occurs in minimally invasive heart surgery, too.

Confidence Measure Confidence measures can be divided up into *local* and *global* measures. A local measure takes information into account that can be derived from the surroundings of the detected minimum only (e.g. curvature), whereas a global measure considers the whole search area. In [70] several confidence measures are discussed and judged for robust calculation of optical flow in mobile robotics. As the calculation of optical flow in [70] is based on pattern matching, using the SAD (sum of absolute difference) as similarity measure, the tracking model is similar to the one used in this work (see Sec. 4.2). According to [70] the following confidence measure is best suited and takes local as well as global information into account: When calculating J not only the global minimum

$$e_{\mathrm{min}} = J(r, p'(\boldsymbol{\mu}_{\mathrm{opt}})) \tag{4.15}$$

82

is determined, but also the second lowest value of J: e_2. The difference

$$d = e_2 - e_{\min} \qquad (4.16)$$

is then used as a confidence measure. In case of the white wall problem as well as the aperture problem, d is small. In case of periodicity, with the error plot having similar minima, d is small, too. Considering the first case (significance), the curvature is taken into account: flat curvature leads to small d, whereas large curvature leads to high values for d. A problem occurs, if the global minimum lies between two grid points of the CCD chip of the camera. As shown in Fig. 4.22 the minimum has the same curvature in the left and the right parts of the figure, but the confidence measure for the left part is significantly higher than for the right part.
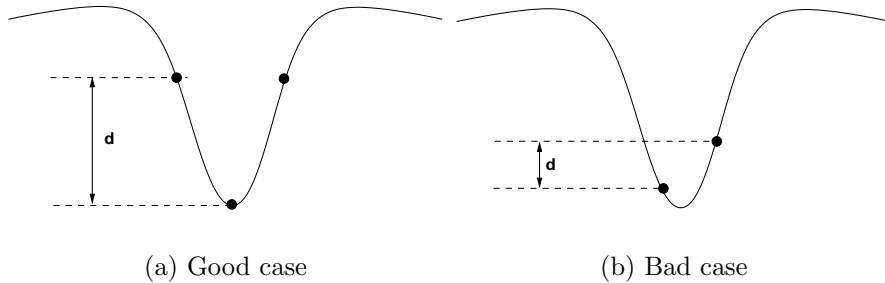

(a) Good case   (b) Bad case

Figure 4.22: Global minimum between grids of the CCD chip [70].

According to [70] this is the main drawback of this measure. To avoid this disadvantage, the following modification is proposed: Not simply the second lowest value e_2 is chosen, but the second lowest value e_2 which has a minimum spatial distance $\delta_{\min}$ from the position of the global minimum $e_{\min}$ is chosen instead: Assuming that $e_{\min}$ is located at $\boldsymbol{\mu}_{\mathrm{opt}} = [t_{x,1}, t_{y,1}]^T$, the second lowest value taken into account (located at $\boldsymbol{\mu}_2 = [t_{x,2}, t_{y,2}]^T$) has to fulfill the following condition:

$$\delta = \sqrt{(t_{x,1} - t_{x,2})^2 + (t_{y,1} - t_{y,2})^2} > \delta_{\min} \qquad (4.17)$$

with $\delta_{\min}$ being the minimum spatial distance. The example given in Fig. 4.23 shows, that this modification leads to high values for d in both cases.

4.6.2 Experimental Results

For this experiment 332 landmarks inside the stabilized area are considered. They have a spatial distance of 5 pixels from each other and parts of the mechanical stabilizer are not included. The search space is limited to translation only. The 332
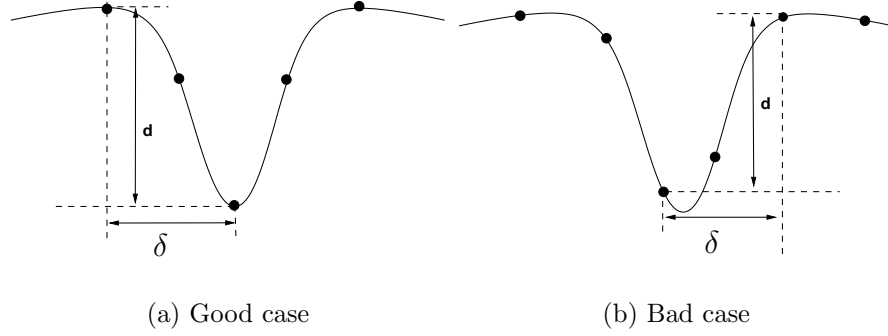
(a) Good case   (b) Bad case

Figure 4.23: Modification of the confidence measure.

landmarks considered are split up into two parts: The large part (282 landmarks) is used to calculate a threshold for d. This threshold is expected to distinguish between good and bad landmarks. The small part (50 landmarks) is used later as a test set to evaluate the performance of the threshold calculated before.

The measure introduced above is calculated for each of the 282 arbitrarily chosen landmarks for the first 10 seconds (250 frames). Thus for each landmark 250 confidence measurements are available. The minimal spatial distance according to Eq. 4.17 was set to $\delta_{\min} = 2$ px. The mean d_i of the 20 lowest confidence measures for each landmark i is calculated and used for subsequent evaluation. Since the translational trajectories are (quasi-) periodic it is assumed (and in the following paragraphs also justified) that d_i describes not only the reliability of the computed trajectory for the first 10 seconds, but also for the entire trajectory (931 frames).

To quantify the reliability of the trajectory, a zero phase forward and reverse filter is applied to the two dimensional trajectory $t = (t_x, t_y)$. At every frame the deviation between the smoothed trajectory $\bar{t} = (\bar{t}_x, \bar{t}_y)$ and the unfiltered trajectory is calculated:

$$d(t, \bar{t}) = \sqrt{(t_x - \bar{t}_x)^2 + (t_y - \bar{t}_y)^2} \ . \tag{4.18}$$

Trajectory points at time j are identified as outliers if the distance function exceeds a certain threshold Θ_d:

$$s(j) = \begin{cases} 0: & d(t, \bar{t})(j) \leq \Theta_d \\ 1: & d(t, \bar{t})(j) > \Theta_d \end{cases} \tag{4.19}$$

and $j \in \mathrm{dom}(t)$. A threshold value of $\Theta_d = 8$ px proved to be appropriate to detect outliers and to omit deviations due to signal noise. The reliability for the

84

first 10 seconds (250 frames) for each trajectory i is quantified by

$$\hat{s}_i = \sum_{j=1}^{250} s_i(j) \quad \text{with} \quad i \in \{1,..,282\} , \tag{4.20}$$

describing the number of outliers.

Figure 4.24(a) shows the experimental frequencies of the detected outliers for the 282 trajectories considered (first 250 frames only). For confidence measures larger than 2 px almost no outliers occur. For the classification of good landmarks, a

(a) Experimental frequencies   (b) Cumulative frequency distribution
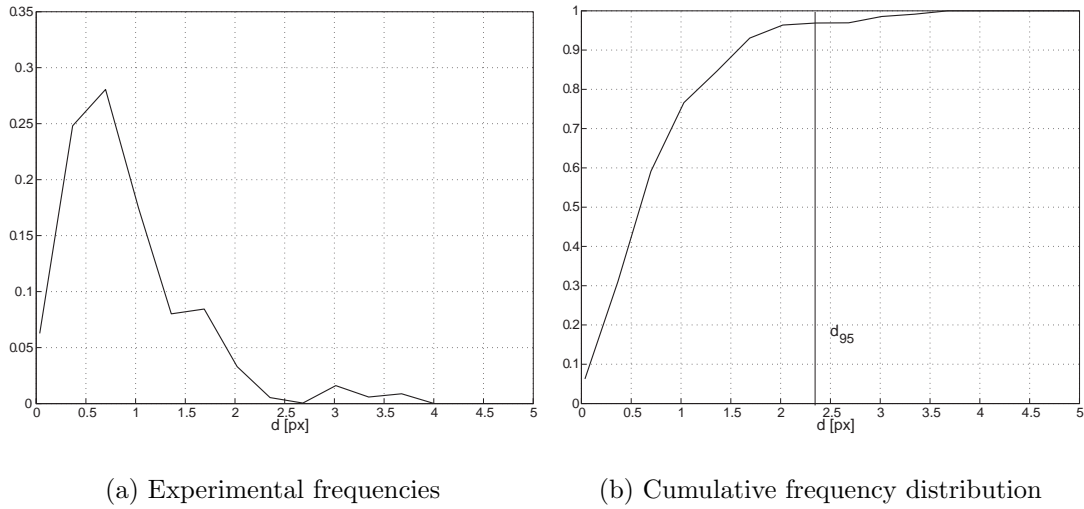
Figure 4.24: Experimental frequencies and their cumulative distribution.

certain threshold d_{95} is searched. Only trajectories with a confidence measure $d > d_{95}$ are regarded as reliable and used for estimating the heart motion. The threshold d_{95} is chosen such that 95% of the outliers (of the first 250 frames) of the 282 trajectories are rejected. Therefore, the cumulative frequency distribution (Fig. 4.24(b)) has to be calculated. The threshold is $d_{95} = 2.41$ px. In this example 48 trajectories (17 % of all trajectories considered) have a confidence measure $d > d_{95}$. If the entire sequence is taken into account (931 frames), then these 48 trajectories are responsible for 3.5% of the outliers of all 282 trajectories. None of the trajectories chosen has more than 25 outliers, 27 trajectories have less than 4 outliers. This leads to the conclusion that the proposed measure describes the reliability of a landmark.

The question arises, how many reliable landmarks have a confidence measure less than d_{95} and are therefore rejected, or in other words, what is the efficiency of the method proposed here? In this section a trajectory is called *reliable*, if it has less than five outliers. There are 88 trajectories with less than 5 outliers and a

confidence measure $d < d_{95}$. 43 trajectories are reliable and have a confidence measure $d > d_{95}$. Therefore, the efficiency is 33%.

Considering the remaining 50 trajectories, 10 trajectories are identified as reliable with the threshold d_{95} being applied. These trajectories account for 0.39% of all outliers. When applied to this test-data, the computed threshold leads to good results, too.

Conclusion A measure describing the reliability of a landmark was proposed. It takes global as well as local structure information into account, has low computational effort and allows the detection of good landmarks in realtime. These landmarks are used in Sec. 5.5 to build a global prediction scheme, enhancing the robustness of the tracking algorithm.

4.7 3D Investigations

To compensate the motion of the heart surface, not only the trajectories lying in the image-plane (x- and y-direction) have to be calculated, but also the motion in z-direction has to be captured. Stereo vision seems to be an appropriate way to gain depth information, since stereo cameras are widely used in minimally invasive robotic cardiac surgery. As shown in Fig. 4.25 capturing the motion of a reference pattern in the left and right cameras is possible (here 3rd order pixel interpolation is used to get subpixel accuracy). Due to the small distance of the cameras (approx. 3 mm) and the large distance of the cameras from the heart surface (approx. 10 cm) the disparity is rather small (approx. 3 px). Therefore, small disturbances or small tracking errors lead to large errors when calculating the motion in z-direction. It does not seem to be possible to gain reliable depth information without additional sensors (e.g. laser range sensor).

To gain further information about the z-motion of the heart surface a pattern is tracked over a sequence of 200 frames with translation and scaling as affine motion parameters. As shown in Fig. 4.26 the scaling parameter s contains a lot of noise but changes significantly, although it is not necessary to determine s for tracking itself. This indicates, that motion in z-direction may not be neglected and additional sensors are needed to gain reliable depth information.

4.8 Conclusions

Capturing the remaining motion of a mechanically stabilized beating heart by exploiting natural landmarks is possible. This is fundamental for developing control schemes allowing minimally invasive robotic surgery on a virtually stabilized
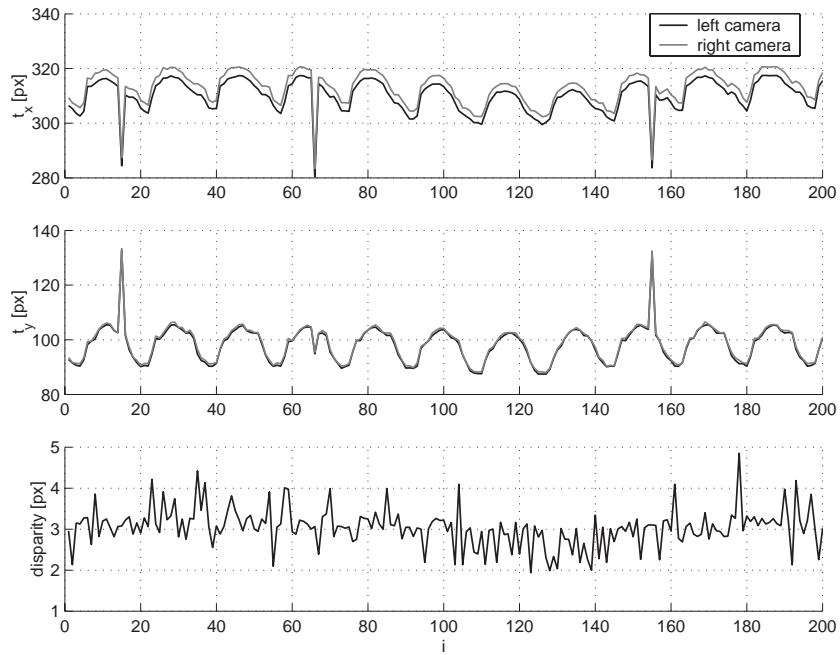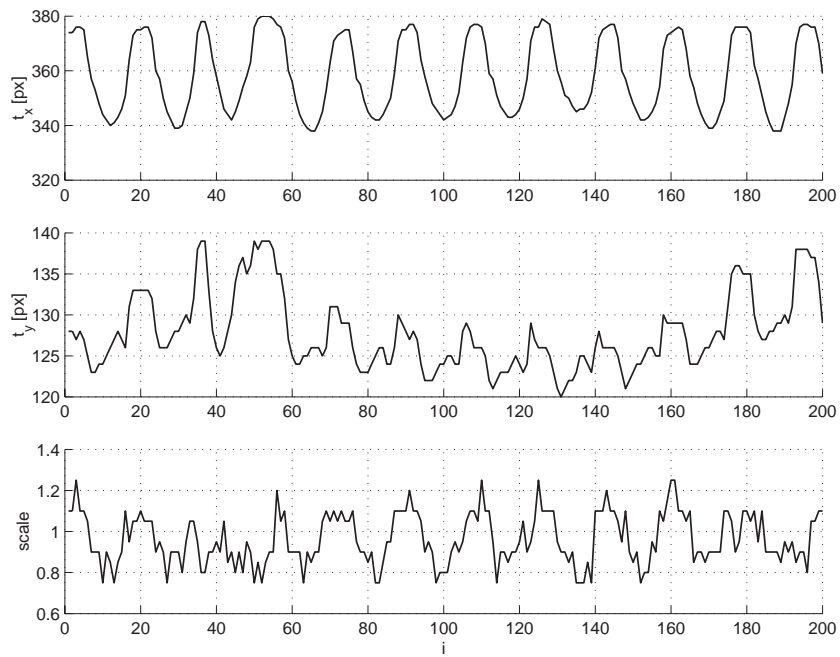
Figure 4.25: Stereo tracking.

Figure 4.26: Tracking with scaling.

87

beating heart (Sec. 1.3).

After appropriate handling of specular reflections, either by structure driven reconstruction or by masking of specularities, a reduced affine similarity model is sufficient for robust tracking: Besides the translational parameters only rotation can be derived reliably without too much noise corrupting the computed trajectory. Visual inspection shows the high accuracy of the tracking scheme proposed here. Analyzing the trajectories with the help of the discrete Fourier transform shows two dominant peaks. They are a result of the patient's heart beat and respiration. Even the first and second harmonic of the heart beat can be derived. An algorithm to detect good landmarks automatically was proposed. Its main features are low computational effort and high significance when applied to real data.

The main problem with visual servoing as proposed here is that it can be disturbed easily (e.g. by occlusions). Therefore, the next chapter establishes a framework allowing robust outlier detection as well as prediction of the expected heart motion. It is shown that these algorithms are suited for a more robust tracking and can handle the above mentioned disturbances.

Chapter 5

Robust Motion Estimation in MIRCS

The motion of a mechanically stabilized beating heart can be captured by exploiting natural landmarks as shown in the preceding chapter. Tracking is robust, even in presence of specular reflections. Nevertheless, landmarks may be occluded (e.g. by surgical instruments) for a short time and therefore tracking fails. To guarantee robust motion estimation under these circumstances, algorithms were developed which are able to predict the heart motion if no tracking information is available. This will not only bridge missing tracking information, but also allow dynamic positioning of the tracking search area. Additionally, prediction is useful for motion compensation: it helps to overcome the delay time of the closed controller loop (including video capturing, data processing, and data transmission) and therefore increases the bandwidth of the robotic system.

The first section of this chapter presents related work which deals with the prediction of signals. Section 5.2 introduces a (local) prediction scheme suitable for quasi-periodic trajectories. This scheme is applied to heart surface trajectories (HSTs), derived from tracking of natural landmarks. Local prediction of HSTs is suited to detect outliers and allows to bridge several subsequent tracking failures. Results of a global prediction scheme exploiting several landmarks at once are shown in Sec. 5.3. In Sec. 5.4 a long-term prediction of HSTs based on electrocardiogram (ECG) signals and respiration pressure signals (RPS) is given. The last section presents an overall tracking scheme that combines the advantages of the prediction algorithms proposed before and thus allows robust, long-term prediction of the heart motion.

5.1 Related Work

Prediction is necessary, if long delay times (e.g. due to data transmission, computation time etc.) occur: They deteriorate the bandwidth of the closed loop dramatically. The ROTEX experiment [31] demonstrated that delay times of up to 6 seconds can be compensated if the remote environment is well known. Furthermore, prediction is important, if unreliable sensor data or low sampling time limits the performance of the closed loop: In [67] prediction of the tumor movement is calculated to compensate for the low sample rate of the X-Ray camera used to capture the tumor motion. This technique allows high accuracy X-Ray cancer treatment. In beating heart surgery no applications or research groups dealing with prediction of the heart motion are known so far.

The main reason why common prediction algorithms fail (e.g. calculation of dominant frequencies and subsequent fitting of sine- and cosine-functions) is due to the quasi-periodicity of the HSTs. Analyzing the amplitude spectrum as in Sec. 4.5.8 gives dominant frequencies, but is only a global view on the trajectories. Local deviations (such as small variations of the heart beat rate) are not taken into account. In contrast, a windowed Fourier transform does not consider low frequencies (i.e. respiration rate). Even frequency trackers fail which might be used to extract dominant frequencies as proposed in [27]. Therefore, a new prediction algorithm which is able to deal with quasi-periodic signals has to be developed.

5.2 Local Prediction

This section presents a local prediction scheme. Here *local* means that information of only one trajectory is used to predict its position one or several steps ahead. Originally, this algorithm was developed to perform one-step predictions to detect outliers reliably and to position the search area of the tracking algorithm. Nevertheless, this method can be easily extended to more-step predictions, although prediction accuracy degrades with increasing prediction interval s (see Sec. 5.2.4).

The prediction scheme presented here does not consider explicit models (e.g. differential equations). This leads to the advantage that no model parameters which might vary over time as well as be patient-dependent have to be determined. Furthermore, it is very difficult to gain an explicit model for HSTs or the correlation between HSTs and ECG or RPS as used in Sec. 5.4 and Sec. 5.5.

5.2.1 Algorithm

Before the prediction algorithm is introduced, the dynamical system considered as well as some definitions are given.

The function $\boldsymbol{F}$ represents the underlying dynamics of the system and $\boldsymbol{x}_k$ represents the inner states:

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}(\boldsymbol{x}_k) \quad \text{with} \quad \boldsymbol{x}_k \in \mathbb{R}^d \ . \tag{5.1}$$

The scalar-valued measurement y_k at time t_k can be written as follows:

$$y_k = m(\boldsymbol{x}_k) \quad \text{with} \quad y_k \in \mathbb{R} \ , \tag{5.2}$$

with m being a smooth measurement function. The goal of the prediction algorithm is to predict the next y_{k+s} (with $s \geq 1$).

As the inner states $\boldsymbol{x}_k$, the functions $\boldsymbol{F}$ and m as well as the dimension d are unknown and cannot be identified easily, they cannot be used for prediction. Therefore, one has to find an alternative description of the dynamical system, based on the known measurements y_k only: A sequence of measurements $\boldsymbol{y}$, called *time series*, is used to capture the unknown inner states of the underlying dynamical system producing the measured output y_k. The time series $\boldsymbol{y}$ is:

$$\boldsymbol{y} = [y_n, y_{n-1}, ..., y_1]^T \ . \tag{5.3}$$

With the measurement function m and the inner states $\boldsymbol{x}_k$ of the underlying dynamical system, the time series can be written as:

$$\boldsymbol{y} = [m(\boldsymbol{x}_n), m(\boldsymbol{x}_{n-1}), ..., m(\boldsymbol{x}_1)]^T \ . \tag{5.4}$$

If $\boldsymbol{F}$ is smooth (at least C^2) and the dynamical system moves on an attractor (as it is assumed in the cases considered here), then the prediction can be achieved with the help of Takens Theorem [71]: Taking a sufficiently long vector built of past values of a time series enables the reconstruction of the underlying structure of the system dynamics which produced the sequence [81]. Mapping from the unknown inner states to these vectors and vice versa is unique.

The sufficiently long vector is called *embedding vector* $\boldsymbol{D}_k$ and is made of p past measurements with lag h between two subsequent components (for the concept of embedding [39]):

$$\boldsymbol{D}_k = \left[y_k, y_{k-h}, ..., y_{k-(p-1)h}\right]^T \ , \tag{5.5}$$

which is:

$$\boldsymbol{D}_k = \left[m\left(\boldsymbol{x}_k\right), m(\boldsymbol{x}_{k-h}), ..., m(\boldsymbol{x}_{k-(p-1)h})\right]^T \ . \tag{5.6}$$

According to Takens, p has to satisfy the condition:

$$p \geq 2d + 1 \ . \tag{5.7}$$

Unfortunately, the dimension d is unknown. Additionally, the embedding (time) lag h has to be chosen carefully: If the embedding lag is too small, then the values y_k and y_{k-h} are almost identical and therefore y_{k-h} contains hardly any new information. Computational effort rises due to the additional amount of data which has to be processed, as the dimension of the embedding vector has to be increased to capture reliably the inner states of the dynamical system. If the embedding lag is too large, y_k and y_{k-h} are almost uncorrelated and the reconstructed dynamics may become very complicated [39]. If the embedding dimension p is too small, then not all of the underlying dynamics is captured, whereas a too large p leads to uncorrelated data in $\boldsymbol{D}_k$. The product $H = p\,h$ stands for the size of the interval which is considered by $\boldsymbol{D}_k$. If H is too large local deviations are not considered and prediction fails (see Sec. 5.1). If H is too small low frequencies are not included and prediction fails, too.

The prediction algorithm itself is introduced before an algorithm to determine optimal values for p and h is presented. Therefore, it is assumed that p and h are chosen appropriately.

Prediction algorithm It has to be shown that the dynamical system moves on an attractor which is a subset of the state space of the system. This means in particular that there are no transient inner states any more. As this can be guaranteed, the following scheme can be applied: The current embedding vector is compared with embedding vectors lying in the past. If a similar embedding vector is found, then according to Takens, similar inner states of the dynamical system have been found. As the inner states are similar and $\boldsymbol{F} \in C^2$, the dynamical system will produce outputs similar to the ones detected in the past (see also [39] for a short description). To be less sensitive to noise when calculating the prediction, not only one similar embedding vector is searched, but M similar embedding vectors lying in the past are taken into account.

A s-step prediction at time k is calculated, by comparing the past vectors

$$\boldsymbol{D}_{k-i} \in \mathbb{R}^p \quad \text{with} \quad \mathrm{i} \in \mathbb{I} \tag{5.8}$$

with the reference vector $\boldsymbol{D}_k$. The values for i have to be chosen in a way to fit the borders of the known time series ($\mathbb{I} = \{s, s+1, .., k-1-(p-1)h\}$). The past vectors $\boldsymbol{D}_{k-i}$ are also called *memory* of the prediction algorithm. Depending on the Euclidean distance

$$\delta_i = \|\boldsymbol{D}_k - \boldsymbol{D}_{k-i}\|_2 \quad \text{with} \quad i \in \mathbb{I} \tag{5.9}$$

the prediction

$$y'_{k+s} = y'_{k+s}(s, M) \tag{5.10}$$

is calculated by the M best fitting vectors $\{\tilde{\boldsymbol{D}}_j\}_{j=1,..,M}$ from $\{\boldsymbol{D}_{k-i}\}_{i\in\mathbb{I}}$, found at the positions f_j with $j = 1, .., M$. The $\{\tilde{\boldsymbol{D}}_j\}_{j=1,..,M}$ corresponding to the M best

matches (M smallest δ_i) can be written as

$$\tilde{\boldsymbol{D}}_j = \boldsymbol{D}_{k-f_j} \ . \tag{5.11}$$

The distances are:

$$\tilde{\delta}_j = \|\boldsymbol{D}_k - \tilde{\boldsymbol{D}}_j\|_2 = \delta_{k-f_j} \ . \tag{5.12}$$

The estimation y'_{k+s} for y_{k+s} is calculated as

$$y'_{k+s} = \sum_{j=1}^{M} w_j y_{k-f_j+s} \ . \tag{5.13}$$

The weights w_j are calculated according to:

$$w_j = \frac{1}{N}\frac{1}{\tilde{\delta}_j} \quad \text{with} \quad N = \sum_{j=1}^{M} \frac{1}{\tilde{\delta}_j} \ . \tag{5.14}$$

Figure 5.1 shows the prediction scheme applied to a HST: Matching of the embedding vector $\boldsymbol{D}_{k=930}$ with $p = 4$ and $h = 25$ with the memory of the time series is given. One match ($M = 1$) at $k - f_j = 570$ is used to predict y'_{960}, the prediction interval $s = 30$ (this large prediction interval is chosen for illustration purposes only). The index i is taken from the interval $\mathbb{I} = \{30, .., 854\}$.
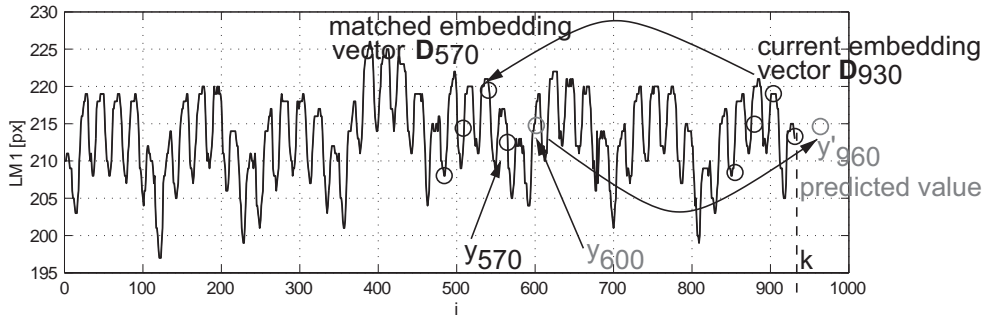

Figure 5.1: Illustration of the local prediction scheme.

Determination of p and h The following paragraphs present an algorithm to determine optimal values for the unknown embedding parameters p and h. In [39] several algorithms for calculating p and h are proposed, such as mutual information or autocorrelation: The autocorrelation $g(h)$ of the time series $\boldsymbol{y}$ is calculated as follows:

$$g(h) = \sum_{k} y_k y_{k-h} \ . \tag{5.15}$$

The autocorrelation function should be (almost) zero at all lags equal or greater than h [39]. Then the redundancy between the components of the embedding vector $\boldsymbol{D}_k$ is least. Unfortunately, this leads to completely uncorrelated elements in the embedding vector $\boldsymbol{D}_k$, which is undesired (see Sec. 5.2.1). As a thumb rule, the time, where the autocorrelation function decays to $1/e$ is chosen as embedding lag h. Unfortunately, the autocorrelation function captures linear correlations only, nonlinear correlations are not taken into account. Therefore, the embedding lag h computed with the autocorrelation function is likely to be too small.

Nonlinear correlation can be computed with the mutual information:

$$I(h) = \sum_{i,j} p_{i,j}(h)\ln(p_{i,j}(h)) - 2\sum_{i} p_i \ln(p_i) \ , \qquad (5.16)$$

with p_i as probability that y_k lies in the i-th bin of the signal's histogram and $p_{i,j}(h)$ as probability that y_k is in bin i and the delayed signal y_{k-h} is in bin j. At the first minimum the optimal lag h has been found: The delayed signal adds maximal information to the knowledged gained from the undelayed signal [39]. Unfortunately, according to [39], the mutual information approach is theoretically well founded for $p = 2$ only.

Therefore, a different solution is proposed here: Since $\boldsymbol{D}_k$ is used for prediction the prediction quality is taken into account to decide if appropriate values for p and h have been chosen. To evaluate the prediction quality, it is necessary to consider not only one predicted value, but m predictions have to be calculated. The estimated mean μ_{err} and the (unbiased) estimation of the standard deviation σ_{err} of the prediction error (of the underlying prediction process)

$$e_l = \|y_l' - y_l\|_2 \qquad (5.17)$$

are calculated to evaluate the prediction. (For simplicity in the following sections mean refers to μ_{err} and standard deviation refers to σ_{err}.) The index l is taken from the interval $l \in \{\bar{n} + s, .., n\}$, with $\bar{n}$ denoting an arbitrarily chosen start point. The mean μ_{err} can be written as:

$$\mu_{\mathrm{err}} = \frac{1}{m} \sum_{l} e_l \ , \qquad (5.18)$$

and the standard deviation σ_{err}:

$$\sigma_{\mathrm{err}}^2 = \frac{1}{m-1} \sum_{l} (e_l - \mu_{\mathrm{err}})^2 \ . \qquad (5.19)$$

In order to compare the prediction quality of different signals, the normalized mean $\hat{\mu}_{\mathrm{err}}$ and the normalized standard deviation $\hat{\sigma}_{\mathrm{err}}$ are used, because they take

the different variability of the signals into account:

$$\hat{\mu}_{\text{err}} \;=\; \frac{1}{m}\sum_l \frac{e_l}{\sigma(\boldsymbol{y})} = \frac{\mu_{\text{err}}}{\sigma(\boldsymbol{y})}\;, \qquad\qquad (5.20)$$

$$\hat{\sigma}_{\text{err}}^2 \;=\; \frac{1}{m-1}\sum_l \left(\frac{e_l}{\sigma(\boldsymbol{y})} - \hat{\mu}_{\text{err}}\right)^2 = \frac{\sigma_{\text{err}}^2}{\sigma^2(\boldsymbol{y})} \qquad\qquad (5.21)$$

and $\sigma(\boldsymbol{y})$ denoting the standard deviation of the considered signal.

Keeping in mind $\hat{\mu}_{\text{err}} = \hat{\mu}_{\text{err}}(p, h, ...)$ and $\hat{\sigma}_{\text{err}} = \hat{\sigma}_{\text{err}}(p, h, ...)$, the parameters p and h are varied to find the best parameters p_{opt} and h_{opt} corresponding to the smallest $\hat{\mu}_{\text{err}}$. A small $\hat{\mu}_{\text{err}}$ indicates a small prediction error and shows that $\boldsymbol{D}_k$ captures the inner states of the time series well.

Advantages For the algorithm presented here, no fitting of model parameters is necessary. Moreover, explicit models describing the trajectories of HSTs, ECG, and RPS signals are not known. The prediction can start as soon as the memory for matching the reference vector is long enough. There is no training necessary, such as with artificial neural networks. The prediction quality is, as shown in subsequent sections, not very sensitive to p and h, so the choice of the embedding parameters p and h is not critical. As the prediction is a weighted average of M values, the influence of noise is reduced.

Disadvantages No upper bound for the prediction error can be given. Nevertheless, the prediction quality with the trajectories considered here is very good. Computation time increases linearly with the length of the memory. Local prediction is limited to a relatively small prediction interval s (up to $s = 5$ for HST prediction, see Sec. 5.2.4) which is sufficient to handle single outliers but makes special care for longer disturbances such as occlusions with surgical instruments necessary. Therefore, the use of additional information is included in the prediction scheme (Sec. 5.3 and Sec. 5.4).

ECG and RPS are used in Sec. 5.4 and Sec. 5.5 for the prediction of HSTs. Since it is not known which inner states of the underlying dynamic systems producing the ECG and RPS are important to predict the HSTs, it has to be guaranteed that all inner states of these dynamic systems are well captured by the corresponding embedding vectors. Therefore, one-step prediction is performed on the trajectories of the ECG and RPS: A small prediction error indicates that the inner states have been captured well. In the following section the local prediction scheme presented here is applied to HSTs. Outlier handling as well as predicting the position of a landmark up to five steps ahead is shown.

5.2.2 ECG Prediction

In Sec. 5.4 and Sec. 5.5 the ECG and RPS signals are used to predict the motion of an occluded landmark. It is not known which inner states of the dynamical system producing the ECG are important for predicting the HST. It has to be guaranteed that all inner states are captured by the embedding vector. Therefore, one-step predictions of the ECG signal are used to find out if all inner states are captured by the corresponding embedding vector. This is ensured by a small prediction error μ_{err}.

A simulated ECG was recorded with a sample frequency of $f_{\mathrm{sample}} = 50$ kHz at the Klinikum Grosshadern (Munich). Here simulated means that the ECG signal was created by an ECG generator, which is usually used to test ECG monitors (ECG signals derived during operation are considered in Sec. 5.3 and Sec. 5.4). Two ECG signals are considered, one with $f_{\mathrm{ECG}} = 60$ bpm (bpm=beats per minute) and the second with $f_{\mathrm{ECG}} = 80$ bpm (see Fig. 5.2). As these signals are simulated, no irregularities, such as atrial flutter or atrial fibrilation, occur. Additionally, they have a perfectly constant frequency, so there are no transient inner states of the dynamical system violating the conditions of the prediction algorithm presented in Sec. 5.2.1. To obtain reasonable computing times, the signals are downsampled, after applying the discrete Fourier transformation and determining the Nyquist frequency. The Nyquist frequency is similar for both signals $f_{\mathrm{Nyquist}} \approx 35$ Hz (Fig. 5.3). As the new sample frequency $f_{\mathrm{sample}} = 80$ Hz is chosen. In the following, all indices of y_i are in $1/f_{\mathrm{sample}}$ time steps.
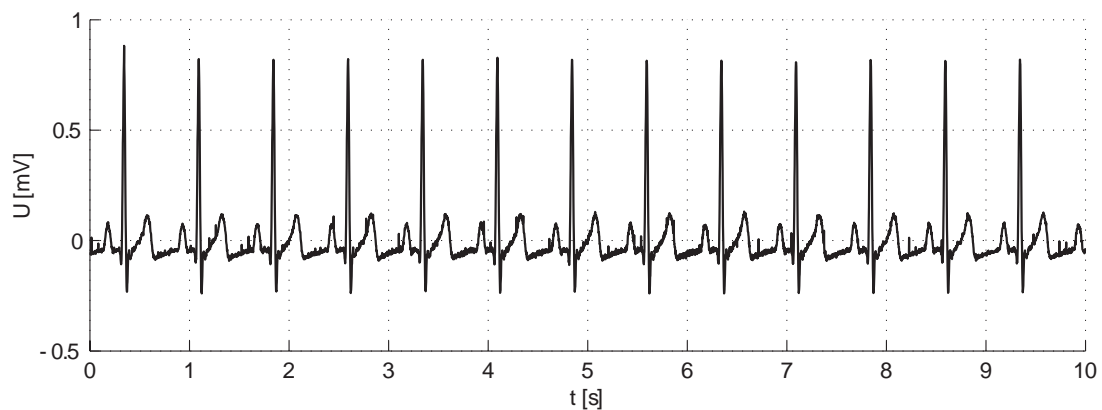

Figure 5.2: Simulated ECG signal, $f_{\mathrm{ECG}} = 80$ bpm.

To capture the dynamics of the signal, p and h have to be chosen in a way such that $\boldsymbol{D}_k$ always contains a characteristic part of the signal. For ECG signals containing long parts with almost constant voltage (Fig. 5.2), this is very important, otherwise the prediction mainly depends on signal noise. The product
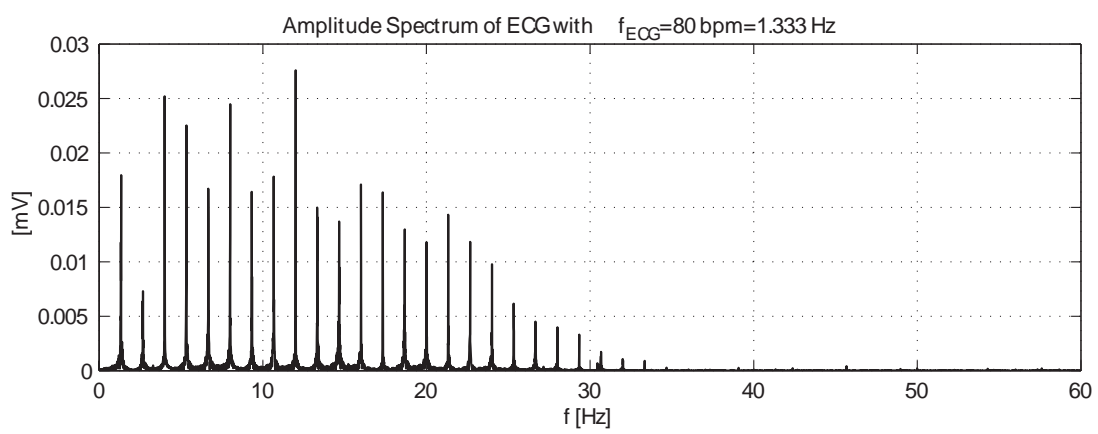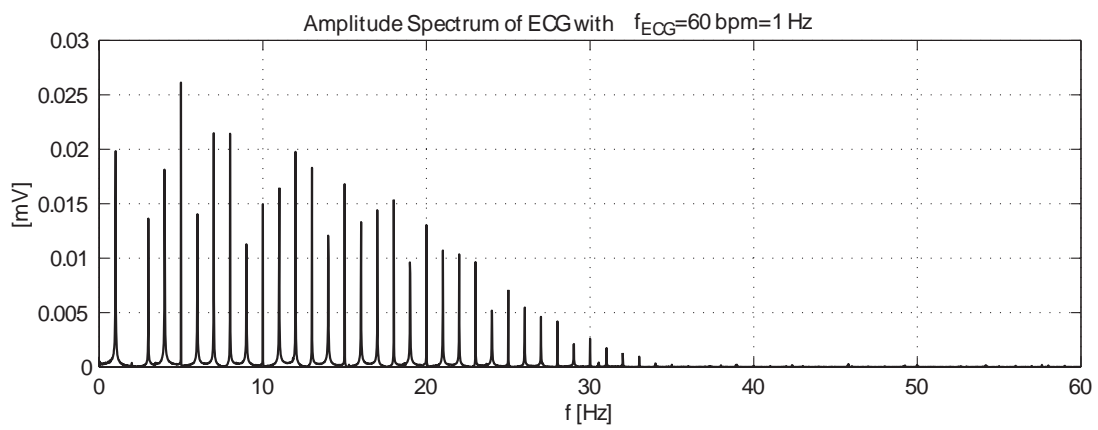
$$H = p \, h \tag{5.22}$$

96

Figure 5.3: Amplitude spectrum of simulated ECG data.

describes how far in the past the elements used for building the reference vector $\boldsymbol{D}_k$ are lying. See Fig. 5.4 for a good and a bad choice of p and h.
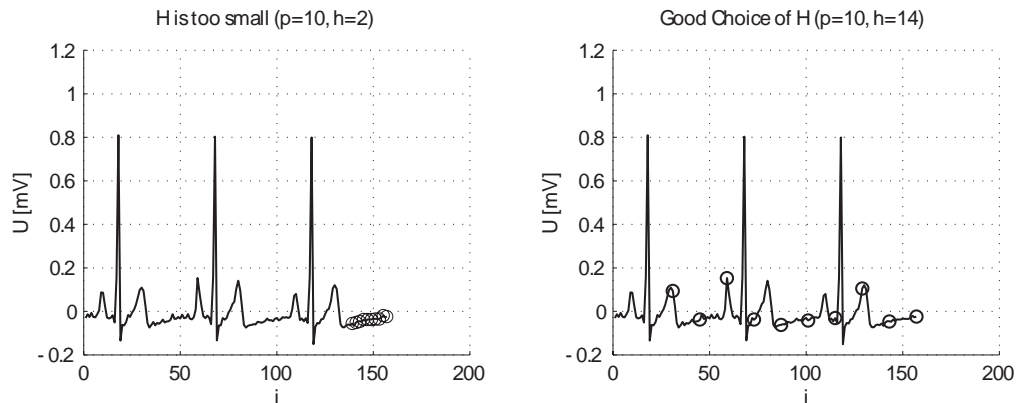

Figure 5.4: Example of bad (left) and good (right) embedding vector $\boldsymbol{D}_k$.

In Fig. 5.5 the performance for different p and h measured by μ_{err} and σ_{err} is summarized for the simulated ECG signal. It can be seen that the embedding dimension $p = 4$ is too small to capture all the underlying dynamics, whereas $p = 8$ leads to much better predictions; increasing p further does not lead to significant better results. This is due to the fact that embedding vectors with dimension $p = 8$ capture the inner states very well, so increasing p does not lead to additional information improving the quality of the prediction. A large error occurs for $h = 30$, almost independent of p: the elements of $\boldsymbol{D}_k$ are chosen in a way such that nearly no dynamics is included; therefore the prediction fails (see Fig. 5.6).

The optimal parameters which lead to the smallest prediction error μ_{err} for an ECG with $f_{\mathrm{ECG}} = 80$ bpm are:

$$p_{\mathrm{opt}} = 8 \quad \text{and} \quad h_{\mathrm{opt}} = 14 \qquad (5.23)$$

and for an ECG with $f_{\mathrm{ECG}} = 60$ bpm (not shown here):

$$p_{\mathrm{opt}} = 8 \quad \text{and} \quad h_{\mathrm{opt}} = 20 \ . \qquad (5.24)$$

The embedding dimension p_{opt} is equal for both ECG signals considered. This is due to the fact that the structure of the underlying dynamical system which produces the ECG signal is equal and only the parameters of the dynamical system differ. The product $f_{\mathrm{ECG}} h_{\mathrm{opt}}$ is similar for both ECG frequencies: As the ECG frequency increases the embedding lag decreases and vice versa. The measurements which are contained in the embedding vectors are therefore similar for both cases.

Figure 5.7 shows the results of a one-step prediction with the optimal embedding parameters for $f_{\mathrm{ECG}} = 80$ bpm. It can be seen, that the prediction quality is very
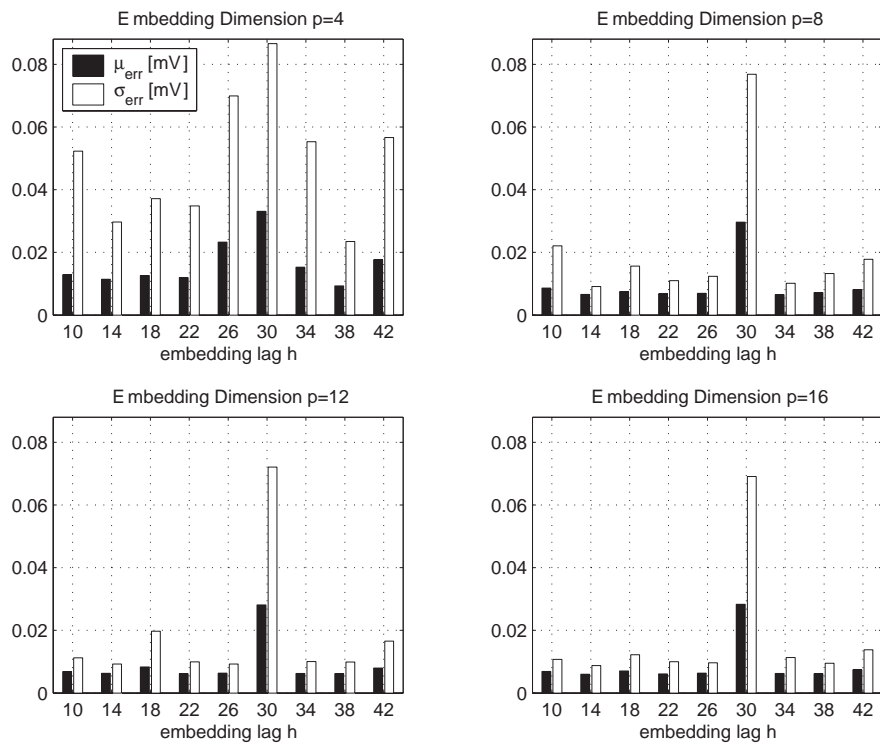
Figure 5.5: Evaluation of μ_{err} and σ_{err} on a simulated ECG with $f_{\mathrm{ECG}} = 80$ bpm for different p and h. Parameters: $f_{\mathrm{sample}} = 80$ Hz, $s{=}1$, $M{=}10$, $\bar{n}{=}3680$, $n{=}4370$.
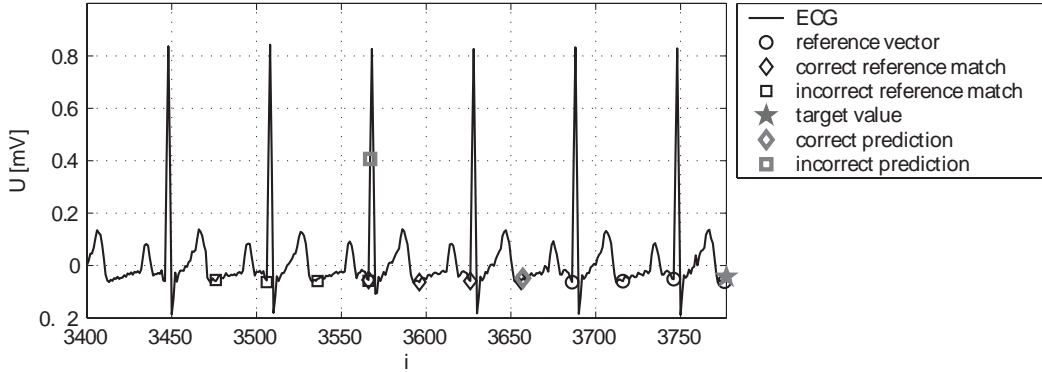
Figure 5.6: Illustrating the problem of choosing h=30 for an ECG with $f_{\mathrm{ECG}} = 80$ bpm. Parameters: $s = 1$, $p = 4$, $h = 30$, $\bar{n}$=3777.

good, or in other words, that the underlying dynamics is captured very well. The results of $f_{\mathrm{ECG}} = 60$ bpm are similar and therefore not presented here.

5.2.3   RPS Prediction

The ECG and RPS signals are used to predict the motion of occluded landmarks. To build the prediction algorithm described in Sec. 5.4 and Sec. 5.5 all inner states of the dynamical system producing the RPS have to be captured by the corresponding embedding vector. One-step predictions of the RPS are calculated to determine if the inner states have been captured well and to calculate the optimal values of p and h for the embedding vector. Therefore, data (see Fig. 5.8) was recorded at the Klinikum Grosshadern (Munich), using an artificial lung. A sample frequency of $f_{\mathrm{sample}} = 40$ Hz is sufficient, because the Nyquist frequency can be determined with $f_{\mathrm{Nyquist}} = 1.3$ Hz. The RPS has a frequency of about $f_{\mathrm{RPS}} = 20$ cpm (cpm=cycles per minute).

Figure 5.9 shows the mean error μ_{err} and the standard deviation σ_{err} (according to Sec. 5.2) for different parameter combinations of p and h. The prediction error does not strongly depend on these parameters. This leads to the conclusion that $p = 5$ is already sufficient to capture the inner states of the dynamic system producing the RPS very well. Increasing p leads to a slightly better prediction of the RPS, because more y_i are included in the embedding vector $\boldsymbol{D}_k$ and therefore the influence of noise while calculating the M best fitting vectors is reduced.

The results of the one-step prediction with the optimal parameters $p_{\mathrm{opt}} = 10$ and $h_{\mathrm{opt}} = 20$ are given in Fig. 5.8. The main prediction errors occur at heavy oscillations. These oscillations are due to mechanical properties of the artificial lung. Fortunately, these oscillations do not occur, if RPS recorded during operation is considered (Sec. 5.4).
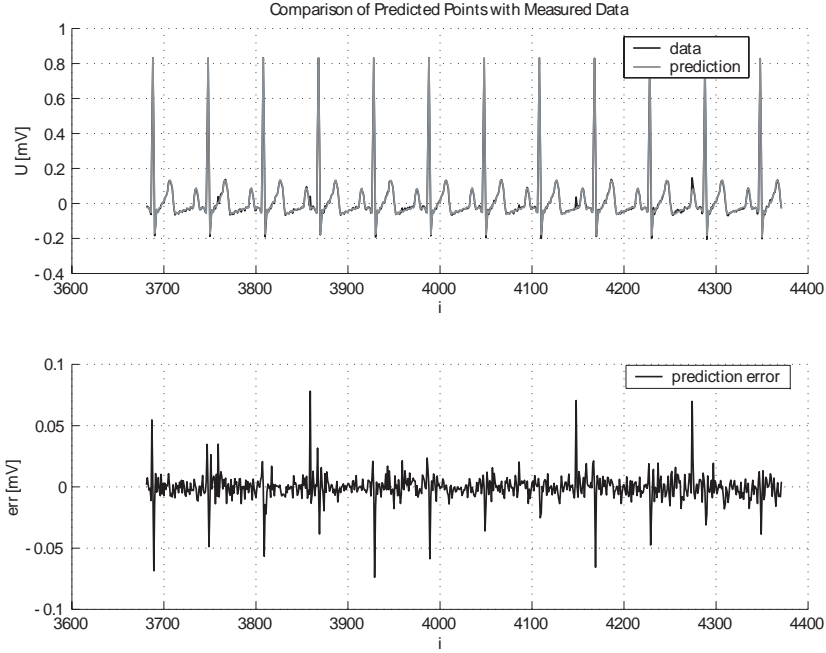
Figure 5.7: Good prediction of a simulated ECG. Parameters: $f_{\mathrm{ECG}} = 80$ bpm, $f_{\mathrm{sample}} = 80$ Hz, $s=1$, $M=10$, $p=8$, $h=14$, $\bar{n}=3680$, $n=4370$.

5.2.4 HST Prediction

Prediction of the heart surface trajectory (HST) is essential for improving the reliability of the tracking scheme (Chap. 4). With one-step predictions it is possible to detect outliers and to position the search area of the tracker. The results presented here are based on trajectories already considered in Sec. 4.5.8. Results of different HSTs, computed from other operations are similar and are shown in Sec. 5.4.

The sample rate $f_{\mathrm{sample}} = 25$ Hz corresponds to the frame rate of frame grabbers. As it can be seen in Fig. 5.10 (here $p = 7$ and $h = 4$ is used), the local prediction scheme works well on HSTs. Nevertheless, in case of outliers the quality heavily deteriorates and large disturbances occur (for $i = 855, .., 865$). This is due to the mismatch of the reference vector $\boldsymbol{D}_k$, which includes outliers, with the memory $\boldsymbol{D}_{k-i}$ of the algorithm. Additionally, outliers themselves are unpredictable and therefore increase the prediction error μ_{err}. Values for μ_{err} and σ_{err} are given in units of pixels [px] and can be seen in Tab. 5.1.

The prediction algorithm has to detect and replace outliers reliably. As result, prediction accuracy is increased, because outliers are removed from the memory $\boldsymbol{D}_{k-i}$ and from the embedding vector $\boldsymbol{D}_k$. The following scheme detects and replaces outliers reliably: If the prediction error e_k is larger than a given threshold C, then y_k is regarded as an outlier and replaced by the predicted value y'_k. An
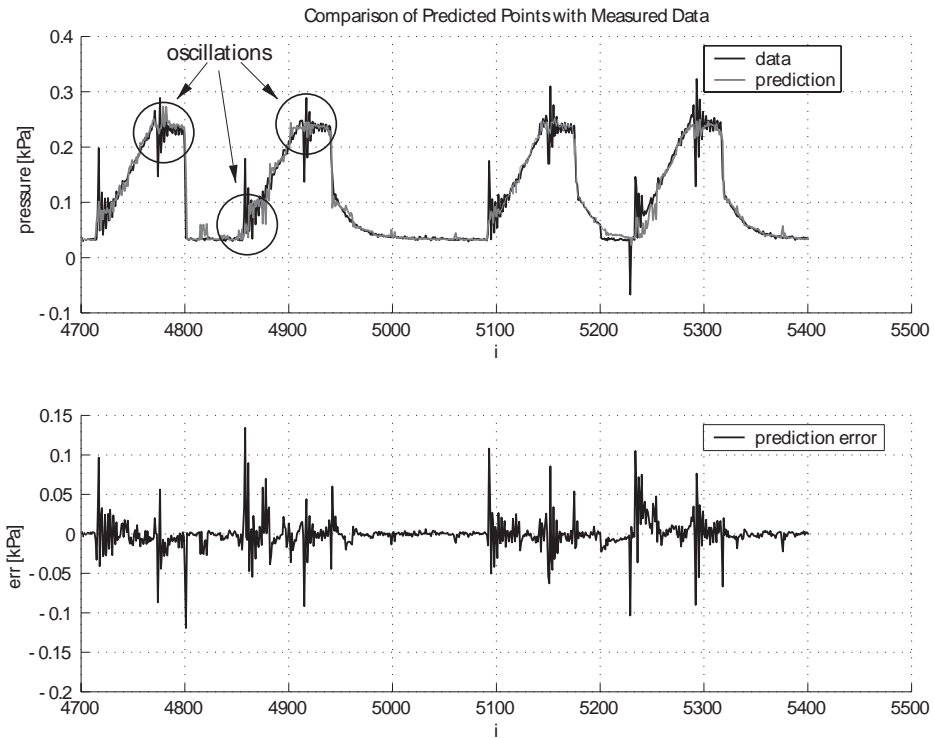
Figure 5.8: Good prediction of RPS. Parameters: $f_{\text{sample}} = 40$ Hz, $s = 1$, $M = 10$, p=10, h=20, $\bar{n} = 4700$, $n = 5400$.
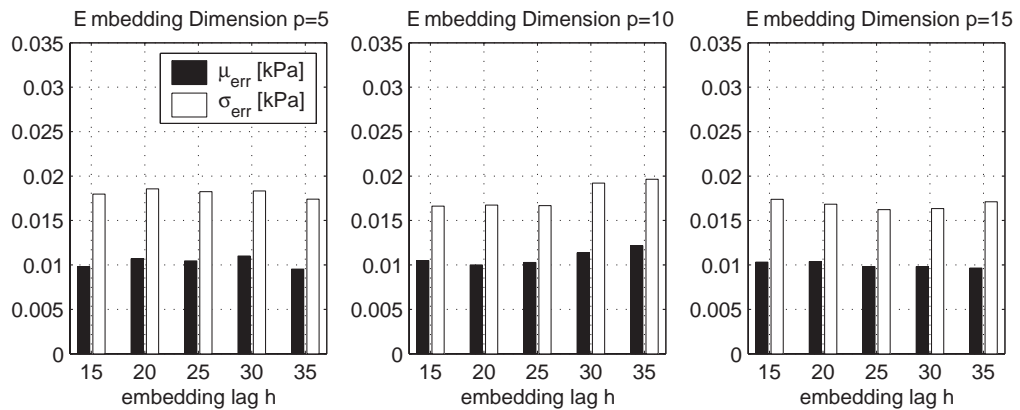

Figure 5.9: Evaluation of μ_{err} and σ_{err} for different p-h-combinations. Parameters: $f_{\text{sample}} = 40$ Hz, $s = 1$, $M = 10$, $\bar{n} = 4700$, $n = 5400$.
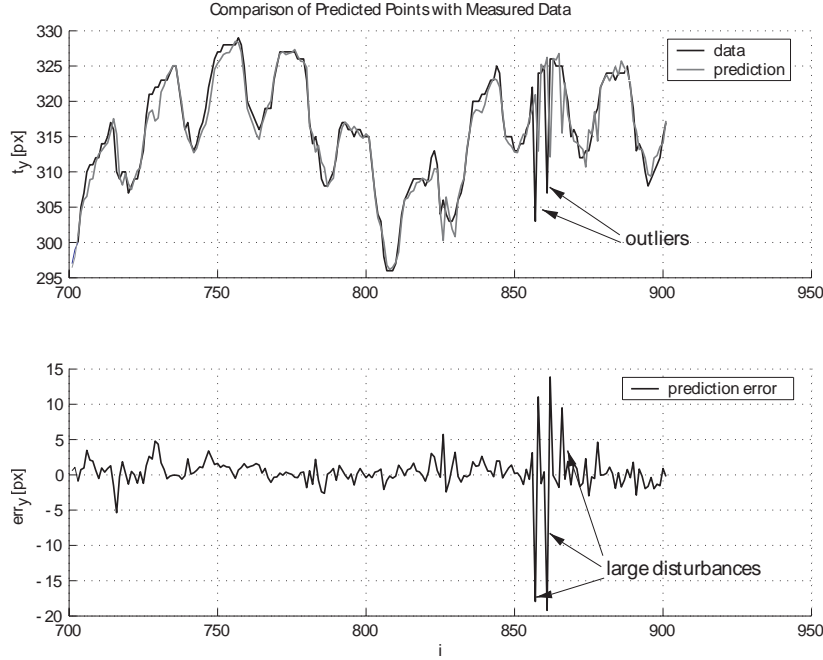
Figure 5.10: Prediction of HST without outlier detection. Parameters: $f_{\text{sample}} = 25$ Hz, $s = 1$, $M = 10$, $p=7$, $h=4$, $\bar{n} = 700$, $n = 900$.

appropriate value for C is the standard deviation (a measure for the variability of HSTs) of the past tracking results. If this scheme is applied to the same trajectory as before, μ_{err} and σ_{err} are reduced (see Tab. 5.1). Handling not only single outliers, but longer disturbances is shown in Sec. 5.3.

Table 5.1: Comparison of prediction quality for HST.

Settings	μ_{err}	σ_{err}
no outlier detection	1.42 px	2.37 px
with outlier detection	1.05 px	0.97 px

Prediction with outlier detection ($p_{\text{opt}} = 7$ and $h_{\text{opt}} = 4$) in Fig. 5.11 shows that no large disturbances occur. Results of different p and h combinations are given in Fig. 5.12. The sensitivity of the prediction quality μ_{err} is, within a certain range for p and h, very low, and almost all parameter combinations lead to small prediction errors. This indicates that good values for p and h were found: Once a sufficiently long vector $\boldsymbol{D}_k$ has been built, adding new components to the vector does not lead to additional information.

It is possible to predict HSTs further than for one step only, but as the prediction interval s increases, prediction quality degrades (Tab. 5.2). This happens because the correlation between the last valid point y_k and the point y_{k+s} to be predicted
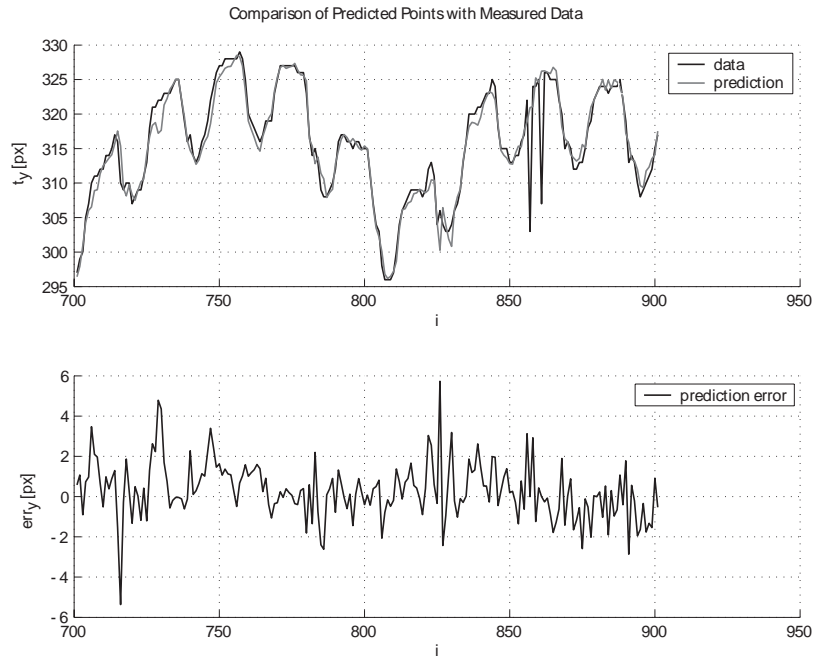
103

Figure 5.11: Prediction of HST with outlier detection. Parameters: $f_{\mathrm{sample}} = 25$ Hz, $s = 1$, $M = 10$, p=7, h=4, $\bar{n} = 700$, $n = 900$.

Figure 5.12: Prediction of HST with outlier detection for different p and h. Parameters: $f_{\mathrm{sample}} = 25$ Hz, $s = 1$, $M = 10$, $\bar{n} = 700$, $n = 900$.

degrades. The global scheme given in Sec. 5.3 avoids this drawback, by taking into account additional information. Results of a five-step prediction ($s = 5$) with $p_{\text{opt}} = 12$ and $h_{\text{opt}} = 2$ providing the minimal prediction error μ_{err} can be seen in Fig. 5.13. Results of different combinations of p and h are summarized in Fig. 5.14.

Table 5.2: Comparison of more-step prediction quality for HST.

Parameter	μ_{err}	σ_{err}
$s = 1$, $h_{\text{opt}} = 4$, $p_{\text{opt}} = 7$	1.05 px	0.97 px
$s = 5$, $h_{\text{opt}} = 2$, $p_{\text{opt}} = 12$	1.49 px	1.31 px
$s = 5$, $h = 4$, $p = 7$	1.51 px	1.21 px

If the optimal values of the one-step prediction are used for the five-step prediction ($p_{\text{opt}} = 7$ and $h_{\text{opt}} = 4$), then the prediction quality is only slightly degraded (see Tab. 5.2). This leads to the conclusion that μ_{err} does not strongly depend on the chosen embedding parameters p and h. This eases the determination of p_{opt} and h_{opt} in realtime applications, because the same values for p and h can be used for different prediction intervals s without losing too much accuracy.

Figure 5.13: Five-step prediction of HST with outlier detection. Parameters: $f_{\text{sample}} = 25$ Hz, $s = 5$, $M = 10$, p=12, h=2, $\bar{n} = 700$, $n = 900$.
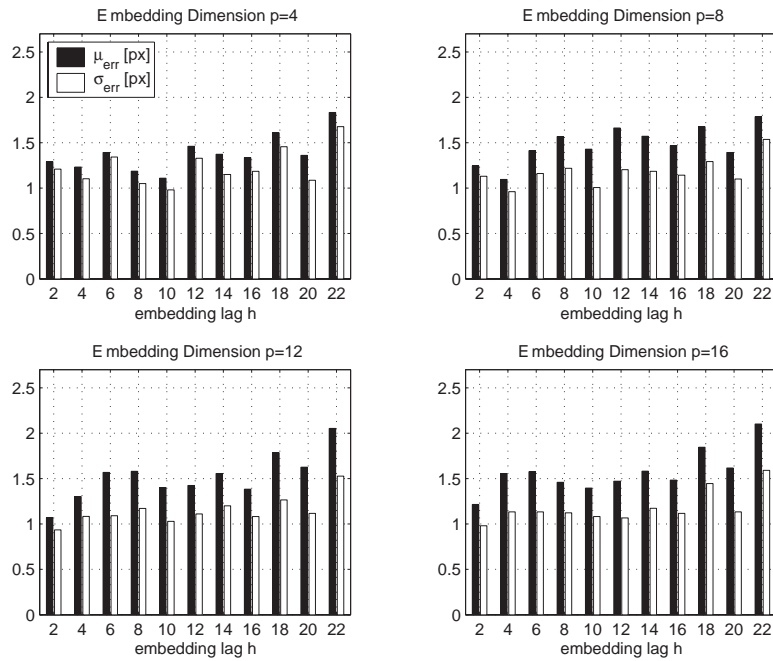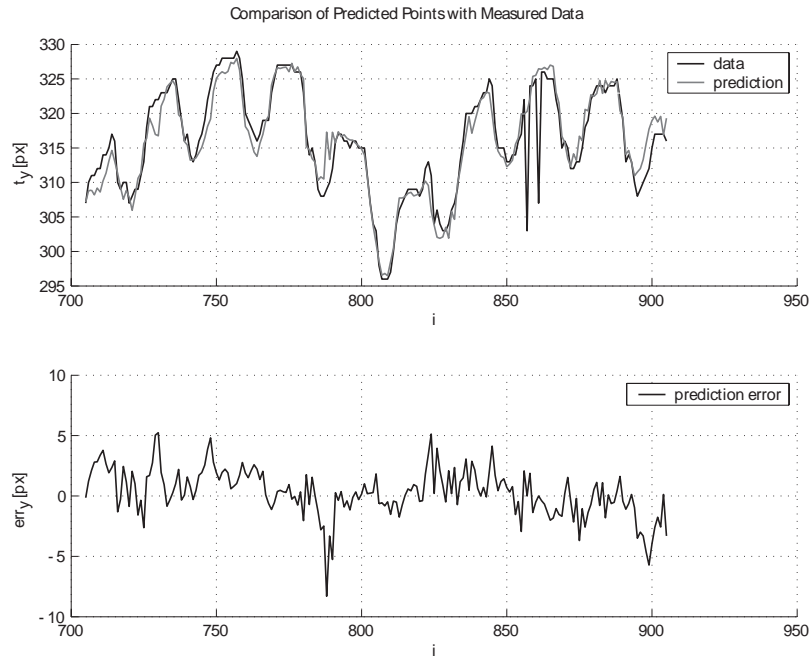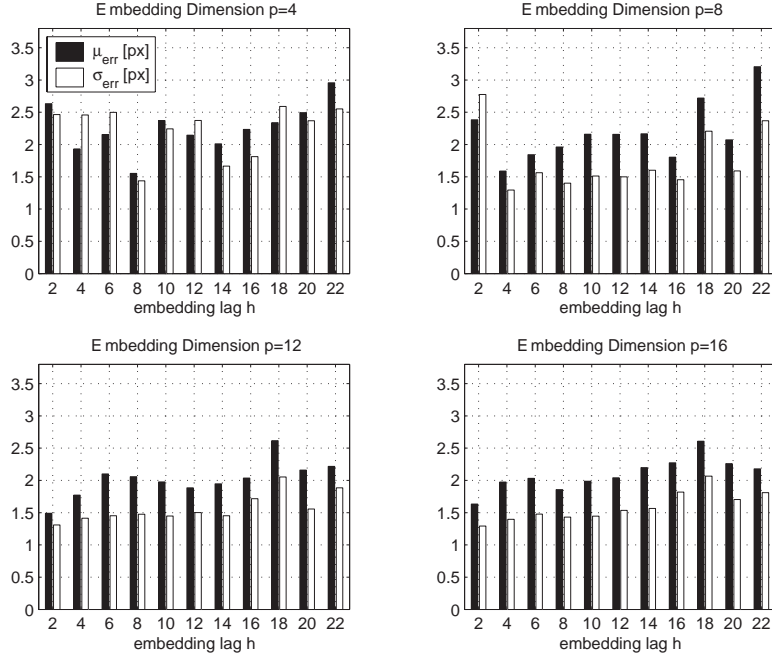
Figure 5.14: Prediction of HST with outlier detection for different p and h. Parameters: $f_{\mathrm{sample}} = 25$ Hz, $s = 5$, $M = 10$, $\bar{n} = 700$, $n = 900$.

5.2.5 Conclusion

It was shown that local prediction for all three types of trajectories (ECG, RPS and HST) is possible. The algorithm presented in Sec. 5.2.1 is therefore able to capture the inner states of these signals. The importance of a one-step or more-step prediction for the HST is evident (outlier detection, positioning of the search area for the tracking scheme). The use of ECG and RPS for the prediction of HSTs will be shown in Sec. 5.4.

Table 5.3: Comparison of prediction quality.

signal	$\hat{\mu}_{\mathrm{err}}$	$\hat{\sigma}_{\mathrm{err}}$
ECG $f_{\mathrm{ECG}} = 60$ bpm	$60 \cdot 10^{-3}$	$79 \cdot 10^{-3}$
ECG $f_{\mathrm{ECG}} = 80$ bpm	$54 \cdot 10^{-3}$	$75 \cdot 10^{-3}$
RPS $f_{\mathrm{RPS}} = 20$ cpm	0.11	0.18
HST $s = 1$, $p_{\mathrm{opt}} = 7$, $h_{\mathrm{opt}} = 4$	0.12	0.11
HST $s = 5$, $p_{\mathrm{opt}} = 12$, $h_{\mathrm{opt}} = 2$	0.17	0.15
HST $s = 5$, $p = 7$, $h = 4$	0.18	0.14

Table 5.3 summarizes the results of predicting different trajectories. The increased values for $\hat{\mu}_{\mathrm{err}}$ and $\hat{\sigma}_{\mathrm{err}}$ when predicting the RPS signal are mainly due

to the heavy oscillations of the RPS signal. Additionally, the simulated ECG signal is perfectly periodic, whereas the RPS signal is not. Predicting the HSTs further than one step degrades the prediction quality, whereas μ_{err} and σ_{err} are quite insensitive to changes of p and h.

In general, it is better to increase p when the embedding vector is built. This degrades $\hat{\mu}_{\mathrm{err}}$ only slightly, as shown in the previous sections. The advantage is that increasing p allows a more robust matching of $\boldsymbol{D}_k$ with the memory of the trajectory: The influence of disturbances, which might be included either in $\boldsymbol{D}_k$ or in the memory $\boldsymbol{D}_{k-i}$, is reduced, because the matching is based on more data points. Nevertheless, p cannot be arbitrarily large due to computation time limitations and the quasi-periodic properties of the HSTs. A too large $H = p\,h$ leads to uncorrelated data in $\boldsymbol{D}_k$ and therefore to inaccurate matching with the memory.

5.3 Global Prediction of HSTs

The quality of the local prediction degrades with increasing prediction interval s, so other solutions to bridge longer occlusions have to be found.

Usually, not the entire workspace is occluded (e.g. by an instrument), but only parts of it. Whereas one or a few landmarks are not visible, other landmarks still are. The motion of the visible landmarks can be used to estimate the unknown motion of the occluded ones. The main advantage of this global prediction scheme, compared to the local one presented in Sec. 5.2, is that in case of longer disturbances always new information about the heart motion is available. As a drawback, several landmarks have to be tracked simultaneously. This leads to an increased amount of computation time. Some details of this section have been published by the author in [52, 53].

5.3.1 Algorithm

The algorithm presented here has two prerequisites: The conditions shown in Sec. 5.2.1 have to be fulfilled for each landmark. Additionally, the embedding vectors of the landmarks have to be functionally coupled. This guarantees a unique mapping from the embedding vectors of the visible landmarks to the (unknown) embedding vector of the covered landmark. The mapping itself is computed with past observations of the trajectories. As all landmarks lie on one organ and move with the same frequencies, this is fulfilled and can be easily justified with the trajectories considered here.

Instead of one landmark as in Sec. 5.2.1, N landmarks are considered simultane-

ously. Their time series can be written as

$$\boldsymbol{y}^r = \left[y_n^r, y_{n-1}^r, ..., y_1^r\right]^T \quad \text{with} \quad 1 \le \text{r} \le \text{N} . \tag{5.25}$$

For each landmark the embedding vector $\boldsymbol{D}_k^r$ with

$$\boldsymbol{D}_k^r = \left[y_k^r, y_{k-h}^r, ..., y_{k-(p-1)h}^r\right]^T \tag{5.26}$$

can be built. As before, p denotes the embedding dimension and h the embedding lag. Prediction quality is not very dependent on the embedding parameters p and h (see Sec. 5.2.4), so they are chosen equal for all N landmarks. The measurements y_i considered here, are not scalar-valued any longer, but contain x-position $(t_{x,i})$ and y-position $(t_{y,i})$ of the tracked landmark:

$$y_i = \left[t_{x,i}, t_{y,i}\right]^T . \tag{5.27}$$

The prediction algorithm presented in Sec. 5.2.1 can be extended to vector-valued measurements easily and is applied in the same way as it is to scalar-valued measurements. The outlier detection given in Sec. 5.2.4 is applied to each component of y_i separately. If an outlier is detected in one direction (either x- or y-direction), then the landmark is predicted in both directions.

Assuming that landmark a is occluded, then the trajectories of the other (unoccluded) landmarks are used to estimate the current position of landmark a: Suppose the landmark was lost at time k_0, then a part of the time series of each landmark which is sufficiently long to allow matching of the corresponding embedding vector $\boldsymbol{D}_k^r$ (see Sec. 5.2.1), has to be saved:

$$\boldsymbol{y}_{k_0}^r = \left[y_{k_0-1}^r, y_{k_0-2}^r, ..., y_{k_0-1-l}^r\right]^T \tag{5.28}$$

with

$$l \le k_0 - 2 . \tag{5.29}$$

The embedding vectors $\boldsymbol{D}_k^r$ for the visible landmarks have to be built:

$$\boldsymbol{D}_k^r = \left[y_k^r, y_{k-h}^r, ..., y_{k-(p-1)h}^r\right]^T \tag{5.30}$$

with

$$1 \le r \le N \quad \text{and} \quad \text{r} \ne \text{a} . \tag{5.31}$$

As in Sec. 5.2.1 these embedding vectors have to be matched with their memory

$$\boldsymbol{D}_{k_0-i}^r \quad \text{with} \quad \text{i} = 1, 2, .., \text{l} - (\text{p} - 1)\text{h} + 1 \tag{5.32}$$

built from the corresponding time series $\boldsymbol{y}_{k_0}^r$ and the M best fits for each visible landmark have to be searched. These M best fits are found at the positions f_j^r and can be given as

$$y_{k_0-f_j^r}^r . \tag{5.33}$$

Their weights w_j^r are computed as shown in Eq. 5.14. Since some HSTs might provide better results in predicting the position of the lost landmark y^a, weights u^r for each valid landmark are introduced:

$$\sum_{r=1,r\neq a}^{N} u^r = 1 \quad \text{and} \quad u^r \in \mathbb{R} \; . \tag{5.34}$$

To reflect the individual prediction reliability, the weight u^r may depend on the prediction quality $\mu_{\text{err}}^{a,r}$ of the position of the lost landmark y^a, where y^r (with $r \neq a$) denotes a particular visible landmark used for prediction. Another possibility to determine the weights u^r is to calculate the distances between the visible landmarks and the lost landmark, which takes the spatial correlation into account. Finally, the outlier measures of the HSTs of the landmarks used for prediction reflect the reliability of these HSTs and are therefore another appropriate measure to determine u^r.

The estimation for

$$y_{k_0+s}^a \quad \text{with} \quad s \geq 0 \tag{5.35}$$

can be calculated by:

$$y_{k_0+s}^a = \sum_{r=1,r\neq a}^{N} u^r \left(\sum_{j=1}^{M} w_j^r y_{k_0-f_j^r}^a \right) \; . \tag{5.36}$$

Figure 5.15 illustrates the prediction scheme for two landmarks (for one dimension only), where LM1 remains valid over the tracking sequence and LM2 is lost at $k_0 = 750$.

This estimation is based on current values contained in the embedding vectors $\boldsymbol{D}_k^r$ of the valid landmarks, no estimation error accumulates. Therefore, this method is suitable for long-term estimations. Of course not only one landmark might be lost; the scheme presented here works also if several landmarks are disturbed simultaneously.

The global prediction scheme is now combined with the local one given in Sec. 5.2. The Petri-net describing the different discrete states of this scheme as well as the transition conditions are given in Fig. 5.16. The following paragraph describes the Petri-net in detail:

The net starts at state s_{OK}: For each landmark considered a local one-step prediction in x- and y-direction is computed and the result is compared to the measured position of the landmark. If an outlier is detected according to Sec. 5.2.4, with C being the standard deviation of the past tracking results, then transition t_1 is fulfilled and the new state of the net is s_{OUT}. The measured position (which is an outlier) is replaced by the predicted position. Transition t_2 is satisfied, if less than n_{OUT} subsequent outliers occur; the new state is s_{OK} (in the implementation used: $n_{\text{OUT}} = 5$). If more than n_{OUT} subsequent outliers are detected, then
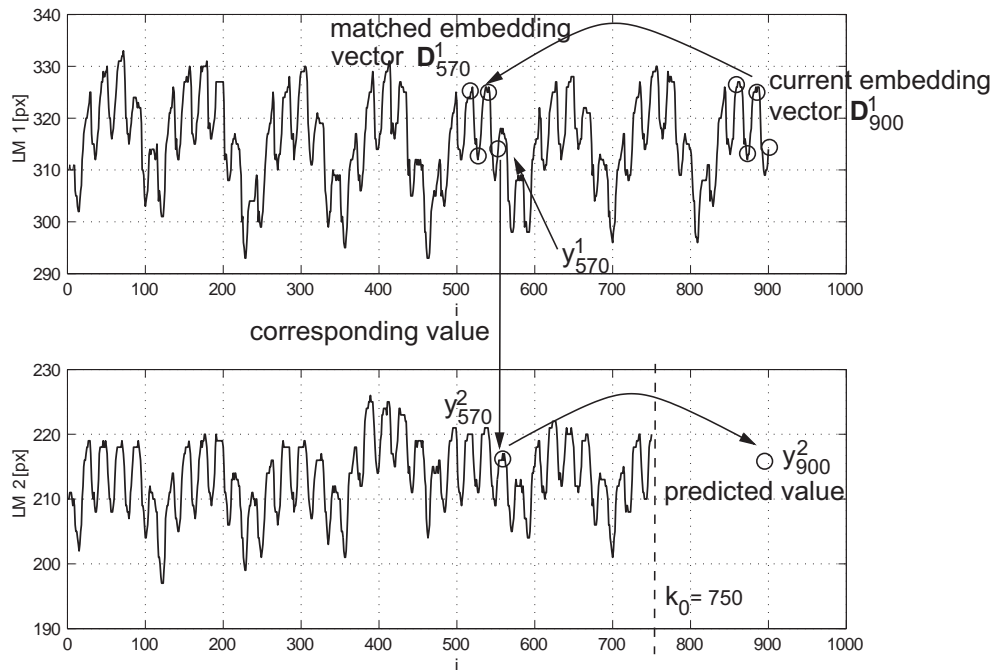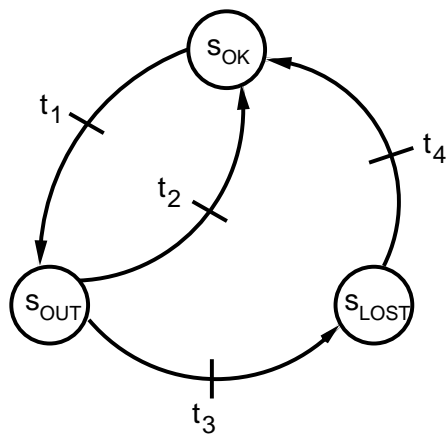
Figure 5.15: Illustration of the global prediction scheme.

Figure 5.16: Petri-net describing the prediction scheme.

transition t_3 is fulfilled, and the new state is s_{LOST}: The landmark is regarded as lost and the global estimation scheme, as described above, is applied. Transition t_4 requires, that n_{OK} subsequent measurements are no outliers (in the implementation used: $n_{\mathrm{OK}} = 15$). If this is fulfilled the landmark is regarded as valid and the new state of the Petri-net is s_{OK} again.

5.3.2 Experiments

This section shows experimental results of a performance evaluation of the algorithm presented above, using landmarks LM1, LM2 and LM8 as already presented in Fig. 4.10. Landmark 8 (lying between LM1 and LM2) is disturbed by three subsequent outliers at $i_{\mathrm{OUT}} \in \{825, 826, 827\}$ and during $i_{\mathrm{LOST}} \in \{835, .., 875\}$ the trajectory is disturbed for almost 2 seconds (41 frames), as it happens during short occlusions by a surgical instrument.

Estimation for LM8 itself is given in Fig. 5.17, additionally the states of the Petri-net are shown. The weights $u^1 = u^2 = 1/2$ are chosen equally, such that both landmarks have the same influence on the prediction of LM8. It can be seen that the estimation of the x-direction is better than of the y-direction. This is confirmed by the normalized mean prediction error $\hat{\mu}_{\mathrm{err,x}} = 0.28$ and the corresponding normalized standard deviation $\hat{\sigma}_{\mathrm{err,x}} = 0.25$ in x-direction and $\hat{\mu}_{\mathrm{err,y}} = 0.39$ and $\hat{\sigma}_{\mathrm{err,y}} = 0.46$ in y-direction, respectively. In order to increase the robustness of the prediction scheme with respect to noise, the embedding dimension p and the embedding lag h are chosen higher than in Sec. 5.2.5. This leads to the fact that more points of the trajectories are contained in the corresponding embedding vectors $\boldsymbol{D}_k^1$ and $\boldsymbol{D}_k^2$.

Figure 5.18 shows the prediction error in detail, as well as the states of the described Petri-net. The three subsequent outliers at $i_{\mathrm{OUT}} \in \{825, 826, 827\}$ are correctly detected, indicated by the state s_{OUT}, followed by s_{OK}. After that, disturbance occurs for 41 frames, starting at $i = 835$: For the next $n_{\mathrm{OUT}} = 5$ steps, the state of the net is s_{OUT} and the local prediction is used for estimation. The global prediction is applied afterwards and the state of the net becomes s_{LOST}. If at least 15 subsequent correct estimations occur, the net switches back to s_{OK}. In this example transition t_4 is violated for a longer period, because of small prediction errors, occuring after $i = 875$.

The experimental results show (together with long-term estimations calculated on different trajectories from other operations as shown in Sec. 5.4.2) that long-term prediction of the motion of natural landmarks lying on the beating heart is possible, even for 40 frames and more. This is an important step towards reliable motion compensation in MIRCS.

In order to increase the reliability of the estimation procedure, it is useful to track parts of the mechanical stabilizer. As the stabilizer is a rigid body, an
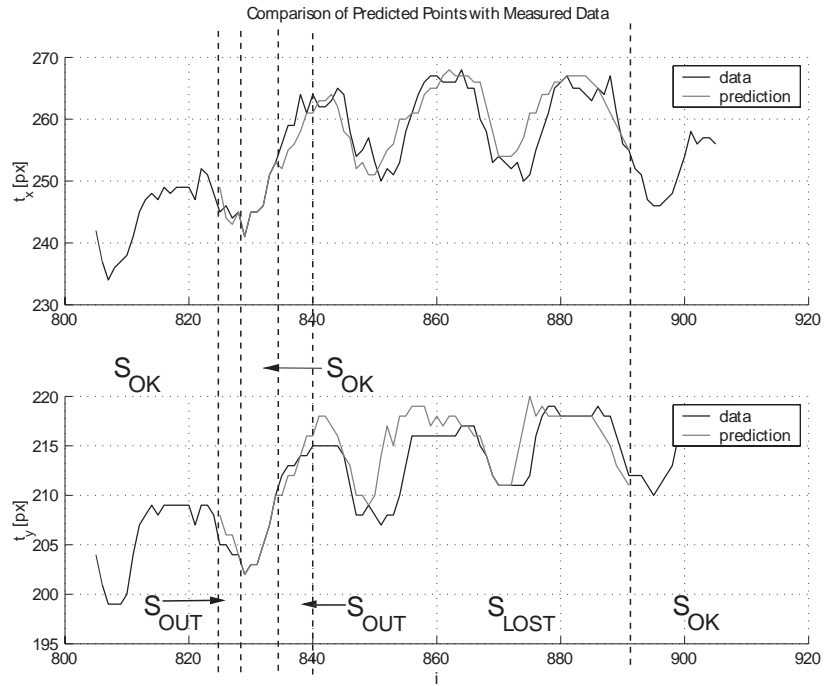
Figure 5.17: Prediction of landmark 8. Parameters: $f_{\mathrm{sample}} = 25\mathrm{Hz}$, p=22, h=13, M=3, $\bar{n}$=750, n=931.
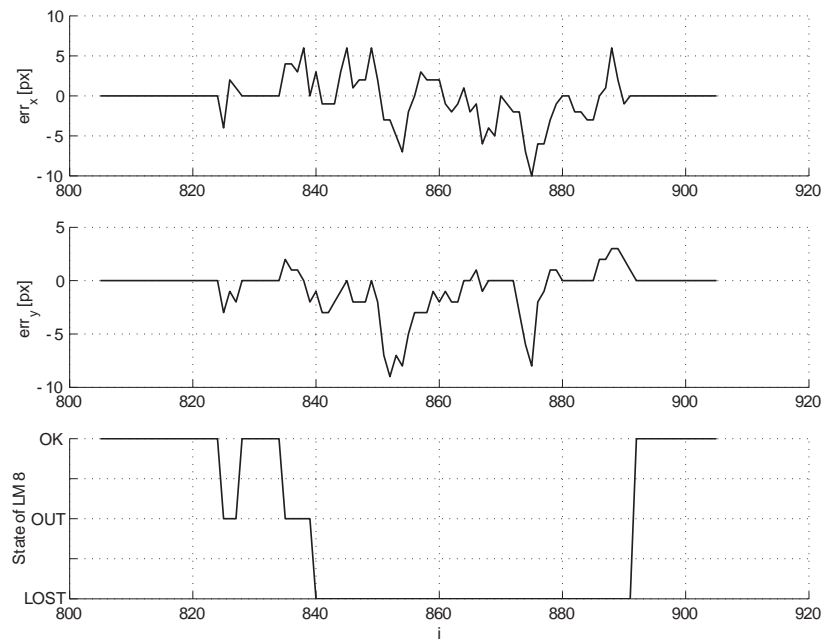

Figure 5.18: Prediction error of landmark 8.

affine motion model is well suited and thus tracking is quite robust. These land-marks can be used to estimate the motion of landmarks lying inside the stabilized area, when they are e.g. covered by an instrument. Here, as in the experiment presented above, a short monitoring of all landmarks is necessary, before the long-term estimation can be applied. The landmarks considered (small rectan-gles) can be seen in Fig. 5.19. The landmarks located on the upper end of the stabilizer (LM10) and on the lower end (LM11) are used to estimate the motion of the landmark inside (LM8). Additionally, the search space of the tracking area is shown (big rectangle). The disturbances as well as the parameters of the Petri-net are identical to the experiment presented before.
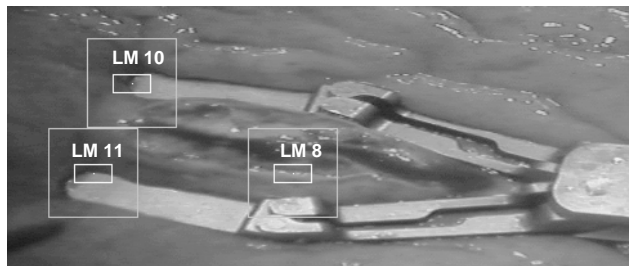

Figure 5.19: Position of landmarks.

The normalized mean prediction error in x-direction is $\hat{\mu}_{\mathrm{err,x}} = 0.25$ and the normalized standard deviation is $\hat{\sigma}_{\mathrm{err,x}} = 0.21$. The corresponding values in y-direction are $\hat{\mu}_{\mathrm{err,y}} = 0.31$ and $\hat{\sigma}_{\mathrm{err,y}} = 0.31$. These values are better than those of the experiment presented before (see Tab. 5.4). This is due to the robust tracking of the motion of the mechanical stabilizer, which can be performed with higher accuracy than tracking of natural landmarks lying on the heart surface. To increase reliability of the tracking further, color markers, as they are used for automatic camera guidance in MIS, seem to be well suited [2, 79].

Table 5.4: Comparison of global prediction of LM8.

	$\hat{\mu}_{\mathrm{err}}$	$\hat{\sigma}_{\mathrm{err}}$
No stabilizer used		
LM8 t_x	0.28	0.25
LM8 t_y	0.39	0.46
Stabilizer used		
LM8 t_x	0.25	0.21
LM8 t_y	0.31	0.31

The next section (Sec. 5.4) presents a long-term prediction scheme that makes use of additional signals which are correlated with the heart motion. Using additional

signals avoids the drawback of being dependent on visual information only when estimating the heart motion.

5.4 ECG and RPS based Prediction

In this section it is shown, how ECG and RPS signals can be used to calculate a robust long-term prediction of HSTs. Therefore, it has to be ensured that HSTs and RPSs as well as HSTs and ECG signals are correlated, which is investigated first. The prediction algorithm itself is shown afterwards.

5.4.1 Correlation of Data

In Sec. 4.5.8 it is assumed that the two dominant frequencies of the HSTs are due to the patient's respiration and heart beat. To verify this assumption, the ECG signal, the RPS data as well as the video stream of the laparoscope were recorded simultaneously during a beating heart surgery performed at the University Hospital Grosshadern (GH) in Munich[1].

As the patient has two pacemakers, the ECG signal given in Fig. 5.20 shows an additional peak per heart-beat and looks therefore different if compared to a standard ECG. Heart motion induced by pacemakers is very regular, which leads to good prediction results. Figure 5.20 shows, besides the ECG signal, the RPS signal as well as the translational motion of a tracked landmark (t_x and t_y). It can be seen that motion in x-direction (t_x) is dominated by the ECG frequency, whereas motion in y-direction (t_y) shows the influence of both signals, ECG and RPS, respectively. The shape of the trajectories is also influenced by the mechanical stabilizer, which restricts the motion of the heart surface.

The correlations of these signals are shown in Fig. 5.21. Correlation between ECG and RPS is hardly there (see $\rho_{\mathrm{ECG,RPS}}$), as pacemaker and lung machine are two completely independent components. The ECG signal and the motion of the landmark in x-direction as well as in y-direction, show a significant correlation, see $\rho_{\mathrm{ECG,t_x}}$ and $\rho_{\mathrm{ECG,t_y}}$, respectively. The correlation between RPS and t_x is very low (see $\rho_{\mathrm{RPS,t_x}}$), because the motion of the landmark in x-direction is hardly influenced by the patient's respiration. The $\rho_{\mathrm{RPS,t_y}}$ trajectory shows the strong influence of the RPS on the y-motion of the considered landmark.

Computing the discrete Fourier transform of the ECG signal and regarding the amplitude spectrum shows a dominant peak at $f_{\mathrm{ECG}} = 1.4$ Hz ($f_{\mathrm{sample,ECG}} = 100$ Hz), as given in Fig. 5.22. This corresponds to a heart beat rate of 84 bpm. The amplitude spectrum of the RPS shows a dominant peak at $f_{\mathrm{RPS}} = 0.17$ Hz

---

[1]The author would like to thank Dipl.-Ing. Hans-Jürgen Sedlmayr (DLR), Dr. Dieter Boehm (GH), and Dieter Schmid (GH) for their support during data acquisition.

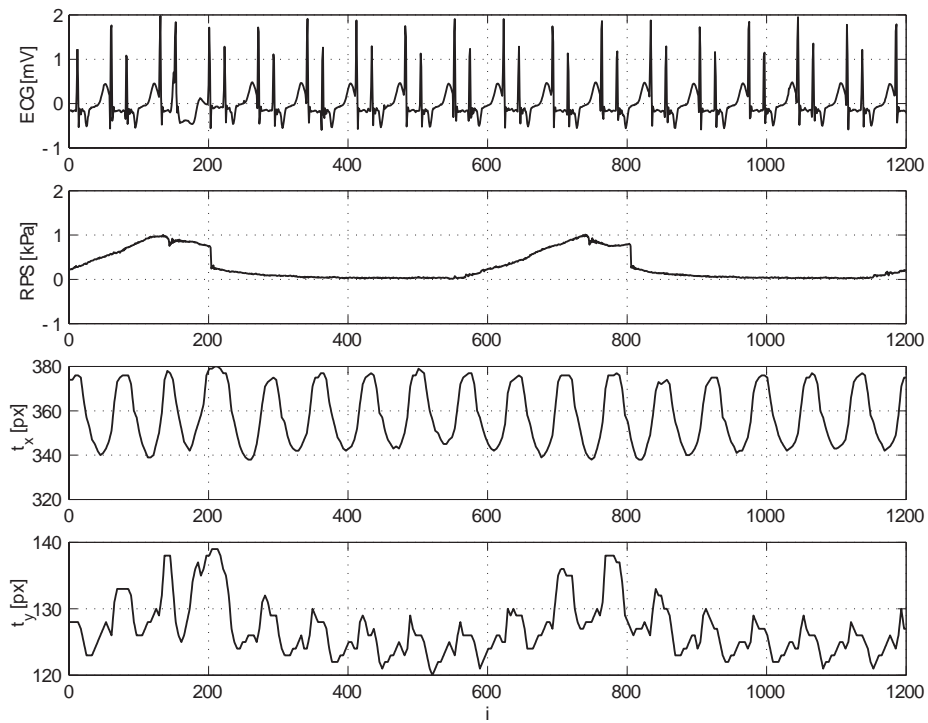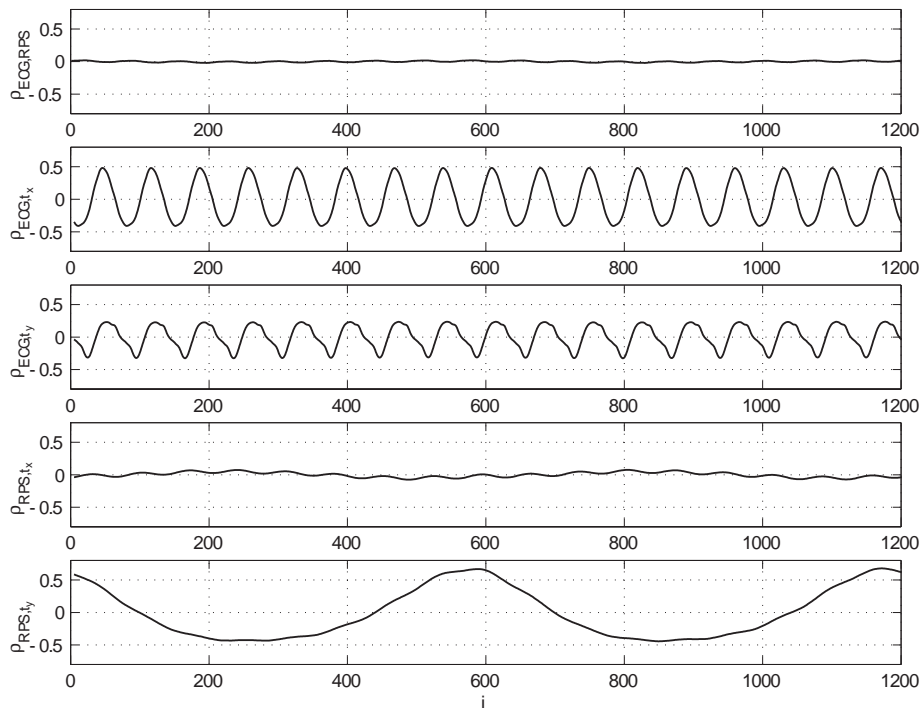Figure 5.20: ECG, RPS, t_x and t_y measured simultaneously.
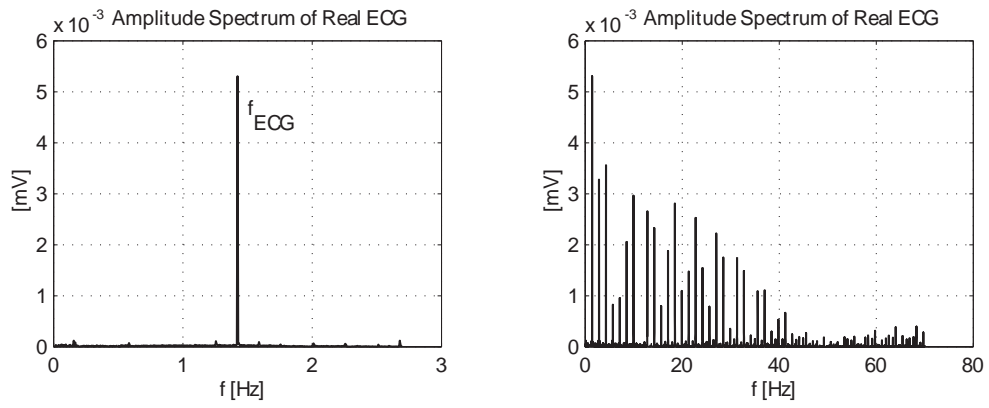

Figure 5.21: Correlation of ECG, RPS and HST.

115

Figure 5.22: Amplitude spectrum of the patient's ECG signal.
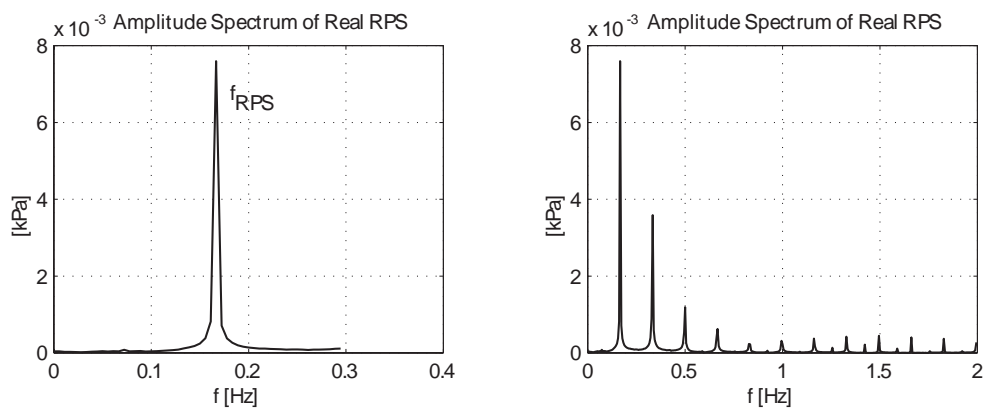

Figure 5.23: Amplitude spectrum of the patient's RPS signal.

116

($f_{\text{sample,RPS}} = 100$ Hz), as given in Fig. 5.23. This is equal to a respiration rate of 10.2 cpm. The amplitude spectrum of the HST (see Fig. 5.24) shows a dominant peak in x-direction at $f_{\text{t}_\text{x}} = 1.4$ Hz, which corresponds to f_{ECG}, whereas the amplitude spectrum of the trajectory in y-direction has two dominant peaks. The first at $f_{\text{t}_\text{y},1} = 0.17$ Hz being equal to f_{RPS}, the second peak can be found at $f_{\text{t}_\text{y},2} = 1.4$ Hz, being equal to f_{ECG}.
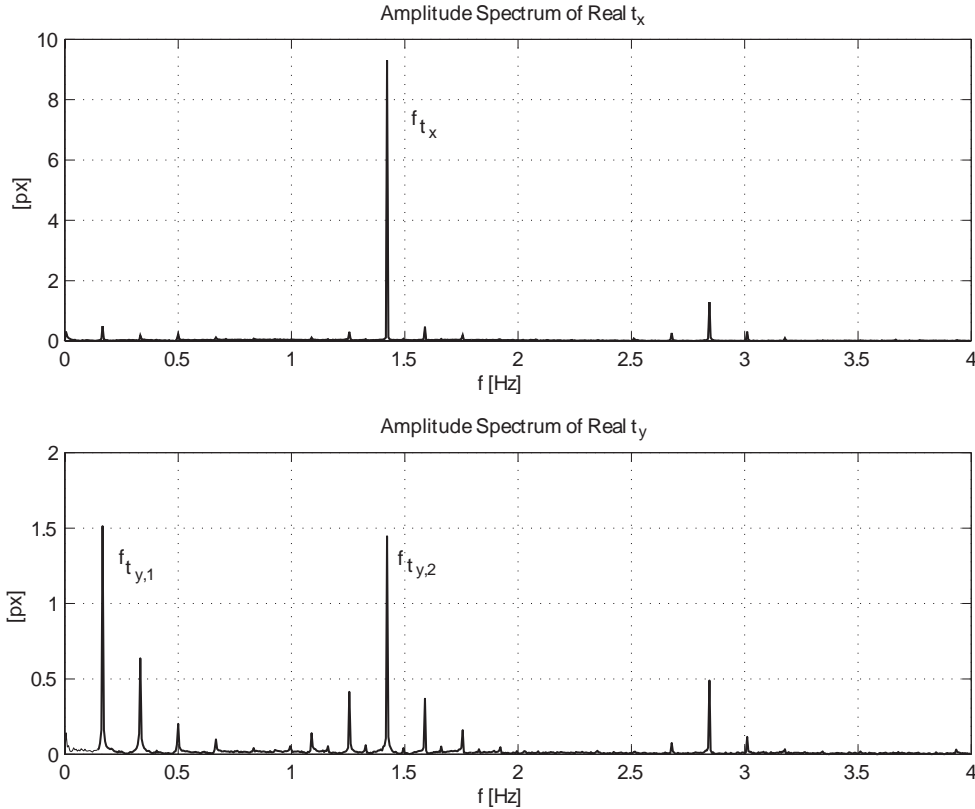
Figure 5.24: Amplitude spectrum of the patient's HST signal.

These investigations justify the assumption made in Sec. 4.5.8: The motion of the landmarks is strongly correlated with the RPS and ECG signal.

The subsequent section presents an algorithm that makes use of the ECG signal and RPS to estimate the motion of landmarks in case of occlusions or other disturbances and is based on the algorithm given in Sec. 5.3.1.

5.4.2   Time Series Embedding

To exploit the correlation between HST, RPS, and ECG signals the algorithm presented in Sec. 5.3.1 is applied. The ECG and the RPS signal are regarded as an additional landmark with high reliability. No outlier detection is applied, as

it is for the HSTs. The measurements $y_{\mathrm{RPS,ECG,i}}$ are vector-valued and can be written as:

$$y_{\mathrm{RPS,ECG,i}} = [y_{\mathrm{RPS,i}}, \xi \, y_{\mathrm{ECG,i}}]^T \; . \tag{5.37}$$

As the ECG and the RPS signals have different units, they have to be scaled by the factor ξ with $[\xi] = \frac{\mathrm{kPa}}{\mathrm{mV}}$, such that both components of $y_{\mathrm{RPS,ECG,i}}$ have the same units. Otherwise the Euclidean distance used to calculate the M best matches according to Sec. 5.2.1 is meaningless. Only appropriate scaling guarantees that the comparison of $\boldsymbol{D}_k^{\mathrm{RPS,ECG}}$ with the memory $\boldsymbol{D}_{k-i}^{\mathrm{RPS,ECG}}$ leads to correct results. Furthermore, the optimal value for ξ which leads minimal prediction errors is unknown.

Calculation of Embedding Parameters Before the optimal ξ can be determined, the optimal embedding dimension $p_{\mathrm{RPS,ECG}}$ and time lag $h_{\mathrm{RPS,ECG}}$ have to be found: Therefore, the embedding dimensions p_{ECG} and p_{RPS} as well as the time lags h_{ECG} and h_{RPS} have to be determined: According to Sec. 5.2.2 for an ECG signal, sampled with $f_{\mathrm{sample}} = 80$ Hz and $f_{\mathrm{ECG}} = 80$ bpm, the following optimal parameters can be found:

$$p_{\mathrm{ECG}} = 8 \quad \text{and} \quad h_{\mathrm{ECG}} = 14 \; . \tag{5.38}$$

Since the sample frequency has a linear impact on the time lag, and the new sample frequency is $f_{\mathrm{sample}} = 100$ Hz and the new heart beat rate (having a linear impact on the time lag too) is $f_{\mathrm{ECG}} = 84$ bpm, the new time lag can be easily determined:

$$h_{\mathrm{ECG}} = 18 \; , \tag{5.39}$$

whereas the embedding dimension remains unchanged. For the RPS the same considerations lead to

$$p_{\mathrm{RPS}} = 10 \quad \text{and} \quad h_{\mathrm{RPS}} = 100 \; . \tag{5.40}$$

The ECG signal has higher frequencies than the RPS, so $h_{\mathrm{RPS,ECG}}$ is chosen in order to capture enough dynamics of the ECG signal for robust matching, as

$$h_{\mathrm{RPS,ECG}} = h_{\mathrm{ECG}} = 18 \; . \tag{5.41}$$

Calculating $p_{\mathrm{RPS,ECG}}$, $H^{\mathrm{RPS}} = h_{\mathrm{RPS}} p_{\mathrm{RPS}}$ as introduced in Eq. 5.22, is considered. The value of H^{RPS} describes, how far values for building the embedding vector $\boldsymbol{D}_k^{\mathrm{RPS}}$ lie in the past. Here $H^{\mathrm{RPS}} = 1000$, with $h_{\mathrm{RPS,ECG}} = 18$, $p_{\mathrm{RPS,ECG}}$ can be calculated:

$$p_{\mathrm{RPS,ECG}} = \frac{H^{\mathrm{RPS}}}{h_{\mathrm{RPS,ECG}}} = \frac{1000}{18} \approx 56 \quad \text{with} \quad f_{\mathrm{RPS}} = 10 \text{ cpm} \; . \tag{5.42}$$

The value $p_{\mathrm{RPS,ECG}} = 56$ guarantees, that enough past values are contained in $\boldsymbol{D}_k^{\mathrm{RPS}}$ to capture the underlying dynamics of the RPS.

Calculation of ξ   The global prediction scheme presented in Sec. 5.3.1 is used to estimate the motion of the landmark already considered in Sec. 5.4.1. Therefore, the landmark is regarded as lost and the ECG signal and the RPS are used for its prediction. The normalized mean prediction error $\hat{\mu}_{\mathrm{err}}$ and the normalized standard deviation $\hat{\sigma}_{\mathrm{err}}$ are calculated for different values of ξ. The results are shown in Fig. 5.25, for x- and y-direction. Predicting other landmarks leads to similar results. The curves decline sharply until $\xi \approx 0.6 \; \frac{\mathrm{kPa}}{\mathrm{mV}}$; for greater values, the curves are rather flat. If only RPS is used for predicting the HST in x- and y-direction ($\xi = 0$), prediction quality is

$$\hat{\mu}_{\mathrm{err,x}} = 0.65 \quad \text{and} \quad \hat{\sigma}_{\mathrm{err,x}} = 0.41 \tag{5.43}$$

$$\hat{\mu}_{\mathrm{err,y}} = 0.46 \quad \text{and} \quad \hat{\sigma}_{\mathrm{err,y}} = 0.40 \; . \tag{5.44}$$

If only ECG is taken into account ($\xi \to \infty$), the prediction quality is

$$\hat{\mu}_{\mathrm{err,x}} = 0.087 \quad \text{and} \quad \hat{\sigma}_{\mathrm{err,x}} = 0.063 \tag{5.45}$$

$$\hat{\mu}_{\mathrm{err,y}} = 0.52 \quad \text{and} \quad \hat{\sigma}_{\mathrm{err,y}} = 0.37 \; . \tag{5.46}$$

These results reflect the different influence of ECG and RPS on the HST in x- and y-direction: Motion of this landmark in x-direction is more influenced by the ECG signal than motion in y-direction. On the other hand RPS is less important for motion in x-direction than in y-direction. Nevertheless, both signals have to be taken into account to achieve optimal prediction results. Since for $\xi \geq 0.6 \; \frac{\mathrm{kPa}}{\mathrm{mV}}$ the choice of ξ is not critical for the prediction accuracy of the HSTs, $\xi = 1.0 \; \frac{\mathrm{kPa}}{\mathrm{mV}}$ is chosen for the subsequent calculations. This works well for other landmarks, too. Prediction for $\xi = 1.0 \; \frac{\mathrm{kPa}}{\mathrm{mV}}$ can be seen in Fig. 5.26. The estimation error (see Fig. 5.27) is remarkably small. Here, the discrete time i is a multiple of $1/f_{\mathrm{sample,ECG}}$=10 ms.

Another advantage of the estimation scheme presented here is that it can be easily included in the global prediction scheme given in Sec. 5.3. This is shown in the following section.

5.5   Robust Motion Estimation

In this section an algorithm as well as experimental results of robust motion estimation in MIRCS is presented. This framework (see Fig. 5.28) is a straight-forward combination of the methods presented in Sec. 5.3 and Sec. 5.4.2. The calculation of the weights in Fig. 5.28 is similar to Sec. 5.3.1. They reflect the individual prediction quality $\mu_{\mathrm{err}}^{a,r}$, where a particular visible landmark y^r is used to predict the position of the lost landmark y^a (lower part in Fig. 5.28). Additionally, the prediction quality $\mu_{\mathrm{err}}^{a,\mathrm{ECG,RPS}}$, where the ECG and RPS signals are used for prediction (upper part in Fig. 5.28), is taken into account. The proposed
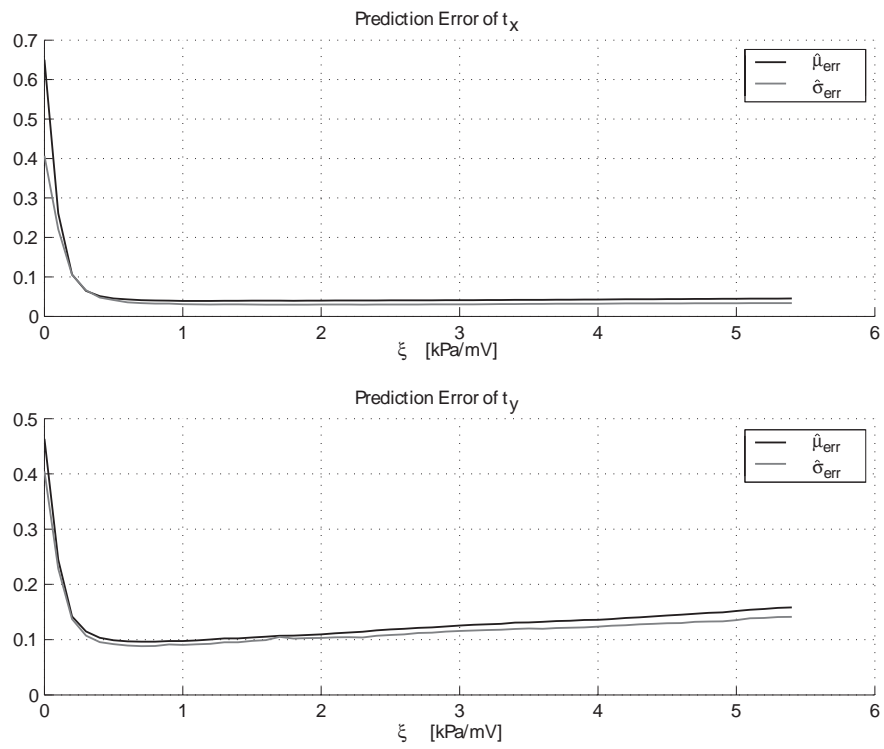
Figure 5.25: Determination of best ξ. Parameters: s=1, p=56, h=18, M=8, $\bar{n} = 4000$, n=8100.

Figure 5.26: Prediction of HST using ECG and RPS. Parameters: $\xi = 1.0 \ \frac{\text{kPa}}{\text{mV}}$, $s=1$, $p=56$, $h=18$, $M=8$, $\bar{n} = 4000$, $n=8100$.

calculation of the weights allows an adaption of the robust motion estimation scheme according to the reliabilities of the individual prediction results.

The main difference to the global prediction scheme given in Sec. 5.3 is, that N landmarks lying on the heart surface are considered, together with the ECG signal and the RPS as an additional landmark. So, overall $N + 1$ landmarks are regarded simultaneously. If d landmarks, with $d \leq N$ are lost, the other landmarks, including the ECG signal and RPS are used for estimation, according to Sec. 5.3.

As the patient has two pacemakers (see Sec. 5.4) the ECG signal is very regular and therefore the resulting HSTs are regular, too. This leads to good prediction results. The position of the landmarks considered (together with the search area) can be seen in Fig. 5.29. Parts of the HSTs of LM1 is shown in Fig. 5.30. The ECG signal and the RPS are the same as already used for the prediction shown in Sec. 5.4.2, which also holds for the parameter $\xi = 1.0 \ \frac{\text{kPa}}{\text{mV}}$. Overall $N + 1 = 4$ landmarks are available. In this section the discrete time i is a multiple of $1/f_{\text{sample,ECG}}=10$ ms.

The following parameters are used to build the embedding vectors of the HSTs:

$$p_{\text{HST}} = 12 \quad \text{and} \quad h_{\text{HST}} = 96 \ . \tag{5.47}$$

121
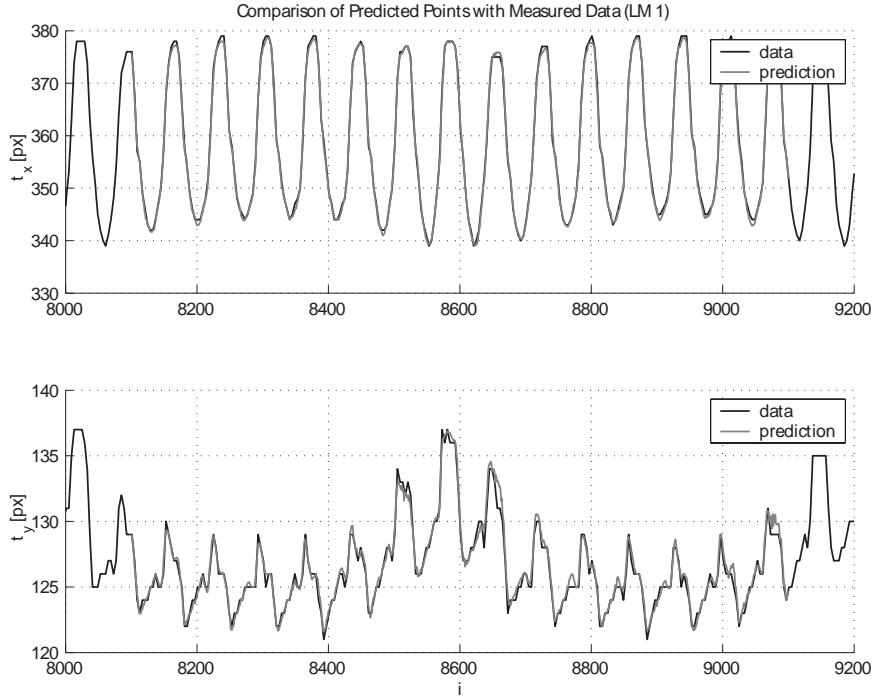
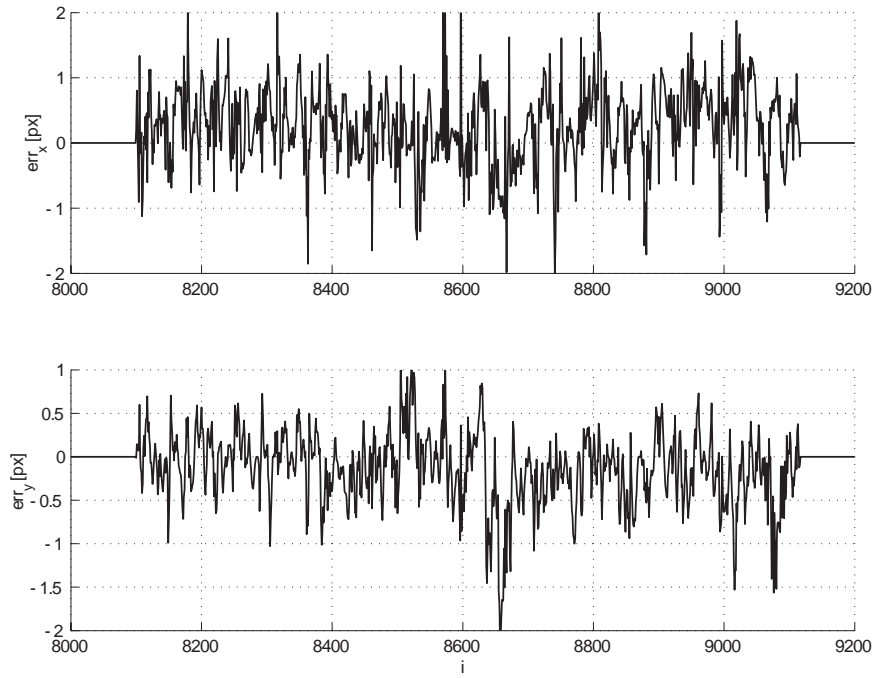Figure 5.27: Prediction error using ECG and RPS. Parameters: $\xi = 1.0 \frac{\text{kPa}}{\text{mV}}$, $s=1$, $p=56$, $h=18$, $M=8$, $\bar{n} = 4000$, $n=8100$.
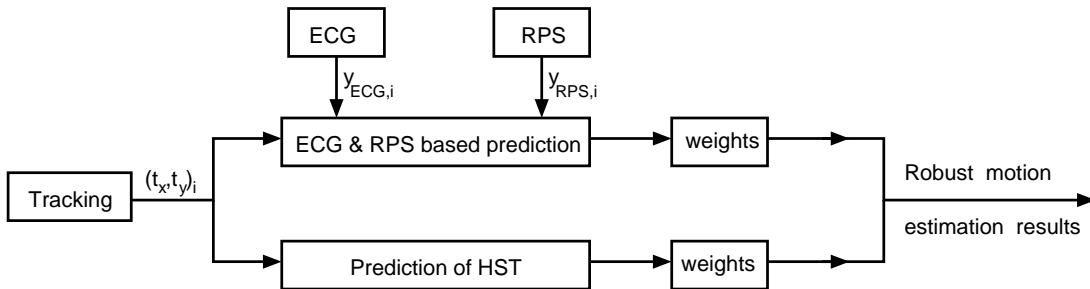

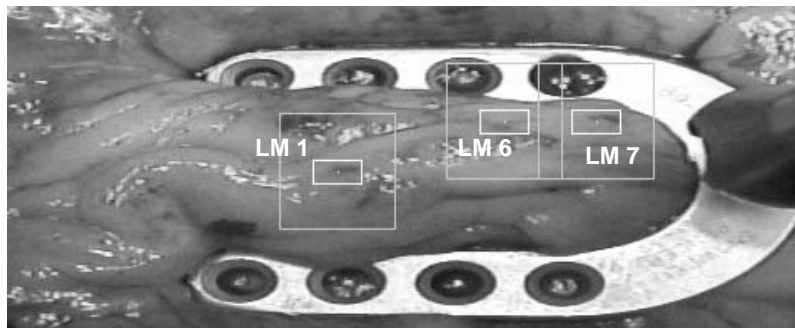Figure 5.28: Robust motion estimation scheme.

Figure 5.29: Position of landmarks.

Calculation of these optimal parameters is according to the algorithm presented in Sec. 5.2.4. For the ECG and the RPS signal, the embedding parameters are:

$$p_{\text{RPS,ECG}} = 56 \quad \text{and} \quad \text{h}_{\text{RPS,ECG}} = 18 , \tag{5.48}$$

as already derived in Sec. 5.4.2. Both embedding vectors $H^{\text{RPS,ECG}} \approx H^{\text{HST}}$ cover about 1100 samples, which is about 10 sec. The memory in which the embedding vectors $\boldsymbol{D}_k^{\text{RPS,ECG}}$ and $\boldsymbol{D}_k^{\text{HST}}$ are matched, contains 8000 samples, and the best $M_{\text{RPS,ECG}} = 8$ and $M_{\text{HST}} = 5$ matches are chosen for prediction.

The artificial disturbances (no valid position of the landmark considered is available) for the HSTs (ECG signal and RPS remain undisturbed) are summarized in Tab. 5.5:

Table 5.5: Disturbances of HSTs.

Landmark	Start (i_{start})	End (i_{end})	Duration
LM1	9000	10000	10 sec
LM6	9500	11000	15 sec
LM7	9800	11400	16 sec

After $i = 9800$ no valid landmarks are available, only the ECG and the RPS signal can be used for estimation. After $i = 10000$ the landmarks become valid again. The performance of the prediction of landmark LM1 is given in Fig. 5.30. The prediction results of landmark LM6 and landmark LM7 are not shown here in detail, but summarized in Tab. 5.6. Figure 5.31 shows the prediction error of LM1 in detail.

Table 5.6: Prediction quality.

	LM1 t_x	LM1 t_y	LM6 t_x	LM6 t_y	LM7 t_x	LM 7 t_y
$\hat{\mu}_{\text{err}}$	0.056	0.17	0.13	0.20	0.10	0.35
$\hat{\sigma}_{\text{err}}$	0.045	0.15	0.45	0.45	0.081	0.20

The high quality of the prediction, compared to Sec. 5.3.2, is mainly due to three reasons: First, additional information, which is strongly correlated with the heart motion, is used. Second, the two pacemakers are responsible for a very periodic ECG signal and therefore for very periodic HSTs. Third, the long memory which is available here (8000 samples) allows a very accurate matching of the $\boldsymbol{D}_k^{\text{RPS,ECG}}$ and $\boldsymbol{D}_k^{\text{HST}}$, which leads to a very good subsequent prediction of the heart motion.

Of course one can think of other, approaches to describe the correlation between ECG and RPS signal as the input signals and the HSTs as the output signal. An
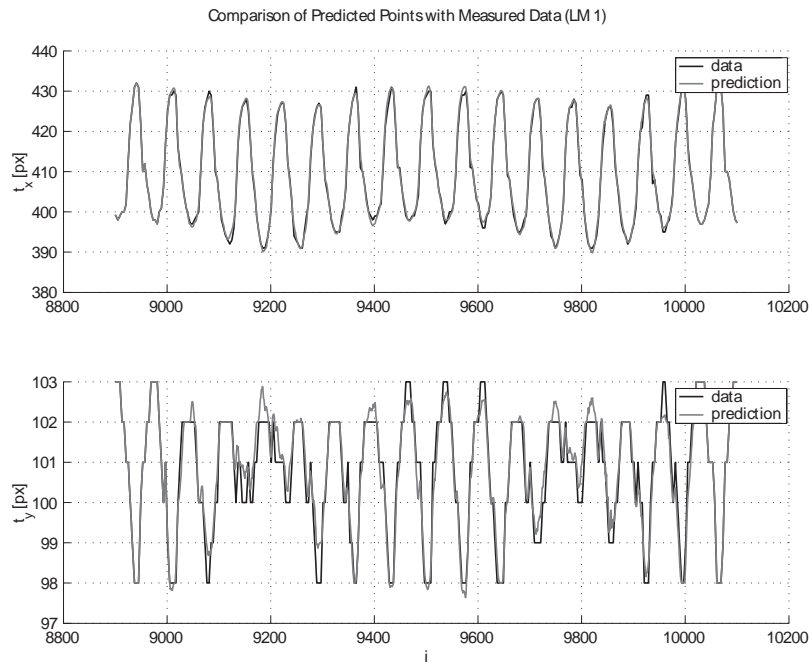
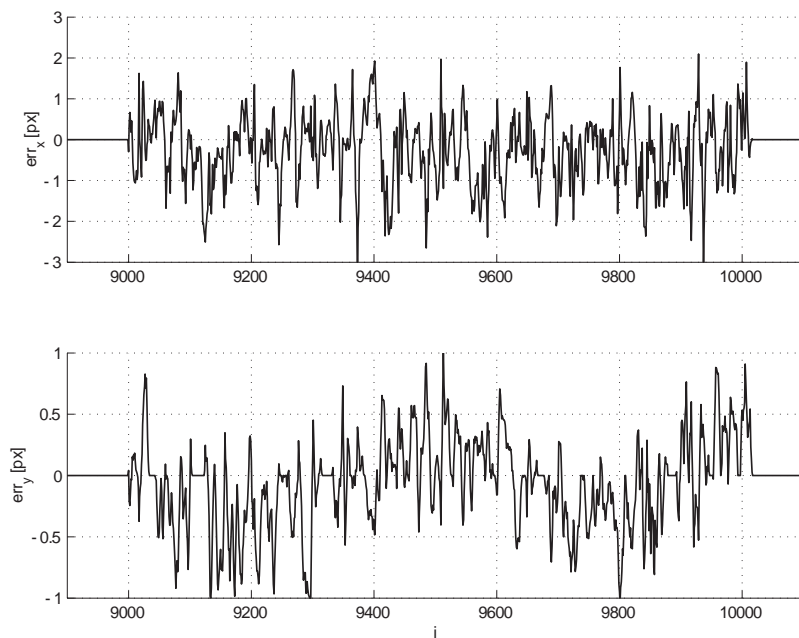Figure 5.30: Trajectory and prediction of landmark 1.

Figure 5.31: Prediction error of landmark 1.

124

artificial neural network (ANN) e.g. might be able to capture this correlation. During this work two types of ANNs have been examined: Perceptron Backpropagation Networks with one hidden layer, as described in [63] and Elman Networks [46], with a so called *context unit*, which holds a copy of the hidden layer. This context unit can be interpreted as a short-term memory that builds the time lag of the ANN. As the results using ANNs for prediction are not as good as the results presented in Sec. 5.4.2 and as ANNs additionally have the drawback of a long training period, they are not presented here.

5.6   Conclusions

The previous sections presented a variety of algorithms to bridge disturbances of the tracking scheme: A framework, based on Takens Theorem, was introduced, which allows to capture the (observable) inner states being responsible for HSTs, ECG as well as RPS signals. This was shown by high accuracy one-step predictions. With one-step prediction of HSTs, robust outlier detection, as well as positioning of the search area of the tracker are possible. More-step predictions can be used to bridge a few subsequent outliers. Additionally, more-step predictions increase the bandwidth of the closed control loop (here closed control loop means the complete MIRS framework that is necessary to compensate the heart motion), because the delay time (communication time, computing time, etc.) can be reduced.

To handle longer disturbances reliably, a global prediction scheme is introduced in Sec. 5.3. The HSTs of visible landmarks are used to estimate the motion of landmarks which are occluded or disturbed. Tracking parts of the mechanical stabilizer, which can serve as prominent landmarks themselves, increases the quality of the global prediction scheme.

Section 5.4 showed that ECG as well as RPS signals are strongly correlated with HSTs. Therefore, these signals can be included in a new tracking scheme. The experimental results validate the performance of an overall tracking scheme that exploits several natural landmarks in combination with the ECG signal and the RPS. This scheme allows robust detection of single outliers as well as a robust long-term prediction of occluded landmarks.

The methods presented here are a prerequisite to establish motion compensation with a surgical robot in minimally invasive beating heart surgery. Robust motion compensation simplifies the surgical task and will lead to a safer and more secure surgery and reduce the patient's risks to a minimum.

Chapter 6

Conclusion and Perspectives

Conclusion   Minimally invasive surgery is a new technique, established in the 1980's. Due to the small incisions, which are necessary to insert the surgical instruments, it is far less traumatic than open surgery. As the access to the operating area is restricted, new operation techniques have been developed. Besides the advantages for the patient, the surgeon has to cope with numerous disadvantages, such as restricted motion inside the body and loss of haptic feedback. To overcome these drawbacks robotics plays an important role: Scaling of the input leads to more accurate motion of the instruments. Tremor filters reduce the undesired jitter of the surgeon's hand and new surgical instruments reestablish full manipulability inside the human body. Nevertheless, only few advantages of MIRS systems are currently exploited: Intelligent control laws, establishing shared control, are hardly used. Force feedback and force control have not been introduced into the operating room. Motion estimation and motion compensation allowing a more accurate therapy is investigated at research institutes mainly.

This work takes a closer look at motion estimation and motion compensation in beating heart surgery. Once the motion has been captured reliably, it can be compensated by the robot with high accuracy and bandwidth. The surgeon cares about complicated tasks, which require special knowledge and are performed with low bandwidth. The goal of motion compensation in beating heart surgery is to enable the surgeon to work on a virtually stabilized beating heart as he got used to in on-pump surgery.

Motion compensation has two prerequisites: motion estimation and control laws to command the surgical robot. Therefore, several Cartesian control laws have been developed, providing a standard interface for motion estimation algorithms and reestablishing the correct hand-eye coordination for the surgeon (see *control* module of Fig. 6.1). Position and velocity control of surgical instruments are a very common way to command a surgery robot. Unintentional damage of tissue can be avoided by force control, which is why new surgical instruments

with miniaturized force/torque sensors have to be built. It is expected that the application of different control strategies depending on the current situation of the operation increases safety and reduces the operation duration. However, an evaluation by surgeons is necessary.

To capture the motion of the heart surface by exploiting natural landmarks, a reduced affine motion model is sufficient. After appropriate handling of specular reflections, tracking can be performed with high accuracy, even if the motion model is reduced to pure translation (this compuation is carried out in the *tracking* module of Fig. 6.1). Considering the amplitude spectrum of the computed trajectories, respiration rate and heart beat rate can be identified.

Tracking is more robust with respect to short occlusions or other disturbances if additional information is considered. Therefore, several landmarks are tracked simultaneously. The position of the unoccluded landmarks can be used to estimate the position of the occluded ones. To be not only dependent on visual information, the electrocardiogram and the respiration pressure signal can be included in the tracking framework. This leads to a robust motion estimation framework which is able to bridge longer disturbances of the visual tracking (see *prediction* module of Fig. 6.1).
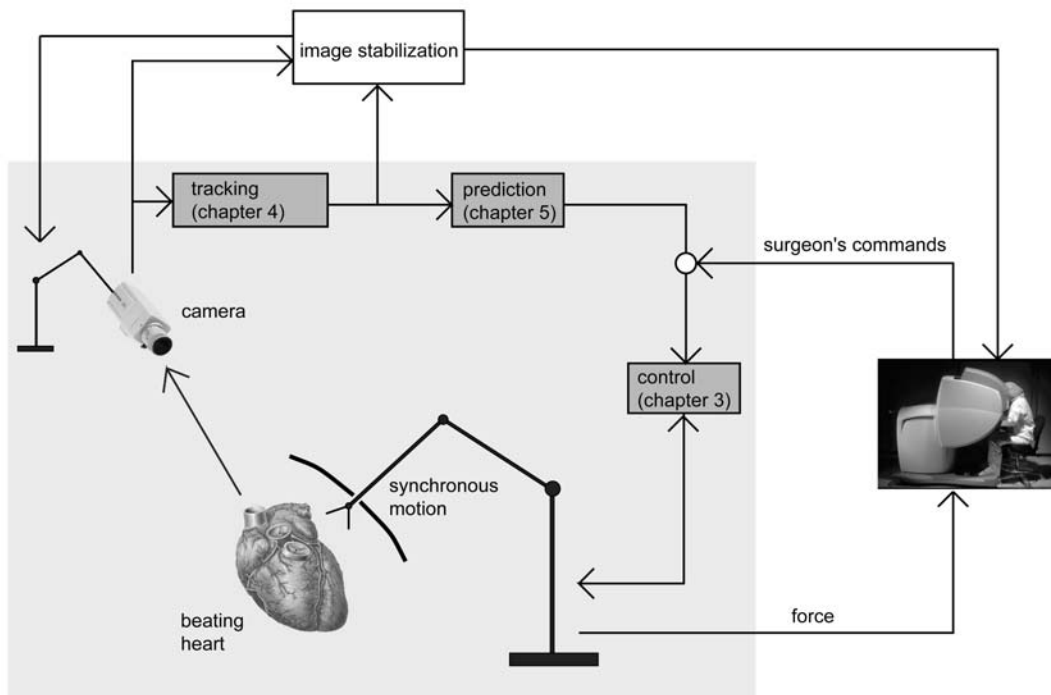

Figure 6.1: Overall Motion Compensation Scheme (heart picture from [50], console picture from [34]).

Perspectives To increase the reliability of the motion estimation scheme further, additional markers, such as small (colored) clips placed near the anastomosis seem to be appropriate. They might allow robust motion estimation, even under severe disturbances and provide additional information about the heart motion, that can be considered by the proposed motion estimation scheme.

A new experimental setup has to be built, allowing the evaluation of the control and motion estimation algorithms proposed here. The evaluation should take special care of ergonomics, accuracy, operation time, and overall reliability of the system. The question if force feedback is important for MIRS systems has to be answered. In addition, the benefit of different control laws can be quantified. And last but not least, the importance of motion compensation has to be justified. Of course, other applications, such as tumor therapy and biopsy, are a promising field for motion estimation and compensation, too.

Furthermore, online motion estimation plays an important role in the registration of preoperative planning data, because deviations between preoperative data and the current situation in the operating room can be reduced. The combination of motion estimation with intraoperative navigation promises high accuracy therapy of the patient, as an online update of the work space is possible. Overall, motion estimation in combination with intelligent control laws allow a variety of new therapies, reducing the patient's risk and enabling a fast recovery from the surgical intervention.

In order to support the surgeon who sits in front of the console (see Fig. 6.1) and to improve the quality of immersion it is important to stabilize the laparoscopic images: The surgeon needs a stabilized image of the area of interest at the remote side, where almost no motion occurs. This can be either achieved electronically by appropriate image warping algorithms or mechanically by moving the camera which is attached to a robot. The commands of the surgeon are superimposed to the motion compensation commands computed by the prediction module.

The following paragraphs present the advanced DLR MIRS system which is currently under development.

Advanced DLR MIRS System Taking into account the insights gained from the preliminary DLR MIRS system, an advanced robotic surgery system is currently designed: It is based on technology developed for the DLR light weight robot [30] and the DLR four finger hand [8].

The electronics of the robot is fully integrated into the light weight structure. This allows easy and fast handling of the robot by one person if the robot has to be removed from the patient in case of emergency. All joints of the robot are motor driven, therefore the robot is designed for both open and endoscopic surgery. The passive joints, which make sure that no forces are exerted at the entry point, can be realized by torque controlled joints. Setting the desired torque value of these

joints to zero ensures that the forces at the entry point remain small and the patient is not injured. Additionally, torque controlled joints allow a variety of advanced control laws, such as high quality impedance control which are useful if the robots cooperate with humans [1].

The new robot has seven degrees of freedom, which allows null space motion to avoid collision between two robots or between robots and OR equipment. In combination with port planning software which considers the individual anatomy of the patient optimal dexterity can be provided to surgeons.

In telepresence applications such as bypass grafts the Cartesian position control loop is closed via the surgeon. Therefore, the absolute positioning accuracy of the robot is not critical whereas the relative position accuracy is. In the applications considered here, the highest demands occur during suturing the anastomosis in bypass surgery, where the relative position accuracy has to be less than 0.2 mm. Absolute position accuracy is important if autonomous tasks have to be executed by the robot, such as positioning screws in spinal surgery or motion compensation in beating heart surgery, and should be 0.2 mm at least. The maximum work space occurs in abdominal surgery and is about 6 cm high, 9 cm wide and 14 cm long.

According to [59] the bandwidth of input trajectories generated by the human (so called non-voluntary arm motion which is typically used to perform tracking or sensor controlled tasks) is about $1 - 2$ Hz. As shown before, the heart beat rate is about 1 Hz, the first harmonic can be found at 2 Hz, the second at 3 Hz. Respiration rate is less than 0.2 Hz. Compensating heart motion up to the first harmonic leads to the same dynamic demands for the telesurgery system as pure telesurgery without motion compensation: The bandwidth of the input trajectories is about 2 Hz.

New surgical instruments provide two additional degrees of freedom inside the patient and enable full dexterity for the surgeon. Miniaturized force sensors are a prerequisite for force feedback and allow supervision of contact forces during surgery.

The complete advanced DLR MIRS system is sketched in Fig. 6.2 and consists of (at least) five identical robots: two hold sensorized and actuated instruments as described above, one moves the camera, and two robots are used as a force feedback master system. Furthermore, a second master system (located in the same operating room or in different hospitals) connected to the first one enables new training and surgical procedures. An unexperienced surgeon can get support by his supervisor during difficult situations or complicated tasks: The experienced surgeon takes control of the surgical instruments as well as of the trainee's master arms. As result, the trainee does not only see the surgical intervention, but his hands are also moved according to the surgical procedure.

Autonomous functions such as automatic camera guidance or motion compensa-
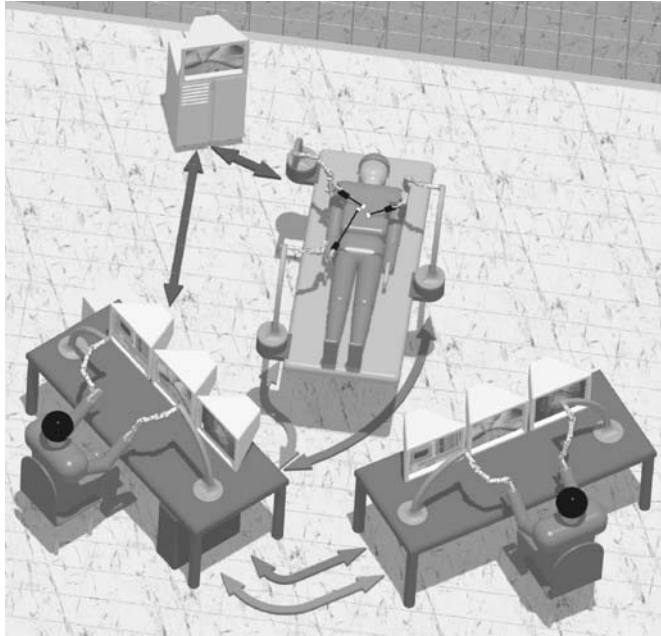
Figure 6.2: Advanced DLR MIRS System

tion make surgical interventions safer and faster. By these efforts the applications of robotic surgery are expected to increase dramatically.

Bibliography

[1] A. Albu-Schäffer. *Regelung von Robotern mit elastischen Gelenken am Beispiel der DLR-Leichtbauarme*. PhD thesis, Technical University Munich, April 2002.

[2] K. Arbter and G.-Q. Wei. Method of tracking a surgial instrument with a mono or stereo laparoscope. US Patent Number: 5,820,545, Oct. 13. 1998.

[3] Armstrong Healthcare Inc. http://www.armstrong-healthcare.com/, December 2001.

[4] Institut für angewandte Informatik ARTEMIS, Forschungszentrum Karlsruhe. http://wwwserv2.iai.fzk.de/∼artemis, December 2001.

[5] P.N. Belhumeur and D.J. Kriegman. What is the set of images of an object under all possible lighting conditions. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 270–277. IEEE, 1996.

[6] C. Bolger, C. Wigfield, T. Melkent, and K. Smith. Frameless stereotaxy and anterior cervical surgery. *Journal of Computer Aided Surgery*, 1999.

[7] Cornelius Borst. Operieren am schlagenden Herz. *Spektrum der Wissenschaft*, Januar 2001.

[8] J. Butterfaß, M. Grebenstein, H. Liu, and G. Hirzinger. DLR-hand II: Next generation of a dextrous robot hand. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, May 2001.

[9] M. Caversaccio, R. Bächler, K. Lädrach, G. Schroth, LP. Nolte, and R. Häusler. The Bernese frameless optical computer aided surgery system. *Journal of Computer Aided Surgery*, 1999.

[10] M. C. Cavusoglu. *Telesurgery and Surgical Simulation: Design, Modeling, and Evaluation of Haptic Interfaces to Real and Virtual Surgical Environments*. PhD thesis, UC Berkeley, August 2000.

[11] M. C. Cavusoglu, F. Tendick, M. Cohn, and S. Sastry. A laparoscopic telesurgical workstation. *IEEE Transactions on Robotics and Automation*, 1999.

[12] M. C. Cavusoglu, W. Williams, F. Tendick, and S.S. Sastry. Robotics for telesurgery: Second generation Berkeley/UCSF laparoscopic telesurgical workstation and looking towards the future applications. In *Proceedings of the 39th Allerton Conference on Communication, Control and Computing*, October 2001.

[13] Computer Aided and Image Guided Medical Interventions Homepage. http://cimint.ethz.ch, March 2002.

[14] Computer Motion Inc. http://www.computermotion.com, January 2002.

[15] I. Cox, S. Rao, and B. Maggs. A maximum likelyhood stereo algorithm. *Computer Vision and Image Understanding*, 63(3):542–567, 1996.

[16] B. Davies, S. Starkie, S.J. Harris, E. Agterhuis, V. Paul, and L.M. Auer. Neurobot: A special-purpose robot for neurosurgery. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, 2000.

[17] A. DiGioia. Surgical navigation and image guided reconstructive hip surgery. June 1998.

[18] R. Drzymala and S. Mutic. Stereotactic imaging quality assurance using an anthropomorphic phantom. *Journal of Computer Aided Surgery*, 1999.

[19] V. Falk, A. Diegeler, T. Walther, B. Vogel N. Löscher, C. Ulmann, T. Rauch, and F. W. Mohr. Endoscopic coronary artery bypass grafting on the beating heart using a computer enhanced telemanipulation system. *Heart Surg Forum*, 2:199–205, 1999.

[20] C. Franciosi, R. Caprotti, F. Romano, G. Porta, G. Real, G. Colombo, and F. Uggeri. Laparoscopic versus open splenectomy: a comparative study. *Surg Laparosc Endosc Percutan Tech*, Oct. 2000.

[21] M. Ghodousssi, S. Butner, and Y. Wang. Robotic surgery – the transatlantic case. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, 2002.

[22] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing.* Addison-Wesley, 1992.

[23] M. Gröger, T. Ortmaier, W. Sepp, and G. Hirzinger. Tracking local motion on the beating heart. In Seong K. Mun, editor, *SPIE Medical Imaging: Visualization, Image-Guided Procedures, and Display*, volume 4681, San Diego, USA, February 2002.

[24] M. Gröger, W. Sepp, T. Ortmaier, and G. Hirzinger. Reconstruction of image structure in presence of specular reflections. In B. Radig and S. Florczyk, editors, *Pattern Recognition, Proc. 23rd DAGM Symposium*, pages 53–60, Munich, Germany, September 2001.

[25] G. Guthart and J. Salisbury. The intuitive telesurgery system: Overview and application. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, 2000.

[26] Gregory D. Hager and Peter N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(10), October 1998.

[27] P. Händel and P. Tichavski. Adaptive filtering for periodic signal enhancement and tracking. *Int. journal of Adaptive Control and Signal Processing*, 1994.

[28] M.A. Helmy. A comparative study between laparoscopic versus open appendicectomy in men. *J Egypt Soc Parasitol*, Aug. 2001.

[29] G. Hirzinger. Robot-Teaching via Force-Torque-Sensors. In *Proceedings of the sixth European Meeting on Cybernetics and Systems Research*, 1982.

[30] G. Hirzinger, A. Albu-Schäffer, M. Hähnle, I. Schaefer, and N. Sporer. On a new generation of torque controlled light-weight robots. In *IEEE International Conference of Robotics and Automation*, pages 3356–3363, 2001.

[31] G. Hirzinger, B. Brunner, J. Dietrich, and J. Heindl. Sensor-based space robotics - ROTEX and its telerobotic features. *IEEE Transactions on Robotics and Automation*, 9(5), October 1993.

[32] M.T. Huang, P.L. Wei, C.C. Wu, Lai, R.J. Chen, and W.J. Lee. Needlescopic, laparoscopic, and open appendectomy: a comparative study. *Surg Laparosc Endosc Percutan Tech.*, Oct. 2001.

[33] Integrated Surgical Systems Inc. http://www.robodoc.com, April 2002.

[34] Intuitive Surgical Inc. http://www.intuitivesurgical.com, May 2000.

[35] S. Jacobs and V. Falk. Pearls and pitfalls: Lessons learned in endoscopic robotic surgery–the daVinci experience. *Heart Surg Forum*, 2001.

[36] Bernd Jähne. *Digitale Bildverarbeitung*. Springer, 4th edition, 1997.

[37] Erik W.L. Jansen. *Towards minimally invasive coronary artery bypass grafting*. Brouwer Uithof, Utrecht, 1998.

[38] KAIST. http://robot.kaist.ac.kr/research/main/telesurgery.html, April 2002.

[39] Holger Kantz and Thomas Schreiber. *Nonlinear Time Series Analysis*. Cambridge University Press, 1997.

[40] Harald Kirchner. *Bewegungsdetektion in Bildfolgen: Ein mehrstufiger Ansatz*. Dt. Universitätsverlag, 1993.

[41] K. Kodera, B. Kiaii, R. Rayman, R. Novick, and W. Boyd. Closed chest CABG on the beating heart with a computer-enhanced articulating system: Case report. *Heart Surg Forum*, 2001.

[42] A. Krupa, C. Doignon, J. Gangloff, M. de Mathelin, L. Soler, and G. Morel. Towards semi-autonomy in laparoscopic surgery through vision and force feedback control. In *Proceedings of the International Symposion of Experimental Robotics*, pages 191–201, 2000.

[43] D.S. Kwon, K.Y. Woo, S.K. Song, W.S. Kim, and H.S. Cho. Microsurgical telerobot system. In *Proceedings of the IEEE/RSJ Int. Conf. on Intelligent Robots and Control Systems, 1998*.

[44] T. Lehmann, W. Oberschelp, E. Pelikan, and R. Repges. *Bildverarbeitung für die Medizin*. Springer, 1997.

[45] K.H. Long, M.P. Bannon, S.P. Zietlow, E.R. Helgeson, W.S. Harmsen, C.D. Smith, D.M. Ilstrup, Y. Baerga-Varela, and M.G. Sarr. A prospective randomized comparison of laparoscopic appendectomy with open appendectomy: Clinical and economic analyses. *Surgery*, Apr. 2001.

[46] W. T. Miller, R. S. Sutton, and P. J. Werbos, editors. *Neural Networks for Control*. The MIT Press, 1990.

[47] V.F. Munoz, C. Vara-Thorbeck, J.G. DeGabriel, J.F. Lozano, E. Sanchez-Badajoz, A. Garcia-Cerezo, R. Toscano, and A. Jiminez-Garrido. A medical robotic assistant for minimally invasive surgery. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, 2000.

[48] C. Natale, R. Koeppe, and G. Hirzinger. An automatic procedure for force controller design. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 1999.

[49] Shree K. Nayar, Xi-Sheng Fang, and Terrance E. Boult. Separation of reflection components using color and polarization. *International Journal of Computer Vision*, 1996.

[50] NetDoctor. http://www.netdoctor.co.uk, February 2003.

[51] Object Management Group (OMG). *A Discussion of the Object Management Architecture*, 1997. OMG Document 00-06-41.

[52] T. Ortmaier, M. Gröger, and G. Hirzinger. Robust motion estimation in robotic surgery on the beating heart. In *Proc. of Computer Assisted Radiology and Surgery - CARS*, Paris, France, June 2002.

[53] T. Ortmaier, M. Gröger, and D. Kotzor. Bewegungsschätzung in der minimal invasiven Herzchirurgie. *at-Schwerpunktheft: Neuroprothetik und Medizinrobotik*, 2002.

[54] T. Ortmaier and G. Hirzinger. Cartesian control issues for minimally invasive robot surgery. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS 2000*, Takamatsu, Japan, October 2000.

[55] T. Ortmaier and G. Hirzinger. Cartesian control of robots with working-position dependent dynamics. In *Proc. of the 6th International IFAC Symposium on Robot Control - Syroco 2000*, Vienna, Austria, September 2000.

[56] T. Ortmaier, D. Reintsema, U. Seibold, U. Hagn, and G. Hirzinger. The DLR minimally invasive robotics surgery scenario. In *Workshop on Advances in Interactive Multimodal Telepresence Systems*, Munich, Germany, March 2001.

[57] T. Ortmaier, U. Seibold, U. Hagn, D. Böhm, H. Reichenspurner, and G. Hirzinger. Autonomy and haptic feedback in minimally invasive robotic surgery. In *Fourth Annual Meeting of the International Society for Minimally Invasive Cardiac Surgery (ISMICS)*, Munich, Germany, June 2001.

[58] A.G. Pedersen, O.B. Petersen, P. Wara, H. Ronning, N. Qvist, and S. Laurberg. Randomized clinical trial of laparoscopic versus open appendicectomy. *Br. J. Surg.*, Feb. 2001.

[59] E.C. Poulton. *Tracking Skill and Manual Control*. New York: Academic Press, 1974.

[60] C. Preusche and G. Hirzinger. Scaling Issues for Teleoperation. In *Proc. of the 5. PHANToM Users Group Workshop*, Aspen, Colorado, 2000.

[61] C. Preusche, T. Ortmaier, and G. Hirzinger. Teleoperation Concepts in Minimal Invasive Surgery. In *Proceedings of 1. IFAC Conference on Telematics Application in Automation and Robotics*. VDI/VDE - GMA, July 2001.

[62] D. Reintsema, J. Vogel, G. Hirzinger, and T. Unterschütz. CORBA - Ein standardisierter Software-Bus für verteilte Anwendungen in der Robotik.

In *Industrielle Automation und Internet/Intranet-Technologie*, VDI-Berichte 1515, pages 181–192. VDI Verlag, 1999.

[63] Raul Rojas. *Theorie der neuronalen Netze*. Springer Verlag, Berlin, 1993.

[64] Jonathan M. Sackier and Yulun Wang. *Computer-Integrated Surgery*, chapter Robotically Assisted Laparoscopic Surgery: From Concept to Development, pages 577–580. MIT Press, 1995.

[65] S. Sauerland, R. Lefering, and E.A. Neugebauer. Laparoscopic versus open surgery for suspected appendicitis. *Cochrane Database Syst Rev*, 2002.

[66] E. Schippers and V. Schumpelick. *Computer-Integrated Surgery*, chapter Requirements and Possibilities of Computer-Assisted Endoscopic Surgery, pages 561–565. MIT Press, 1995.

[67] A. Schweikard, G. Glosser, M. Bodduluri, M. Murphy, and J. Adler. Robotic motion compensation for respiratory motion during radiosurgery. *J. Computer Aided Surgery*, pages 263–277, Sept. 2000.

[68] L. Sciavicco and B. Siciliano. *Modeling and Control of Robot Manipulators*. The McGraw-Hill Companies, Inc., 1996.

[69] K. Sharma, W. Newman, M. Weinhous, G. Glosser, and R. Macklis. Experimental evaluation of a robotic image-directed radiation therapy system. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, 2000.

[70] Norbert O. Stöffler. *Realzeitfähige Bestimmung und Interpretation des optischen Flusses zur Navigation mit einem mobilen Roboter*. PhD thesis, Technical University Munich, 2001.

[71] Floris Takens. Detecting strange attractors in turbulence. *Lecture Notes in Mathematics*, 898:366–381, 1981.

[72] Russel H. Taylor, Stéphane Lavallée, Grigore C. Burdea, and Ralph Mösges, editors. *Computer-Integrated Surgery*, chapter Minimal-Access Surgery, pages 555–557. MIT Press, 1995.

[73] Michael R. Treat. *Computer-Integrated Surgery*, chapter A Surgeon's Perspective on the Difficulties of Laparoscopic Surgery, pages 559–560. MIT Press, 1995.

[74] UC Berkeley. http://robotics.eecs.berkeley.edu/, April 2001.

[75] URS Universal Robot Systems. http://www.urs-group.com/orth/, April 2002.

[76] Vista Technology Development. http://www.vistadevelopment.com/, Jan 2003.

[77] C. Wagner, N. Stylopoulos, and R. Howe. The role of force feedback in surgery: Analysis of blunt dissection. In *Proc. of the 10th Annual Haptics Symposium*, March 2002.

[78] Y. Wang, K. Grace, and D. Uecker. Motion minimization and compensation system for use in surgical procedures. US Patent Number: 5,971,976, Jan. 28. 1998.

[79] G.-Q. Wei, K. Arbter, and G. Hirzinger. Automatic tracking of laparoscopic instruments by color-coding. In *Proc. First Int. Joint Conf. CVRMed-MRCAS '97, Springer Verlag: Lecture Notes in Computer Science (1997)*, Grenoble, France, March 20-22 1997.

[80] Joachim Weickert. *Anisotropic Diffusion in Image Processing*. B.G. Teubner, Stuttgart, 1998.

[81] A.S. Weigend and N.A. Gersehnfeld. *Time Series Prediction. Forecasting the Future and Understanding the Past.* Addison Wesley, 1993.

[82] D. Winkeler, HE. Vitzthum, and V. Seifert. Spinal markers: A new method for increasing accuracy in spinal navigation. *Journal of Computer Aided Surgery*, 1999.

[83] Laurence B. Wolf. On the relative brightness of specular and diffuse reflection. In *Proceedings Conference Computer Vision and Pattern Recognition*, pages 369–376. IEEE, 1994.

[84] W. Zylka and J. Sabczynski. Effects of localization devices and registration methods on the accuracy of stereotactic frame systems predicted by the Gaussian approach. *Journal of Computer Aided Surgery*, 1999.