Christoph H. Staub

Micro Endoscope based Fine Manipulation in Robotic Surgery

Dissertation

Technische Universität München

TECHNISCHE UNIVERSITÄT MÜNCHEN
Lehrstuhl Robotik und Echtzeitsysteme

Micro Endoscope based Fine Manipulation in Robotic Surgery

Christoph H. Staub

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:		Univ.-Prof. Dr. Nassir Navab
Prüfer der Dissertation:	1.	Univ.-Prof. Dr. Alois Knoll
	2.	apl. Prof. Dr. Robert Bauernschmitt

Die Dissertation wurde am 22.04.2013 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 09.11.2013 angenommen.

Acknowledgement

I would like to thank

Prof. Alois Knoll and Prof. Robert Bauernschmitt for supervising my thesis
Dr. Gerhard Schrott, Dr. Hermann Mayer, Dr. Reinhard Lafrenz, Amy Bücherl, Gisela Hibsch, Monika Knürr, and Gertrud Eberl for supporting me in all affairs

all my colleagues, especially the crew of the robot lab

my parents for their constant support

Abstract

Robotic surgery has pioneered minimally invasive interventions with respect to the recovery of dexterity and visiomotor control. In addition to the technical progress made in telemanipulation technology, the accomplishment of partially autonomous performed sequences shows promising results in supporting surgeons during delicate and time-consuming tasks. However, the established workflow of preoperative planning, registration, and ensuing task execution suffers from the inherent calibration inaccuracies of surgical robots. Currently, planned tasks are merely replayed, since an adaption of the mapping between the execution plan, the patient, and the manipulators is difficult due to calibration uncertainties. In the scope of this thesis we aim to transit toward a more flexible approach by acquiring task-relevant information in situ at the time of execution. In this, *in situ knowledge acquisition* plays the major role to perceive the environment, in which visual information is the driving force. To further strengthen autonomy, *interactive system control* supports the surgeon during task execution.

A key concept of *in situ knowledge acquisition* is to enhance the limited view of the conventional laparoscope with a task-specific view that provides sensor data optimized for the relevant perspective. Therefore, an instrument-mounted camera is introduced: a surgical instrument is augmented with a miniaturized stereo endoscope to recover the relationship between tool and tissue. Since camera and instrument share a common coordinate frame, most calibration uncertainties of the system can be avoided and tasks can be executed with machine precision. A newly developed micro projector enhances the prevalent homogeneous texture of tissue by projecting a pattern onto the surface. The mask is encoded with a globally unambiguous Hamming pattern, thus enabling a combination of stereo matching and structured light. Further, we present an approach for markerless surgical instrument detection and tracking. The method fuses image-based tracking with additional sensor input and is independent from the instrument's appearance.

By *interactive system control*, we propose a novel approach for human machine interaction with the aim to quickly and intuitively access system commands. To this end, the user is given the possibility to start robotic assistance by triggering the relevant functionality by gesture-based commands, performed with haptic devices at the master console. Given the acquired situs knowledge, image-based control laws are derived that allow an autonomous alignment with the target region. The task of tissue dissection was chosen to explain the procedure. We further introduce methods to assist the surgeon during the cutting process by providing haptic guidance during this fine manipulation task. The necessary precision in motion planning and feedback generation becomes possible with the hand-mounted micro endoscopes.

A realistic setup for minimally invasive robotic surgery has continuously be enhanced and served as platform for the experiments conducted.

Zusammenfassung

Der Einsatz von Robotern in der Chirurgie hat minimal-invasive Eingriffe hinsichtlich der Handhabung von Instrumenten und der Hand-Auge-Koordination erheblich vereinfacht. Neben dem technischen Fortschritt auf dem Gebiet der Telemanipulatoren lässt die Durchführung von (teil-) autonomen Sequenzen auf eine Unterstützung der Chirurgen bei komplexen und zeitaufwändigen Aufgaben hoffen. Der derzeit etablierte Arbeitsablauf von präoperativer Planung, Registrierung, und anschließender Durchführung der Aufgabe leidet jedoch unter inhärenten Kalibrierungsungenauigkeiten des Robotersystems. Geplante Aufgaben werden derzeit lediglich wiedergegeben, da eine Anpassung des Aufsführungsplans bezüglich des Patienten und der Roboter aufgrund von Kalibrierungsproblemen schwierig ist. Im Rahmen dieser Dissertation wird ein flexibler Ansatz angestrebt, welcher aufgabenrelevante Informationen innerhalb des Operationsgebietes während der Durchührung erfasst. Hierbei spielt der *Erwerb von Szenenwissen* eine bedeutende Rolle, mit dem Hauptaugenmerk auf visuell erfassbaren Informationen. Um den Gedanken eines autonomen Verhaltens weiter zu stärken, wird der Chirurg mittels *interaktiver Systemsteuerung* unterstützt.

Ein wesentliches Konzept des *Erwerbs von Szenenwissen* ist es, das eingeschränkte Sichtfeld des herkömmlichen Laparoskops durch eine aufgabenspezifische Perspektive zu ergänzen. Hierzu wird eine Mikro-Kamera, welche an einer Instrumtenspitze befestigt wird vorgestellt: ein chirurgisches Instrument wird mit einem miniaturisierten Stereo-Endoskops ausgestattet, welches das Verhältnis zwischen Instrument und Gewebe erfasst. Da sich Kamera und Instrument auf das gleiche Koordinatensystem beziehen, können viele der systembedingten Kalibrierungsprobleme vermieden werden. Ein neuartiger Mikroprojektor verbessert das weitgehend homogene Erscheinungsbild des Gewebes durch Projektion eines Musters. Positionen innerhalb des Musters sind durch einen global eindeutigen Hamming-Code identifizierbar, welcher eine Kombination von Korrespondenzsuche und "structured light" zur Stereorekonstruktion ermöglicht. Des Weiteren wird ein Ansatz zur markerlosen Erkennung und Verfolgung von chirurgischen Instrumenten vorgestellt. Die Methode kombiniert ein bildbasiertes Verfahren mit zusätzlichen Informationen von Sensoren, wobei das Erscheinungsbild des Instrumentes irrelevant ist.

Mittels einer *interaktiven Systemsteuerung* wird ein neuartiger Ansatz in der Mensch-Maschine-Kommunikation vorgestellt, welcher das Ziel hat, Systembefehle schnell und intuitiv auszulösen. Hierfür wird dem Benutzer die Möglichkeit gegeben Assistenzfunktionen durch Gesten, welche an den haptischen Eingabegeräten der Masterkonsole ausgeführt werden, zu starten. Anhand des gewonnenen Szenenwissens werden verschiedene visuelle Regler vorgestellt, welche den Arzt während der Ausführung von Aufgaben unterstützen. Der Ansatz wird anhand des chirurgischen Schneidens exemplarisch umgesetzt. Durch haptische Rückkopplung wird der Chirurg während des Schnittes geführt. Die für die Bewegungsplanung und Feedbackgenerierung erforderliche Genauigkeit wird durch die am Instrument befestigten Mikro-Endoskope ermöglicht.

Ein realistisches Robotersystem für minimal-invasive Chirurgie wurde fortlaufend verbessert und diente als Platform für die durchgeführten Experimente.

Contents

1 Introduction to Medical Robotics

This thesis is conducted in the interdisciplinary field of minimally invasive robotic surgery. We present a novel approach, termed online surgery, which dynamically generates the necessary execution plan at the time of task execution. The approach goes beyond traditional surgery, where static "offline"-planned tasks are merely replayed without adequate adaption to the dynamic environment. At this end, we augment surgical instruments with stereoscopic micro endoscopes that provide images always from the task-relevant perspective. Therewith, we aim to overcome current restrictions of the offline paradigm, particularly evident during assisted and autonomous control. Situs knowledge acquisition, specifically depth perception and instrument localization, plays a major role in online surgery. The knowledge gained is employed for interactive system control, which comprises the two aspects of corrective motion planning and intuitive system control.

1.1 From Offline toward Online Surgery

In the last two decades, medical robotics has undergone an astonishing development from basic research and feasibility studies to commercial products that found their place in the daily routine work of physicians. Besides the development of new hardware, research has branched out to a variety of areas. New aspects of micro-scale manipulation, innovative instrument concepts, and rehabilitation robotics have emerged. The entire scope of the area is reflected in the survey "Special Issue on Medical Robotics" [48, 49]. Taylor and Stoianovici provide a broad overview of about 35 different computer-integrated systems [192]. In [90, 56, 70] the reader is sent to a three-part journey with emphasis on the technological challenges and system design considerations of today's surgery systems.

The application of robotic surgery can roughly be classified into "image-guided" procedures and those aimed to obtain minimal "invasiveness" [47], while the boundaries are becoming increasingly blurred. Image-guided procedures pioneered robotic surgery. The alignment of diagnostic data with the patient's anatomy in combination with machine precision enabled a planned and targeted treatment of previously iden-

tified structures with a new level of accuracy. In its original definition, image-guided procedures rely on static execution plans which are executed by the robot. The plans are created *offline*, based on preoperative diagnostic imaging, such as computer tomography or magnet resonance tomography. The acquired knowledge about the patient's anatomy is then used to generate robot-executable trajectories. In the operating theater, the plan is merely replayed, without giving the surgeon the possibility to actively intervene in the execution. With the transition to minimally invasive robotic surgery (MIRS), the reply of offline created plans is difficult, since the robot trajectories need to be adjusted to a dynamic environment. We take up this challenge and propose *online surgery*. This approach aims to adjust robot motions necessary for an (partial) autonomous task execution dynamically, based on information perceived directly at the surgical site in parallel to the actual task execution. Before we explain the concept of online surgery more detailed, we consider the drawbacks of offline surgery.

Image-guided surgery, which relies on offline generated plans, can predominantly be found in surgical domains that allow for a reliable registration between preoperative data and the patient. Rigid structures are suitable by nature, but also whole-body patient tracking during radiotherapy is in everyday clinical practice [177]. A prominent representative is the Robodoc™ assistant for orthopedic interventions [191]. The lack of ability to perceive detailed information about the currently processed tissue and to adapt the generated plan accordingly has already led to complications with this system. Because Robodoc™ could not differentiate bone from soft tissue, nerves were harmed occasionally during drilling.

offline workflow

The currently established workflow of offline surgery is divided into *sequentially* performed steps. The steps specify a loop of "preoperative planning → intraoperative registration → and plan execution". According to Yaniv et al., six key enabling technologies play an important role [214]. The identified driving devices and visualization technologies are *preoperative imaging*, *segmentation*, *registration*, *tracking*, *data visualization*, and *human-machine interaction* (HMI). While we do not dwell on medical imaging, we would like to briefly introduce the essential technological aspects to clarify the established workflow.

- *Image segmentation* is inherently coupled with the planning phase of image-guided procedures. The acquired image data is partitioned into non-overlapping connected regions. Identified regions can then be matched to distinct anatomical structures to quantify the dimensions of structures and to define instrument trajectories.
- *Registration* helps to combine the gained knowledge by aligning multiple corresponding structures into a single reference frame, such that spatial correspondences between all frames coincide. Regarding robotic surgery, registration links the execution plan to both the patient's anatomy and the manipulators. This step decisively determines the achievable accuracy and is currently the limiting factor. This is especially true for minimally invasive interventions, where image-based registration methods, which search for natural landmarks or fiducial markers, are the only applicable tracking technique.

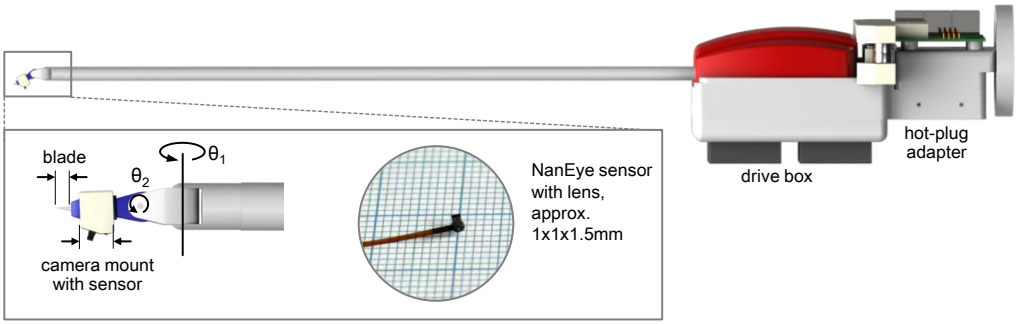


Figure 1.1: Concept of camera-augmented instrument, illustrated on the basis of an EndoWrist SnapFit™ scalpel and NanEye™ micro cameras.

- *Human-machine interaction* deals with the question of how functionalities offered by the surgical system can be operated by, or in cooperation with, the surgeon. Early systems relied on traditional computer interfaces such as mice, keyboards, and foot pedals. More recently, researchers investigate methods such as voice activated control, gesture-based input, and gaze contingent control to intuitively trigger commands.

This definition of Yaniv is strongly influenced by the application of augmented reality, which aims to enhance the limited field of view with the consistent presentation of virtual patient data. Here, the visualization of poses of tracked instruments, indication of target areas such as tumorous regions or lesions, facilitate surgeons in completing their spatial ability of the presented anatomical structure. However, the definition applies in the same way to robotic surgery.

In summary, the traditional workflow of image-guided medical procedures is currently driven by the strong separation of data acquisition and an ensuing task execution. The offline planning phase of an intervention has the objective of creating robot-executable trajectories, based on preoperative imaging. To date, a mapping of the different information sources into a uniform representation is necessary to transfer the generated plan to the operating theater. Various registration routines between the patient, the manipulators, and the generated model, play a crucial role to accurately superimpose the relevant data on anatomy. The uncertainties inherent in minimally invasive interventions pose, however, strong limitations on this approach. The non-rigid anatomy associated e.g. with thoracic and abdominal surgery makes it difficult to apply static plans. Because direct visual feedback is replaced by indirect laparoscopic feedback, registration can only be performed by means of the camera.

Motivated by these challenges, we aim to achieve a transition from the current offline surgery toward the dynamic approach of online surgery. It drives our methodology of assisting surgeons in that context and environment information necessary for planning are acquired in situ, thus *online* at the time of task execution. Hence, we do not only replay preplanned trajectories, but are able to provide up-to-date situs knowledge in order to reliably implement task assistance. In doing so, *adaptive* ***in situ knowl-***

online surgery

edge acquisition and ***interactive system control*** constitute the approach. Adaptive knowledge acquisition aims to model exactly the situs part of interest for task planning, i.e. by depth perception and tool tracking. For this purpose, we introduce a camera-augmented instrument that is equipped with a miniaturized stereo endoscope, as illustrated in Fig. 1.1. The resulting perspective provides sensor data always from the relevant (task-specific) position. Since the sensor is aligned with the instrument coordinate frame, the mentioned error-prone registration steps can largely be avoided. Based on these precise measurements, we derive control laws that allow correcting fine-scaled instrument movements and guidance of the operator. Being a part of interactive system control, this kind of *instrument control* aims to reduce the mental workload of surgeons and assists in managing complexity of interventions through contextual systems. Beyond, *system control* focuses on providing intuitive input channels to invoke system commands.

To gain a better understanding of the challenges and constraints associated with the application domain we focus on, let us first illustrate the general concept of minimally invasive surgery. A more technical perspective of telesurgery can be found in Sec. 5.1. Afterwards, we continue to address the methods of adaptive situs modeling and interactive system control in greater detail.

1.1.1 What is Laparoscopy?

The technique of minimally invasive surgery pioneered its way into various surgical disciplines at the beginning of the 1980s. MIS differs to conventional surgery in using long instruments through small incisions inserted into the patient. These so-called "key holes" or "trocars" are approximately 1cm in length. Three instruments with trigger-like handles are typically used: two surgical tools (one for each hand) and the laparoscope, a camera that is usually guided by an assistant. Patients benefit, in addition to cosmetic advantages such as less trauma and scarring, from reduced pain, shorter hospitalization, and shorter rehabilitation [125]. These facts are obviously a potential for hospitals to reduce costs. The patients benefit came, however, at the expense of surgeons.

drawbacks

Many drawbacks of the approach can be attributed to the loss of spatial perception and require much effort and training to be controlled: the tip of the instruments can not be oriented arbitrarily, restricting the movements to four degrees of freedom (DoF) inside a conical workspace. With the loss of the "wrist" at the distal end, surgeons had to cope with reduced dexterity, which is particularly evident during complex movements as they are typical for delicate tool-tissue interaction. Visual and motor spaces are not consistent anymore, resulting in cumbersome hand-eye coordination [36]. The trocar causes a leverage effect at the fulcrum point, reversing the surgeon's motion and leading to an unequal acceleration of instrument tip and handle, which is dependent on the insertion depth of the tool, cf. Fig. 1.2. Friction disturbs the perception of forces and the palpation of vasculature is impossible due to the lack of tactile feedback. Surgeons are faced with reduced depth perception due to a two-dimensional endoscopic view. Since the patient's anatomy can not be observed directly anymore, the reduced sight yields

severe orientation conditions. The quest for decreasing the size of surgical openings led to the advanced concepts of single port and natural orifice transluminal endoscopy surgery (NOTES) [40]. While single port surgery is carried out through one abdominal incision, NOTES uses esophagus, stomach or vagina to access the abdominal cavity. Needless to say that instrument control therewith gets even more challenging.

To ease operation, the introduction of master-slave technology strikes to recover manual dexterity and visio-motor control. Instead of directly manipulating the surgical instruments, the surgeon sits at a master console and directs the movements of a remote telemanipulator [1]. Two haptic input devices, one for each hand, allow for an intuitive operation of two surgical instruments. Mechanical wrists at the distal end of the surgical instrument recover mobility in six degrees of freedom and assist the surgeon with motion scaling for reduced gross hand movements and suppression of human tremor. The master console, also called medical workstation, typically has the capability to display 3D images of the situs, provided by stereoscopic optics at the slave.

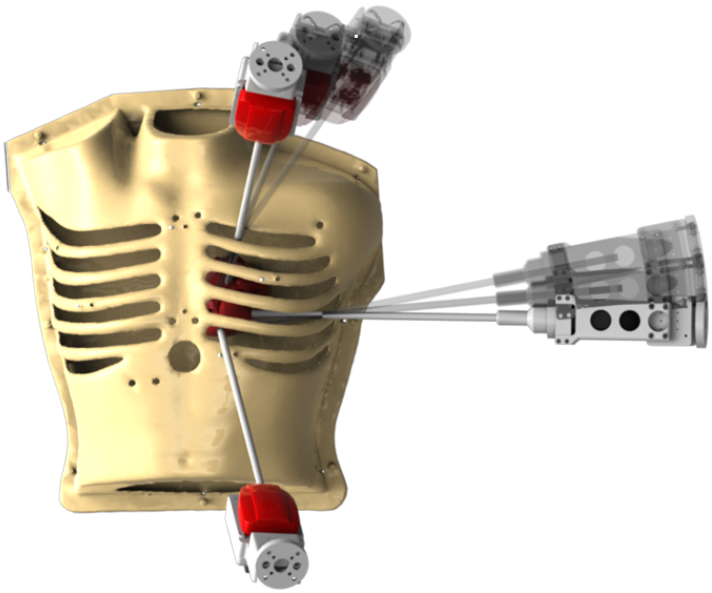


Figure 1.2: Instrument movements in MIRS with remote center of motion.

In order to enhance safety, accuracy, and task completion time, researchers investigate in autonomous control [127]. The level of autonomy is governed by the degree of user interaction. While in manually performed telesurgery the surgeon compensates for any tissue movement by visually closing the servo-control loop, selectively automated (sub-) tasks allow an active intervention of the system in this control loop, moving the surgeon up in the "hierarchy of controllers". For instance, fully automated actions are performed solely by the robot, interpreting higher-level directives and assigning the surgeon a supervisory role. In doing so, the control-loop is closed locally at the slave by substituting sensory information and perceiving the environment. In shared control the surgeon and the robot achieve task completion in working cooperatively.

autonomy

The system aids the surgeon while he remains in full control at all times. The control loop is closed at the master-side, incorporating sensory information of the slave in combination with human input commands. Online surgery focuses heavily on the latter type.

1.1.2 The Challenge of Autonomy

Autonomous control is difficult to implement for the predominantly present soft tissue of MIRS. The limited view and the dynamic environment impede access to relevant information that models the relation between the surgical instrument and the situs. In the sequel, we consider different autonomy modes to elucidate the issues of acquiring task-relevant knowledge in situ and using the information gained to deduce system control laws.

As aforementioned, full autonomy enables a robot to perform a certain task independently, e.g. tying a knot, assuming access to sufficient background knowledge about the task to be performed, the operator's intentions, and the environment. In this case, the system would respond to all kind of incidents during the execution.

Partial autonomy requires a more refined differentiation, with a distinct and universally acceptable description seems to be difficult. The most generic definition is the joint execution of a task by man and machine. However, the question of whether the task is carried out in alternation between human operator and robot, or simultaneously, remains to be clarified. Consider for instance performing a running suture with three needle insertions. [143] execute the task in alternation with the system by breaking it up into different subtasks. The operator takes those task portions that require fine manipulation skills, i.e. grasping the needle and punctuating tissue. The system recognizes transitions between the individual task steps and automatically proceeds with transportation movements, such as pulling the thread and handing the needle to the second instrument. No environment cues are considered, disqualifying the method for subtasks that involve interaction with tissue or tools. Consequently, the operator needs to grasp the needle manually before the robot automatically completes the task. The automated movements are acquired by temporal averaging of multiple user demonstrations and superimposed on haptic input devices during replay. Learning by demonstration often serves as concept in human-machine skill transfer [7, 19]. Many approaches are, however, limited to the acquisition and reproduction of movements without considering environment dynamics [203]. Although methods for trajectory adaption have been proposed [117], their implementation is difficult due to incomplete or obsolete situs knowledge. This fact explains the current restriction on transportation movements, which are easier to implement since the uncertainties arising are mainly attributed to the system. For a generalized execution an accurate model of the situs is essential.

If tasks are processed simultaneously by human and robot, the latter can either solve its own subtask concurrent to human input or assist the operator to improve his task performance. [145] use automated scissors to cut a thread, at which the loose ends are held by operator controlled micro grippers. Assistance, seen on the background of improving manual task performance in terms of accuracy, speed, and quality, is often

achieved by employing the concept of "virtual fixtures" (VF). Introduced by Rosenberg, virtual fixtures overlay telemanipulation tasks with abstract sensory information [168]. A fixture can essentially be characterized in two ways: forbidden-region virtual fixtures restrict the operator from entering certain areas, while guiding virtual fixtures augment the surgeon's ability to perform complex procedures. Visual or auditory signals can convey the fixture information. In remote surgery it is useful to implement the fixture by means of haptic feedback. Hence, haptic virtual fixtures are also referred to as "active constraints" or "virtual walls".

A major drawback, shared by all types of haptic virtual fixtures is the precise generation of the constraint information. More precisely, the question of how to obtain the geometric distance between the manipulator and a potential fixture and how to relate this measurement to the master-side haptic device is essential and significantly affects the quality of the feedback. Although observing the objective defining the virtual fixture visually has the potential to cope with the dynamic nature of the situs, the approach suffers from strong inherent registration inaccuracies. The measurements can solely be acquired by either recovering depth information from laparoscopic images or by establishing correspondences between preoperative data and the patient [108]. Preoperative registration of the patient's anatomy with atlas models is often proposed, but as we have seen in the last section, it is less suitable for dynamic environments. Surgical tools restrict the view and relevant parts of the workspace might not be reconstructible due to the limited maneuverability of the camera. In addition, many error sources of telemanipulation systems contribute to a comparably poor overall calibration between endoscopic camera, surgical instruments, situs, and the haptic devices on the master-side. All uncertainties mentioned sum up if the endoscopic camera is used to derive a vision-based fixture for surgical instruments that are mounted on a different manipulator than the camera. The achievable precision is then not sufficient for small-scaled fine manipulation tasks. On this account, the application of virtual fixtures is in many cases still limited to delineate larger regions [149, 62].

trajectory generation

1.1.3 In Situ Knowledge Acquisition

Partial autonomy, in particular virtual fixtures, aims to combine the advantages of human and robot in a common remote workspace. By definition, this implies access to common knowledge. During time-consuming preprocessing steps, telling the system the difference between safe (or target) regions and forbidden regions, the workspace can be segmented preoperatively. The transfer to the system involves an error-prone transformation chain, making it difficult to apply the procedure to fine manipulation tasks. Advanced control will only be realized if the traditional loop of planning, registration, and execution can be overcome and the robot itself becomes an integral part of the planning phase. To do so, we give instant access to relevant and task specific situs details by **augmenting surgical instrument with micro endoscopes**, therewith instantiating our definition of online surgery.

Most surgical activities can either be described by tool-tissue or tool-tool interaction

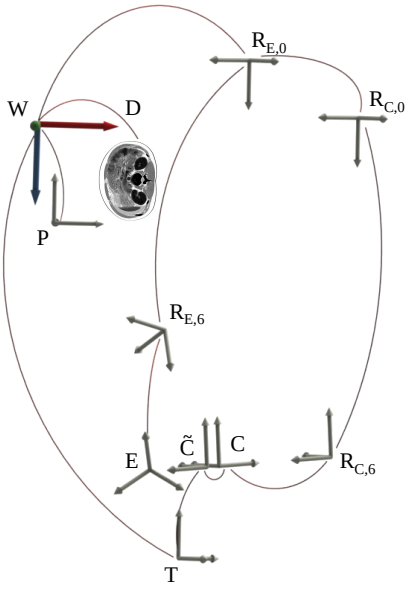


Figure 1.3: Important coordinate frames: $\mathrm{R}_{\mathrm{E},0}$ instrument robot base, $\mathrm{R}_{\mathrm{E},6}$ instrument robot wrist, $\mathrm{R}_{\mathrm{C},0}$ camera robot base, $\mathrm{R}_{\mathrm{C},0}$ camera robot wrist, C camera, $\tilde{\mathrm{C}}$ calibrated camera image, E instrument, T task frame, P haptic device, W world frame, D execution plan.

[134]. Recovering the spatial relationship between surgical tools and anatomical structures, a second tool, or surgical supplies, becomes therewith essential to plan tasks online. The basic steps toward this goal are *tool tracking* (cf. Sec. 3.2) and *surface reconstruction* (cf. Sec. 3.3). The difficulty of their application lies in the nature of minimally invasive interventions, which inhibits any direct view of the surgical field. The use of external tracking devices and imaging modalities, such as fluoroscopy, computer tomography, or magnetic resonance imaging, are often incompatible with robotic systems, emit prolonged radiation, and face serious challenges in achieving sufficient accurate registration results. Deriving situs information directly from the laparoscope is one of the few remaining options, but suffers from strong calibration inaccuracies. The problem is further tightened if the endoscopic image stream needs to be registered with preoperative models. More detailed, major drawbacks can be identified as follows:

situs knowledge

1. **Erroneous transformation chains.** Telemanipulation systems are operated by means of Cartesian control. The camera and the instrument are attached to two different manipulators, with the consequence being that long and erroneous transformation chains are introduced (see also Fig. 1.5 and Sec. 5.2.3). Fig. 1.3 illustrates the problem by introducing the important coordinate frames. Frame C indicates the pose of the endoscopic camera and $\widetilde{\mathrm{C}}$ refers to a calibrated camera image. The frame indicating the posture of the micro end effector at the instrument's distal end is denoted with E. The task specific coordinate frame T is independent of the system's kinematics, and finally the origin of the world is denoted as W. Image-guided object manipulation now demands an accurate and gapless transformation chain from the global pose of the object to be manipulated,

to the camera and to the surgical tool used for manipulation. This transform in turn is also dependent on the robot base frame $\mathrm{R}_{\mathrm{E},0}$ and the tool center position $\mathrm{R}_{\mathrm{E},6}$ of the instrument robot, respectively $\mathrm{R}_{\mathrm{C},0}$ and $\mathrm{R}_{\mathrm{C},6}$ of the camera robot. If the task execution is based on preoperative imaging, the generated plan D additionally needs to be aligned with the target reference frame. This image-based registration involves the entire transformation chain mentioned above. When it comes to haptic guidance, an accurate alignment of the plan and the world is of particular importance. Due to the Cartesian control, the coordinate frame of the input devices P is typically coincident with the world frame. Thus, smallest deviations yield to misdirected haptic feedback, making the assistance impractical and misleading.

2. **Fixed perspective.** The remote center of motion restricts movements of the laparoscope to a conical workspace. As a consequence, parts of the situs may be viewed poorly and always from a similar perspective. Repositioning the laparoscope for situs exploration is impracticable, since it is time consuming and interrupts the physician in his work routine. Perceiving detailed up-to-date depth information for model building is severely restricted due to occlusions caused by organs and instruments.

3. **Improper field of view.** The relatively fixed perspective of traditional laparoscopes in combination with their wide-angle optics, mentioned under item 2., have direct impact on in situ task planning. Various tasks, in particular fine manipulation of tissue, require the relevant field of view for proper trajectory planning associated with the relationship between the instrument and the target region. During tissue dissection, for instance, the important information is how well the blade follows the structure to be cut. This task-dependent visualization cannot be provided by the conventional laparoscope.

To tackle these drawbacks, we propose to augment surgical instruments with miniaturized endoscopes, which are shown in Fig. 1.4. With the recent advances in sensor technology it has now become possible to extend medical instruments beyond their actual functionality, upgrading them to imaging devices. The concept is illustrated in Fig. 1.1. When mounting an image source to a surgical instrument's tip, it is found that the image is obtained from the relevant task-specific perspective, keeping the distal end of the tool always in field of view. In this way, the relationship between tissue and tool can be derived straightforward, without elongated transformation chains. The combination of imaging and surgical instrument also strengthens autonomy by incorporating the information pertaining to the area of interest. Naturally, the image can be combined with the conventional laparoscope. The creation of new views improves navigation and orientation of the instrument within the operative field. Unlike conventional endoscopes, it can be positioned freely in space. The micro camera shows a very limited, though microscope-like view of the situs. This property enables detailed visual inspection of e.g. suture quality or formerly tumorous tissue.

micro endoscopes

Other surgical domains already benefit from the advances in sensor technology. Lüth for example explored the paranasal sinus by attaching a micro camera on fine-manipulation forceps [113]. When percutaneous body structures are to be visualized,

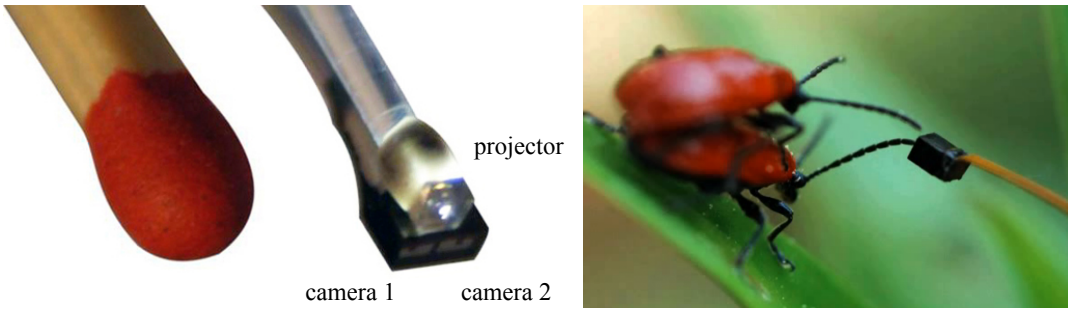


Figure 1.4: Proposed micro camera system. (a) stereo camera pair with additional pattern projector; (b) single NanEye™ camera [image courtesy of Awaiba GmbH].

intraoperative ultrasonography and optical coherence tomography can be used. While [103] aims at minimizing the risk of an accidental dissection of sub-surficial vessels by ultrasound probing, the latter method is currently used mainly in retinal surgery. The Medigus SRS™ system for intraluminal treatment of gastroesophageal reflux disease combines a surgical stapler, ultrasonic sights for accurate positioning, and a video camera in a single flexible tube [24].

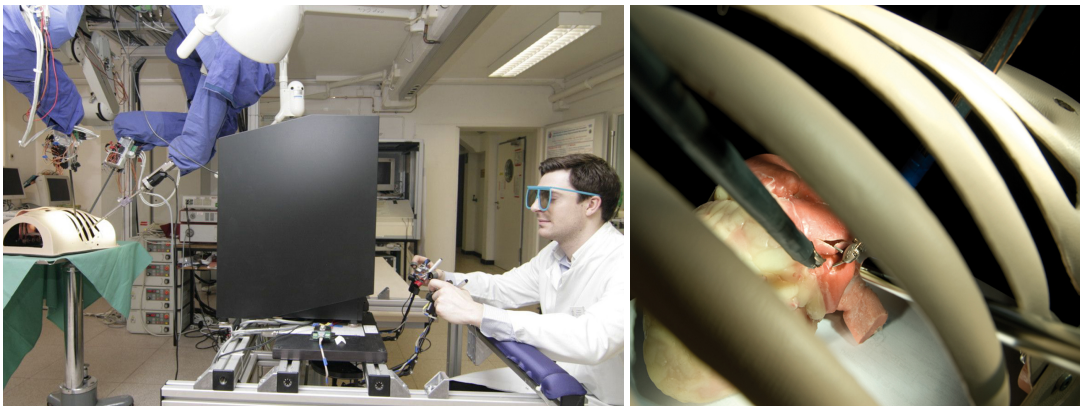
1.1.4 Interactive System Control

The demand for human interaction and supervision decreases with an increasing level of autonomy. Many surgical tasks require machine precision, especially in terms of accuracy, but human intelligence and supervision at the same time. In our efforts to transit MIRS toward a more intuitive and contextual responsive system, we integrate task-specific information to ease task execution. The information can either be based upon explicit user directives or it can be derived from implicit cues. The cues in turn arise either directly from the situs model obtained or are set by the operator.

control types

The type of parameters necessary to specify a certain system behavior can be very different in nature. Environment-dependent parameters, i.e. those introduced in the preceding section, are typically required by tasks that involve tool interaction. We group those tasks into *instrument guidance* and *automated instrument control*. Instrument guidance assists the user in achieving superior task performance in terms of accuracy or execution speed. This includes hybrid control schemes. [58] for instance, controls the position of a manipulator by means of visual servoing based on ultrasound images, while the velocity of the motion is manually set by the surgeon. Similarly, the control of individual degrees of freedom of a manipulator can be split between system and user. In contrast, automated instrument control generates motion commands based on the perceived situs knowledge without any user input. Both control schemes usually require knowledge about a tool's position, the relationship between tool and target, or both to adjust the control loop.

The specific *call of a system function* that triggers system behaviors, including those de-



(a) (b)

Figure 1.5: Setup of our telesurgery system at the German Heart Center Munich: (a) slave-side robots (covered with blue fabric), equipped with surgical instruments and master-side medical workstation, comprising haptic devices and a stereoscopic screen; (b) manipulation scene illustrating the patient-side configuration of the instruments and the trocars (two needle drivers and one stereoscopic laparoscope).

scribed above, obviously depends exclusively on some user choice. However, such a call possibly requests additional information for execution that needs to be defined by the operator, e.g. a target position intended for execution. With the intention to reduce the mental workload of surgeons and to manage complexity of interventions, the question arises which modalities are well-suited to guarantee a smooth integration of such contextual information into the surgical workflow. In general, we ask the following requirements for interactive control:

- shortest possible distraction of the surgeon from the operative situs,
- little cognitive burden and mental stress,
- fast and seamless integration into the surgical workflow,
- correct interpretation and execution of commands,
- little training effort.

The integration of operator knowledge into the surgical workflow plays a central role in enhancing dexterity and usability of surgical robots. We consider the introduced types of system control and assess the required amount of interaction in order to propose draft solutions, which are dealt with in the course of this treatise in more detail.

1. **System commands.** The growing number of functionalities offered by MIRS systems is more and more imbalanced with the currently available input options. This development demands for new interfaces that facilitate the handling. Sign language and haptic gesturing promise to be intuitive and simple to understand. We borrow the concept from everyday life and transfer the idea of interpreting gestures to the movements performed by the operator at the master devices. In

this way, we offer a customizable input modality with fast interaction times. Since haptic gestures define a spatio-temporal context, the method allows integrating location-dependent information, e.g. the user can define the incision points of a suturing task in advance.

2. **Instrument guidance.** This type of system control behavior assists the operator during task execution by consuming the situs knowledge gained. In particular the performance of delicate fine manipulation tasks, such as tissue dissection, benefits from machine precision through system aid. We therefore instantiate shared control by employing haptic virtual fixtures to convey force information. The accurate and task-relevant information derived from our new micro camera-augmented instruments allows smart behaviors, i.e. "snap to" an online generated trajectory and overcomes the calibration limitations mentioned.

3. **Instrument control.** Instrument control adopts autonomous control to relieve the surgeon from small subtasks. Instrument control can be driven by both user input or on intrinsic system knowledge. Assisted targeting for instance helps the user to align the surgical tool with a manually chosen target. While the selection of the target is based on operator input, the manipulator motions necessary for the alignment are derived from situs knowledge. In a similar way, automating the displacement of the camera helps the surgeon to concentrate on his primary task. We investigate two modes to reveal a proper endoscope position. On one hand, the surgeon can direct the robot by means of gaze control. Gaze provides a strong cue and incorporates the user's knowledge of the relevant field of view into the control loop. On the other hand, the instrument's pose can be used to keep the dominant instrument in the field of view.

1.2 Problem Statement and Application

So far, we identified the drawbacks related to traditional offline surgery and autonomy. Although methods of preoperative planning have shown potential in acquiring and learning tasks, less progress has been made for the transfer and execution phase. The adaption of tasks to only slightly different environments currently exceeds most capabilities of current surgical robots. Thus, a certain degree of autonomy could be achieved, but it lacks the necessary adaptivity. To overcome these restrictions, we introduced the method of online surgery that aims to answer three fundamental questions:

- How can system commands be triggered at the master-console in an ergonomic and intuitive way to start assistance, while providing the possibility to pass task specific information, such as the location intended for execution, along with the command?

- How can task trajectories be planed and corrected without relying solely on preoperative data, but taking the dynamic environment of the surgical field into account?

- How can the inherent system uncertainties be overcome, which arise from error-prone transformation chains, to enable autonomous task execution with a high accuracy?

Online surgery tries to refrain from predefined execution plans and obtains the necessary information in situ, in parallel to the actual task execution. A significant and challenging step in this direction is adaptive surgical field modeling. Smart tools that combine imaging and surgical instrument are expected to support this process, i.e. by providing task-specific views. Because the camera is aligned with the instrument's coordinate frame, prolonged transformation chains are bypassed and intrinsic system errors minimized. Therefore, we augment our instruments with endoscopic micro cameras. As a result, task execution can be accomplished by relying on image-derived control laws, which increase machine-precision. We illustrate the steps of online surgery based on the generic use case of **guided tissue dissection**.

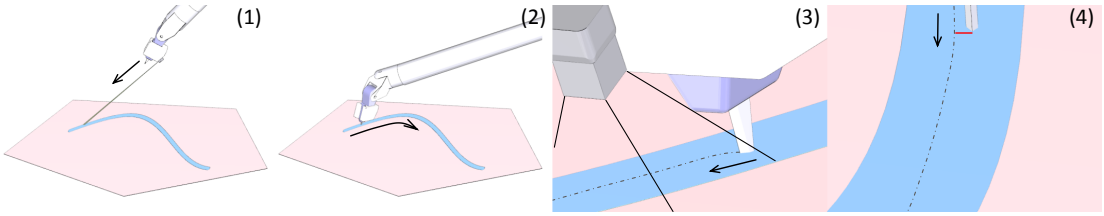


Figure 1.6: Steps of guided tissue dissection: (1) indication of incision point and automatic alignment of blade tip with the selected target; (2) the field of view provided by the conventional laparoscope is commonly inadequate for task planning: important structures cannot be identified and visibility is limited; (3) micro endoscopes (highlighted in gray) are attached at the distal end of the tool. The mounting is colored white. The camera provides images always from the task-relevant perspective, as illustrated in (4). This allows for reconstructing the cut path (dashed line) and measuring the deviation between blade and cut path (red line).

This dexterous manipulation task requires micro-scaled manipulation skills. However, the conventional laparoscope often provides a poor perspective, which is obstructed by the scalpel and impedes a precise trajectory definition. We start the task of guided tissue dissection by calling the corresponding system function using our gesture-based input interface (cf. Sec. 4.2). The procedure can then be divided into independent steps, as illustrated in Fig 1.6.

tissue dissection

1. The surgeon selects the desired incision point of the scalpel within the surgical field. For precision reasons, this is done using a laser beam, where the projected dot indicates the penetration point on the surface. The laser diode is attached to the surgical instrument.

2. A visual servoing scheme maneuvers the scalpel to the assigned target. If necessary, position-based servoing first brings the instrument into the field of view of the laparoscope. Afterwards, image-based servoing is employed to perform the precise alignment with the target (cf. Sec. 4.3). The image features necessary to generate the corresponding motion commands of the manipulators are obtained by tracking the surgical instrument (cf. Sec. 3.2).

3. Once the scalpel is aligned with the structure to be cut, the dissection is started by the surgeon. The image center of the laparoscope is always automatically aligned with the scalpel, while the cut itself is performed by the surgeon under haptic guidance. The system calculates the optimal blade orientation to guarantee a smooth cut (cf. Sec. 4.4). Our camera-augmented instrument with tool-tip mounted micro endoscope recovers the deviation between the blade and the cut path from the task-relevant perspective (cf. Sec. 3.3). In this way, the approach enjoys similar advantages as traditional visual-servoing techniques, but becomes applicable to delegate processes that are not adequately observable by the conventional laparoscope.

In contrast to the traditional laparoscope, the pair of miniaturized laparoscopes can be positioned freely to generate new three-dimensional views of the situs. To facilitate depth recovery on homogeneous surfaces, a micro projector that projects a globally unambiguous pattern onto the scene was developed. Experiments are conducted on our realistic telesurgery system, depicted in Fig. 1.5 and discussed more detailed in Sec. 5.2.1.

2 Terminology

Gaining knowledge about in situ conditions is extensively done by deducing geometry from camera images. Before describing the projective relationship between the three-dimensional world and the two-dimensional images, we define some general terminology with respect to rigid body transformations. We model the problems of instrument tracking and depth perception from a probabilistic point of view. In the center is Bayes' theorem, which we briefly introduce accompanied by the Kalman filter as a frequently used case of application.

2.1 Coordinate Frames and Transformations

In the next sections, we frequently deal with various coordinate frames and their transformation. Therefore, we begin by establishing a uniform notation. Extending the representation of a point $\mathbf{x} \in \Re^3$ in the three dimensional Euclidean space to its homogeneous form $\mathbf{x} = [x_x, x_y, x_z, 1]^{\mathrm{T}}$ allows us to express any rigid transformation $\tilde{\mathbf{x}} = \mathbf{R}\mathbf{x}+\mathbf{t}$ by a single transformation matrix $\mathbf{T}$ as

$$\tilde{\mathbf{x}} = \mathbf{T}\mathbf{x} \quad \text{, with} \quad \mathbf{T} = [\mathbf{R} \mid \mathbf{t}] = \begin{bmatrix} \mathbf{R}_{[3\times3]} & \mathbf{t} \\ \mathbf{0}^{\mathrm{T}}{}_3 & 1 \end{bmatrix}, \tag{2.1}$$

where $\mathbf{R} \in SO(3)$ is the 3×3 rotation matrix and $\mathbf{t} \in \Re^3$ the translation vector. There remains the questions, which coordinate frame the transformation refers to as a reference. For instance, a camera sensor captures images relative to its own coordinate frame. In order to relate the measurement to any other frame, the involved frames need to be aligned. We introduce some dedicated notation for coordinate frames that we are frequently using, i.e. W denotes the world coordinate system and $\mathrm{R_{i,j}}$ refers to the j-th link of robot number i in our system. Frame C is usually associated with a camera. Each couple of frames can be aligned using a transformation ${}^{\mathrm{A}}_{\mathrm{B}}\mathbf{T}$. In this mapping, the reference frame A is superscribed and the target system B is subscribed. Multiple transformations can be composed through non-commutative multiplications, i.e. the posture of a robot's end effector can be expressed relative to its base frame with ${}^{\mathrm{R_i}}_{\mathrm{R_0}}\mathbf{T} = {}^{\mathrm{R_1}}_{\mathrm{R_0}}\mathbf{T} \cdots {}^{\mathrm{R_{i-1}}}_{\mathrm{R_{i-2}}}\mathbf{T}\ {}^{\mathrm{R_i}}_{\mathrm{R_{i-1}}}\mathbf{T}$. Transformations between individual robot joints are

denoted by ${}^{\mathrm{R_{i,j}}}_{\mathrm{R_{i,k}}}\mathbf{T}$, where $j \neq k$. Reference frames associated with vectors are indicated likewise, e.g. ${}^{\mathrm{A}}\mathbf{x}$. If the assignment of coordinate frames is non-ambiguous, we refrain from indexing the variables.

Velocities are represented by the 6-vector $\boldsymbol{\xi} = [\boldsymbol{\upsilon}, \boldsymbol{\omega}]^{\mathrm{T}}$, where $\boldsymbol{\upsilon} = [\upsilon_x, \upsilon_y, \upsilon_z]^{\mathrm{T}}$ is the linear velocity portion and $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^{\mathrm{T}}$ the angular velocity respectively. In many cases it is necessary to relate the velocity of two rigidly connected frames A and B, i.e. between the end effector of a manipulator and an attached tool. Velocities between two rigidly attached frames can be transformed with

$$ {}^{\mathrm{A}}\boldsymbol{\xi} = \begin{bmatrix} {}^{\mathrm{B}}_{\mathrm{A}}\mathbf{R} & {}^{\mathrm{B}}_{\mathrm{A}}[\mathbf{t}]_{\times}\,{}^{\mathrm{B}}_{\mathrm{A}}\mathbf{R} \\ \mathbf{0}_{[3\times3]} & {}^{\mathrm{B}}_{\mathrm{A}}\mathbf{R} \end{bmatrix} {}^{\mathrm{B}}\boldsymbol{\xi}, \tag{2.2} $$

where the quantity ${}^{\mathrm{B}}_{\mathrm{A}}[\mathbf{t}]_{\times} = \mathbf{S}(t)$ is the skew-symmetric matrix defined by the equation

$$ \mathbf{S}(\mathbf{t}) = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}, \tag{2.3} $$

representing the cross product with ${}^{\mathrm{B}}_{\mathrm{A}}\mathbf{t}$ [46] .

2.2 Projective Geometry

Projective geometry describes the mapping between an image sensor and the three-dimensional world. In the sequel, we consider both directions, the relationship of the world to the sensor and the reverse mapping, which describes how image coordinates can be projected back to the world.

3D to 2D Projections: From World to Sensor Space

To describe a three dimensional scene in the two-dimensional image plane of a camera, it is necessary to model the projection geometry of the camera and the spatial relationship between the camera and the world reference frame. Two sets of parameters are used, the *intrinsic* and the *extrinsic* camera parameters.

Intrinsic parameters describe how metric 3D points form camera space project to 2D pixel coordinates. A general acquisition model, which is frequently employed, is the pinhole camera. A small pinhole defines the camera center C through which optical rays enter the camera body, as illustrated in Fig. 2.1(a). The rays are incident on the retinal plane and form a 180 degree rotated representation of the observed reality. The principle axis of the camera is orthogonal to the retinal plane, passing C and intersecting the retinal plane in the principle point $\mathbf{c} = [c_x, c_y]^{\mathrm{T}}$. The distance between the camera center and the principle point is denoted as focal length f. To provide a more convenient representation of the projection, with a reference frame attached at the upper-left corner of the image, a virtual image plane in front of the camera is

usually introduced at the same distance f. The projection matrix $\mathbf{K}$ projects a homogeneous 3D point ${}^{\mathrm{C}}\mathbf{x}$, given in the reference frame of the camera, to ${}^{\mathrm{I}}\mathbf{x} = \mathbf{K}{}^{\mathrm{C}}\mathbf{x}$ in image space with

$$\mathbf{K} = \begin{bmatrix} f_x & \sigma & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{2.4}$$

To scale the metric value of the focal length to pixel space $f \mapsto (f_x, f_y)$, it is normalized with the metric width and height of a single pixel on the CCD camera p_x and p_y respectively, to $f_x = f/p_x$ and $f_y = f/p_y$. The same normalization applies for the metric principle point, which can be expressed in pixels by normalizing its value with the sensor's dimensions along the $x-$ and $y-$ axis. A skew angle σ describes the axis displacement, if the sensor's pixels are not rectangular,

$$\sigma = (\tan \alpha) \cdot f_y. \tag{2.5}$$

Since for CCD and CMOS sensors σ is always zero, we can finally describe the projection model with

$$x = f_x \frac{{}^{\mathrm{W}}x_x}{{}^{\mathrm{W}}x_z} \qquad \text{and} \qquad y = f_y \frac{{}^{\mathrm{W}}x_y}{{}^{\mathrm{W}}x_z}. \tag{2.6}$$

Particular optics with a short focal length suffer from additional nonlinear distortion effects, which are more pronounced with increasing distance from the image focus point. Distinction is usually made between radial distortion, which locally alters the scale and causes straight world lines to project onto curves in the image, and tangential distortion, which results from an off-centered lens alignment. To account for the distortion effects, first a nonlinear transformation function $D(\cdot)$ is applied, followed by the calibration matrix, therefore ${}^{\mathrm{I}}\mathbf{x} = \mathbf{K}D({}^{\mathrm{W}}\mathbf{x})$. Also compare appendix A.1.

The spatial relationship between the camera frame C and the world reference frame W is governed by the extrinsic camera parameters, expressed as an Euclidean transform ${}^{\mathrm{W}}_{\mathrm{C}}\mathbf{T}$. Combining both intrinsics and extrinsics, an arbitrary scene point ${}^{\mathrm{W}}\mathbf{x}$ projects from the world frame to image space with

$$\begin{aligned} {}^{\mathrm{I}}\mathbf{x} &= \mathbf{K}_{[3\times3]}\ {}^{\mathrm{W}}_{\mathrm{C}}\mathbf{T}\ {}^{\mathrm{W}}\mathbf{x} = \mathbf{K}_{[3\times3]}\ {}^{\mathrm{W}}_{\mathrm{C}}[\mathbf{R} \mid \mathbf{t}]\ {}^{\mathrm{W}}\mathbf{x} && (2.7) \\ &= {}^{\mathrm{W}}_{\mathrm{I}}\mathbf{P}_{[\mathbf{3}\times\mathbf{4}]}\ {}^{\mathrm{W}}\mathbf{x}, && (2.8) \end{aligned}$$

where $\mathbf{K}_{[3\times3]}$ is the left (3×3) sub-matrix of $\mathbf{K}$ and $\mathbf{P}_{[3\times4]}$ is denoted as the projection matrix.

2D to 3D Projections: Depth by Triangulation

Reconstructing the actual 3D position of an image point, given in at least two different camera views, is known as stereo matching. For stereoscopic setups, the spatial relationship between the two cameras is known and the world coordinate can be recovered by triangulation, as illustrated in Fig. 2.1(b). The distance between the two camera frames, called *baseline* b, causes a horizontal displacement of the projection in

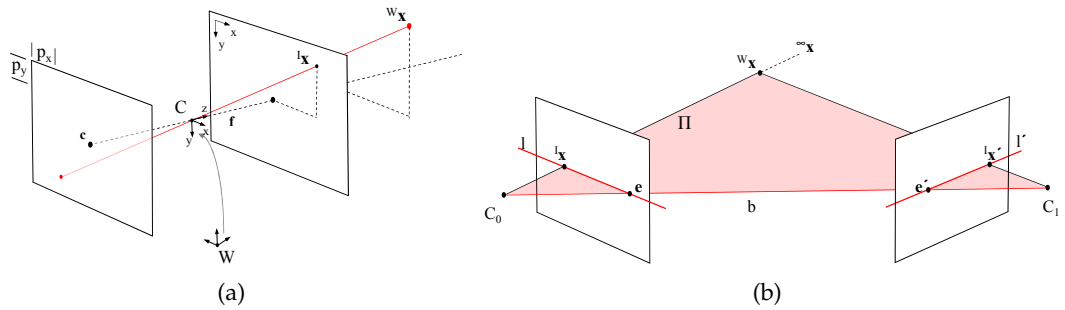


Figure 2.1: (a) camera geometry; (b) epipolar geometry.

one image with respect to the second image. This horizontal *disparity* value d describes the relationship between corresponding pixel coordinates as

$$x' = x + sd(x, y) \quad \text{with} \quad y' = y, \tag{2.9}$$

where $s = \pm 1$ always ensures a positive disparity value, regardless of whether d is calculated from the left to the right image or vice versa. The depth x_z of a world coordinate can then be expressed with

$$x_z = \frac{fb}{d}, \tag{2.10}$$

where f is the focal length measured in pixels.

The problem of recovering depth from a scene reduces therewith to finding corresponding pixel matches in both images. Equation (2.9) describes the horizontal disparity value, assuming that images are coplanar and corresponding pixels can be found in the same horizontal scanline of both images. This constraint speeds up matching and increases its reliability, since instead of looking for vectors between points only scanlines need to be compared. For non axis-aligned camera configurations, with known calibration parameters, images need to be *rectified*, i.e. pre-warped, so that corresponding *epipolar* lines are coincident in both images. This characteristic is defined by the epipolar geometry: the projection of a scene point ${}^{\mathrm{W}}\mathbf{x}$ to its corresponding image point ${}^{\mathrm{I}}\mathbf{x}$ in the left camera image defines a ray between the left camera's center and the projection on the image plane. All points along this ray project to the same pixel in the left image. The right camera observes all of these possible point correspondences as a line. The epipolar plane Π is passing through the cameras center and the scene point. Its intersection with both image planes defines the epipolar line. The correspondence search in the right camera image can therefore be restricted to a segment of the epipolar line, bounded at one end by the projection $\mathbf{e}'$ of the left camera's optical center in the right image, referred to as *epipole*, and the projection $\mathbf{x}'$ of the original world point at infinity ${}^{\mathrm{W}}\mathbf{x}_{\infty}$.

As above-mentioned, input images need to be rectified to arrange corresponding epipolar lines. For a calibrated stereo rig, with known intrinsics and extrinsics, the relationship between the two camera frames C_0 and C_1 is given by a rotation $\mathbf{R}$ and a translation $\mathbf{t}$. An image point given in C_0 can be expressed in C_1 as $\mathbf{x}' = \mathbf{R}\mathbf{x} + \mathbf{t}$. The epipolar

constraint implies that the vectors $\mathbf{x}, \mathbf{x}'$ and $\mathbf{t}$ are coplanar, thus one of the vectors must lie in the plane spanned by the other two, or

$$\mathbf{x}\left[\mathbf{t} \times (\mathbf{R}\mathbf{x}')\right] = 0, \tag{2.11}$$

which can also be written as

$$\mathbf{x}^{\mathrm{T}}\mathbf{E}\mathbf{x}' = 0. \tag{2.12}$$

This equation defines the relationship of all pairs of point correspondences, where $\mathbf{E}_{[3\times3]}$ is called the *essential* matrix

$$\mathbf{E} = [\mathbf{t}]_{\times}\mathbf{R}. \tag{2.13}$$

Note that for two corresponding image coordinates the essential matrix describes the epipolar line for $\mathbf{x}$ in the other image, on which $\mathbf{x}'$ must lie, thus

$$l' = \mathbf{E}\mathbf{x}. \tag{2.14}$$

Given the corresponding measurement points $\mathbf{x}$ and $\mathbf{x}'$ we can recover the depth of the original world point by intersecting the two rays originated in each optical center and passing through the image point. In consequence of calibration uncertainties, the rays are not expected to intersect in exactly one point. Instead, the 3D coordinate is estimated as a minimization of the distance between both rays.

2.3 Bayesian Probabilities

Many problems are considered from a probabilistic point of view. Thus, let us consider two random variables X and Y, defined on the same probability space. The joint probability that X will take the value x_i and Y will take the value y_i is written as $\rho(X = x_i, Y = y_j)$, with $i = 1, \cdots, M$ and $j = 1, \cdots, N$. In brief, we simply write $\rho(X, Y)$ for the probability of X and Y. The quantity $\rho(X|Y)$ is the probability of X given Y, referred to as the conditional probability. The quantity $\rho(X)$ is simply the probability of X, also known as the marginal probability.

The product rule of probability $\rho(X, Y) = \rho(Y|X)\rho(X)$ together with the symmetry property $\rho(X, Y) = \rho(Y, X)$ immediately gives us an equation that is well-known as *Bayes' theorem*

$$\rho(X|Y) = \frac{\rho(Y|X)\rho(X)}{\rho(Y)}. \tag{2.15}$$

The theorem describes the dependency of a conditional property on its inverse [33].

In Bayesian probability theory, probabilities are interpreted as a quantification of uncertainties instead of being treated as frequencies of random, repeatable events. From this perspective, we can express the uncertainties in model parameters that describe a certain observation. A set of assumed model parameters $\mathbf{s}$ then replaces the event represented by the random variable X, and Y is now an observation of data $\mathcal{D}$ that we try to explain with the model. We first evaluate $\mathcal{D}$ and then express the uncertainty in

the model parameters $\mathbf{s}$ for the observation as *posterior* probability $\rho(\mathbf{s}|\mathcal{D})$. The evaluation of the observed data on the right-hand side of (2.15), given the model parameters, is called *likelihood* distribution $\rho(\mathcal{D}|\mathbf{s})$. It expresses how well the parameters describe the current observation. It is weighted with a *prior* distribution $\rho(\mathbf{s})$. This prior knowledge reflects an assumption about the parameter distribution in advance of the actual observation. The denominator of (2.15) ensures the integral of $\rho(\mathcal{D}|\mathbf{s})$ to be one. In short,

$$\text{posterior} \propto \text{likelihood} \times \text{prior} \tag{2.16}$$

converts a prior distribution into a posterior distribution by incorporating the measurement of how well a certain parameter set explains the observation. Maximizing the posterior term in order to find the best parameters that fit the data is called a maximum posterior probability (MAP) estimation, written as

$$\mathbf{s}^* = \arg\max_{\mathbf{s}} \rho(\mathbf{s}|\mathcal{D}). \tag{2.17}$$

Likewise, maximizing only the likelihood without integrating the prior is called maximum likelihood (ML) estimation.

2.4 Kalman Filter Revisited

Inter alia, the Bayesian theory finds its application in the Kalman Filter [212]. The filter estimates the optimal discrete-time state $\mathbf{s} \in \Re^n$ of a linear dynamic process, given as difference equation

$$\mathbf{s_t} = \mathbf{F}_t \mathbf{s}_{t-1} + \mathbf{w}_t \tag{2.18}$$

with a measurement $\mathbf{z} \in \Re^m$

$$\mathbf{z}_t = \mathbf{H}_t \mathbf{s}_t + \mathbf{v}_t. \tag{2.19}$$

The transition matrix $\mathbf{F}_t$ describes the dynamics of the process and relates the previous state $\mathbf{s}_{t-1}$ to current the state $\mathbf{s}_t$. Matrix $\mathbf{H}_t$ is the measurement model matrix, relating the state to the measurement. The white Gaussian noise variables $\mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q}_t)$ and $\mathbf{v}_t \sim \mathcal{N}(0, \mathbf{R}_t)$ represent the process and measurement noise respectively, with time-depending covariance $\mathbf{Q}_t$ and $\mathbf{R}_t$.

Starting with an initial state distribution as Gaussian $\mathcal{N}(\mathbf{s}_0, \mathbf{\Sigma}_0)$, the *prediction* step computes the a priori state distribution $\mathcal{N}\left(\mathbf{s}_t^-, \mathbf{\Sigma}_{s,t}^-\right)$

$$\mathbf{s}_t^- = \mathbf{F}_t \mathbf{s}_{t-1}, \tag{2.20}$$

$$\mathbf{\Sigma}_{s,t}^- = \mathbf{F}_t \mathbf{\Sigma}_{s,t-1} \mathbf{F}^{\mathrm{T}}{}_t + \mathbf{Q}_t. \tag{2.21}$$

Given a new measurement $\mathbf{z_t}$, the error, denoted as measurement residual $\mathbf{r}_t$, is computed between the actual and estimated measurement

$$\mathbf{r}_t = \mathbf{z}_t - \mathbf{H}_t \mathbf{s}_t^-. \tag{2.22}$$

In a similar fashion the residual covariance can be obtained

$$\mathbf{\Sigma}_{r,t} = \mathbf{H}_t \mathbf{\Sigma}_{s,t}^- \mathbf{H}^{\mathrm{T}}{}_t + \mathbf{R}_t. \tag{2.23}$$

Using $\mathbf{r}_t$ and $\mathbf{\Sigma}_{r,t}$, the *correction* step, which integrates the new measurement with the propagated state in the a posteriori distribution $\mathcal{N}(\mathbf{s}_t, \mathbf{\Sigma}_{s,t})$, is performed

$$\mathbf{s}_t = \mathbf{s}_t^- + \mathbf{K}_t \mathbf{r}_t, \tag{2.24}$$

$$\mathbf{\Sigma}_{s,t} = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t)\, \mathbf{\Sigma}_{s,t}^-, \tag{2.25}$$

where $\mathbf{K}_t$ is called the *Kalman gain*

$$\mathbf{K}_t = \mathbf{\Sigma}_{s,t}^- \mathbf{H}^{\mathrm{T}}{}_t \mathbf{\Sigma}_{r,t}^{-1} \tag{2.26}$$

that either emphasizes the measurement in case of a higher gain or follows more closely the prediction for a lower gain.

3 In Situ Knowledge Acquisition

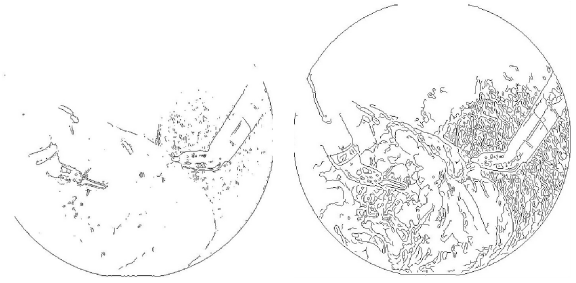
Building an adaptive task-specific model of the surgical field is the first step in tackling the challenges of online surgery. Once we have dealt with the characteristics of endoscopic images, we start with localizing surgical instruments. To manage a variety of existing tools without employing tracking markers, we propose a method that adapts to the instrument's appearance, independent from the shaft's color and the shape of the functional instrument part. We then address the problem of perceiving the environment with miniaturized micro endoscopes. To facilitate the reconstruction of poorly textured areas, we introduce a micro projector that uses a globally unambiguous encoded texture pattern to enhance the correspondence search during the stereo matching.

3.1 Endoscopic Image Characteristics

Endoscopic camera images show certain characteristics that are rarely found in other scenarios and make it difficult for machine vision to analyze their content. Image quality is affected by the camera itself and the conditions of the surgical site. Rigid endoscopes convey light rays typically by means of rod lenses to a camera mounted at the end. Newer devices, especially those with flexible tubes, offer tip-mounted sensors with improved image quality. Fiber optic bundles are used as light guides to illuminate the situs. Although the resolution and image quality of endoscopic systems were substantially improved in recent times, many laparoscopes still operate with the analog PAL signal. PAL uses interlaced line scanning, reading even and odd line numbers alternately. Thus, moving objects cause artifacts when being captured between individual half frames. In the sequel, images are always deinterlaced before processing, meaning that two half frames are combined to create the final image.

Depending on the interventional application, the surgical environment might appear highly cluttered, as illustrated in Fig. 3.1. The homogeneous appearance of tissue makes it difficult to identify spatial structures as they are i.e. required during the correspondence search in depth reconstruction. Main difficulties of in situ knowledge acquisition are also emanating from organ movement and respiration, non-uniform

and time-varying lightning conditions, and specular reflections, which change the appearance of background and surgical instruments.



(a) (b)

Figure 3.1: (a) Response of the Sobel filter; and (b) the Canny filter.

3.2 Tool Localization

Localizing surgical instruments, in particular the ability to track the movements of the tip, is a prerequisite for the development of techniques that assist surgeons with higher level functionality. Performing the localization in situ is important, because instruments can be exposed to high forces at the trocar points, yielding to aberration and deflection of the flexible shaft. Since approaches that model the tool's dynamic flexion to correct the forward kinematics are rather system specific [29], we focus on visual tool tracking [17]. The next section reviews the state of the art.

3.2.1 Basic Techniques

mechanical accuracy

Some studies investigated the feasibility of the daVinci™ robot as "mechanical tracking system" and validated its position accuracy using joint encoder readings. While Kast et al. specified an accuracy of about 6mm for the daVinci classic™ [89], [105] found the accuracy of the daVinci-S™ to be approximately 1mm. Experiments conducted on our telesurgery system (cf. Sec. 5.2.1), which is equipped with accurate industrial manipulators that carry Intuitive EndoWrist™ instruments, could however not affirm similar results. While the absolute accuracy of industrial manipulators is indeed about 1mm, the flexible shaft and error-prone transformation chains deteriorate results noticeable. Nickel et al. investigated magnetic tracking in a realistic operating room environment [135]. Static interference sources, such as a metallic operating table, have been considered during the calibration procedure. The found deviation was less than 1cm within a well-defined workspace. According to the authors, the activity of the manipulators itself had little influence. [106] measured a kinematic error of approximately 10.6±22.9mm with the daVinci™. Additional optical markers, attached to the manipulator links, improved the precision to approximately 1.5mm. Notwithstanding that non-image-based tracking is capable to provide measurements even when instruments are obscured in the situs, the above mentioned studies find no consistent conclusions regarding accuracy, applicability and transferability. Specifically, the deflection

of the shaft is often omitted.

In image-based tracking, blob-detection is frequently applied in combination with tracking markers. Usually, colored bio-compatible markers are attached at the distal end of the instrument. The blob detector then groups pixels with similar properties into larger regions, which differ from their surrounding. The HSV color space is particularly suited, since it separates pixel color information (Hue, Saturation) from luminance (Value). Therefore, the classification is more robust against illumination changes. Analysis of typical laparoscopic image sequences has found cyan be a suitable marker color [67, 178]. While the former authors performed a segmentation in the H-S color plane at 17Hz, the latter restricted their analysis to the hue channel, but simultaneously identify the instrument type by a multi-colored marker. Depth information was derived as the size ratio of individual marker parts with respect to each other were known. A similar approach was proposed by Zhang and Payandeh [218]. They considered the ratio of marker and shaft diameter. Tobergte et al. used a dot pattern to extrapolate the instrument's pose [194]. After segmentation and topological ordering, the dots were matched agains the well-known 3D model of the marker. Kruppa et al. retrofitted surgical instruments with laser diodes that project a pattern structure on the tissue surface [101]. In conjunction with three circular LEDs, mounted at the instrument tip, the projected dot pattern was used to recover the relationship between organ and instrument.

artificial markers

To avoid artificial markers, other authors segment instruments based on color information [51]. Bayes classifiers are frequently used to learn the color statistics of the tool with respect to the background, making the segmentation process more robust against varying lightning conditions. The color distributions are represented as Gaussians and the classifier assigns, according to Bayes' theorem, the observed pixels to the most probable class. Evidently, manual labeling of the classes is required to learn their probability density distribution. Speidel et al. [181] and Kim et al. [92] combined a Bayesian color classifier with conditional density propagation of the instrument state to integrate object dynamics. The latter authors implemented a two-stage approach, first locating the instrument itself and subsequently detecting the forceps with an adapted color model. The employed particle filter [85] propagates a set of randomly generated and weighted hypotheses for each image frame, calculated on the basis of a dynamic model that depends on the previous time steps. The weight is updated according to a likelihood function, comparing measurement and prediction. The new tracking position is then estimated as the weighted mean of all samples. In [111], the method was combined with additional visual cues, such as changes in specular highlights and tissue deformation, for the application of surgical event classification.

color

In addition to color, shape information is a possible criterion to minimize the risk of misclassification. However, detecting the tool solely with gradient images is difficult due to the cluttered appearance of the situs, as depicted in Fig. 3.1. Ueckert et al. considered the elongated shape of the instrument shaft to fit a rectangular bounding box after color classification [201] . To cope with lens distortion, especially visible at small distances between camera and object, a trapezoid is used for the near-field case.

shape

Thresholding of the two principal second-order moments then indicated the match of a shaft. McKenna et al. analyzed adjacent pixels in the background of a presumed shaft position: a reasonable instrument state minimizes the background, while the shaft region is maximized [120]. Line structure concepts, e.g., the instrument's length-to-diameter ratio, are often combined with a color search. For instance, Hessian matrix analysis [213] and the Hough transform [197, 45] were applied. Casales and Amat applied a window operator to assign parallel line pairs to a tool marker [41].

additional constraints

Geometric considerations, such as workspace and movement restrictions of the instruments, can be derived based on the configuration of the trocar point with respect to the camera pose. The relationship allows to limit the search space in the image domain. Voros et al. manually labeled the trocar position in images, using their so-called "vocal mouse" [205]. The point was assumed to have a fixed location during the intervention. The constraint was then used to restrict the search of a gradient filter that works in conjunction with a Hough transform. With images showing little specular reflections and a resolution down-sampled to 200×100px, the method operates in near-realtime with an average error of 11px. Doignon et al. applied seeded region growing, initialized automatically at the image boundary [50] . Recognized candidate regions are then be classified by shape. They simultaneously estimate the trocar position, however, it seems that the information is currently not used for seed initialization. The approach operates at 13fps on a two-level image pyramid with the even field providing a resolution of 320×120px.

machine learning

The Center for Computer Integrated Surgical Systems and Technology (CISST, Johns Hopkins University, Baltimore) presented articulated tracking of the EndoWrist™ tools. They fused kinematic information of the daVinci™ robot with a template tracker, which minimizes the sum of squared differences of source and target image [38]. More recently, they reported a general purpose articulated object tracker and demonstrated its application to surgical scenarios [152]. Geometry and kinematics of the object to be tracked have to be known in advance. The appearance of different body parts was modeled with a class-conditional probability and matched with the input image after rendering the target object geometry. The appearance model was trained with manually labeled images. The computational complexity limits the method currently to 0.2fps. Two different articulated tool trackers have also been proposed by Reiter et al. [161, 162]. One approach learns features offline by rendering a CAD model of the instrument in different poses and matches the templates nearby live kinematic data with 3fps. The second approach learns landmarks on the tool's forceps, such as the manufacturer's logo, as well as different wheels and pins of the mechanics. The authors evaluated SIFT, HoG, and Region Covariance features to train a support vector machine and two different types of randomized trees. The current performance is restricted to 0.8fps.

3.2.2 Hybrid Instrument Localization

The diversity of different instruments employed during a surgical intervention, such as scissors, scalpels, forceps, or needle drivers, differ in their kinematics and in the

appearance of the functional part. With respect to our application scenario, we define the following requirements for instrument localization:

- **Real-time capability**. A high update rate is required, since the instrument's location will be further processed by control algorithms that generate robot trajectories.
- **Markerless**. The approach should forgo any instrument modifications.
- **Reusability**. The surgical system should detect instrument changes automatically to adapt the tracking to the respective tool kinematics and the appearance of the tool used. No prior training should be necessary.
- **Adaptivity**. The method should be robust to changes of the instrument appearance during surgery, e.g. caused by bloodstains.

Concerning these requirements, we propose a hybrid tracking approach. First, the pose of the instrument is determined by means of a position sensor in Cartesian space. This world coordinate is then related to the perspective of the laparoscope by projecting it into image space. Based on this initial guess, the estimate is refined during a visual optimization. Since the prior sensor information restricts the image search space we can perform visual tracking locally. In principal, all types of position sensors can be used, e.g. optical fiber brackets that are embedded into the instrument. Regarding our particular setup, we deduce the position of both the instrument and the laparoscope from robot joint readings.

hybrid tracking

Without loss of generality, we assign the tracking reference frame $\mathrm{T_W}$ always to the projection of the sensor-based instrument measurement in image space (cf. Fig. 3.3). After feature matching, the recovered model state $\mathbf{s}$ corresponds to frame $\mathrm{T_{ref}}$. Taking the time-averaged error between the last k kinematic observations $\tilde{\mathbf{s}}_{t-k}$ and the corresponding outputs of the visual tracker $\mathbf{s}_{t-k}$ into account, we calculate a corrected state parameter prediction $\hat{\mathbf{s}}_{t-1}$, which is then Kalman-filtered to obtain the posterior state probability distribution. Also compare Fig. 3.2.

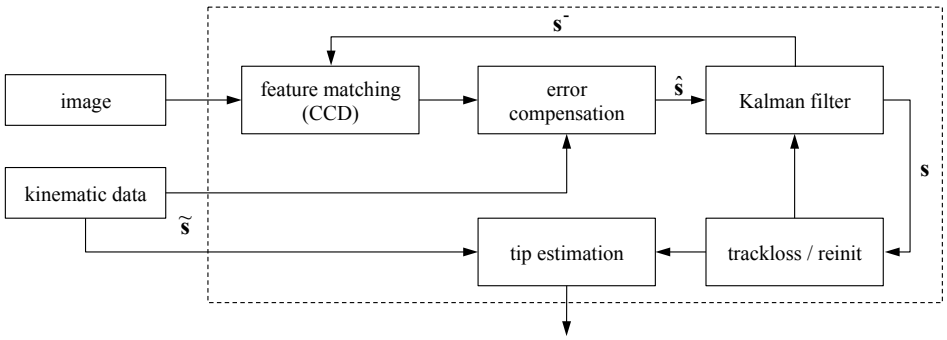


Figure 3.2: Tracking pipeline.

As visual tracking modality the CCD (contracting curve density) algorithm is applied [73]. It allows recovering the contour of the shaft without knowing its coloration in

advance and can cope with color changes over time to a certain degree. A detailed explanation is given in the sequel. Further, we detect the transition between the shaft end and the articulated instrument tip. The pose of the articulated instrument part is estimated from encoder readings and related to the tip position. Next, the individual steps of the algorithm are described.

Sensor-based State Space Prediction

Our surgical system is equipped with Intuitive Surgical EndoWrist™ instruments. The system can identify the type of the instrument that is currently connected to the robot by means of a hot-plug (cf. Sec. 5.2.1), i.e. its kinematics and its individual calibration parameters are known. The instruments are composed of a long shaft, wrist joint, and a tool-specific functional part. Forceps, for instance, have two independently movable jaws. A scalpel consists of only one "finger". Fig. 3.3 illustrates the kinematics of a standard needle driver. We will use

$$\mathbf{q}_e = \left[q_{e_1}, \cdots, q_{e_i}, q_{e_{i+1}}, \cdots, q_{e_j}\right] \tag{3.1}$$

to describe the kinematic chain, whereas $q_{e_1}, \cdots, q_{e_i}$ refer to the joints of the manipulator that carries the tool, and joints $q_{e_{i+1}}, \cdots, q_{e_j}$ are related to the surgical instrument, i.e. $i = 6$ and $j - i = 4$ in case of the needle driver. Since the laparoscope is a straight tool without additional joints we can simply modify the kinematics of the corresponding robot by translating the tool center position to the center of the carried camera, thus $\mathbf{q}_c = [q_{c_1}, \cdots, q_{c_i}]$ and $\mathbf{q} = [\mathbf{q}_e, \mathbf{q}_c]$. We choose three "virtual" reference points ${}^{\mathrm{E}}\mathbf{x}_k$ ($k = 1, 2, 3$) distally located on the instrument. These points are chosen according to the instruments (kinematic) model and are no visual features. Features k_1 and k_2 are located at the end corners of the shaft. Feature k_3 is located at the shaft's center, above the two other features (cf. Fig. 3.3). The projection of the points into image space is denoted as $\mathbf{x}_k$ and defines the instrument orientation, the distal end of the shaft, and the width of the shaft in the image domain. Function f_k concatenates the two kinematic models of instrument and camera, taking the transformation between the two robot bases into account. Given the joint values, each of the k features can be expressed as

$$f_k : \Re^{(e_j + c_i)} \rightarrow \Re^3;\ \mathbf{q} \mapsto {}^{\mathrm{W}}\mathbf{x}_k. \tag{3.2}$$

Without loss of generality, we choose the world reference frame at the camera's optical center. The mapping between the robot bases is described as the homogeneous transform ${}^{\mathrm{R_E}}_{\mathrm{R_C}}\mathbf{T}$. The resulting transform of the forward kinematics is ${}^{\mathrm{R_{E,0}}}_{\mathrm{R_{E,j}}}\mathbf{T}$ for the instrument and ${}^{\mathrm{R_{C,0}}}_{\mathrm{R_{C,i}}}\mathbf{T}$ for the camera respectively, which yields

$${}^{\mathrm{W}}\mathbf{x}_k = {}^{\mathrm{R_{C,0}}}_{\mathrm{R_{C,i}}}\mathbf{T}\ {}^{\mathrm{R_E}}_{\mathrm{R_C}}\mathbf{T}\ {}^{\mathrm{R_{E,0}}}_{\mathrm{R_{E,j}}}\mathbf{T}^{-1}\ {}^{\mathrm{E}}\mathbf{x}_k. \tag{3.3}$$

The mapping from world to sensor space is modeled as $\mathbf{x}_k = g({}^{\mathrm{W}}\mathbf{x}_k)$, with the projection function

$$g : \Re^3 \rightarrow \Re^2;\ {}^{\mathrm{W}}\mathbf{x}_k \mapsto remap\left(\frac{f}{{}^{\mathrm{W}}\mathbf{x}_{k_z}} \begin{bmatrix} {}^{\mathrm{W}}\mathbf{x}_{k_x} \\ {}^{\mathrm{W}}\mathbf{x}_{k_y} \end{bmatrix}\right), \tag{3.4}$$

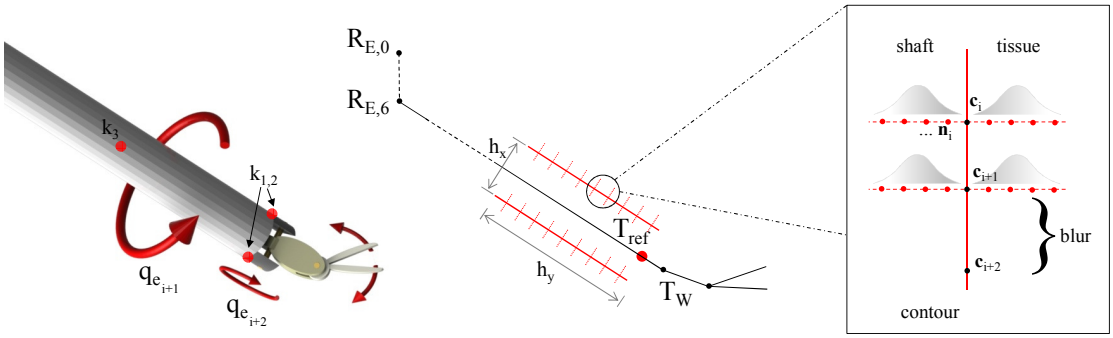


Figure 3.3: Kinematics of the EndoWrist™ needle driver and tracking model representation with coordinate frames $\mathrm{R}_{\mathrm{E},0}$ robot base, $\mathrm{R}_{\mathrm{E},6}$ robot wrist, T_{W} tracking reference frame (equals kinematic prediction), $\mathrm{T}_{\mathrm{ref}}$ object reference frame. The contour model is illustrated with a red line —, the sampling normals with red dashed lines - - -.

where $remap(\cdot)$ is an optional remapping that accounts for lens distortions. When combining the kinematics model (3.2) and the camera model (3.4) we can relate the feature points ${}^{\mathrm{E}}\mathbf{x}_k$ to the image space with

$$\mathbf{x}_k = (g \circ f_k)(\mathbf{q}). \tag{3.5}$$

The projection of the feature points gives an initial guess of the instrument's pose in the image and restricts the visual search space. The prediction accuracy mainly depends on the quality of the sensor readings, that is, in our case on the system calibration and the tool's shaft deflection (cf. Sec. 5.2.3).

The initial estimate is now corrected by means of a model-based visual measurement. To be independent of the appearance of different instrument types, we neglect the tool-specific functional instrument part in this step and represent the shaft with a rectangular two-dimensional shape. The planar roto-translational model pose $\mathbf{s}$ is parameterized by rotation θ, translation t_x respectively t_y, and the two scaling factors h_x and h_y:

$$\mathbf{s} = [t_x, t_y, h_x, h_y, \theta]\,. \tag{3.6}$$

As mentioned, we treat the image-based tracking as a *local* refinement step of the kinematic prediction. The kinematic-based estimate of the model parameters $\tilde{\mathbf{s}} = \left[\tilde{t}_x, \tilde{t}_y, h_x, h_y, \tilde{\theta}\right]$ is updated according to the cycle time of the robots, except the scaling parameters. The scaling is set once during tracking initialization according to the kinematic observation, afterwards it is kept in accordance to the tracker output.

Given the tracker output $\mathbf{s}$ and the kinematic estimate $\tilde{\mathbf{s}}$, the error of the mapping $(g \circ f_k)(\mathbf{q})$, averaged over the last k frames, is

$$\overline{\mathbf{e}} = \sum_k \left(\tilde{\mathbf{s}}_{t-k} - \mathbf{s}_{t-k}\right). \tag{3.7}$$

Since the kinematic error changes smoothly within the workspace, we can correct the estimate according to (3.7) and define a corrected state parameter prediction as

$$\hat{\mathbf{s}}_{t-1} = \tilde{\mathbf{s}}_{t-1} - \overline{\mathbf{e}}. \tag{3.8}$$

In order to account for process dynamics and for uncertainties in $\hat{\mathbf{s}}_\mathbf{t}$ we generate the final prior state hypothesis $\mathbf{s}_t^-$, by applying a motion model to $\hat{\mathbf{s}}_\mathbf{t}$. Specifically, a Kalman filter is used in conjunction with Brownian motion. Brownian motion is given by a linear and time-invariant autoregressive process of the form $\mathbf{p}_t = \mathbf{F}^1\mathbf{p}_{t-1} + \mathbf{F}^2\mathbf{p}_{t-2} + \cdots + \mathbf{F}^n\mathbf{p}_{t-n} + \mathbf{W}^0 w_t$ with $n = 1$ [148]. The matrices $\mathbf{F}^n$ model the transitions between the time steps $(n-1)$ and n. Since Brownian motion is a first order model, it neglects any derivatives (e.g. $\dot{\mathbf{p}}_t$ and $\ddot{\mathbf{p}}_t$, with $\mathbf{F}^1 \equiv \mathbf{I}$, $\mathbf{W}^0 \equiv \mathbf{I}$). Thus, the state probability distribution depends on the pose and on Gaussian process noise w_t only. The final prior state hypnosis is then

$$\mathbf{s}^- = \hat{\mathbf{s}}_\mathbf{t} + w_t. \tag{3.9}$$

The state probability distribution is centered at the corrected kinematic guess. We trust our space-time corrected kinematic prediction and only allow for little dynamics of the position, though permit a higher scaling dynamic (i.e., $\sigma_x, \sigma_y \ll \sigma_\theta \lll \sigma_s$) to quickly adapt to changes in the depth direction, thus

$$\left[\tilde{t}_{x_t}, \tilde{t}_{y_t}, \tilde{\theta}_t\right] \sim \mathcal{N}\left(\left[\tilde{t}_{x_{t-1}}, \tilde{t}_{y_{t-1}}, \tilde{\theta}_{t-1}\right], diag\left(\sigma^2_{\tilde{t}_x}, \sigma^2_{\tilde{t}_y}, \sigma^2_{\tilde{\theta}}\right)\right) \tag{3.10}$$

$$\left[h_{x_t}, h_{y_t}\right] = s\left[h_{x_{t-1}}, h_{y_{t-1}}\right],\ s \sim \mathcal{N}\left(1, \sigma_s^2\right) \tag{3.11}$$

Next, we discuss the image-based model fitting process to close the loop of sensor-based state space prediction and the visual correction step.

Visual Measurement Modality

The visual measurement process is performed under the state hypothesis $\mathbf{s}^-$ and the Kalman filter is used with a target-related likelihood working on a real-time capable implementation of the contracting curve density (CCD) algorithm [72], which is implemented in the OpenTL framework [148]. The CCD algorithm is a contour tracker, which fits a geometric model across the image to describe the screen contour projection of the model under the given pose hypothesis and camera view as best as possible. Instead of looking for sharp edges along the model boundary, the algorithm maximizes the separation between color regions, i.e. the instrument (hereinafter referred to as foreground region F) and tissue (referred to as background region B). This kind of object separation is favorable in the cluttered environment of a surgical site, where edge maps are difficult to interpret (again, cf. Fig. 3.1). Since the color distribution of both object classes are learned online, the method is applicable to a broad variety of surgical instruments without a preceding training phase. To a certain degree, changes of the color appearance of the shaft can be accounted for during the intervention. On the algorithmic side, CCD is implemented by iterating two steps until convergence. First, local color statistics in the vicinity of the two sides of the contour are collected. Afterwards, the observed pixels are assigned to either the foreground- or the background class, according to the respective statistics, and a minimization of the classification error is performed with a Gauß-Newton step. While the two steps are alternated, the region considered for computing the statistics as well as the fuzzy assignment are reduced, thus contracting the likelihood function with each iteration.

We sample the color statistics at the geometric model presented in Fig. 3.3 along the two edges of the shaft's main axis. Hence, orientation and scale are determined in this step. The transition between shaft and functional part, more precisely t_x and t_y, needs to be recovered in a separate step, as described later in the section.

Typically we operate in RGB color space. Depending on the instrument used, also HSV color space might be suitable. After identifying a set of i uniformly distributed contour points $\mathbf{c}_i$ under the current pose hypothesis, color statistics m_i^o up to the second order ($o = 0, 1, 2$) are collected along the contour point normals $\mathbf{n}_i$ (cf. Fig. 3.3, right illustration). A number of D points is evaluated on each normal up to a maximum distance L, specifically $D/2$ on each side of the contour, according to

$$m_i^{0,B/F} = \sum_{d=1}^{D} w_{id} \tag{3.12}$$

$$m_i^{1,B/F} = \sum_{d=1}^{D} w_{id} \cdot I\left(\mathbf{c}_i \pm \bar{d}L\mathbf{n}_i\right) \tag{3.13}$$

$$m_i^{2,B/F} = \sum_{d=1}^{D} w_{id} \cdot I\left(\mathbf{c}_i \pm \bar{d}L\mathbf{n}_i\right) \cdot I\left(\mathbf{c}_i \pm \bar{d}L\mathbf{n}_i\right)^{\mathrm{T}}, \tag{3.14}$$

with $I(\mathbf{x})$ the raw image values at position $\mathbf{x}$ and $\bar{d} = d/D$ the normalized distance to the contour. The $\pm$-sign relates the normal direction to the instrument F or the background B respectively. Note that attention has to be paid in limiting the search distance to be less the width of the shaft to avoid sampling the wrong area. The local weights w_{id} decay exponentially with the distance to their contour point, thus giving a higher confidence to pixels close to the object boundary.

In contrast to the original implementation of CCD, which collects statistics of connected image regions, the speed-up version samples only local line statistics. As a consequence, a single Gaussian is sufficient for representation instead of mixtures of Gaussians. However, the single line statistics are blurred with their j respective neighbors to receive contributions from larger areas:

$$\widetilde{m}_i^{o,B/F} = \sum_j \exp\left(-\lambda \left\|\mathbf{c}_i - \mathbf{c}_j\right\|\right) m_j^{o,B/F}; \quad o = 0, 1, 2. \tag{3.15}$$

The factor $\lambda < 1$ influences the amount of contribution of the neighboring statistics. In doing so, the entire image region around the shape contour is accounted for. Finally, the obtained area statistics are normalized to receive the means $\bar{\mathbf{I}}_i^{B/F}$ and 3×3 covariance matrices $\bar{\mathbf{\Sigma}}_i^{B/F}$ of the pixel values for the two-sided silhouette:

$$\bar{\mathbf{I}}_i^{B/F} = \frac{\widetilde{m}_i^{1,B/F}}{\widetilde{m}_i^{0,B/F}}, \tag{3.16}$$

$$\bar{\mathbf{\Sigma}}_i^{B/F} = \frac{\widetilde{m}_i^{2,B/F}}{\widetilde{m}_i^{0,B/F}}. \tag{3.17}$$

Given mean and covariance, the classification likelihood for the pixels $I(\mathbf{c}_i \pm \bar{d}L\mathbf{n}_i)$ is computed by comparing their values with the respective statistics $(\bar{\mathbf{I}}_i^{B/F}, \bar{\mathbf{\Sigma}}_i^{B/F})$. A

multi-resolution approach is applied to overcome local minima by performing the classification with a fuzzy assignment. The fuzzy membership function

$$a(\bar{d}) = \frac{1}{2}\left[erf\left(\frac{\bar{d}}{\sqrt{2}\sigma}\right) + 1\right] \tag{3.18}$$

assigns the observed pixel value to the foreground, with $erf(\cdot)$ the Gauss error function and σ governs the sharpness of the assignment. The final classification is obtained by mixing foreground and background statistics according to

$$\hat{\mathbf{I}}_{id} = a\left(\bar{d}\right)\bar{\mathbf{I}}_i^F + \left(1 - a\left(\bar{d}\right)\right)\bar{\mathbf{I}}_i^B \tag{3.19}$$

$$\widehat{\mathbf{\Sigma}}_{id} = a\left(\bar{d}\right)\bar{\mathbf{\Sigma}}_i^F + \left(1 - a\left(\bar{d}\right)\right)\bar{\mathbf{\Sigma}}_i^B. \tag{3.20}$$

The color residuals are then given with

$$\mathbf{e}_{id} = I(\mathbf{c}_i + \overline{d}L\mathbf{n}_i) - \hat{\mathbf{I}}_{id} \tag{3.21}$$

and organized in vector form as $\mathbf{E}$. With the corresponding covariance matrix $\mathbf{\Sigma} = blockdiag(\widehat{\mathbf{\Sigma}}_{id})$ the likelihood of a correct classification, expressed as a Gaussian, is

$$\rho_{ccd}(\mathbf{z}|\mathbf{s}_t) \propto \exp\left(-\frac{1}{2}\mathbf{E}^{\mathrm{T}}\mathbf{\Sigma}^{-1}\mathbf{E}\right). \tag{3.22}$$

The likelihood contracts after each pose update of $\mathbf{s}$, since the normal length factor L is exponentially decayed.

Within a Gauss-Newton loop, the derivatives of $\mathbf{E}$ are computed by differentiating (3.19) and (3.18) with respect to the shaft's pose parameters $\mathbf{s}$ and stacking them into the global Jacobian $\mathbf{J}$

$$\mathbf{J}_{id} = \frac{\partial\hat{\mathbf{I}}_{id}}{\partial\mathbf{s}} = \frac{1}{L}\left(\bar{\mathbf{I}}_i^F - \bar{\mathbf{I}}_i^B\right)\frac{\partial a}{\partial\bar{d}}\left(\mathbf{n}^{\mathrm{T}}{}_i\frac{\partial\mathbf{c}_i}{\partial\mathbf{s}}\right). \tag{3.23}$$

The actual state update is then performed according to

$$\mathbf{s} = \mathbf{s} + \Delta\mathbf{s}, \tag{3.24}$$

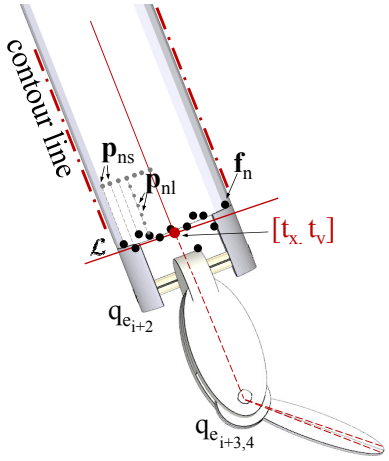
with the pseudo-inverse $\mathbf{J}^+$ of $\mathbf{J}$ and

$$\Delta\mathbf{s} = \mathbf{J}^+\mathbf{E}. \tag{3.25}$$

The optimization is stopped for the termination criteria $\Delta\mathbf{s} \approx 0$ or after a fixed number of iterations.

At this point, we recall that the visual measurement constitutes a local refinement step and the state space refers always to the space-time corrected estimate of an external sensor source. Hence, the tracking results is transformed back to global image coordinates.

Figure 3.4: Determination of the shaft's end point (t_x, t_y) and pose estimation of functional part.



Full Pose Estimation

Since the geometric model used for sampling color statistics for the CCD modality does not consider the transition between instrument shaft and functional part, the final values of the translational components t_x and t_y of the state vector need to be calculated in a separate step. Otherwise, movements that are collinear with the instrument's main axis are not detected correctly and yield a drift of the contour model along the shaft. We distinguish shaft and functional part by local intensity edges. Intensity edges can be observed at the discontinuities between shaft and tip, along a number of N rays. The rays extend linearly from a starting point $\mathbf{p}_{ns}$, located at the distal end of the contour model, toward the shaft's orientation $\mathbf{n}$, up to a reasonable search distance L. The positions $\mathbf{p}_{ns}$ are chosen to uniformly sample the width of the shaft, taking the current scaling factor of the model into account. Intensity gradients are sparsely evaluated at the pixel positions $\mathbf{p}_{nl}$ with a sampling rate l, along each of the rays

$$\mathbf{p}_{nl} = \mathbf{p}_{ns} + d\mathbf{n}, \tag{3.26}$$

with $d = L/l$. If a intensity gradient exceeds a threshold ϕ_n, it is marked as feature point $\mathbf{f}_n$ and the algorithm continues with the next ray. If the maximum search length is reached without finding a feature, the ray is omitted:

$$\mathbf{f}_n = \begin{cases} \mathbf{p}_n & \text{if } \Delta\mathbf{p} > \phi_n \text{ and } d \leq L, \\ \mathbf{0} & \text{otherwise.} \end{cases} \tag{3.27}$$

Both the maximum search length and the threshold ϕ_n are dynamically adjusted dependent on the averaged values of the preceding rays. Image noise is reduced beforehand by applying a Gaussian filter.

The set of candidate feature points is used to find the transition between shaft and functional part by line fitting. The feature set possibly contains few gross errors. To be robust against outliers, we apply the iterative RANdom SAmple Consensus (RANSAC) paradigm [57]. In each iteration, the algorithm randomly chooses a minimum number

of points to generate an instance of the model to be fitted. Accordingly, a line with the implicit representation

$$\mathcal{L} : \theta_1 f_x + \theta_2 f_y + \theta_3 = 0 \tag{3.28}$$

requires a subset of two points. The residual points are checked for consistency with the found parameterization θ, with an absolute fitting error of

$$e_{\mathcal{L}}(\theta) = \sum_x \left| \frac{\theta_1 f_x + \theta_2 f_y + \theta_3}{\sqrt{\theta_1^2 + \theta_2^2}} \right| \tag{3.29}$$

and split into inliers, which agree with model parameterization within an acceptable threshold, and outliers. The procedure is performed until the highest number of inliers is probably a good fit. In the final step, we vote the intersection of the fitted line and the skeleton of the shaft as the transition to the functional instrument part. The translational components t_x and t_y of the state vector are updated accordingly. Fig. 3.4 illustrates the approach.

Given the final instrument state, we estimate the pose of the articulated instrument part. Using (3.5) the kinematics of the articulated tool tip is related to the camera view and projected into image space. The projection is corrected according to the error between sensor-based prediction and image measurement at the current time step to provide an adequate alignment between shaft and tip part:

$$\mathbf{x}_e = (g \circ f_e) - \mathbf{e}_e, \tag{3.30}$$

where $\mathbf{e}_e = [\tilde{s}_{t_x} - s_{t_x}, \tilde{s}_{t_y} - s_{t_y}]^{\mathrm{T}}$. Function f_e projects the current configuration of the tip kinematics from the model to $\mathbf{x}_e$ into image space, with $e = (e_{i+1}, \cdots, e_j)$ as defined in (3.1).

Pose Initialization and Coherence Check

Since the utilized measurement modality operates merely on local image regions, the last remaining aspect to be treated is automatic initialization and reinitialization of the tracking system, both at the beginning of the tracking and in case of tracking loss.

As we have seen in the previous section, the relationship (3.5) constitutes an initial guess of the instrument location in image space, associated with space-time corrected error values. During successful tracking we remember these parameters with respect to the sensor-based Cartesian prediction. To reduce the computational demands, the workspace is partition with an octree representation and parameters are assigned to the corresponding grid nodes according to the tree's resolution. Values are averaged in case of multiple grid occupancy. During a tracking loss this spatial indexing is used to recall suitable parameters that reinitialize the measurement subsystem. If no data is available for exactly the requested node, the spatial neighborhood is searched up to a certain distance. If the procedure fails, a global visual search is performed, which is based on color classification, i.e. the reddish surgical background is segmented. More specific, identified pixels are combined to connected regions in a binary image

by means of blob detection. An ensuing elliptic approximation of the connected components that exceed a certain size defines the instrument shaft. We then compute the major axis of the ellipses and vote the one with minimum distance to the projection of the sensor-based estimate to be the candidate used to reinitialize the system.

After each tracking step, a coherence module constantly checks the quality of the measurements and binds the local tracking with the (re-)initialization module in case of a tracking loss. We have chosen a histogram-based approach, operating on the hue and saturation channels of the image portion located within the contour boundaries of the CCD tracking model under the computed pose. The normalized dissimilarity between the histograms of the expected model appearance $\mathcal{H}_M$ and the current surface observation $\mathcal{H}_O$ is calculated using the Bhattacharyya distance

$$\mathcal{D}_{BHA}(\mathcal{H}_M, \mathcal{H}_O) = \sqrt{1 - \frac{1}{\overline{\mathcal{H}}_M \overline{\mathcal{H}}_O N^2} \sum_J \sqrt{\mathcal{H}_M(J) \cdot \mathcal{H}_O(J)}}, \tag{3.31}$$

with N being the total number of bins and $\overline{\mathcal{H}}_i = \frac{1}{N} \sum_J \mathcal{H}_i(J)$. The pixels used for evaluation are sparsely sampled with a uniform distribution over the image region in question. If the dissimilarity exceeds a threshold $\phi_{\mathcal{H}}$ the system is reinitialized, otherwise the model histogram is updated according to $\mathcal{H}_M^* = \mathcal{H}_O$.

Evaluation

The proposed method was experimentally evaluated on our telesurgery setup. The left needle driver of the system was tracked inside of a medical mockup model. Fig. 3.5 shows the scene. The projection of the kinematic prediction in image space is illustrated blue, while the actual tracking result is depicted yellow. The two jaws of the forceps are not modeled individually, but the center position is shown. The pose of the functional instrument part is derived from encoder readings and adapted to the tracked shaft position. Due to mechanical inaccuracies of the Bowden-wire driven forceps, this method can give only an approximate estimate. Introducing an additional

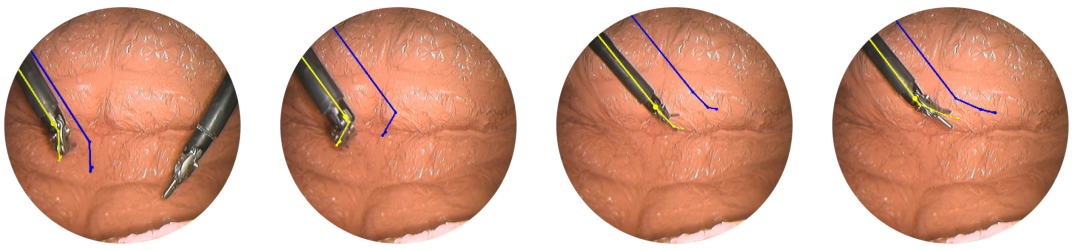


Figure 3.5: Laparoscopic instrument tracking: the kinematic prediction of the instrument pose is depicted in blue, the tracked position in yellow.

feature matching at the tip can further improve the result. To determine the accuracy of the approach, about 700 frames were hand-labeled. The results of the corresponding measurements are given in Fig. 3.6. The instrument pose as derived from the kinematic prediction, the ground truth position, and the actual tracking position are

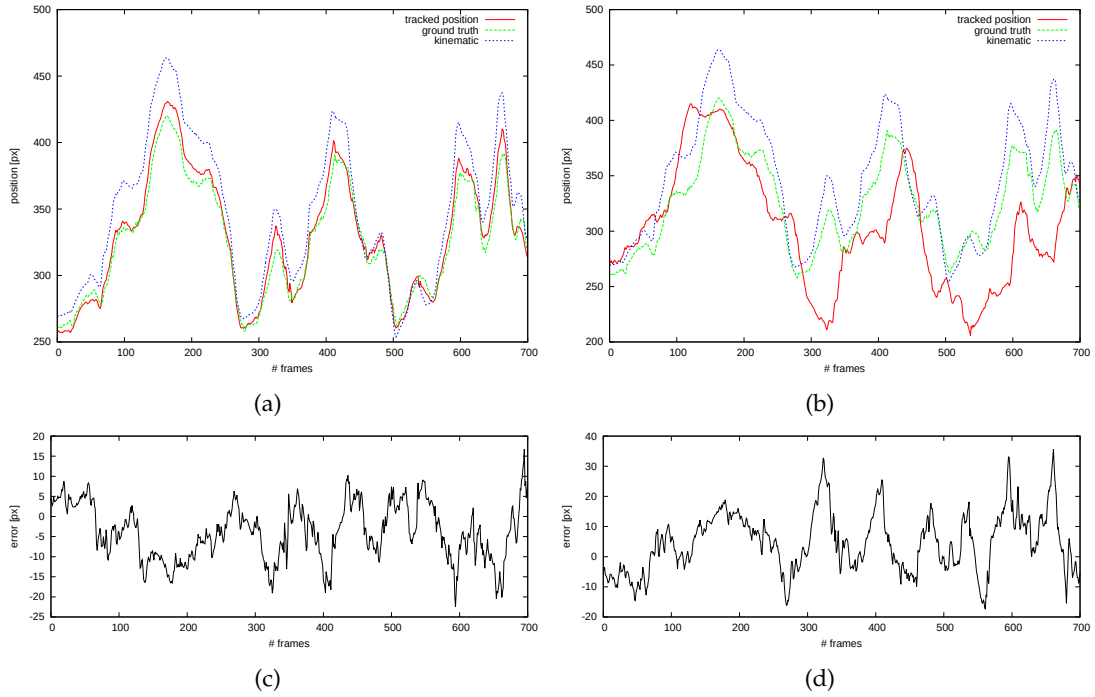


Figure 3.6: (a) shows the tracked instrument position, the manually labeled ground truth position, and the kinematic prediction for the $x-$ component, (b) for the $y-$ component. (c) respectively (d) illustrates the corresponding tracking error.

separated for the $x-$ and $y-$ component. Since the instrument used exhibits an overall grayish/metallic appearance, the current main difficulties of the method arise in distinguishing the transition between shaft and functional part. This can yield an erroneous offset of the tracked position toward the tip. While the applied Kalman filter operates well during automated instrument guidance, where the instrument is moved in a continuous motion, sudden changes in the instrument direction yield a track loss. A particle filter that evaluates multiple hypothesis might improve this behavior during manual operation. At the PAL resolution of our endoscope, we achieve a refresh rate of approximately 23fps.

3.3 Depth Perception with Micro Endoscopes

Besides knowing the instrument's location, autonomous in situ task planning and execution also requires knowledge about the tissue geometry. Depth can be acquired using many different approaches and at least as many ways exist to classify them. For an extensive overview we refer to [37, 189, 77, 34] and to [183, 38] for a particular focus on depth perception in minimally invasive interventions. With respect to our application of augmenting surgical instruments with miniaturized stereo cameras, we consider only optical methods that are based on triangulation, that is, stereo matching and structured light (SL). Keep in mind that also direct tactile methods have been used to recover depth in situ [103, 199]. Also the improved resolution of recent time-

of-flight (ToF) cameras makes them interesting to enhance conventional laparoscopes with depth perception capabilities, but results are noise and calibration difficult [84].

3.3.1 Combining Stereo Matching and Structured Light

The challenge of passive stereo is to find matching pixels in the images. This is the correspondence problem, as introduced in Sec. 2.2. Given a calibrated camera system, the search is constraint to the respective epipolar lines. Feature-based approaches are capable of producing only sparse disparity maps or require elaborated gap interpolation, since they assign distinctive feature points. A variety of feature descriptors are available, such as the Scale-Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF), Binary Robust Invariant Scalable Keypoints (BRISK), as well as edge and line descriptors, to name typical candidates [200, 6]. Dense stereo methods estimate a set of pixel-wise correspondences based on the correlation of the pixels intensity or color value. The (dis-)similarity of pixels is rated with a metric that computes a cost value for each match. Following the taxonomy of Scharstein, costs are then aggregated by summing over a support region, and finally the best match is found in an optimization step [173]. While local stereo approaches simply vote the support region with the lowest cost, global stereo optimization makes explicit smoothness assumptions. Typically, the problem is formulated in an energy minimization framework [188]. Semi-global methods mimic global methods in considering only a spatial neighborhood, resulting in a shorter computation time [81].

Regardless of the chosen method, passive stereo is susceptible to illumination changes and has problems in poorly textured areas, since no correspondences can be found. Most errors occur on depth discontinuities, where either the corresponding matching area is not visible in the other image or the ordering constraint is violated. The latter usually implies that objects in one image have the same spatial order in the second image. This order might be permuted if objects are partially occluded by other objects.

Structured light can alleviated some of the issues mentioned. SL is an active triangulation method, where one camera is replaced by a light projector that emits an artificial structure onto the scene. The projection of this well-known pattern improves photo-consistency and facilitates the correspondence search. Triangulation is then performed between the deformation of the imaged pattern and the projected pattern. Therefore, the pattern needs to be unambiguous to avoid misinterpretation. The coding of patterns can refer to different strategies:

- *Temporal multiplexing* encodes pixel locations by a sequence of patterns. E.g., stripe indexing uses binary coding, where each mask consists of a different spatial series of black and white stripes (e.g. a Gray code [86]). Therewith, each scene point receives a unique illumination order, specifically N masks encode $2^N - 1$ stripes. Fringe patterns (e.g. sine waves) are used for phase unwrapping and typically require less pattern sequences than binary encoded masks.
- *Spatial multiplexing* uses individually recognizable elements, called primitives or characters of a codeword. A primitiv can e.g. be characterized by shape, color,

or brightness. The pattern is created by grouping the primitives to globally non-recurring codewords. Therewith, a specific position can be extracted by considering the local neighborhood. To enforce the global uniqueness constraint De Bruijn graphs and M-Arrays are frequently applied [217, 25, 116].

- *Frequency multiplexing* performs phase unwrapping of fringe patterns in the frequency domain rather than in the spatial domain, e.g. by Fourier transformation [39].

At this point, we will not consider the multitude of proposed pattern designs more detailed, but refer to the surveys [171, 146, 61]. Rather, we make some considerations regarding our particular camera setup. For the miniaturized stereo endoscope, Awaiba NanEye™ cameras with a resolution of 250×250px are used. The sensor's package size measures 1×1×1.5mm. A detailed description of the sensor is provided in Sec. 5.2.2. The camera is designed for use in close range to the scene, approximately 3–15mm. The resulting perspective shows only a small detail of the scene, exhibiting a homogeneous surface. Adding artificial texture is therewith essential.

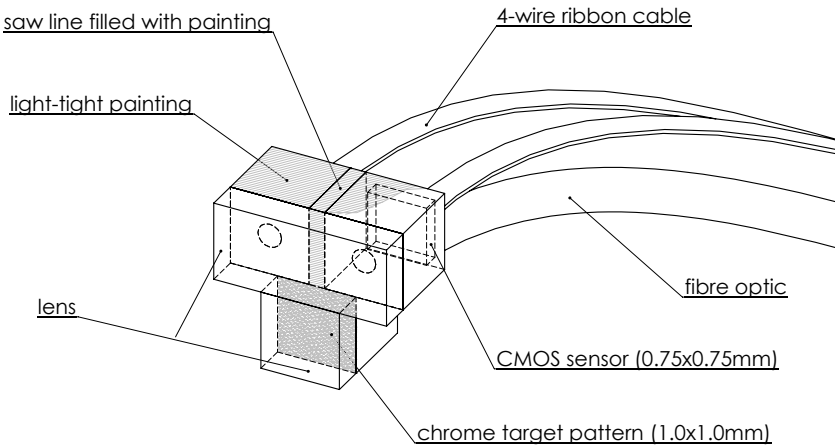


Figure 3.7: Proposed endoscopic micro stereo camera setup with additional micro projector mounted below the two cameras.

chromium mask

In order to enable depth perception with the miniaturized cameras, we developed a micro projector that enhances the surface with additional structure. Our final reconstruction system, which is illustrated in Fig. 3.7, features two micro cameras and the projector, resulting in an overall size of approximately 2×2×1.5mm. Clearly, the mask of the projector needs to be manufactured highly accurate. More precisely, the layout of the designed pattern structure must be transferred to a mask area of 0.75mm^2. To meet this requirement, the pattern was formed in chromium on a glass substrate. For the final assembly, the chromium mask is mounted behind the same type of lens as used for the cameras, therewith featuring the same projection properties. Illumination is provided with a 1mm fiber optic, which is mounted behind the mask, and a LED light source. However, the production process of the mask poses restrictions on the design of the pattern:

1. The structure of the pattern needs to be *binary*, since the mask is realized as a *graphical optical blackout* mask (GOBO). The chromium blocks light, while light can pass through the glass substrate for the remaining regions.

2. The system is a fixed-pattern projector, thus the pattern layout is *static* and cannot be changed over time.

3. The pattern structure has to be realized by means of a *regular pixel grid*.

Considering these limitations, we designed a binary one-shot pattern with globally non-recurring spatial encoding. The pattern is presented more detailed in Sec. 3.3.4.

For now, it remains the question of how much information can be encoded with a binary one-shot pattern to disambiguate potential matches. The pattern is assembled from primitives, where the set of all primitives is called the alphabet. Unambiguous codewords are spatially arranged arrays of primitives. A larger alphabet makes it easier to assemble larger non-recurring pattern structures, since more primitives allow for more codeword variations. However, decoding the pattern gets more complex, since more individual primitives need to be distinguished. Likewise, longer codewords reduce the similarity between two words. That is, a higher Hamming distance is required. In return, recovering longer codewords demands for the correct decoding of more primitives, which makes them susceptible to noise.

Binary primitives can only be varied by shape. Consequently, the spatial extent of the primitives define the size of codewords, therewith determining the entropy of the pattern. While larger shapes can be resolved easier in the camera image, they encode only sparse locations, therewith limiting the resolution of the range image. According to the Nyquist–Shannon sampling theorem, the aspect ratio of the mask's grid spacing and the sensor resolution needs to be at least twice as high to recover the primitives, when sensors without Bayer patterns are used [140]. For example, see the shapes pro-

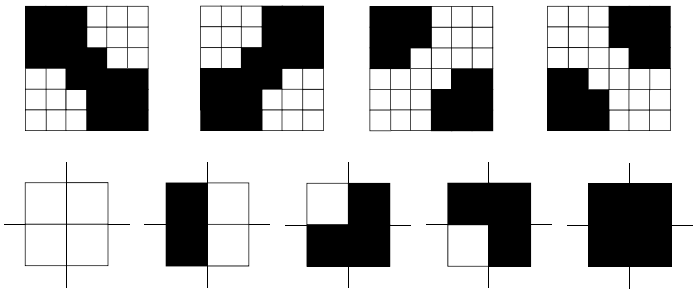


Figure 3.8: Primitive shapes proposed by Vuylsteke [206] (first row) and Griffin [64] (second row) to encode grid positions. The spacing of the mask and the spatial extent of the shapes with respect to the camera resolution define the number of distinguishable positions.

posed by Vuylsteke and Griffin [206, 64], depicted in Fig. 3.8. They assign only a single codebit to every location, which is more robust to perspective distortion than using large shapes. Indexing the grid points is then achieved by distributing the binary in-

formation in the neighborhood.

stereo + SL

To provide dense range maps, we aim to combine stereo matching with structured light. A binary pattern with an alphabet size of only two primitives, where a pixel is either set or unset, is used. The codewords are two-dimensional globally unambiguous sub-windows of the pattern, centered at each pixel position. Being a pseudo-random noise pattern (PSM), the layout represents also a suitable texture to support the correspondence search using window-based dissimilarity measures in stereo matching. Clearly, the projected pattern is subject to perspective deformation, which is why the decoding success of the pattern also depends on the spatial extent of the sub-window. I.e., smaller codewords are less prone regarding distortions. In our particular setup, camera and projector have the same lens distortion properties. That is, for a small baseline the imaged pattern shows relatively little distortion in the camera image that is caused by the lens and most distortion is caused by the scene.

We treat each decoded position as a ground control point (GCP) that is assigned to an unambiguous pattern position. The GCPs are integrated as a prior jointly with the costs of stereo matching into a global energy minimization framework, illustrated in Fig. 3.9.

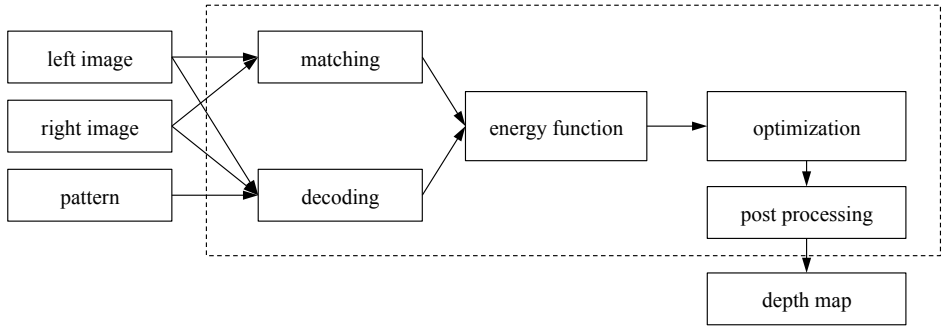
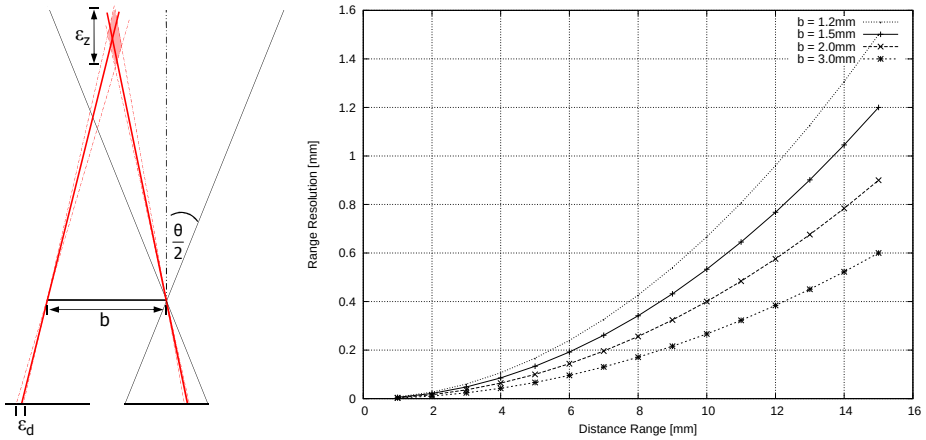


Figure 3.9: Procedure of depth recovery: stereo matching costs are aggregated with GCP costs, which result from pattern decoding. The final energy function is minimized in a global optimization framework.

In the proposed approach, the spatial extent of the symbols and the codeword length suitable to construct the pattern, do ultimately not only depend on the properties required for structured light. Rather, the requirements of stereo matching, specifically the properties of the dissimilarity measure used, must be considered. To investigate both aspect, we first define the arrangement of the stereo cameras and introduce a simulation environment that helps us to find good parameters. Afterwards, attention is payed to the actual pattern design.

3.3.2 Sensor Arrangement

The camera arrangement of a stereoscopic setup strongly influences the quality of depth reconstruction. In particular, two factors play an important role: the depth range



(a) (b)

Figure 3.10: (a) depth error and influence of the baseline on the common field of view; (b) range resolution with $e_d = 1$ for different baselines.

that is required for the intended application, i.e. $z_{near} < z < z_{far}$, and the baseline b. The depth error can be separated into correspondence error ϵ_d and geometric resolution $\frac{z^2}{bf}$. The correspondence error is caused by mismatches during the disparity search and increases e.g. with sensor noise. Applied to the general depth expression $z = \frac{fb}{d}$, the reconstruction error can be formulated with the Taylor approximation [59]

$$\epsilon_z = \frac{fb}{d} - \frac{fb}{d+\epsilon_d} = \frac{z^2\epsilon_d}{fb + z\epsilon_d} \approx \frac{z^2}{fb}\epsilon_d. \tag{3.32}$$

Finding a suitable camera configuration that meets the technical and environmental conditions of our setting best, means to minimize the geometric error. Given fixed camera properties, specifically focal length and resolution, the baseline can be chosen with respect to the expected working distance of the sensor to the scene. In (3.32), the geometric resolution $\frac{z^2}{fb}$ is quadratic in depth. For a fixed focal length f, it can be reduced only by increasing the baseline. Camera calibration revealed a focal length of $f = 216$px for the micro cameras.

Using (3.32) and assuming $\epsilon_d = 1$ the range resolution for different baselines was evaluated for our setup. Results for the range of b =1.2...3.0mm are given in Fig. 3.10(b). Note that this theoretical assessment is based on a pinhole model and neglects distortion effects. Within the typical range of operation between 4 – 15mm, the expected error for a baseline of 1.2mm is about twice as high as the error for a baseline of 3mm. However, with the increase of the baseline also the disparity search range increases. To avoid incorrect correspondences and to keep the computational complexity low, the disparities should not significantly exceed 10 – 15 percent of the image width. Table 3.1 gives reference disparity values when distortion is neglected.

Wider baselines result in a shift of the overlapping image region toward higher depth values, causing a loss of the near range (cf. Fig. 3.10(a)). Both camera images are

aligned next to each other for

$$z = \frac{b}{2\tan\left(\frac{\theta}{2}\right)}. \tag{3.33}$$

One possibility to increase the baseline artificially is to tilt the cameras towards each other. However, we aim to realize the stereo camera pair directly on wafer level during the production process. That is, two adjacent sensors are directly cut as one piece from the silicon. This guarantees an optimal alignment, which is important at the small scale of the setup, but modification of the sensor orientation becomes impossible.

	distance					
baseline	2	4	6	8	10	15
1.2	75	38	25	19	15	10
1.5	94	47	31	23	19	13
2.0	125	63	42	31	25	17
3.0	188	94	63	47	38	25

Table 3.1: Baseline over distance: theoretically resulting disparity in pixels. Distance and baseline in mm.

1.2mm baseline

Based on the above investigations and the expected range resolution, we have chosen a baseline of 1.2mm for our setup. A $200\mu m$ saw line with a depth to the cover glass of the lenses separates the sensors on silicon. The saw line as well as the housing are painted light-tight to protect the sensors from interfering light.

Now that the geometry of the stereo system is set, we introduce a realistic simulation environment, which mimics the setup. The simulation serves as the basis for our experiments, since our access to the hardware is limited.

3.3.3 Sensor Simulation by Ray Tracing

Acquiring data sets with known ground truth is a difficult and time-consuming process that typically requires an elaborated hardware setup, such as laser-range scanners. In MIRS, in vivo ground truth can only be obtained through post-operative registration with high-speed CT volume scans [185], but the approach is limited by the registration accuracy. Once all data is captured, no changes can be made to the original configuration, i.e. the camera parameters and the baseline are fixed. We introduce a ray tracing emulation environment to test and optimize the proposed stereo system [11].

Being based on physical principles, ray tracing is capable of producing realistic-looking images along with corresponding depth data. By GPU-acceleration, interactive real-time applications can be implemented despite the computational complexity. As we will see in the next sections, such a simulation environment also allows optimizing the hardware setup, e.g. by evaluating different projection pattern layout. Our simulation is based upon the Nvidia OptiX™ ray tracer [150]. The engine abstracts ray tracing in a general purpose pipeline and is not pre-configured for a specific rendering method. Instead of capturing light that is emitted from the scene, ray tracing inverts

the imaging process and traces the path light has taken through the scene backwards. The simulation process can be divided into three steps: while the SDK handles tracing a ray throughout the scene by traversing a node graph representation of the scene model, the user has to specify the ray direction and the reflectance of the light by defining object surface properties and ambient light. To benchmark the quality of our depth reconstruction approach, we particularly need to implement the characteristics of camera and projector. An arbitrary number of cameras and projectors with corresponding intrinsic and extrinsic parameters can be defined. The projector emits light through a user-defined pattern, illuminating scene objects with the projection of the pattern.

To specify the direction of outgoing rays into the scene, the Bouguet lens model is used, which accounts for tangential and radial distortion effects. The model is widely employed for camera calibration, thus enabling a simple transfer of real-world parameters to the simulation. Since ray tracing reverses the light transfer process, the model needs to be inverted [79]. In a similar manner, the regular (non-inverted) model is used to describe the geometry of the projector.

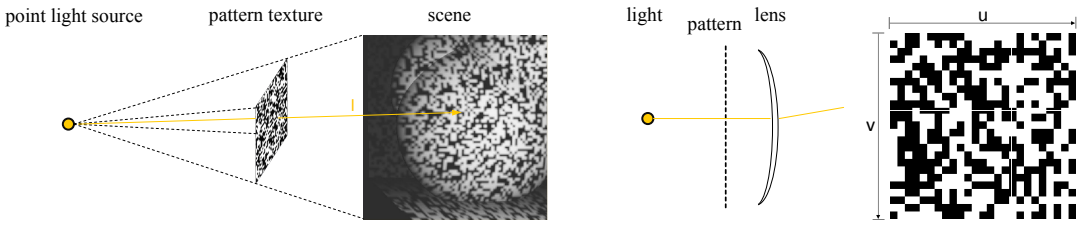


Figure 3.11: Simulation process of the pattern projector: (a) pattern projector geometry; (b) example pattern texture.

projector simulation

The pattern projector is implemented as a combination of a point light source with an additional texture plane located in between the light source and the lens. The intensity of the projector is determined by first sampling the light without considering the pattern and attenuating the intensity according to the pattern texture afterwards. Black pixels block a ray of light, as illustrated in Fig.3.11(a). This yields a two-step approach: first the pattern position that corresponds to a certain scene point is located in order to calculate the light intensity of a ray. Then, the distortion model is applied to account for the projector's lens. Texture coordinates are obtained by normalizing the direction vector $\mathbf{l}$ between the light source and the ray intersection with a scene object's z-component $[u, v]^{\mathrm{T}} = [l_x/l_z, l_y/l_z]^{\mathrm{T}}$. Due to the inverted light tracing process, handling the projector's lens distortion becomes analog to the task of correcting lens distortions of cameras in the real world. Instead of mapping the coordinates of a distorted image to an undistorted one, when asked to sample the attenuation value of a distorted projection at coordinates $[u, v]^{\mathrm{T}}$, the coordinates are remapped to the equivalent coordinates of an undistorted projection. With the substitution $r^2 = u^2 + v^2$, we get the undistorted coordinates $\tilde{u}$ and $\tilde{v}$ with the Bouguet terms accounting for radial

and tangential distortion

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \end{bmatrix} = \left(1 + k_1 r^2 + k_2 r^4 + k_5 r^6\right) \begin{bmatrix} u \\ v \end{bmatrix} + \mathbf{dx} \tag{3.34}$$

and the tangential distortion term

$$\mathbf{dx} = \begin{bmatrix} 2k_3 uv + k_4(r^2 + 2u^2) \\ k_3(r^2 + 2v^2) + 2k_4 uv \end{bmatrix}, \tag{3.35}$$

where $k_1, \cdots, k_5$ are the distortion coefficients. Therewith, emulating the projector can be summarized as follows:

Data: pattern texture $\mathbf{T}$
Result: pixel intensity
foreach *3D scene point* $\mathbf{s}$ **do**
  `//start with the point light model`
  $intensity \leftarrow \text{sample\_point\_light}(light, \mathbf{s})$
  $\mathbf{l} \leftarrow \mathbf{s} - light.position$
  $\mathbf{x} \leftarrow \text{projectToPlane}(\mathbf{l})$
  $\mathbf{x} \leftarrow \text{distort}(\mathbf{x}, light.instrinsics)$ `//apply distortion`
  `//check for valid coordinates and calculate intensity`
  if $\mathbf{x} \in \text{sizeof}(\mathbf{T})$ **then**
    return $intensity * \text{tex2D}(\mathbf{T}, \mathbf{x})$
  else
    return 0
  end
end

Simulating the camera is treated in a similar fashion, in which the point light is replaced with a sensor model and as a matter of course the texture pattern is omitted. However, the ray tracing principle request an inversion of the Bouguet model. The derivation of the model inversion can be found in appendix A.1. The sensor model specifies how the image sensor reacts to incoming light. E.g. specific sensor characteristics can be simulated, such as noise performance. Note, that we also define a standard deviation of the camera parameters to avoid a "perfect" calibration within the simulation. Therewith, camera parameters are slightly altered before the simulated images are rectified for stereo reconstruction. Camera images resulting from different simulation steps are illustrated in Fig. 3.12.

ground truth

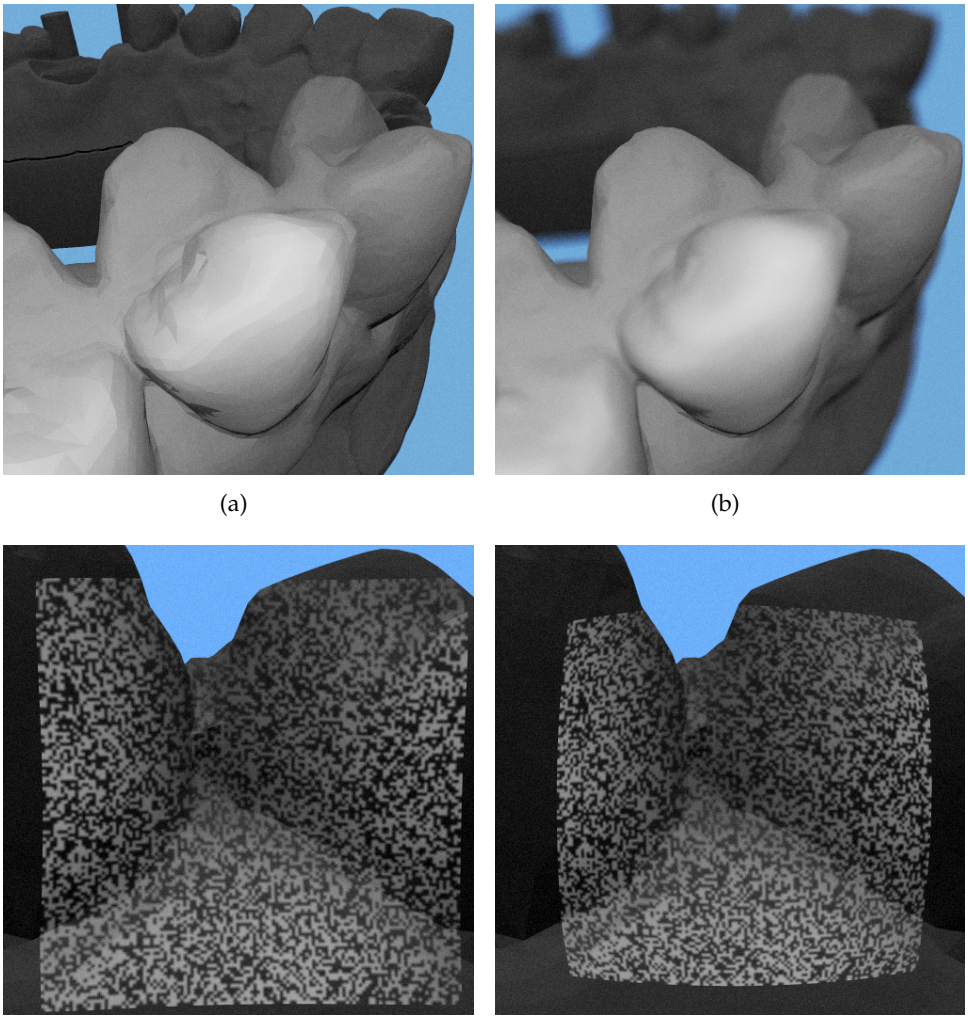
Since ray tracing calculates the intersection of all rays sent into the scene with the scene's objects, obtaining ground truth range information can be seen as a co-product, where the length of a ray is output. To evaluate the quality of the depth reconstruction, we use the quality metrics proposed by Scharstein and Szeliski [173]. Specifically the root-mean-squared error (RMS)

$$R = \sqrt{\frac{1}{N} \sum_n |D_n - \widehat{D}_n|^2} \tag{3.36}$$

and the percentage bad matching pixels

$$B = \frac{1}{N} \sum_{n} |D_n - \widehat{D}_n| > \delta_b \tag{3.37}$$

are calculated between the computed disparity map D and the ground truth disparity $\widehat{D}$, where N is the total number of pixels and δ_b is a disparity error tolerance.



(a) (b)

(c) (d)

Figure 3.12: Examples of the ray tracing simulation: (a) simulation with Bouquet model; (b) additional simulation of aperture; (c) projector with same intrinsic parameters as camera. Due to the small baseline pattern distortion is mainly affected by the scene; (d) projector with same focal scale and principle point, but zero lens distortion. All images were gamma-adjusted for better visibility.

3.3.4 Projection Pattern Design and Optimization

So far, we have derived the hardware-related parameters of the proposed miniaturized stereoscopic system. Taking this into account, attention is now payed to the actual layout of the pattern mask. Bear in mind that the design has to support stereo matching and the encoding of spatial unambiguous grid positions equally. Hence, the following questions need to be answered:

1. How can we design the pattern to support window-based stereo matching and structured light alike?
2. How can we maximize the number of encoded grid positions on the pattern?
3. How can we minimize decoding and matching errors, therewith maximizing the difference between individual codewords?

Pattern Design

Let us first assume a pattern that is solely employed for intensity-based stereo matching. A pattern supports the correspondence search best, if pixels along epipolar lines differ strongly within the disparity range d. Because of the static binary projection property of the GOBO mask, we are forced to consider sub-windows to establish uniqueness of pixel locations. Therewith, a pixel neighborhood defines a codeword with a block size $N = n \times n$ and is matched against the d other blocks in the second image. The larger the difference between the individual blocks, the better the matching score. Konolige studied the design of ideal projection patterns used for passive stereo with respect to the imperfections of camera and projector [99]. Specifically, he compared different *locally* unambiguous patterns, i.e. local Hamming patterns and non-recurring De Bruijn sequences [109]. Further, he investigated the influence of camera/projector phase, blur, and aspect ratio on the reconstruction quality obtained with specific patterns. However, neither extrinsic nor intrinsic parameters were accounted for. Image processing was performed directly on artificial pattern images to simulate the effects. We expand the idea of considering system specific parameters in order to find suitable patterns. Instead of simply processing the pattern image, we simulate our entire hardware setup within the ray tracing emulation introduced in the last section. Therewith, real-world lens parameters as well as the geometric arrangement of the stereo camera pair and the projector can be considered.

Summarized, intensity-based reconstruction requires patterns that vary as much as possible *locally* within the disparity search range. In contrast, structured light approaches require *globally* unambiguous information to identify pixel coordinates. Our pattern needs to combine both properties. Therefore, we propose a pattern design that has a guaranteed minimum Hamming distance between all codewords, while the codewords are globally non-recurring. Fig. 3.13 illustrates the difference. The Hamming distance is defined as

global Hamming pattern

$$d_H(\mathbf{x}, \mathbf{y}) = \sum_{n=1}^{N} x_n \oplus y_n, \tag{3.38}$$

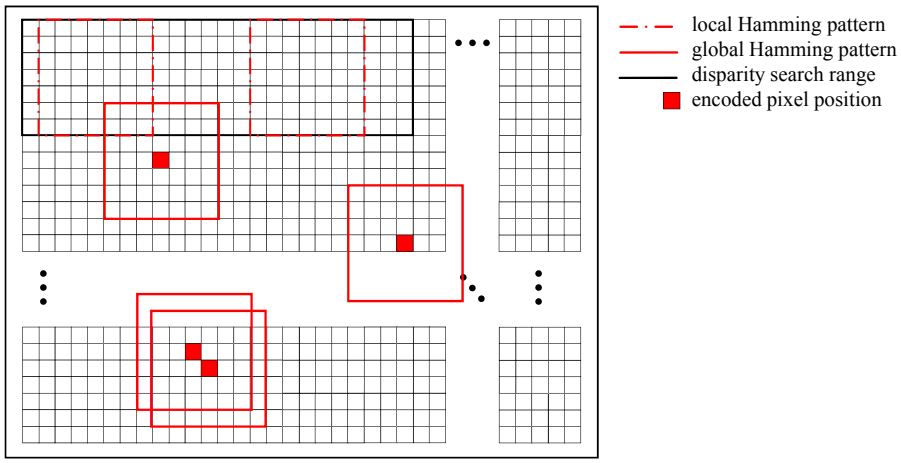


Figure 3.13: Difference between local and global Hamming patterns: for local Hamming patterns only the blocks within the disparity search range are non-recurring. Global Hamming patterns guarantee a minimum Hamming distance between all codewords, while each codeword is unambiguous. It encodes the centered pixel position of each block.

where $\mathbf{x}$ and $\mathbf{y}$ are two words with length N.

A natural bound on the information content representable with the global Hamming pattern is its resolution. However, the chosen projector resolution also influences the quality of window-based stereo matching. To estimate a suitable aspect ratio of the camera's and the projector's pixel size, we employ the ray tracing simulation. The aspect ratio

$$\alpha = \frac{\text{projector resolution}}{\text{camera resolution}} \tag{3.39}$$

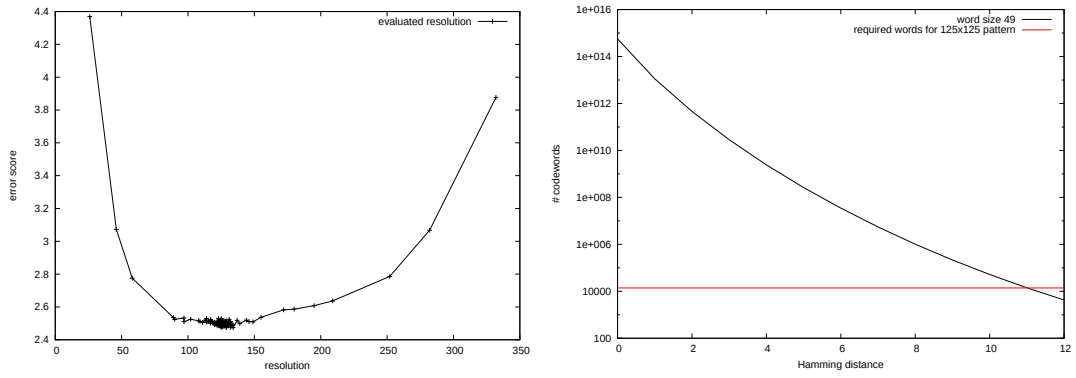
is optimized, while the reconstruction quality obtained with a particular pattern resolution is rated. During the optimization process, random noise patterns were employed. Local block matching stereo [100] is applied and the dissimilarity between pixels is measured with the Sum of Absolute Differences (SAD)

$$\Phi_{SAD}(I, J) = \frac{1}{N} \sum_{n \in N} |I_n - J_n|, \tag{3.40}$$

where I_n and J_n are the pixel values of the left and right image respectively and N defines the window size. The overall reconstruction quality is evaluated according to

$$\mathcal{E} = \lambda \cdot B + R, \tag{3.41}$$

where B is the percentage of bad pixels, R is the root-mean-squared error, and λ is a gain factor. Both errors are computed with respect to the ground truth disparity $\widehat{D}$ obtained from ray tracing. As scene mode, the dental impression depicted in Fig. 3.12 was used. Fifteen different positions within the workspace of the camera are sampled and the reconstruction quality is evaluated. Afterwards, a new projector resolution is chosen according to an optimization strategy. The Covariance Matrix Adaption Evolution Strategy (CMAES) [75] is applied to minimize the objective function $\mathcal{E}$ and the



(a) (b)

Figure 3.14: (a) pattern resolution optimization; (b) word set over Hamming distance for a word size of 7×7px (—) and the required number of words for the 125×125px projection pattern (—).

evaluation of all camera poses is repeated until convergence. The resulting errors are illustrated in Fig. 3.14(a). Based on this theoretical assessment, half the image sensor resolution was chosen as a starting point for the projector prototype. The specific imaging capabilities are tested on the actual hardware, cf. Sec. 3.3.7.

Next, the codeword size $N = n \cdot n$ needs to be determined. This final step also answers the question of the degree of fault tolerance of the pattern. Let us start by considering the number of codewords that are available for pattern generation for a m-ary alphabet $\mathcal{A}$, where $|\mathcal{A}| = m$. All codewords are designed to provide a guaranteed minimum Hamming distance d to all other words of the pattern. In $\mathcal{A}^N$, the Hamming sphere with radius d centered at a codeword $\mathbf{x}$ is the set of all words where the Hamming distance to $\mathbf{x}$ is at most d [165]:

$$S_d(\mathbf{x}) = \left\{ \mathbf{y} \in \mathcal{A}^N | d_H(\mathbf{x}, \mathbf{y}) \leq d \right\}. \tag{3.42}$$

codewords

The cardinality of some sphere $S_d(\mathbf{x}) \subseteq \mathcal{A}^N$ is

$$|S_d(\mathbf{x})| = \sum_{i=0}^{d} \binom{N}{i} (m-1)^i, \tag{3.43}$$

where $m = |\mathcal{A}|$. There are $|C|$ spheres, thus

$$|\mathcal{A}^N| = m^N \geq |C| \sum_{i=0}^{d} \binom{N}{i} (m-1)^i. \tag{3.44}$$

Solving for $|C|$ yields to a bound for the number of codewords with minimum Hamming distance d:

$$|C| \leq \frac{m^N}{\sum_{i=0}^{d} \binom{N}{i} (m-1)^i}. \tag{3.45}$$

Clearly, the set of possible codewords with a greater Hamming distance is exhausted more quickly for short sequences and increasing $|\mathcal{A}^N|$. However, the risk of decoding

errors also increases with the word length, since noise impedes a correct classification of pixels. Intensity-based matching shows exactly the opposite behavior. Here, a small block size is susceptible to noise, but yields sharp depth boundaries. The depth confidence increases for larger blocks, since more information is considered, but tends to blur depth discontinuity. We have chosen a word size of 7×7 and a pattern size of 125×125px. Fig. 3.14(b) illustrates the progression of the available word set over Hamming distance. For this size, a total number of $118 \cdot 118 = 13,924$ globally non-recurrent codewords is required.

Pattern Generation

Unfortunately, the above-introduced globally non-recurrent projection pattern with guaranteed minimum Hamming distance between all codewords cannot be created in a descriptive way. This means that there is no known rule, after which the matrix can be created. A generative generation of the matrix in a trial-and-error manner is computationally expensive. In particular, the set of codewords decreases with increasing Hamming distance and makes it difficult to fill the matrix (cf. Fig. 3.14(b)). Morano et al. created a small 45×45 pseudo random structure by checking all existing codewords for consistence with the constraints after adding a new letter of the alphabet used [124]. We introduce an accelerated version of this method. The algorithm starts

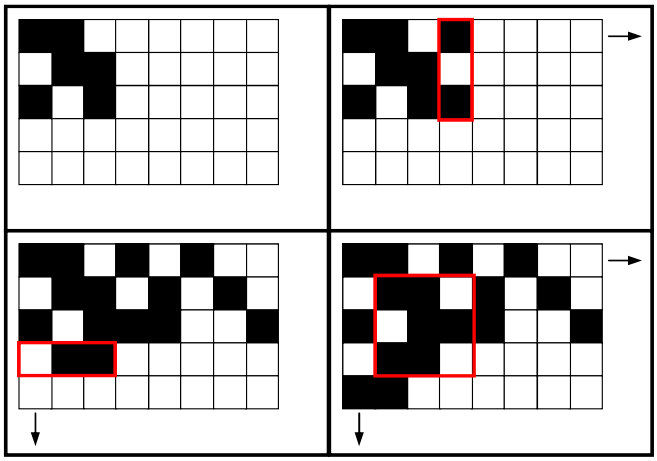


Figure 3.15: Process of generating a binary globally non-recurring pseudorandom pattern with word size 3×3.

by seeding one $n \times n$ codeword at the top left corner of the pattern with a random pixel assignment (cf. Fig. 3.15). The rest of the matrix is filled by iterating between adding a new codeword and checking it against *all* existing words. A word is considered to be valid if and only if the Hamming distance is exactly d and the word appears exactly once in the entire matrix. If either condition is violated, the word is rejected and the algorithm proceeds with a new random assignment. For the first row of the pattern, adding a random column vector of height n generates a new word. Equally, adding a row vector of size n yields pattern growth at the left margin. For the rest of the matrix, adding a single pixel generates a new codeword each time.

Algorithm 1: Validate codeword

```
def contains(node, word, dist = d)
begin
    if (node == null) then
        return false                                   ▷ word does not exists
    end
    if (length(word) == 0) then
        return true                                    ▷ word found
    end
    letter, rest = head(word), tail(word)              ▷ split word
    if contains(left ? node.left : node.right, rest, dist) then
        return true                                    ▷ word found
    else if (dist == 0) then
        return false
    else
        return contains(left ? node.right : node.left, rest, dist − 1)
    end
end
```

The bottleneck of this approach is the comparison of new codewords against all others. A patter of size $s \times s$ requires $q = (s - n + 1)^2$ words to fill the matrix. Thus, the Hamming distance must be computed at least

$$c = \sum_{i=1}^{q-1} i = \frac{q(q-1)}{2} \tag{3.46}$$

Regarding our pattern of size 125×125px and codewords of 7×7 this yields to $q = 14,161$ and $c = 100,259,880$ codeword comparisons.

To efficiently generate larger binary patterns, we boost the codeword comparison using a binary tree. Two main operations are performed on the tree: insertion of a new codeword and validation of a codeword. The *insertion* operation is trivial: while iterating over the positions of the letters of a codeword, we add a new left subtree for the letter 0 and a right subtree for the letter 1. The *contains* operation recursively checks if a codeword is valid, thus it does not exists in the tree and differs in d positions from all other words (cf. algorithm 1). The method starts at the root node with $dist = d$. If the node is unset, the tree does not contain the word. Otherwise, the word length is checked. If it is zero, the current node represents the word, otherwise we recursively descent in the left respectively right node. If the word has not been found up to this point, it must be checked whether it differs $dist$ positions from all other words. This is the case for $dist == 0$. Otherwise we validate the rest of the word with the Hamming distance set to $dist = dist - 1$.

3.3.5 Energy Formulation

To efficiently combine stereo matching and the decoded pattern information, we formulate the depth reconstruction as an energy minimization problem. A high matching

confidence is assumed for pixel positions that are recovered from the projected pattern structure. We treat them as a sparse set of ground control points with known disparity. The GCPs are used as regularization term to constrain the disparity search in a global Markov Random Field (MRF) stereo framework. From a Bayesian perspective the posterior probability, including the GCP regularization prior, can be written as [207]

$$\rho(D|I_s, G) \propto \rho(I_s|D)\rho(G|D)\rho(D), \tag{3.47}$$

where D is the dense disparity map, G are sparsely distributed GCPs, and $I_s = (I, J)$ is the stereo image pair. Taking the negative logarithm

$$-\log \rho(D|I_s, G) = -\log \rho(I_s|D) - \log \rho(G|D) - \log \rho(D) \tag{3.48}$$

casts the problem to a Gibbs energy minimization [189] with

$$\mathcal{E}(D) = \mathcal{E}_d(D) + \lambda_g \mathcal{E}_g(D) + \lambda_s \mathcal{E}_s(D). \tag{3.49}$$

The term $\mathcal{E}_d$ is the data energy, $\mathcal{E}_s$ the smoothness prior energy and $\mathcal{E}_g$ the GCP energy. The weights λ_g and λ_s control the influence of the regularization terms on the overall energy. The corresponding graphical model of the MRF is an undirected graph $\mathcal{G} = (V, E)$, where a value d_p from the finite label set of disparity values $d \in \mathcal{L}$ is assigned to every node $p \in V$, as illustrated in Fig. 3.16(a). The nodes correspond to the discrete pixel grid of the image. For the sake of simplicity scalar indices are used to identify pixel positions instead of a vector notation in the sequel.

The likelihood energy of the data term $\mathcal{E}_d$ measures the intensity consistency of pixel correspondences for a potential disparity d_p

$$\mathcal{E}_d(D) = \sum_{p \in I} \psi_p(d_p), \tag{3.50}$$

where

$$\psi_p(d_p) = \min\left(\Phi(I_p, J_{p+d}), \Delta_d\right) \tag{3.51}$$

is a pixel-wise cost between both stereo image pairs for a given disparity d_p using a dissimilarity function $\Phi(\cdot, \cdot)$. The costs are truncated at Δ_d. We employ the non-parametric Census transform [216, 215], since it ideally supports the projected Hamming pattern. The Census transform compares the gray value of the central pixel of a window with all neighbors in a predefined order, i.e. clockwise:

$$\xi(I_i, I_j) = \begin{cases} 1 & \text{if } I_i > I_j, \\ 0 & \text{if } I_i \leq I_j. \end{cases} \tag{3.52}$$

The resulting binary descriptor C represents a mapping of the pixel intensities with respect to the central pixel calculated for both of the stereo images

$$C(I_i) = \bigotimes_{n \in N} \xi(I_i, I_n), \tag{3.53}$$

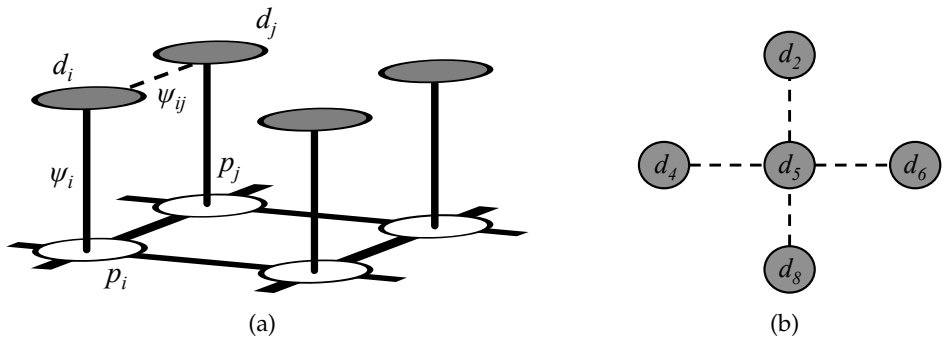


Figure 3.16: (a) graphical model of the stereo Markov random field; and (b) standard 4-connected neighborhood.

where the operator $\otimes$ concatenates the result of the single intensity comparisons. The final dissimilarity is then measured using the Hamming distance (3.38) between the two bit vectors:

$$\Phi_\tau(I_i, J_j) = d_H(C(I_i), C(J_j)). \tag{3.54}$$

It makes sense to choose the window size of the Census transform equivalent to the word size of the projection pattern. By design, mismatches between pixel positions therewith vary by at least the minimum Hamming distance of the pattern codewords. Since no pixel values are compared directly, the Census transform is also capable of handling sensor-related brightness variations, which cannot be avoided with the small cameras despite an illumination sensitivity calibration of the CMOS sensors.

To render the ill-posed stereo matching problem well-posed, regularization is necessary in addition to the likelihood obtained from the dissimilarity measure. Local smoothness is encoded with a regularization prior using the standard 4-connected neighborhood (cf. 3.16(b)). The smoothness function penalizes variations between labels of nodes in the neighborhood, assuming piecewise smoothness of the scene depth

$$\mathcal{E}_s(D) = \sum_{(i,j)\in\mathcal{N}_4} \psi_{ij}(d_i, d_j), \tag{3.55}$$

with

$$\psi_{ij}(d_i, d_j) = \min\left(w_{ij} \cdot |d_i - d_j|^a, \Delta_s\right). \tag{3.56}$$

We employ two widely used models, where penalty costs are either chosen to increase linearly ($a = 1$), or quadratically ($a = 2$). The costs are truncated at a maximum value Δ_s. The scalar w_{ij} allows pairwise weighting of the costs.

Additional regularization helps to improve reconstruction quality. A priori knowledge about the scene can be employed as constraint, e.g. obtained from laser-range finders, image-region segmentation, or reliably matched feature points [60, 207, 35, 187, 211, 219].

In our case, the prior comes from pattern decoding, which allows indexing spatially unique pixel positions. As with any binary-encoded one-shot structured light method,

decoding success depends significantly on the perspective distortion of the projected codewords on the scene's surface. This is also why small codewords are more robust against distortion. Remember that our projector is equipped with the same type of lens than the cameras, therewith most of the distortion is caused by the scene itself. For this reason, successfully decoded words are expected to cluster in spatially connected regions with low surface curvature. In conventional structured light settings, positions decoded in the camera image need to be related to the projector to infer depth. Hence, it is necessary to perform projector calibration. To become independent of any additional calibration other than the cameras, we decode the pattern in both image pairs. After finding corresponding codewords in both images the disparity can be calculated with respect to the stereo camera configuration. [93] proposed this advantageous method to support the correspondence search in stereo matching by projecting a continuous color pattern. Here, the projector can be positioned freely in space, while we need to keep the distance to the cameras as small as possible to minimize perspective distortion.

GCPs

Prior to pattern decoding, the input images are down-sampled with the aspect ratio α between camera and projector, since only pixels up to the resolution of the pattern mask can be encoded unambiguously. Dynamic thresholding is applied and decoding is performed according to algorithm 1, whereat the word input parameter is build from binary pixel values. Each sub-window of the chosen codeword size is validated in the images. Thereafter, decoded pixel positions are checked for coherence. The codeword is rejected, if it was not found in both images or the spatial offset indicates a decoding error. Disparity values for the ground control points are calculated by comparing matching codewords in the left and right image respectively. The final ground control point map $\widetilde{D}$ is obtained by up-sampling the set of found GCP disparities to the camera resolution. Assuming piecewise smoothness of the scene depth, gaps resulting from the aspect ration α are filled in this process with the disparity value of the neighboring GCP.

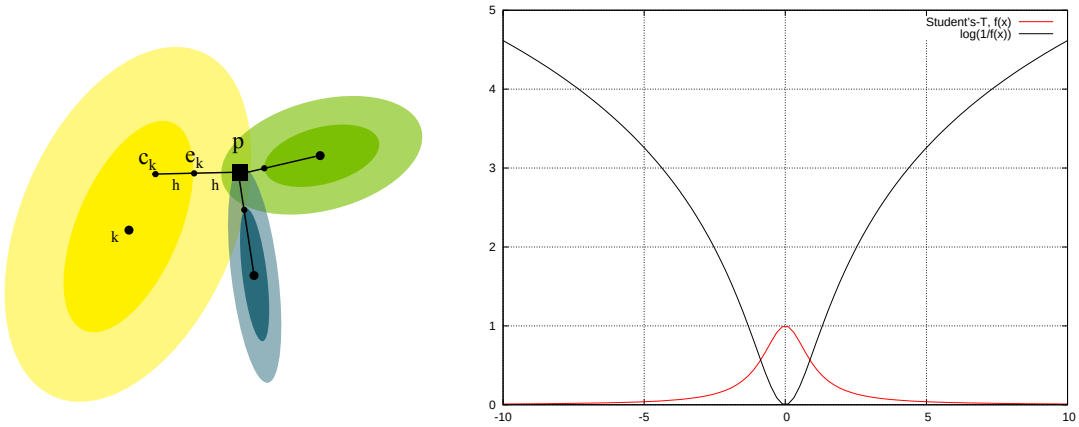
The resulting disparity values of the ground control points $\widetilde{D}$ are propagated within their spatial neighborhood to interpolate a dense GCP map $G(\widetilde{D})$. Decoded GCPs tend to cluster, since the decoding success depends on factors including the degree of local pattern distortion. Contour pixels of recovered GCP clusters are organized in a k-d tree structure. Non-GCP pixel values are predicted based on the distance-weighted values of the k neighboring GCPs with

$$G(\widetilde{D}) = \sum_{p \in \widetilde{D}} \left(\frac{\sum_{q \in \mathcal{N}_{knn}} \tilde{w}_{pq} \widetilde{D}_q}{\sum_{q \in \mathcal{N}_{knn}} \tilde{w}_{pq}} \right). \tag{3.57}$$

The distances are penalized with an inverse quadratic disparity weight

$$\tilde{w}_{pq} = \frac{1}{1 + (\|p - q\|_2^2)^2}. \tag{3.58}$$

Further, we impose a confidence value for interpolated pixels of the dense disparity map of ground control points. We fit a bivariate Gaussian to each of the k GCP clusters. Pixels lying within the range of the Gaussian are scored with a confidence value



(a) (b)

Figure 3.17: (a) calculation of the confidence value c_p for pixel p; (b) Student's-T distribution and derived energy in log-space.

of $c_p = 1.0$. Non-GCP pixels p receive their confidence value from the neighboring GCP's, determined by their corresponding Gaussian. Therefore, we expand the initial distribution such that the new distribution starts at the boundaries of the old Gaussian. This step is illustrated in Fig. 3.17(a), where the new influence region is depicted brighter than the initial distribution. We calculate the distance h to the nearest ellipse point e_k of pixel p and follow that direction beginning from e_k with the same distance, resulting in the new pixel position c_k. The final confidence score for the non-GCP pixel p is then evaluated by

$$c_p = \min\left(\sum_k \exp\left(-\frac{1}{2}\mathbf{E}_{ck}^{\mathrm{T}}\boldsymbol{\Sigma}_k\mathbf{E}_{ck}\right), 1.0\right). \tag{3.59}$$

The GCP energy term

$$\mathcal{E}_g(D) = \sum_{p\in I}\psi_{gcp}(d_p, \tilde{d}_p), \tag{3.60}$$

with $\tilde{d} \in \widetilde{D}$, penalizes diverging disparity assignments according to

$$\psi_{gcp}(d_p, \tilde{d}_p) = \begin{cases} \Phi_s(d_p, \tilde{d}_p, c_p) & \text{, if } \; c_p > \gamma \\ \frac{\Phi_s(d_{min}, d_{max}, 1.0)}{|\mathcal{L}|} & \text{, otherwise,} \end{cases} \tag{3.61}$$

if the pixel received a confidence value that exceeds γ. The energy function Φ_s is derived from a Student's-T distribution

$$\Phi_s(d_p, \tilde{d}_p, c_p) = \log\left(\frac{1}{\left(1 + |d_p - \tilde{d}_p|^2 c_p^2\right)^{-1}}\right). \tag{3.62}$$

An example distribution for $c = 1.0$ is shown in Fig. 3.17(b).

The finally resulting disparity assignment can be obtained by applying existing energy minimization techniques. We implemented belief propagation [55] and a graph cut [96], more precisely the FastPD algorithm [97].

3.3.6 Disparity Refinement

As a last step, we refine the resulting disparities. Occlusion handling is treated by calculating the disparity map from both left and right images and performing a consistency check to eliminate mismatches. That is, disparity values from one image are reprojected into the other image. If the difference of the disparity values exceeds an acceptable tolerance δ_t, the pixel is invalidated. Otherwise, the final disparity value is computed by taking the average of both values.

$$D_p = \begin{cases} 0, & \text{if } |D_{r,p} - D_{l,p}| > \delta_t, \\ \frac{|D_{r,p} - D_{l,p}|}{2}, & \text{else.} \end{cases} \tag{3.63}$$

Local speckle peaks are identified to remove spurious pixels that do not represent a valid structure. Missing disparity values, which result from removing small segments, are interpolated to obtain a complete disparity map. Finally, the disparity map is processed with a bilateral mean filter to smooth depth transitions. The bilateral mean filter is an edge-preserving filter, which treats the spatial and the color domain with two separate Gaussian kernels:

$$\overline{D}_p = \frac{\sum_{q \in N} G_{\sigma_s}(\|p - q\|) \cdot G_{\sigma_c}(D_p - D_q) \cdot D_q}{\sum_{q \in N} G_{\sigma_s}(\|p - q\|) \cdot G_{\sigma_c}(D_p - D_q)}, \tag{3.64}$$

where G_{σ_s} and G_{σ_c} are spatial Gaussians in the coordinate (s) respectively color (c) domain.

3.3.7 Experiments

We start by evaluating the decoding performance under different settings in the simulation environment. Camera parameters were transferred from calibration results obtained with the NanEye™ micro cameras. Similar distortion parameters were assumed for the projector lens. Calibration was performed with a checkerboard pattern, where each checkerboard is 1mm in length. The pattern was exposed to a printed circuit board (PCB) to guarantee the required accuracy. To adjust image appearance of both sensors, a linear illumination scan was performed and lookup calibration tables were

	x	y	z
translation	-1.20088	-0.00424	0.01008
rotation	0.00151	-0.00854	-0.00602

Table 3.2: Result of extrinsic camera calibration. Translation in mm, rotation as Rodrigues vector. Remember that the stereo setup was design with a translation of $(-1.2, 0.0, 0.0)^{\mathrm{T}}$ and zero rotation.

generated. Of particular interest is the result of the extrinsic calibration, since we specified the sensor arrangement in Sec. 3.3.2. As table 3.2 shows, the baseline of 1.2mm was met precisely. As expected, manufacturing the stereo setup directly on silicon yields zero rotation of the two sensors, which allows omitting the rectification step and the camera images can simply be corrected for lens distortion.

With respect to the proposed pattern design, we investigate the effect of Hamming distance, sensor noise, viewing angle and projector rotation in a synthetic scene, rendered with the above-introduced ray tracing framework. As reference, a planar test scene was used, depicted in Fig. 3.18. The distance between plane and camera is 1cm. Two

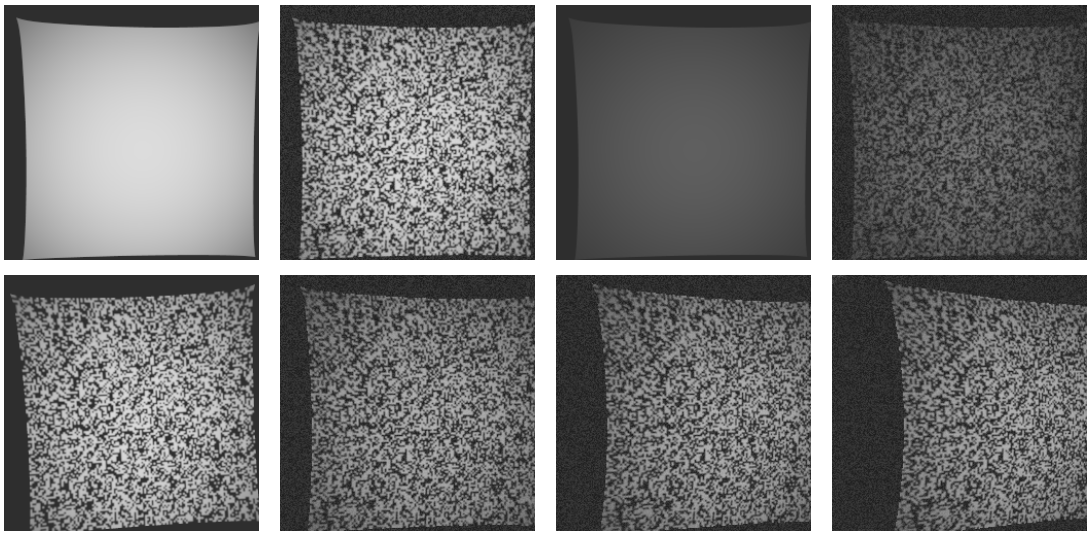


Figure 3.18: Left camera view on the simulated planar test scene. Top row: illustration of the two different illumination settings used during the experiments, each with and without projected pattern. Moderate white Gaussian noise with a standard deviation of $\sigma = 0.16$ was added to the images with simulated pattern. The projector is positioned according to the proposed setup, thus with a baseline of 0.6mm to each camera and with a y-offset of 0.6mm below the cameras. Bottom row, from left to right: projector rotation of 3.0° with respect to the camera (first image), and impact of larger baselines between camera and projector with 1.2mm, 2.4mm, and 3.6mm on pattern distortion.

different brightness settings of the projector, referred to as "level 1" for sufficient illumination and as "level 2" for a significantly reduced brightness, illuminate the scene. Decoding was performed with and without sensor noise. To simulate moderate sensor noise, additional Gaussian white noise with a standard deviation of $\sigma = 0.16$ was added to the images. Imperfect alignment of the projector against the camera was simulated by rotating the projector with respect to the camera. At an angle of approximately 3° decoding performance reduces to zero. This fact indicates that the projector alignment for the final hardware setup needs to be performed with a precise alignment device, since the rotation cannot be determined and compensated for. Fig. 3.18 also demonstrates the effect of larger baselines between camera and projector. An increas-

ing baseline not only significantly reduces the observable projection area, but increases pattern distortion. Table 3.3 summarizes the results with the number of successfully decoded pixel positions.

d_H	noise σ	intensity	projector rotation	decoded	%
1	0.0	level 1	0.0	**739**	**4.73**
1	0.15	level 1	0.0	**618**	**3.96**
1	0.0	level 2	0.0	**722**	**4.62**
1	0.15	level 2	0.0	**712**	**4.56**
1	0.0	level 1	3.0	**340**	**2.18**
1	0.15	level 1	3.0	**291**	**1.86**
1	0.0	level 2	3.0	**334**	**2.14**
1	0.15	level 2	3.0	**131**	**0.84**
4	0.0	level 1	0.0	**2722**	**17.42**
4	0.15	level 1	0.0	**2399**	**15.35**
4	0.0	level 2	0.0	**2693**	**17.23**
4	0.15	level 2	0.0	**1859**	**11.90**
4	0.0	level 1	3.0	**2016**	**12.90**
4	0.15	level 1	3.0	**1875**	**12.00**
4	0.0	level 2	3.0	**1935**	**12.38**
4	0.15	level 2	3.0	**1292**	**8.27**
8	0.0	level 1	0.0	**6412**	**41.04**
8	0.15	level 1	0.0	**5971**	**38.22**
8	0.0	level 2	0.0	**6453**	**41.30**
8	0.15	level 2	0.0	**5051**	**32.33**
8	0.0	level 1	3.0	**5724**	**36.63**
8	0.15	level 1	3.0	**5415**	**34.66**
8	0.0	level 2	3.0	**5792**	**37.07**
8	0.15	level 2	3.0	**4423**	**28.31**

Table 3.3: Evaluation of the decoding performance based on the ray tracing emulation environment. Hamming distance d_H; noise σ of sensor; projector light intensity level; viewing angle on scene in degree; projector rotation about the viewing axis in degree; number of decoded pattern pixels; percentage of decoded pattern pixels.

We recall that we deal with the special case of using the same lens for both camera and projector. If camera and projector would share the same origin, the camera captures the projected pattern distortion-free. Apart from slanted scene objects, the imaged pattern distortion is caused by the offset between camera and projector. Due to the axis alignment of camera and projector, as well as the small baseline in our setting, this distortion is comparably low. Therefore, it is also possible to decode the pattern without prior remapping of the camera image that removes lens distortion. The effect is depicted in Fig. 3.19, second and third row. While images in the second row are decoded as captured by the camera, images in the third row are undistorted beforehand. We use the color coding depicted in the first row of Fig. 3.19 to visualize the pixel displacement with respect to the original pattern mask. Pattern displacement ground truth, obtained from the ray tracing emulation, is provided for some of the scenes. Images in the third row illustrate the effect on a non-planar dental impression model. As a further test scene, we use a down-scaled version of the Stanford bunny. In general, the Hamming

distance chosen for our pattern ($d_H = 8$) provides a satisfactory decoding performance. However, decoding success significantly decreases on slanted surfaces.

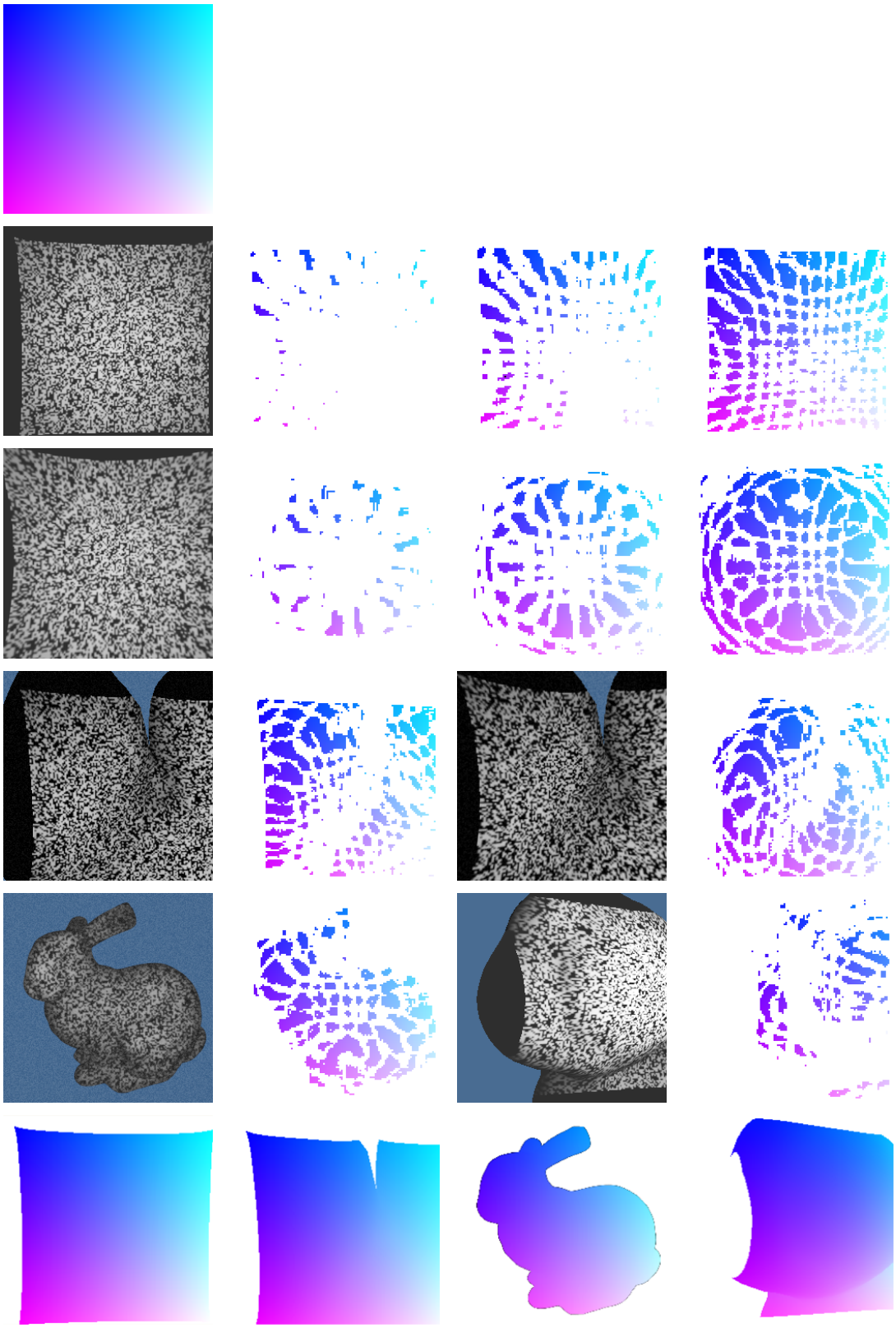
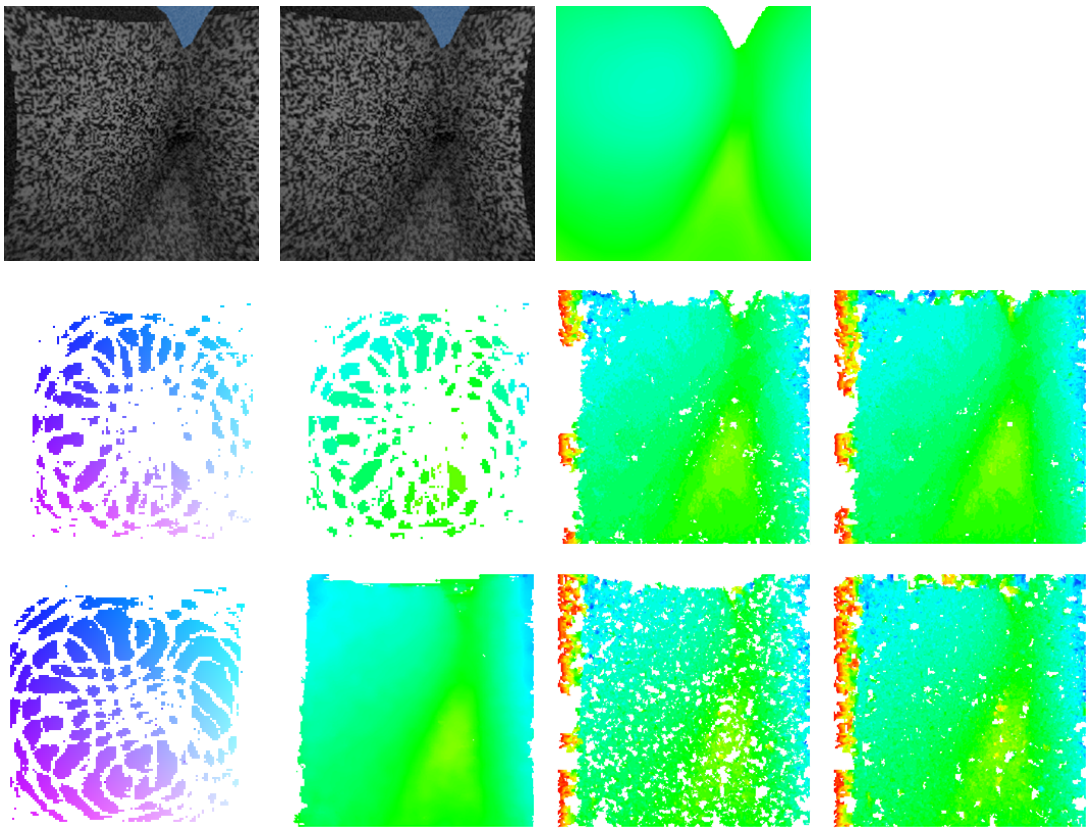


Figure 3.19: Decoded pixel positions after consistency check. First row: color coding scheme. Second row: the camera image was not undistorted before decoding with Hamming distances 1, 4, and 8. Third row: Same scene, but undistorted camera image. Fourth row: decoding of a tooth, without and with undistorting the image. Fifth row: Stanford bunny and closeup of the bunny's head. The model was down-scaled to fit the camera requirements. Sixth row: ground truth pattern displacement.

Next, we investigate the integration of the GCPs into the reconstruction framework. Window-based dissimilarity measures generally perform well under the idealized conditions of the emulation environment. To better elucidate the influence of the GCP prior, we decided to use the absolute differences (AD) of pixels as dissimilarity measure. Since AD does not score an image region but single pixels, it is highly sensitive to noise. Therewith, regularization is important to avoid mismatches and gaps in the range map. Strong weighting of the regularization energy $\mathcal{E}_s$, which considerers the 4-connected neighborhood of pixels, might yield to over smoothing of disparity values. In contrast, the GCP energy $\mathcal{E}_g$ integrates scene-dependent knowledge with already well inferred disparity values from pattern decoding. The information gained from the encoded pattern mask is too sparse to describe the scene. However, it reduces uncertainty during correspondence search.

The experiments were conducted on scenes of different difficulty (cf. Fig. 3.20). Again, noise with a standard deviation of $\sigma = 0.16$ was added to the camera images. No post processing, such as gap interpolation, disparity refinement, or left/right consistency check was performed, expect the removal of speckle. With respect to each test scene, the first row shows the undistorted left and right camera image as well as the ground truth range map. The first image of the second row shows the decoded GCPs after coherence check. Compared to the decoded GCPs of the right camera image, which are depicted in the third row, the number of found GCPs is reduced after matching the left and right ground control points. This effect can mainly be attributed to the undistortion process, which destroys the pattern structure in certain image areas. The finally inferred disparity values of the decoded pattern are shown in the second row, second image. To calculate the final range map, we employed belief propagation and stoped the optimization process after four respectively eight iterations to score results. This allows comparing the influence of the GCP prior with respect to the conventional smoothness prior. While the last two images in the second row show the results with GCP prior, the optimization of the last two range images of the third row was performed without ground control points. Clearly, the results obtained with ground control points after four optimization steps already outperform the results after eight iterations, when only the smoothness prior is used.



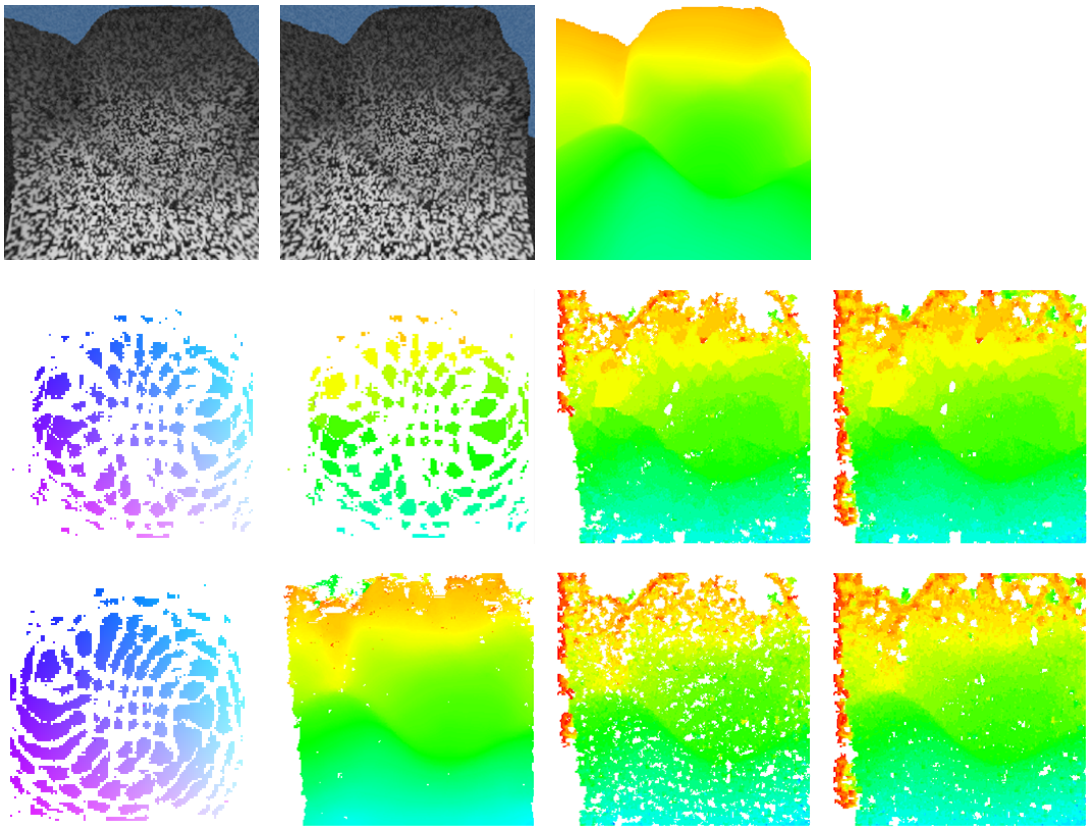


Figure 3.20: With respect to each test scene, the first row shows the undistorted left and right camera image along with the ground truth range map. The second row shows the decoded GCPs after coherence check and the corresponding GCP disparities. The last two images of the second row show the inferred disparities after four respectively eight iterations using belief propagation and the absolute difference dissimilarity measure. The corresponding results without the use of the GCP prior are illustrated in the last two images of the third row. Finally, the first two images of the third row show the GCPs that were decoded in the right camera image, and the resulting disparity map of the proposed framework using the Census transform and the GCP prior.

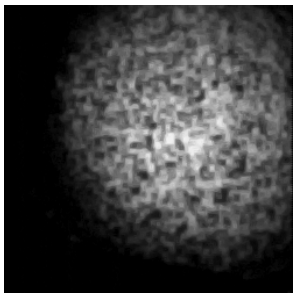


Figure 3.21: Mismatch of current projector and camera resolution: the GOBO mask cannot be imaged by the micro camera when mounted co-planar. For the next prototype, a lower projector resolution is necessary.

The described setup was actually realized on hardware. The pattern is transferred by means of chromium to a glass substrate with a size of 0.75×0.75mm. The substrate was mounted behind the lens using a precise alignment device. A 1mm fiber optic light guide was bonded to provide illumination. The prototype projector was assembled by Awaiba GmbH. Fig. 3.22 illustrates the projector and the achieved quality: the first image shows the entire projection area on a plane surface, where the borders are distorted by the lens. Notice the projector at the upper edge of the image. The last image shows the projection on a dental impression model, whereas the pattern structure is clearly visible.

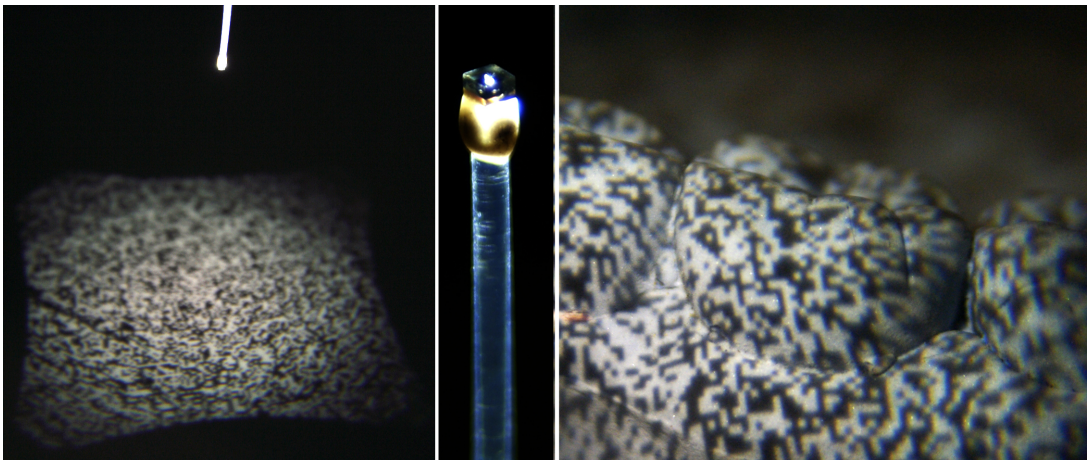


Figure 3.22: The assembled micro projector system (tip dimension 1×1mm) is shown in the center image: the pattern structure was transferred to a GOBO mask and mounted behind a lens that is bonded to a 1mm fiber optic light guide. The first image shows the total projection with border distortion. The last image shows the projection of the pattern on a dental test scene.

Based on the experiments of Sec. 3.3.4, half the camera resolution was used to realize the globally unambiguous Hamming pattern on the GOBO mask. This decision was based on two aspects. On one hand, we wanted to evaluate the imaging performance of the projector with a high resolution. The result can then serve as a reference for future projector prototypes. On the other hand, we wanted to examine the resolution capability of the micro cameras. For this, also a grayscale camera without Bayer pattern was employed, which improves sharpness due to the missing color interpolation. Unfortunately, the manufactured pattern structure is too small to be imaged with the current version of the micro cameras, as depicted in Fig. 3.21. Based on the gained experience, further investigations are necessary to find a matching projector resolution.

micro projector

Therefore, depth perception with the proposed hardware setup is currently performed without decoding the pattern. However, note that the projection is still essential to enable dense depth reconstruction. The projector was offset with respect to the micro cameras so that the pattern was accurately imaged. This yields pattern distortion in the image and a significant brightness decrease. Hence, different reconstructed objects were painted with white color to enhance the texture projection. Specifically, we re-

constructed some wiring on a surgical instrument, a heart symbol, a depression on a plaster cast model, a tooth from a dental impression model, and a screw head, as illustrated in Fig. 3.23. The first two rows show the micro camera view with and without texture projection respectively. An external light source illuminated the scene for images captured without pattern projection. The last two rows show the inferred depth using belief propagation and the FastPD algorithm, as introduced above. Both optimizers provide similar results. Also note the spatial extent of the reconstructed structures. The "wiring" shown on the first object has a spatial extent of only 0.1-0.2mm. When additionally using the GCP prior, we expect depth maps to become smoother.

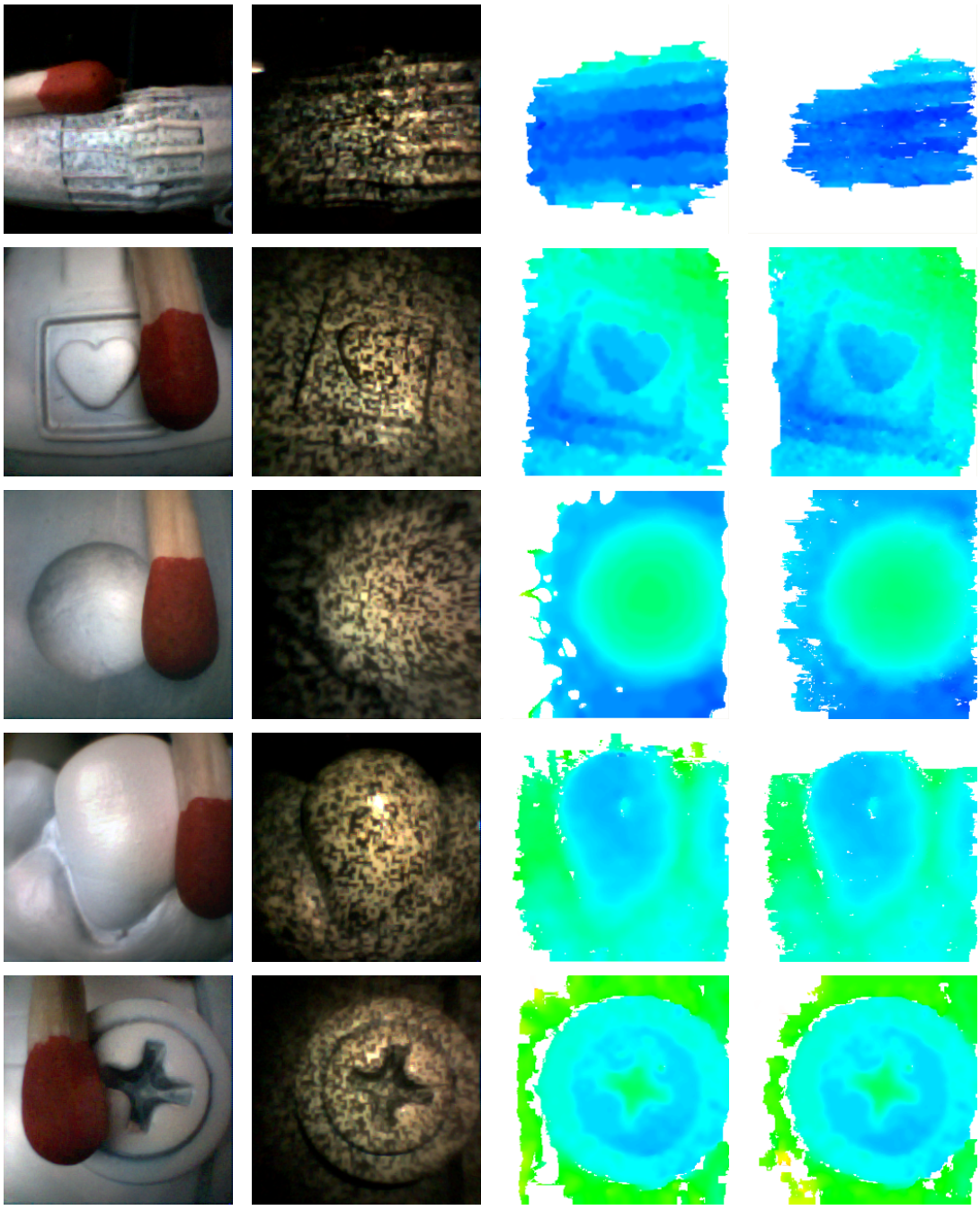


Figure 3.23: Reconstruction of different objects with the miniaturized camera setup. Columns: scene image without pattern projection, captured with micro camera and external illumination; camera image with pattern projection; disparity map obtained with belief propagation; disparity map obtained with graph cut. Rows: wires on a surgical instrument (wire diameter approx. 0.1-0.2mm, diameter of instrument shaft is 8mm); heart symbol; recess in dental impression; tooth of dental impression; screw head.

4 Interactive System Control

The (partial) autonomous execution and assistance of frequently performed, error-prone, and exhausting tasks has shown enormous potential to speed up procedures and to mentally relieve surgeons. To allow autonomous behavior, integration of contextual information about the situs is necessary. Further, intuitive user interfaces are required to interact with the surgical robot. Specifically calling system commands is essential, but has been paid little attention yet. In this chapter, we tackle these problems by introducing a gesture-based input channel. Likewise, we build on the results of the preceding chapter to deduce control methods for corrective motion planning, taking the acquired situs knowledge into account. We investigate in accurately guiding instruments to target regions and in assisting surgeons during small-scaled fine manipulation tasks.

4.1 Instrument and System Control

We distinguish between the control strategies that generate motion commands for the manipulators and the control of system functionality. The two control types differ in the amount of human interaction required. System control, i.e. calling a specific system command, is always activated by the operator. In contrast, instrument control demands previously acquired situs knowledge, but at the same time might also rely on additional meaningful input from the operator. For example, endoscope control uses a defined target position to align the camera. On one hand, this information can be conveyed while the camera follows a surgical tool, whereas the motion command is deduced from the position of the tracked tool. On the other hand, the operator can also manually define a target. Providing intuitive input channels that facilitate the handling of the growing amount of functionality and at the same time tightly integrating the possibility of conveying task-relevant information, which is associated with the triggered function, is challenging. We start by reviewing existing interfaces regarding these manifold requirements.

Manual control is still the most widespread method for both instrument- and system control. It comes with the burden of having the surgeon to actuate a clutch that de-

couples the input devices from the manipulators. After decoupling, the input devices can be used with another instrument or as a pointing device. To resume telepresence control, the posture of manipulator and input device needs to be synchronized. The procedure is time-consuming and inconvenient. Additional foot pedals are typically provided to activate and deactivate control.

Hands-free control entered the operating theater with the emergence of robot assistants. For instance, voice-activated control allows the surgeon to move the camera based on a limited number of voice commands (e.g., "left, right, start, stop"), while still being able to handle the surgical instruments [170]. The surgeon must often wear a dedicated microphone to ensure an adequate voice quality for speech recognition. The recognition rate of such short commands, which are not presented in the context of a sentence, still leaves room for improvement. Other systems aim to detect the motion of body parts. [136] uses head movements: head worn IR-emitting tracking markers were detected and their movement was interpreted as input command. A drawback of many hands-free methods is that a given command usually only allows executing a single manipulator movement at a time (either horizontally, vertically, or depths), prolonging and complicating the alignment. [63] replace head tracking with the detection of "mouse-gesture" commands and therewith made the idea applicable to master-consoles that fix the surgeon's head. [65] explored alternative input devices to remotely control the surgical instruments of the daVinci robot. Using a Microsoft Kinect™ sensor, 3D hand gesture tracking was used for fine manipulation tasks. The instrument pose is derived from the relative positions of both hands.

Gaze contingent control currently develops to an interesting alternative. Mylonas et al. utilize the relationship between horizontal disparity of both eyes and depth perception, which varies with the viewing distance, to deduce depth information at the operator's fixation point [204]. The information gained was e.g. used to adaptive motion stabilization during beating heart surgery [128], to interactively prescribe virtual fixtures [129], and to plan 3D paths in situ during focused energy ablation [184]. Noonon et al. performed gaze contingent articulated robot control and evaluated strategies for joint selection [137, 138]. An eye tracker was integrated into the daVinci™ console, which allows video capturing of the eyes at 50fps. The eyes are illuminated with a fixed infrared light source and the corneal reflection is measured in relation to the position of the pupil. This particular setting comes with the advantage of a fixed head position and a non-obstructing view on the eyes, but is rather device specific and cannot be applied to other setups.

Workflow analysis draws the attention of researchers in order to model and analyze medical procedures [110]. Knowledge about the course of action of a specific type of intervention can be used to detect its current phase. Weede et al. applied the paradigm to predict suitable laparoscope positions with a Markov model [209]. To increase the recognition rate of the individual surgical stages, they extended the approach with an instrument classification based on the visual bag-of-words approach [210]. With respect to the extend of autonomy, workflow analysis is probably the most "intelligent" of the suggested methods. However, the complexity of surgical trajectories represents

a significant obstacle and large data sets are necessary to reliably train the underlying models. This requirement is not easy to comply, since each single type of intervention needs to be modeled individually.

Haptic constraints capitalize on the accuracy of robotic systems, enhancing the operation speed, and reduce mental stress, while permitting the user to retain ultimate control over the system [151]. Haptic virtual fixtures have been implemented in both medical telepresence [23, 22, 28, 163] as well as in cooperative control systems [202, 31, 115] to shape the motion of surgical instruments, e.g. to assist the operator during complex tasks, such as knotting or cutting [88, 155]. The former systems usually offer admittance-type haptic devices. The latter are impedance type robots, therewith maintaining high stiffness.

The reviewed input interfaces reveal that, so far, the vast majority of effort has been put on investigations to enhance instrument control. However, no significant alternatives for user input are available. New input channels need to be naturally integrated into the working environment, offering a pervasive way to call system commands while keeping the interaction time at a minimum. Beyond, the inclusion of ancillary instructions into this process remains a critical factor. To tackle these problems, we introduce **gesture-type input** and interpret movements directed at the master's haptic devices as a user command [15]. The method facilitates the indication of task-relevant in situ coordinates associated with the execution.

system control

Following the individual stages of our initially presented task of tissue dissection, we deduce control laws for **corrective motion planning** [9, 16]. We aim to automatically align surgical tools with a specified target in situ. We guide both, the laparoscope as well as surgical instruments. Visual servo control is well suited for this purpose, since it operates directly on image data, therewith avoiding many of the system-related uncertainties and supporting our approach of online surgery.

instrument control

4.2 Gesture-based Input Interface

Gesture recognition is widely studied as a computer input modality (see e.g., [122, 126]) and increasingly used in the medical field, e.g., [32, 176]. In particular, hand gestures are very intuitive and expressive, whereat two categories are distinguished: static finger configurations, also called postures, and dynamic gestures. In [154], for instance, the authors proposed to instruct a teleoperated manipulator using natural sign language. The signs define a spatial-temporal context for the ensuing robot behavior, e.g. the user points to an object. In a similar fashion, we aim to transfer this easy understandable concept to the medical workstation and *interpret the movements of the haptic devices as input command*. The following advantages compared to the prevalent menu-type interfaces are expected:

- the execution time of gestures is constant compared to menu interfaces, where the interaction time is dependent on the complexity of the menu,

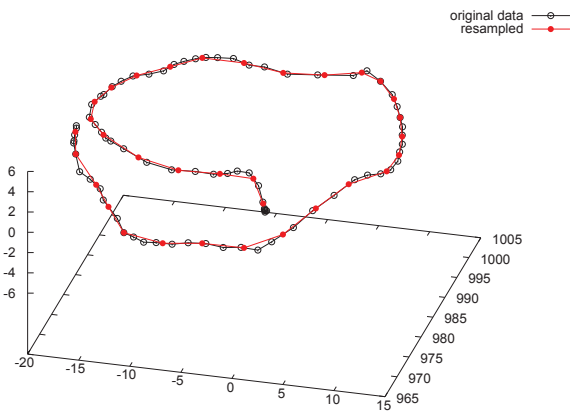


Figure 4.1: Equidistant resampling the recorded instrument trajectory introduces a uniform distribution of sampling points independent of the execution speed.

- the assignment of gestures and commands is customizable to individuals,
- no time-consuming decoupling/resynchronization of the haptic devices with the manipulators is necessary,
- due to the spatio-temporal context of the modality, ancillary location-dependent information, such as the incision points of a suturing task, can be defined after the command using the instrument as a pointing device .

At the master console, the recognition of a gesture command is activated by pressing a foot pedal and completed when the pedal is released again.

4.2.1 Recognizing Gestures

Variations in the execution of gestures occur between different instantiations as well as different users. Due to the close relationship to surgical workflow modeling, we follow the successful application of discrete Hidden Markov Models (HMM) to classify gestures [166, 160, 144, 159]. Discrete models have the advantage of requiring less training and being computationally less expensive during the parameter optimization over continuous Markov models [66, 139]. From an implementation point of view, it is irrelevant whether the features used for training the model are obtained from encoder readings of the haptic device or from the slave manipulators. We opted for the latter.

Markov Modeling

A Hidden Markov Model is a stochastic model with two random processes [156]. A HMM can be described by the quintuple $\lambda = (A, B, N, M, \pi)$, where N is the number of hidden states $\{s_i : 1 \leq i \leq N\}$ of the model, M the number of possible observation symbols, $A = \{a_{ij}\}$ the transition probability matrix between states s_i and s_j, and the observation symbol probability distribution in state i, $B = \{b_i(k)\}$ for $\{k : 1 \leq k \leq M\}$. Variable π denotes a probability distribution over the initial states. The first process of the model is a hidden Markov chain and describes the dependency of the state s_n, reached at time n, from the previous state s_{n-1}. The second stochastic process defines the emission probability of the observation.

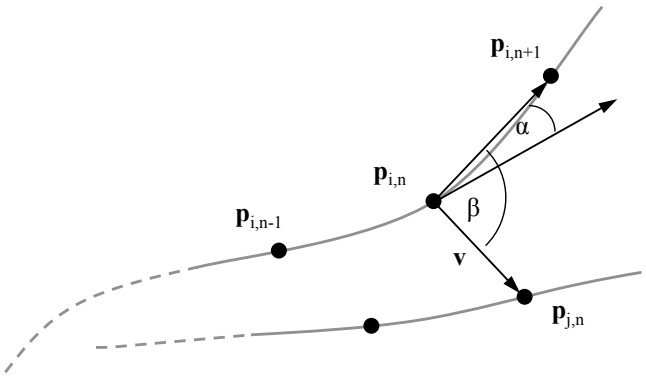


Figure 4.2: Directional change of one instrument (α) and directional change of one instrument wrt. a second instrument (β).

For each of the gestures a model with left-right topology was applied: states are arranged in a linear progression, whereas each state is entered at least once and no transitions to past states are allowed. The HMMs were trained with the Baum-Welch algorithm [156]. Experimental results revealed that four to six states are appropriate to model the gestures. Approximately 25 demonstrations for each gesture were taught by a person who was not involved in the later evaluation [14]. The Viterbi algorithm was used to find the path with the highest likelihood $\rho\left(\Pi|\lambda\right)$ through the topology of λ that would generate the sequence Π [156]. In order to prevent the underflow problem during longer time-series, a $\log$-scaling is used.

Data Acquisition and Preprocessing

The trajectories of both instruments were sampled at a frequency of 10Hz and stored in a data base. The recorded raw data is represented by $\mathbf{s}_{i,n} = (x, y, z, f_x, f_y, f_z, g, t)_{i,n}$, where $(x, y, z)_{i,n} =: \mathbf{p}_{i,n}$ is the Cartesian position of the instrument and $(f_x, f_y, f_z)_{i,n}$ the corresponding force, which occurs at the distal end of the instrument. Index i indicates the associated instrument, n the number of the data point, g denotes the state of the gripper (open/closed), and t is a timestamp. The varying execution speed of gestures yields to an inhomogeneous distribution of sampling points with unequal spacing and requires resampling the input data. For instance, trajectory segments that are executed with low velocity, such as tight turns, comprise more sampling points than fast movements, such as straight lines. To represent the data within a regularly spaced distribution it is resampled position equidistant [87], that is with a uniform spatial spacing l between an old position $\mathbf{p}_{i,n}$ and the new position $\tilde{\mathbf{p}}_{i,n+1}$, as illustrated in Fig. 4.1. The Euclidean distance d_E is used for a linear interpolation, such that

$$d_E(\mathbf{p}_{i,n}, \tilde{\mathbf{p}}_{i,n+1}) \stackrel{!}{=} l. \tag{4.1}$$

In doing so, minor non-meaningful geometrical variations, e.g. caused by human hand tremor, are also removed. In the sequel, the preprocessed data is denoted with $\tilde{\mathbf{s}}$ and used for feature extraction.

Features

After preprocessing, trajectories are represented by sampling points with equal distances along with the recorded instrument forces and the gripper state. To describe the vector between two adjacent sampling points $\mathbf{p}_{i,n}$ and $\mathbf{p}_{i,n+1}$, the substitution $\Delta\mathbf{p}_{i,n} = \mathbf{p}_{i,n+1} - \mathbf{p}_{i,n}$ is applied, where $\|\Delta\mathbf{p}_{i,n}\| = l$ after the resampling step. The model feature vector $\mathbf{f}_t = (f_1, \cdots, f_{12})_t$ is derived from the input data as follows:

1. **Change in direction of the instrument trajectory f_1, f_2:** The change of the instrument direction, defined by two adjacent points $\mathbf{p}_{i,n}$ and $\mathbf{p}_{i,n+1}$ can be described by the angle α_t, enclosed by $\Delta\mathbf{p}_{i,n-1}$ and $\Delta\mathbf{p}_{i,n}$ (cf. Fig. 4.2). To ensure a continuous representation, sin and cosine of the angle are calculated [69]:

$$f_{1,t} = \cos\alpha_t = \frac{\Delta\mathbf{p}_{i,n-1} \times \Delta\mathbf{p}_{i,n}}{\|\Delta\mathbf{p}_{i,n-1}\| \cdot \|\Delta\mathbf{p}_{i,n}\|}, \tag{4.2}$$

$$f_{2,t} = \sin\alpha_t = \frac{\Delta\mathbf{p}_{i,n-1}\Delta\mathbf{p}_{i,n}}{\|\Delta\mathbf{p}_{i,n-1}\| \cdot \|\Delta\mathbf{p}_{i,n}\|}. \tag{4.3}$$

2. **Relative motion direction of one instrument wrt. to the second instrument f_3, f_4:** The feature describes the change of direction of one instrument regarding the second instrument. E.g., it is indicated if the instruments move towards, or away, from each other. Similar to (4.2) and (4.3), sin and cosine of the angle β are calculated between the vectors $\Delta\mathbf{p}_{i,n}$ and $\mathbf{v} = \overline{\mathbf{p}_{i,n}\mathbf{p}_{j,n}}$, where i and j denote the two instruments. Also compare Fig. 4.2.

3. **Velocity of an instrument f_5, f_6:** The velocity of the instrument tip is numerically approximated with the timestamp t of the recorded data. After the linear resampling of the trajectory, the velocity of each instrument is

$$f_{5/6,n} = \frac{l}{\tilde{\mathbf{s}}_{t,n} - \tilde{\mathbf{s}}_{t,n-1}}, \tag{4.4}$$

   where l is the new distance between the two points after resampling. Note that from the position equidistant resampling follows a resampling of the timestamps that yields certain inaccuracies.

4. **Distance between the two instruments f_7:** Depending on the gesture, the Euclidean distance between two instruments varies over time or keeps similar (e.g., in case of parallel moving instruments). The feature represents the current distance and is denoted as

$$f_{7,n} = \|\mathbf{p}_{i,n} - \mathbf{p}_{j,n}\|. \tag{4.5}$$

5. **Distance change between two instruments f_8:** In contrast to feature f_7, the distance change between two instruments over time indicates the movement direction of one instrument wrt. the second one:

$$f_{8,n} = \|\mathbf{p}_{i,n} - \mathbf{p}_{j,n}\| - \|\mathbf{p}_{i,n-1} - \mathbf{p}_{j,n-1}\|. \tag{4.6}$$

6. **State of the gripper** f_9, f_{10}**:** The state of the grippers can directly be taken from the recorded dataset and is represented with

$$f_{9/10,n} = \tilde{\mathbf{s}}_g = \begin{cases} 1, \text{ if the micro gripper is closed,} \\ 0 \text{ else.} \end{cases} \tag{4.7}$$

7. **Force magnitude** f_{11}, f_{12}: The three-dimensional force vector is immediately available from the sensor reading. To be more robust against noise, only the force magnitude is used:

$$f_{11/12,n} = \sqrt{\tilde{\mathbf{s}}_{f_x}^2 + \tilde{\mathbf{s}}_{f_y}^2 + \tilde{\mathbf{s}}_{f_z}^2}. \tag{4.8}$$

All features are normalized over the range of the demonstrations. Discrete observations are obtained by quantizing the feature vector using the k-means++ algorithm [27].

4.2.2 Finding Intuitive Gestures

In principle, gesture-based input has the disadvantage that the gestures need to be remembered and linked mentally to a surgical action, whereas menu entries only need to be recognized. This requires greater cognitive effort and increases the risk of false input commands. This disadvantage can be minimized if the gestures that need to be remembered are intuitive. It is hence necessary to first identify gestures that would feel intuitive for the user with respect to a certain surgical tasks and can therefore be easily remembered and executed. For this purpose, an exploratory user experiment was conducted. A fundamental part of the input method is that the surgeon can freely combine any personalized gesture with any system function. To train the system with individual gestures for each subject, however, would be to time-consuming. Thus, a set of the four most distinct combinations of gestures and actions were used for the later evaluation.

Materials and Methods

The goal of the study was to identify intuitive hand gestures, which are appropriate to trigger a corresponding system function. The surgical actions itself were identified to be recurrent during surgical interventions in advance. The study was conducted with an opportunity sample of 22 participants, 14 of whom were male and 8 female. The average age was 36yrs. with $\sigma = 14$yrs.. During the experiment subjects were asked to spontaneously perform two alternative gestures that they would associate with each of nine pre-selected surgical assistance function. In detail, we had a closer look at the following automatable tasks: surgical knot tying, surgical suturing, retraction of the assistant arm, measuring of the distance between two points, alignment of the endoscopic camera with a predefined position (in our case either “home position” or “position one”), automated alignment of the camera with a surgical instrument, and extraction of liquids by suction. The surgical tasks were chosen at random and each action had to be repeated three times. First, the participants had to present two different possibilities in order to elect their preferred gesture in the third repetition. Thereafter, subjects were asked to give a reason for their choice. A countertop that limits the

(a) (b)

(c) (d)

Figure 4.3: Images taken from the video sequence of gesture "measure distance", showing the following stages: (a) initial position; (b) instrument 1 moves towards instrument 2; (c) intermediate position; (d) end of the gesture reached, once instrument 1 approaches its initial position again. [*original video recorded and analyzed by V. Nitsch and I. Karl*]

workspace to 30×30×30cm and a mockup of the original haptic input devices served as test-bed. Two styli have been complemented with brackets that can be used to indicate the state of the gripper. The forearm was placed on a slabstock foam, what resembles the posture people usually take at the master console. All subjects were shown a video introduction of the system. The experiments have been video taped (see Fig. 4.3) and each gesture was reviewed according to six different criteria:

1. The *handedness* describes whether the left or the right hand is predominately used for the execution.

2. The *symmetry* of the movement indicates if both hands performed a similar movement (symmetrical), or if one hand contributed more to the gesture than the other.

3. The *curvature* describes whether the movement was rather straight, curved, or a combination of both.

4. Analyzing the principal axis of movement gives information on whether the ges-

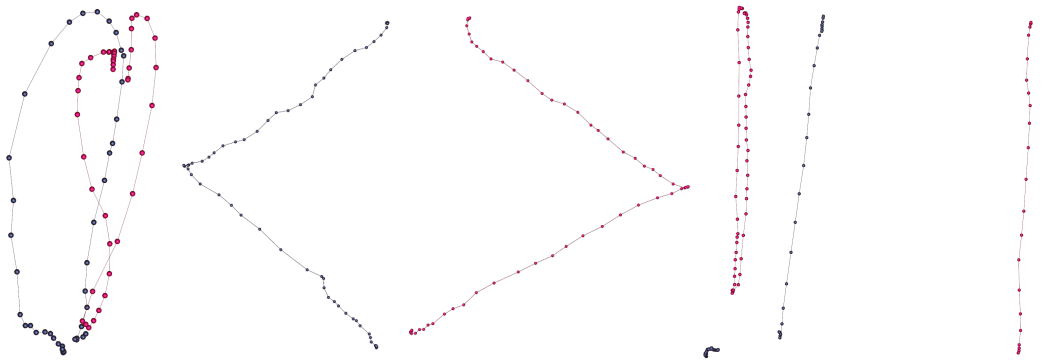


Figure 4.4: Final trajectories of selected gesture instances. Blue lines indicate the left instrument, red lines indicate the right instrument. The trajectories were linked to the following gestures: (a) knot tying; (b) suturing; (c) measure distance (rotated by 90° for better visualization); (d) retract assistance arm.

ture has rather been performed in a *plane* or all 3 dimension have been exploited in space.

5. The *gripper state* can be changed during a gesture several times and indicates whether the gripper at the distal end of the instrument is closed or open.

6. The *quantity* of the movements provides conclusions about how many movements have been performed with one hand.

Results

In the following analysis of the results, the chosen gestures of the first and second round of the experiments were combined, so we come to a total of 44 runs per surgical action. Both evaluations concerning the gesture-based input were carried out in cooperation with V. Nitsch and I. Karl, University of Armed Forces, Human Factors Institute. As part of his MD, J. Haas supported us in medical questions during the second evaluation. The analysis of the video material showed that the state of the gripper was rarely considered by the subjects and can be neglected for all tasks. In all cases, the gesture that was performed first was preferred by the subjects. Thus, we conclude that intuitive gestures, which are associated with a certain task, come to mind faster than others. Subjects have chosen the gestures either because of its close symbolic link to the related action or perceived the movement of the hand to be unique. Neither the differentiation of gestures from each other, nor the simplicity of the execution did play a role for the subjects so far.

In a real-life scenario, it is not only important to use intuitive gestures, but they also have to be well distinguishable and quickly executable. The following four gestures that were highly consisted in the manner of its execution were selected for the later usability evaluation.

- *Surgical knot-tying*: The subjects used both hands in 82% of the runs. Typically, a symmetrical movement (72%) was performed. One, two, or three subsequent circular movements were performed, which were either curved (32%) or a combination of sinuous and straight movements (59%). In most of the cases, the movements included all three spatial dimensions (68%).

- *Surgical suture*: Approximately 50% of the subjects employed either one hand or both hands (45%). Half of the presented movements were curvilinear (50%). Distinctive were 4 up to 8 three dimensional movements (59%) with both hands.

- *Retraction of the assistant arm*: In 68% of the cases the subjects utilized only the right hand. Most of the time only a single movement was executed (75%). The gesture was performed with a linear movement in 54% of the cases. All spacial dimensions were utilized up to 45% while 27% of the subjects utilized only a single plane.

- *Measure the distance between two points*: Most of the subjects utilized both hands (82%) for this gesture, at which 61% symmetrical movements were performed. In 94% of all cases one hand performed only one movement, while the second hand either performed one movement (45%) or two movements (54%). In each case one third of the gestures were carried out by sinuous, straight, or combined movements.

After minor adaptations of the gesture concerning their handling on the surgical robot, the system was taught the four gestured shown in Fig. 4.4.

4.2.3 Haptic-Type Input vs Menu-Type Input

The four gestures elected in the preceding study were used to compared the effectiveness of the proposed input method against traditional menu-based input. Our main hypothesis, which has been proven correct, was that gesture-based input reduces interaction time during the call of system commands, compared to menu-based input. In addition to objective measures of performance, another aspect evaluated in this study was the user experience. User experience is an extension of the concept of usability and has been defined as "all aspects of the user's experience when interacting with the product, service, environment or facility. [...] It includes all aspects of usability and desirability of a product, system or service from the user's perspective" [182].

Materials and Methods

Subject of this study were 24 medical students with an average age of 24yrs. (σ = 3yrs.), half of whom had surgical experience. Eleven participants were female and all but two were right-handed. A 2×4 ((input mode)×(gesture)) within-subject design was implemented, whereby gesture input was tested against menu input. A plausible two-tiered menu design was chosen for this experiment, with which participants had to select two options for each gesture: on the first screen of the menu, a general "surgical action" option had to be activated, which then led to the second screen on which

the appropriate gesture had to be selected and confirmed.

The menu could be operated with two foot pedals: the next menu item was selected by pressing the first pedal, the second pedal confirmed the selection. As the end of the menu was reached, the selection moved back to the first entry. The time that it took people to activate a surgical action with the respective input mode was measured in each trial (input time), as well as the success rate in triggering the correct action (input success). The user experience of both input modes was assessed with the AttrakDiff2 [78], a well-tested questionnaire measuring four different aspects of user experience. The four aspects of user experience measured are: pragmatic quality (PQ), attractiveness (ATT), hedonic quality-stimulation (HQ-S) and hedonic quality-identity (HQ-I). The construct of pragmatic quality refers to the perceived ability of a product to accomplish task goals by offering useful and usable functions and requires participants to rate the system on items such as complicated/simple and unpredictable/predictable. Attractiveness measures the users' global positive/negative evaluation of a product and contains items such as pretty/ugly and attractive/repulsive. Hedonic quality-stimulation refers to the ability of a product to satisfy the user's needs for the development of one's knowledge and skills and is rated with items such as unimaginative/creative and lame/mesmerizing. Finally, the construct of hedonic quality-identity measures the extent to which a product promotes one's self-worth and is comprised of items such as unstylish/stylish and cheap/valuable.

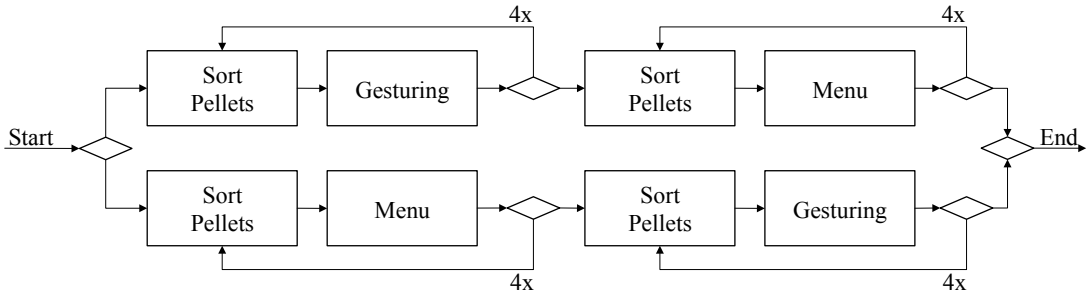
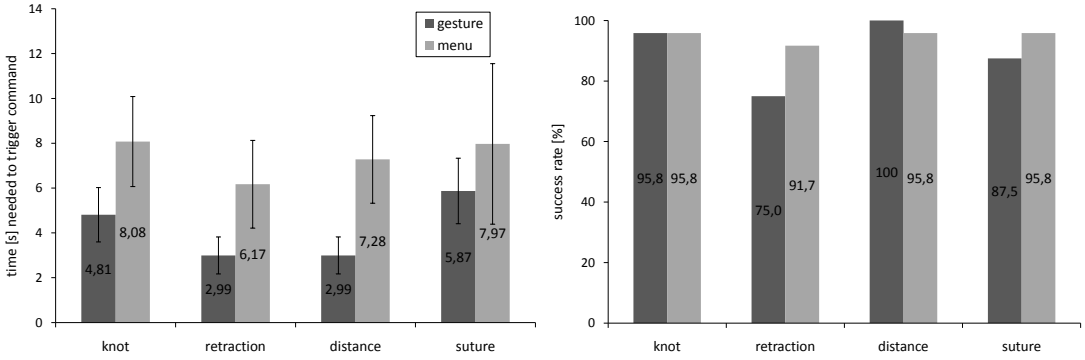


Figure 4.5: Course of evaluation for each subject (without training phases).

Prior to the experiment, participants were trained in the use of both the menu and the gesture input modes in triggering the four gestures according to a standardized training procedure. All subjects had sufficient time to familiarize themselves with the general handling of the system. On average, participants took 6.75min ($\sigma = 2.66$min) to learn the four gestures, whereas it took on average 2.96min. ($\sigma = 1.12$min) to learn how to navigate the menu efficiently. Upon successful completion of the training phase, participants were then asked to either perform a certain gesture or select the appropriate menu items in order to trigger a particular action. The input modes were trained and tested in one block, meaning that participants would first be trained, then perform with one input mode, after which they would be trained and tested with the other input mode (cf. Fig. 4.5). The input mode and the tested gestures were systematically varied for each person in order to avoid learning or fatigue effects. To embed the initiation of surgical actions in a holistic process, the participants had to sort small polymer



(a) Trip time per individual action (b) Comparison of the success rates

Figure 4.6: Statistical plots of the evaluation of haptic-type input vs menu-type input.

pellets of two different colors. First, subjects had to decide, depending on the color of pellet, which pellet type they wanted to transport with either the left or the right surgical instrument. Each color had to be stored in its dedicated petri dish.

Results

A factorial one-way analysis of variance (ANOVA) found a large and statistically significant effect of input mode on input time ($F(1, 22) = 38.44, p < .001, \eta^2 = .64$). The estimated marginal means indicate that, on average, it took significantly less time to trigger the surgical action via gesture input ($\mu = 4.45$sec, $\sigma = 0.86$sec) compared to activation via menu input ($\mu = 7.41$sec, $\sigma = 2.06$sec). The times needed to trigger a certain action are depicted in Fig. 4.6(a). There was also a significant main effect of gesture ($F(2.04, 44.79) = 23.79, p < .001, \eta^2 = .52$), but no significant interaction effect ($F(3, 66) = 2.18, p = .10$). Together, these results suggest that while some surgical actions (e.g. arm retraction) took longer to activate than others (e.g. distance measuring or suturing), input times were consistently shorter with gesture input than with menu input. A look at the input errors suggest that, while it took less time to input a command for a surgical action via gesture, this mode is slightly more error prone with 10.42% of gesture inputs classified as false compared to 5.21% of false inputs via the menu (out of 96 commands). Fig. 4.6(b) shows the success rates for the individual actions. Finally, an ANOVA of the AttrakDiff2 scores indicates a significant main effect of input mode ($F(1, 23) = 23.74, p < .001, \eta^2 = .51$), whereby significantly higher mean user experience scores were given for gesture input ($\mu = 5.40, \sigma = 0.87$) than for menu input ($\mu = 4.21, \sigma = 0.85$). Only the scores to pragmatic quality did not differ significantly between the two input modes ($t(23) = 0.17, p = .87$), whereas gesture input received significantly higher ratings for hedonic quality-identity ($t(23) = 4.67, p < .001, r = .70$), hedonic-quality stimulation ($t(23) = 7.97, p < .001, r = .86$) and attractiveness ($t(23) = 4.39, p < .001, r = .68$). These findings indicate that, while the gesture input system was not necessarily considered to provide greater functionality than the menu, it was perceived to be more comfortable and stimulating. In comparison, the results show that gesture-based input is faster and receives more favorable user experience ratings compared to the tested menu-mockup, even though this input method

is still slightly more error prone. Although the effect of learning on input success has not been explicitly investigated in this study, it seems likely that, despite the rigorous training protocol implemented in this experiment, participants were more practiced in menu-based input than in gesture-based input. Hence, one might assume that the likelihood to commit an error with gesture input would decrease with further practice. In addition, further studies are required to determine the factors that mitigate the effectiveness of gesture-based input. For example, obviously, the superior effectiveness of gesture-based input over the traditional menu input strongly depends on the complexity of the menu, as well as the input mechanisms (e.g. foot pedals vs. mouse-type interaction). Some subjects also had concerns about the feasibility of gesturing during a real intervention, since they were in full control of the robots during gesture-type interaction. Spatially limited gestures might therewith be preferable over more complex ones. The last point we want to highlight is that all gestures were taught the system by a single person. Demonstrations can therewith be biased in favor of this person and influence the recognition rate.

4.3 Visual Instrument Control

In the previous section, we proposed a new method to intuitively trigger a system function, such as our reference task of assisted tissue dissection. In this regard, the next step is the alignment of the scalpel with the defined target on the tissue surface [20, 21]. We indicate the incision point on the surface using an instrument-mounted laser. Next, we discuss how the scalpel can be aligned precisely with this target using visual guidance.

As introduced in Sec. 1, MIRS systems typically suffers from a multitude of different error sources, which affect their overall precision. In particular if the laparoscope is used as a 3D sensor that defines Cartesian task coordinates that are executed by a second manipulator, the resulting accuracy is not sufficient for fine manipulation. This technique is known as position-based visual servoing (PBVS). Although tasks such as automated scissors [145] and grasping of surgical suture material [133] were demonstrated with PBVS, the success can mainly be attributed to the relatively large opening angle of scissors. This fact reduces the demands on the accuracy required to grasp surgical suture material.

Image-based visual servoing (IBVS) overcomes intrinsic system errors, including mechanical play and calibration uncertainties. By integrating visual data directly into the control loop, endoscopic instruments can be accurately aligned with a target position. [102] for instance, guides a surgical instrument based on orientation marks, which were projected by a shaft-mounted optical device and allow recovering the relationship to the tissue surface. Nageotte et al. presented visual three-dimensional path following [130]. Their overall goal is to define sequences for autonomous image-guided suturing, at which the motion necessary for tissue punctuation with circular needles is calculated [131, 132]. [83] manually selected setpoints in stereo images to define trajectories of surgical tasks. Beyond, visual servo control is applied to synchronize the manipulator

motion with organ movements, e.g. during beating heart surgery [142].

We distinguish two cases of visual instrument control: autonomous positioning of surgical tools and autonomous camera control. As a general rule, visual servo control aims to minimize a task function dependent on an time-varying error $\mathbf{e}_t$ between the current pose of the robot and a reference pose [43, 44]. The error is derived by observing visual features $\mathbf{s}_t$ and their desired goal configuration $\mathbf{s}_d$

$$\mathbf{e}_t = \mathbf{s}(\mathbf{m}_t, \mathbf{a}) - \mathbf{s}_d, \tag{4.9}$$

where $\mathbf{m}_t$ comprises visual measurements that build a feature descriptor. The vector $\mathbf{a}$ incorporates additional knowledge, such as camera parameters, into the process. The design of $\mathbf{s}$ specifies the control law. While image-based control relies on features immediately available in pixel coordinates, position-based control treats the camera as a 3D sensor, thus operates in Cartesian coordinates. Accordingly, depth values need to be acquired. This error-prone step makes position-based methods usually more sensitive to calibration uncertainties as image-based methods [54, 82].

Assuming a hand-mounted camera, visual servoing links the relationship of observed features to the velocity $\boldsymbol{\xi} = [{}^{\mathrm{C}}\boldsymbol{\upsilon}, {}^{\mathrm{C}}\boldsymbol{\omega}]^{\mathrm{T}}$ of the camera frame C, where ${}^{\mathrm{C}}\boldsymbol{\upsilon}$ is the instantaneous linear velocity and ${}^{\mathrm{C}}\boldsymbol{\omega}$ is the instantaneous angular velocity. The relationship between $\boldsymbol{\xi}$ and the image feature velocity $\dot{\mathbf{s}}$ is given by the visual Jacobian, or so-called interaction matrix $\mathbf{L}_s(\mathbf{q})$:

$$\dot{\mathbf{s}} = \mathbf{L}_s(\mathbf{q})\boldsymbol{\xi}. \tag{4.10}$$

The interaction matrix, for which we simply write $\mathbf{L}_s$, is function of the manipulator configuration $\mathbf{q}$ and the image features $\mathbf{s}$

$$\mathbf{L}_s(\mathbf{q}) = \mathbf{L}_s = \left[\frac{\delta \mathbf{s}}{\delta \mathbf{q}}\right] = \begin{bmatrix} \frac{\delta s_1(\mathbf{q})}{\delta q_1} & \cdots & \frac{\delta s_1(\mathbf{q})}{\delta q_m} \\ \vdots & \ddots & \vdots \\ \frac{\delta s_n(\mathbf{q})}{\delta q_1} & \cdots & \frac{\delta s_n(\mathbf{q})}{\delta q_m} \end{bmatrix}, \tag{4.11}$$

with a feature vector comprising n distinct features and m is the dimension of the task space. When combining (4.9) and (4.10) we immediately obtain the relationship between the velocity screw $\boldsymbol{\xi}$ and the time variation of the error

$$\dot{\mathbf{e}} = \frac{d}{dt}(\mathbf{s}_t - \mathbf{s}_d) = \dot{\mathbf{s}} = \mathbf{L}_e\boldsymbol{\xi}, \tag{4.12}$$

where $\mathbf{L}_e = \mathbf{L}_s$. Using $\boldsymbol{\xi}$ as input to the manipulator (that carries the camera) and an exponentially decoupled decrease $\dot{\mathbf{e}} = -\lambda\mathbf{e}$ of the task function, the control law is given by

$$\boldsymbol{\xi} = -\lambda\mathbf{L}_e^{+}\mathbf{e}, \tag{4.13}$$

where λ is a proportional gain and $\mathbf{L}_e^{+}$ is the pseudoinverse of $\mathbf{L}_e$.

In the sequel, we further adapt this control law to our needs. We derive a servoing scheme that considers the remote center of motion in MIRS and allows positioning

surgical tools on the tissue surface. A practical difficulty is, however, that the tool has to be in the field of view of the laparoscope. Therefore, a switching control scheme is employed, which first drives the instrument into the field of view using end-point open loop position control and then continues the alignment based on raw image data. During position-based control, most of the time only the target can be observed in image space and calibration errors of the camera, the relationship between the involved manipulators, and the instruments affect the accuracy.

instrument control

With the instrument being located in the field of view of the laparoscope, image-based servo control is applied to meet the demanded accuracy. For a single tracked feature point ${}^{\mathrm{W}}\mathbf{x}$ in world coordinates that projects to image space as ${}^{\mathrm{I}}\mathbf{x} = [x_x, x_y]^{\mathrm{T}}$ the related visual Jacobian is [43]

$$\mathbf{L}_{\mathbf{x}} = \begin{bmatrix} -\frac{1}{z} & 0 & \frac{x_x}{z} & x_x x_y & -(1+x_x^2) & x_y \\ 0 & -\frac{1}{z} & \frac{x_y}{z} & 1+x_y^2 & -x_x x_y & -x_x \end{bmatrix}. \tag{4.14}$$

By stacking the Jacobians of the feature points $\mathbf{m} = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$, we obtain $\mathbf{L} = [\mathbf{L}_{\mathbf{x}_1}, \cdots, \mathbf{L}_{\mathbf{x}_n}]^{\mathrm{T}}$. Variable $\mathbf{a}$ now is the calibration of the stereoscopic laparoscope that is used to estimate $z := {}^{\mathrm{W}}\mathbf{x}_z$. Note that $n \geq 3$ is necessary to control six degrees of freedom. In MIRS, the trocar constraint governs two degrees of freedom of the instrument movement at the incision point P, restricting linear movements at this point to either insertion or retraction. The velocity ${}^{\mathrm{P}}\boldsymbol{\xi}$ at the trocar and the velocity ${}^{\mathrm{E}}\boldsymbol{\xi}$ at the end of the instrument's shaft are related as

$${}^{\mathrm{P}}\boldsymbol{\xi} = {}^{\mathrm{E}}_{\mathrm{P}}\mathbf{V}\,{}^{\mathrm{E}}\boldsymbol{\xi}, \tag{4.15}$$

or more precisely

$$\begin{bmatrix} {}^{\mathrm{P}}\boldsymbol{\upsilon} \\ {}^{\mathrm{P}}\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} {}^{\mathrm{E}}_{\mathrm{P}}\mathbf{R} & {}^{\mathrm{E}}_{\mathrm{P}}[\mathbf{t}]_{\times}\,{}^{\mathrm{E}}_{\mathrm{P}}\mathbf{R} \\ \mathbf{0}_{[3\times 3]} & {}^{\mathrm{E}}_{\mathrm{P}}\mathbf{R} \end{bmatrix} \begin{bmatrix} {}^{\mathrm{E}}\boldsymbol{\upsilon} \\ {}^{\mathrm{E}}\boldsymbol{\omega} \end{bmatrix}, \tag{4.16}$$

with ${}^{\mathrm{E}}_{\mathrm{P}}[\mathbf{t}]_{\times}$ the skew-symmetric matrix associated with ${}^{\mathrm{E}}_{\mathrm{P}}\mathbf{t}$. Assuming a straight instrument shaft (or a calibrated one, cf. Sec. 5.2.3), the rotation matrices of (4.16) are the identity $\mathbf{I}_3$ (or the calibration matrix respectively) and the translation ${}^{\mathrm{E}}_{\mathrm{P}}\mathbf{t} = [0, 0, d]^{\mathrm{T}}$ governs the insertion depth d of the instrument with respect to the trocar. The trocar constraints all movements at P to the shaft's z-axis (that is the direction of the shaft), thus ${}^{\mathrm{P}}\boldsymbol{\upsilon} = [0, 0, {}^{\mathrm{P}}\upsilon_z]^{\mathrm{T}}$. With (4.15) and simple developments we obtain the relationship

$$\begin{bmatrix} 0 \\ 0 \\ {}^{\mathrm{P}}\upsilon_z \end{bmatrix} = \begin{bmatrix} {}^{\mathrm{E}}\upsilon_x - d\,{}^{\mathrm{E}}\omega_y \\ {}^{\mathrm{E}}\upsilon_y + d\,{}^{\mathrm{E}}\omega_x \\ {}^{\mathrm{E}}\upsilon_z \end{bmatrix} \tag{4.17}$$

between the linear and the angular velocity at the distal end of the instrument, which is finally solved for

$${}^{\mathrm{E}}\omega_x = -\frac{{}^{\mathrm{P}}\upsilon_y}{d} \qquad \text{and} \qquad {}^{\mathrm{E}}\omega_y = \frac{{}^{\mathrm{P}}\upsilon_x}{d}. \tag{4.18}$$

The Cartesian position of the trocar points is well-known and defined in our system's software framework. Thus, the insertion depth of the instrument can be calculated using the forward kinematics.

Considering a point-to-point alignment of the surgical tool with a target, the linear velocity ${}^{\mathrm{E}}\boldsymbol{\upsilon}$ and the angular velocities ${}^{\mathrm{E}}\omega_x$ and ${}^{\mathrm{E}}\omega_y$ of the tool need to be calculated. The latter two are governed by the trocar constraint according to (4.18). The tracked instrument position is given in both cameras of the stereoscopic laparoscope and denoted as ${}^{\mathrm{C_l}}\mathbf{x} = \left[{}^{\mathrm{C_l}}x_x, {}^{\mathrm{C_l}}x_y\right]^{\mathrm{T}}$ and ${}^{\mathrm{C_r}}\mathbf{x} = \left[{}^{\mathrm{C_r}}x_x, {}^{\mathrm{C_r}}x_y\right]^{\mathrm{T}}$ for the left $\mathrm{C_l}$ and right $\mathrm{C_r}$ camera frame respectively. Therewith, we control the necessary five degrees of freedom. The corresponding feature vector is $\mathbf{s} = \mathbf{x}_s = \left[{}^{\mathrm{C_l}}\mathbf{x}, {}^{\mathrm{C_l}}\mathbf{x}\right]^{\mathrm{T}}$. After stacking the image feature points, the interaction matrix develops to

$$\dot{\mathbf{s}} = \begin{bmatrix} {}^{\mathrm{C_l}}\dot{\mathbf{x}} \\ {}^{\mathrm{C_r}}\dot{\mathbf{x}} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_l \\ \mathbf{L}_r \, {}^{\mathrm{C_l}}_{\mathrm{C_r}}\mathbf{V} \end{bmatrix} {}^{\mathrm{C_l}}\boldsymbol{\xi} \tag{4.19}$$

$$= \mathbf{L}_{\mathrm{E}} \, {}^{\mathrm{C_l}}\boldsymbol{\xi}, \tag{4.20}$$

where ${}^{\mathrm{C_l}}_{\mathrm{C_r}}\mathbf{V}$ is the spatial motion transform between the two camera frames. Since a single feature point does not allow observing feature rotations, only the left half of $\mathbf{L}_{\mathrm{E}}$ is considered. The remaining two angular velocities are governed by the trocar constraint, as described above. The resulting velocity screw is expressed in the reference frame of the camera. To relate the motion to the instrument, a velocity transform according to the transformation chain presented in Sec. 5.2.3 is applied.

Next, we derive a control scheme to automatically align the laparoscope with an instrument. This is e.g. used to follow the scalpel during tissue dissection. The alignment is performed so that the instrument is always located in the image center. We refrain from an automatic insertion movement of the camera into the patient for safety reasons. Recalling the trocar constraint, linear and angular velocity of the laparoscope are not independent. Therefore, we partition the interaction matrix in their respective velocity portions

camera control

$$\dot{\mathbf{s}} = \mathbf{L}_{\upsilon}\boldsymbol{\upsilon} + \mathbf{L}_{\omega}\boldsymbol{\omega}, \tag{4.21}$$

where $\mathbf{L}_{\upsilon}\boldsymbol{\upsilon}$ gives the velocity component of the translational part and the angular velocity component is given by $\mathbf{L}_{\omega}\boldsymbol{\omega}$. Since all velocities are related to the camera frame we forgo indexing. With the relationship (4.17) we obtain

$$\dot{\mathbf{s}} = \begin{bmatrix} \mathbf{L}_{\upsilon} \, \mathbf{L}_{\omega} \end{bmatrix} \begin{bmatrix} \boldsymbol{\upsilon} \\ \boldsymbol{\omega} \end{bmatrix} \tag{4.22}$$

$$= \begin{bmatrix} -\frac{1}{z} & 0 & \frac{x_x}{z} \\ 0 & \frac{1}{z} & \frac{x_y}{z} \end{bmatrix} \begin{bmatrix} \upsilon_x \\ \upsilon_y \\ 0 \end{bmatrix} + \begin{bmatrix} x_x x_y & -(1+x_x^2) & x_y \\ 1+x_y^2 & -x_x x_y & -x_x \end{bmatrix} \begin{bmatrix} -\frac{\upsilon_y}{d} \\ \frac{\upsilon_x}{d} \\ 0 \end{bmatrix} \tag{4.23}$$

$$= \underbrace{\begin{bmatrix} -\frac{1}{z} - \frac{1}{d}(1+x_x^2) & -\frac{1}{d}x_x x_y \\ -\frac{1}{d}x_x x_y & -\frac{1}{z} - \frac{1}{d}(1+x_y^2) \end{bmatrix}}_{\mathbf{L}_{\mathrm{C},\boldsymbol{\upsilon}}} \begin{bmatrix} \upsilon_x \\ \upsilon_y \end{bmatrix}. \tag{4.24}$$

The final camera interaction matrix is $\mathbf{L}_{\mathrm{C},\boldsymbol{\upsilon}}$. The necessary relationship to transform the velocities from the camera frame to the manipulator's wrist is found by hand-eye calibration.

Both control laws are subject to an adaptive gain λ, which limits the maximum velocity to

$$\lambda = \begin{cases} \frac{\lambda_0}{\|\mathbf{L}^+\mathbf{e}\|} & \text{for } \|\mathbf{L}^+\mathbf{e}\| < a, \\ \lambda_0 & \text{otherwise.} \end{cases} \tag{4.25}$$

The motion is decelerated with an exponential decay for an decreasing error. The threshold a switches the gain and λ_0 defines the maximum velocity. According to (4.13) the final velocity $\boldsymbol{\xi}$ is relate to the manipulator joints

$$\dot{\mathbf{q}} = \mathbf{J}^{-1}\boldsymbol{\xi}, \tag{4.26}$$

where $\mathbf{J}$ is the robot Jacobian. The different update rates of manipulator and camera were considered by applying a Kalman filter.

Experimental Verification

Several experiments were conducted to demonstrate the capability of our system to perform precise alignment tasks with the above-introduced control laws. The accuracy of position-based control was investigated by means of a planar checkerboard calibration plate. The plate was placed freely within the workspace. Cartesian coordinates of about thirty edge points of the checkerboard pattern were reconstructed by means of the stereoscopic laparoscope. The second manipulator was equipped with a calibration trihedron, which was driven to the reconstructed coordinates. Thus, the entire calibration chain between the two manipulators, starting at the laparoscopic camera and ending at the tip of the trihedron was included. Reference values were obtained by manually positioning the trihedron at the corresponding checkerboard corner. The measured error ranged between 3mm and 7mm. When a surgical instrument is used instead if the trihedron, this error further increases due to the flexibility of the shaft.

The image-based control laws were tested in a simulation environment before experiments were conducted on our telesurgery system. The simulation was performed with Matlab Simulink™, where the kinematics, the relationship between the two robot bases, as well as instrument and camera parameters were modeled. Fig. 4.7 illustrates results of an alignment task of a needle driver (red lines) as well the laparoscope (black lines). Simulation results are shown with solid lines, while results obtained with the telepresence system are shown with a dashed line style. Fig. 4.7(a) depicts the resulting trajectories in image space. The corresponding development of the error values is shown in Fig. 4.7(b). Fig. 4.7(c) shows the velocity screw during instrument alignment. The tracked tip of a straight needle was used as feature input during the experiments on the telesurgery system. The needle was green-colored and placed in the forceps of the needle driver. The needle tip was segmented via thresholding, while restricting the search range to the vicinity of the tracked distal end of the tool. The target position was chosen manually within the field of view of the camera. The measured alignment accuracy of the needle tip with the target was about 1mm. Shaft vibrations impede a minimization of the pixel error to exactly zero. Thus, the alignment was stopped for small error values. Compliance with the trocar point was checked using a Polhemus

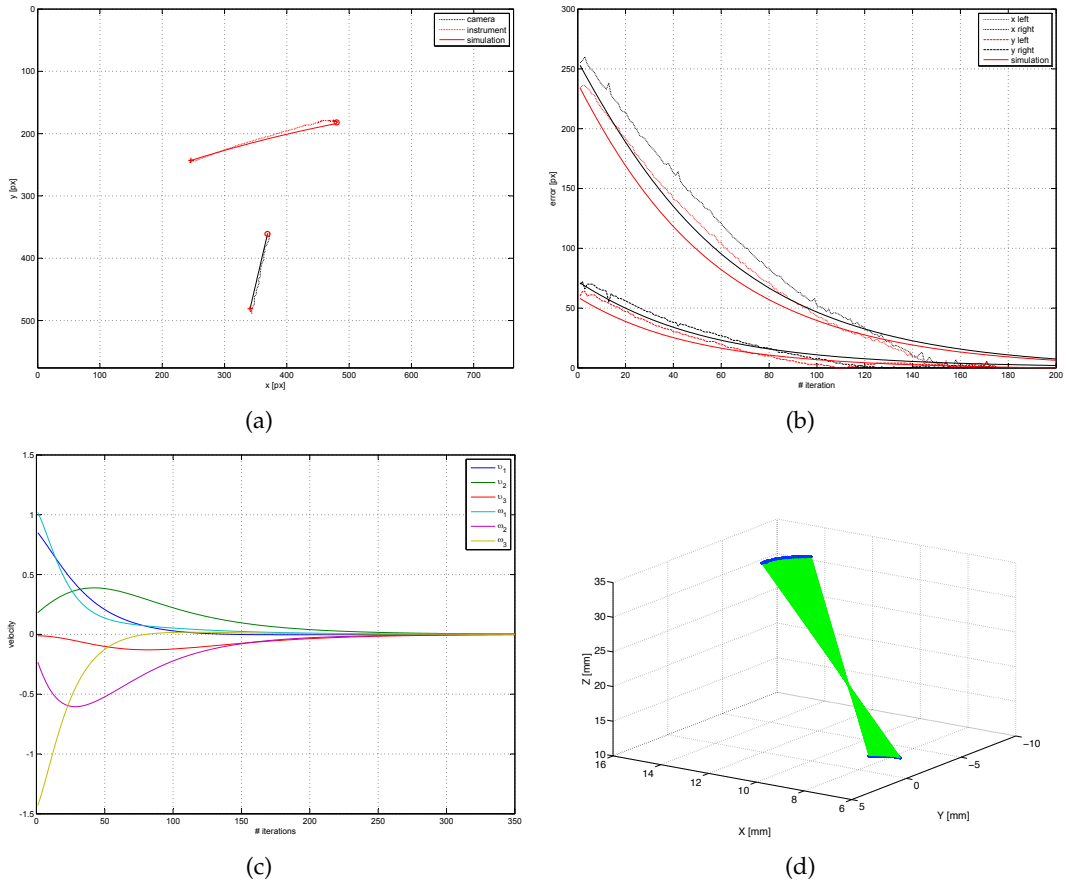


Figure 4.7: (a) Image trajectories for instrument (red) and camera (black) alignment task. Simulation results are illustrated with solid line style, while results obtained on our telesurgery system are shown with dashed line style; (b) Development of the corresponding error values; (c) velocity screw of the instrument; (d) verification of the remote center of motion.

Liberty™ magnetic tracker. The sensors coils of the tracking system were attached at the shaft of the instrument, beyond the disturbance area of the robot's motors. The chosen sensor arrangement allowed deducing the instrument movement. Fig. 4.7(d) shows the remote center of motion, while the blue points show the measured sensor positions.

4.4 Hybrid Instrument Control

Following our initial task description, we continue with the actual tissue dissection after the instrument is aligned with the incision point. Under the proposed method of online surgery the task execution is based on visual information obtained from the micro endoscopes at the instrument. By augmenting the instrument with the above-mentioned miniaturized camera, instead of relying on the conventional endoscope, the performed cut becomes independent from calibration uncertainties of the telemanipulation system, other than the micro camera itself [12]. The approach therewith enjoys

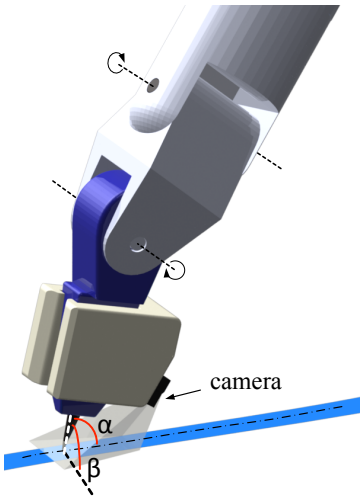


Figure 4.8: Augmentation of a surgical scalpel with the miniaturized camera. The pinpoint of the blade is visible in the camera image.

similar advantages as traditional image-based visual servoing techniques. Regarding our reference fine manipulation task of tissue dissection, the micro camera observes the cut path immediately preceding in order to deduce an optimal trajectory. The cut path is assumed to be visually identifiable (e.g., a vessel or a unique anatomical structure). A hybrid control scheme is applied, where the surgeon retains ultimate control over the scalpel position, while the blade orientation is set automatically. To capitalize on the accuracy of the robotic system, haptic constraints guide the surgeon, reducing mental stress and enhancing safety.

hybrid control

The miniaturized camera is mounted stationary with respect to the blade, capable of observing both the cut path and the blade tip from the relevant perspective, as illustrated in Fig. 4.8. The camera is aligned with the blade, hence the error between blade and cut path can be measured in pixel units. Since the camera moves with the blade, instrument movements are perceived as counter-movements to the observed surface. Due to the Cartesian control of the telemanipulation system, the reference frame of instrument and haptic device is axis aligned. Consequently, haptic virtual constraints can be derived directly from the measured error and expressed in the frame of the input device. In this way, we introduce similar precise virtual constraints to telesurgery that are known from cooperatively controlled handheld devices such as the JHU steady hand robot [31], where the fixture generation and the haptic device is not spatially separated.

4.4.1 Trajectory Generation

When holding a scalpel directly in hand, as in conventional surgery, a self-alignment torque of the blade facilitates guidance by damping unwanted angular motions due to the contact between tissue and the blade sides. In telesurgery, this contact force can usually not be measured and fed back to the operator, thus limiting blade alignment

to a visio-motor mapping of the operator. To mimic this behavior, we decompose the task of tissue dissection into two subtasks:

1. Smooth minimization of the deviation between scalpel and optimal cut path. That is, the error needs to be driven gradually toward zero *during the forward motion* of the blade.

2. Alignment of the blade orientation with the current cut direction to prevent tissue fissures.

During the cut, we keep the instrument wrist perpendicular to the surface (Fig. 4.8, angle β), while the steepness α is kept at a constant value.

We restrict ourselves to movements in the $xz-$plane, which is coplanar to the robot bases of our setup, and define the cut path as the plane parametric curve

$$\mathbf{c}(r) \equiv \begin{pmatrix} x(r) & y(r) \end{pmatrix}^{\mathrm{T}}, \quad r \in [0,1]\,. \tag{4.27}$$

The tip of the blade on the surface is denoted as $\mathbf{q} = [q_x, q_y]$, as illustrated in Fig. 4.9. We define $\mathbf{c}(\hat{r}(\mathbf{q})) =: \mathbf{p}$ as the curve point with minimal distance from the blade as

$$\left\| \mathbf{c}\big(\hat{r}(\mathbf{q})\big) - \mathbf{q} \right\| = \min_{r \in [0,1]} \|\mathbf{c}(r) - \mathbf{q}\|\,. \tag{4.28}$$

As long as the blade follows the optimal trajectory, the direction of the tool tip at that point is the normalized tangent direction

$$\theta_c(\mathbf{q}) = \frac{\mathbf{t}(\mathbf{q})}{\|\mathbf{t}(\mathbf{q})\|} \tag{4.29}$$

with

$$\mathbf{t}(\mathbf{q}) = \frac{d}{dt}\mathbf{c}(r)\bigg|_{r=\hat{r}(\mathbf{q})}\,. \tag{4.30}$$

Once the blade differs from the optimal trajectory, we define a Cartesian error vector $\mathbf{e}$ between the blade tip and the optimal path as

$$\mathbf{e}(\mathbf{q}) = \mathbf{q} - \mathbf{c}\big(\hat{r}(\mathbf{q})\big) = \mathbf{q} - \mathbf{p} \tag{4.31}$$

and an angular error θ with respect to the current blade orientation θ_m and the tangential angle θ_c as

$$\theta = \theta_m - \theta_c. \tag{4.32}$$

Minimizing the lateral deviation of the blade with respect to the cut path without taking the blade's current orientation into account yields to tissue fissures, since the blade is moved contrary to its orientation. Likewise, the blade's orientation should be adapted only during forward motion. The optimal cut is obtained when the error vector $\mathbf{e}$ is zero and the blade orientation equals θ_c.

The development of constraints on the blade motion can be compared to the non-holonomic constraints of differential-driven mobile robots: the motion is governed by the forward velocity and the angular velocity, while any lateral movements are to be avoided. We adapt a solution presented in [180, 121]. The derivation of the kinematic equations can be found in appendix A.2. The development of constraints on the blade motion is treated in terms of a Serret-Frenet formulation, where a virtual target frame F moves tangential along the curve $\mathbf{c}$. Since we follow the cut path in a forward di-

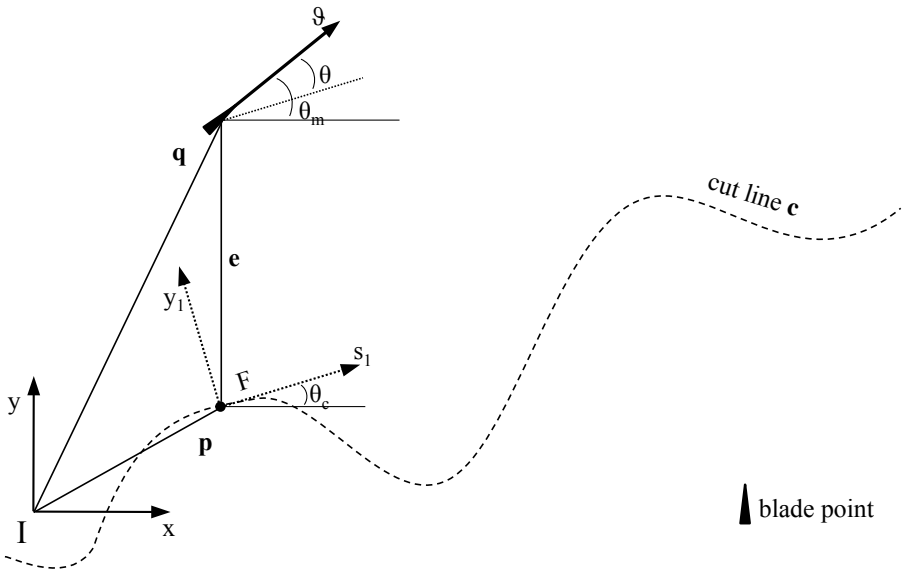


Figure 4.9: Guided cutting: The position of the blade $\mathbf{q}$ is expressed in frame I and a Serret-Frenet frame F rooted in the tangent space of the path. Optimal guidance is achieved by minimizing $\mathbf{e}$ and θ.

rection, the directed tangent equals the signed curvilinear abscissa s_1 of F. Given the instantaneous kinematic model of the blade expressed in frame F

$$
\begin{aligned}
{}^{\mathrm{F}}\dot{q}_{s_1} &= v\cos\theta - \dot{s}\big(1 - \kappa(s)\cdot {}^{\mathrm{F}}q_{y_1}\big), && (4.33)\\
{}^{\mathrm{F}}\dot{q}_{y_1} &= v\sin\theta - \kappa(s)\dot{s}\cdot {}^{\mathrm{F}}q_{s_1}, && (4.34)\\
\dot{\theta} &= \omega - \kappa(s)\dot{s}, && (4.35)
\end{aligned}
$$

where $\kappa(\cdot)$ is the curvature of the path in $\mathbf{p}$, $\dot{s}$ is the tangential velocity in $\mathbf{p}$, v the linear velocity and ω the angular velocity, the blade optimally follows the cut path for

$$
{}^{\mathrm{F}}\dot{q}_{s_1} = {}^{\mathrm{F}}\dot{q}_{y_1} = \dot{\theta} = 0. \tag{4.36}
$$

Otherwise, the blade can be directed towards the optimal cut path by driving the three parameters of the instantaneous kinematic model towards zero. For this purpose, we investigate a control scheme that constrains the surgeon's input motion using haptic virtual fixtures.

Feedback Generation

A correct scalpel angle is most important for a smooth cut and should always be orientated with the cut direction. Due to the missing self-alignment torque of the blade in telesurgery, setting the orientation manually is very difficult and sensitive to user input. In order to facilitate the work of surgeons, we apply hybrid control, where the blade orientation is always set by the system, while the user is assisted in setting the position by a haptic virtual fixture. We started with experiments were the user had full control over the scalpel position. However, the small spatial expansion of the structure to be cut, which we assume with about 1-2mm in size, required a large down-scaling of the input motion to avoid lateral movements with respect to the blade. Therefore, we decided to use only the forward motion of the haptic input device as input command.

virtual fixture

The actual position of the blade is then maintained by the control law. A haptic virtual fixture provides synchronization between the master and the slave. The fixture is implemented to take on a passive role, scaling the user's input force to drive the operator back to the desired path.

To couple the instantaneous kinematic model to the motion of the haptic input device, we recall that the tool tip mounted camera allows evading the error-prone system uncertainties. Since the reference frames of camera and instrument are aligned, the deviation between tool-tip and desired trajectory, measured in image space, can directly be coupled to the input device. During haptic guidance, we are interested in automatically optimizing the angular velocity ω of the blade. Rearranging (4.35) yields to

$$\omega = \dot{\theta}_m + \kappa(s)\dot{s}. \tag{4.37}$$

In order to achieve a smooth convergence toward zero, we follow [180, 121] and choose the error functions

$$\dot{\theta} = \dot{\delta} - \gamma \cdot {}^{\mathrm{F}}q_{y_1} \cdot \upsilon \frac{\sin\theta - \sin\delta}{\theta - \delta} - k_2(\theta - \delta), \tag{4.38}$$

$$\dot{s} = \upsilon\cos\theta + k_1 \cdot {}^{\mathrm{F}}q_{s_1}. \tag{4.39}$$

Function $\delta = k_3 y_1^2$ shapes the transition between desired and current trajectory during the path approach to zero. Variables $k_1, k_2, k_3 > 0$ are scaling factors that influence the error minimization rate. The forward velocity υ is manually controlled by the operator and set according to the motion of the input device.

4.4.2 Feedback Generation

Recalling the last section, we need to update the current blade position $\mathbf{q}$ as well as the angle θ in every time step and receive the corresponding cut direction update ω along with a forward velocity υ. The input can directly be derived from the motion of the haptic device. To cancel involuntary jerky input movements to the haptic devices, the user's input is tremor filtered [53].

The generation of the feedback signals can be divided into a general part, limiting the dynamic behavior of the haptic input device with an artificial non-linear damping and a task specific part, which generates the actual guiding force. The resistance is realized as damping of the commanded stylus movement, therewith preventing high input velocities and facilitating fine-scaled movements during tissue dissection. We implemented an exponential envelope

$$s_d = \left(1 - \exp\left(-al^2\right)\right) \cdot f_{max} \tag{4.40}$$

that restricts the maximum speed of the stylus, where a is related to the stiffness of the system, f_{max} is the maximum force value, and l the distance between the current and the previous stylus position $l = \|\mathbf{q}_{t-1} - \mathbf{q}\|$. The applied force vector is then

$$\mathbf{f}_d = \frac{\mathbf{q}_{t-1} - \mathbf{q}}{\|\mathbf{q}_{t-1} - \mathbf{q}\|} \cdot s_d. \tag{4.41}$$

The application of an exponential function ensures a smooth force progression, as sudden force changes usually lead to unpredictable vibrations at the end effector, while being still capable of simulating high stiffness.

The haptic virtual fixture is realized on a proxy-based implementation, where the proxy represents the calculated optimal position and the master servos to the proxy. The master is controlled by

$$\mathbf{f}_g = k_p(\mathbf{p} - \mathbf{q}) + k_d(\dot{\mathbf{p}} - \dot{\mathbf{q}}), \tag{4.42}$$

where k_p and k_d are the proportional and derivative gains respectively. Haptic rendering has demanding computational requirements, i.e. the haptic device used has an update rate of 1000Hz, which is considered to be the lower limit to provide realistic rendering of rigid contacts. The mismatch between camera frame rate and haptic loop cycle is compensated by means of a Kalman filter. While the position of the blade is predicted in between the camera frames, the prediction is corrected as soon as vision-based fixtures can be derived. The final force is composed of

$$\mathbf{f} = \mathbf{f}_d + \mathbf{f}_g \tag{4.43}$$

and applied to the haptic device.

4.4.3 Implementation

An evaluation to prove the feasibility and power of the proposed approach was conducted within the simulation environment of our telesurgery system. In order to obtain real-world conditions, the simulated micro camera images were artificially delayed to 25 frames per second. The delay introduces a mismatch between haptic loop and image processing which, if not compensated, can result in system oscillations. In order to calculate the deviation between cut path and blade, the skeleton of the observed curve is calculated. Currently, we assume to observe a single curve on a uniform background. After edge detection, the skeleton is fitted by a penalized regression spline [52]. With

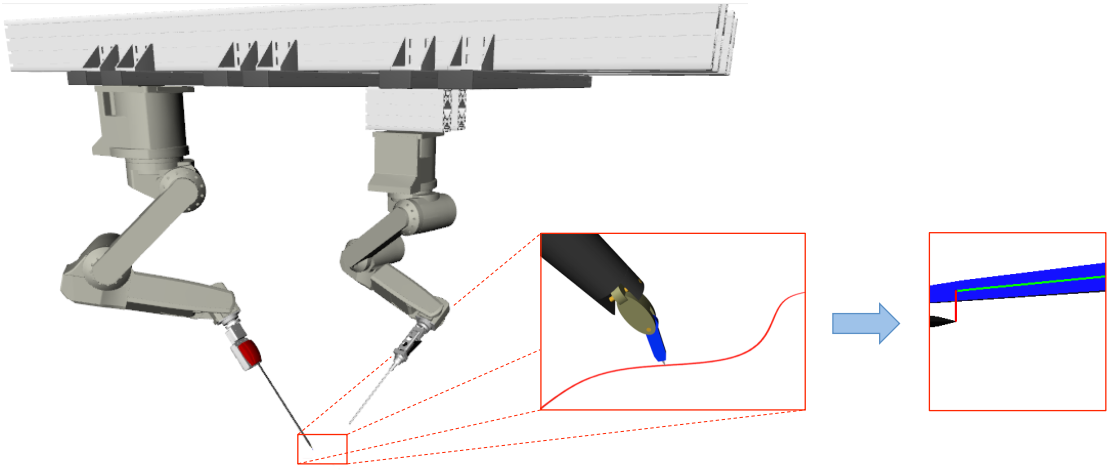


Figure 4.10: Simulation environment used during the experiment. From left to right: manipulators with scalp and endoscopic camera; blade orientated on the cut path; cut path from the perspective of the tool-tip mounted endoscopic micro camera. Visible is the pinpoint of the blade, the cut path, and the skeleton (green) of the path used to measure the deviation from the scalpel (red).

the continuous curve representation we can now compute the curve's derivations as well as the distance to the position of the blade's pinpoint in image space. The cut task was performed in 2D space, hence the user could not alter the height between tool and surface or change penetration depth. The restriction was implemented as virtual fixture on both master and slave. A haptic virtual fixture with high stiffness on the master-side prevented the user from penetrating the plane, while the master neglects commands perpendicular to the cut plane. The specified cut path was a sinusoidal curve (cf. Fig. 4.11). We have compared our method with bimanual control.

Manual control. The user has to follow the trajectory without any guidance. The deviation from the ideal path needs to be minimized while taking care of the blade orientation. The orientation can be adjusted by rotating the stylus of the haptic input device. The user input was tremor-filtered and the instrument's position on the cut plane was automatically maintained with the above-mentioned virtual fixture on master and slave to ensure equal conditions with the second experiment.

Guided dissection. The user is guided along the trajectory with the proposed method. The error between current blade position and optimal path is calculated automatically and the new blade orientation is set accordingly. Haptic feedback is used to synchronize the user's position at the haptic device with the new instrument position.

During the experiments, the control computer recorded the movement of the instrument, the deviation between blade tip and the path in pixel units, and the orientation of the blade. The last parameter can be considered as most meaningful, since it

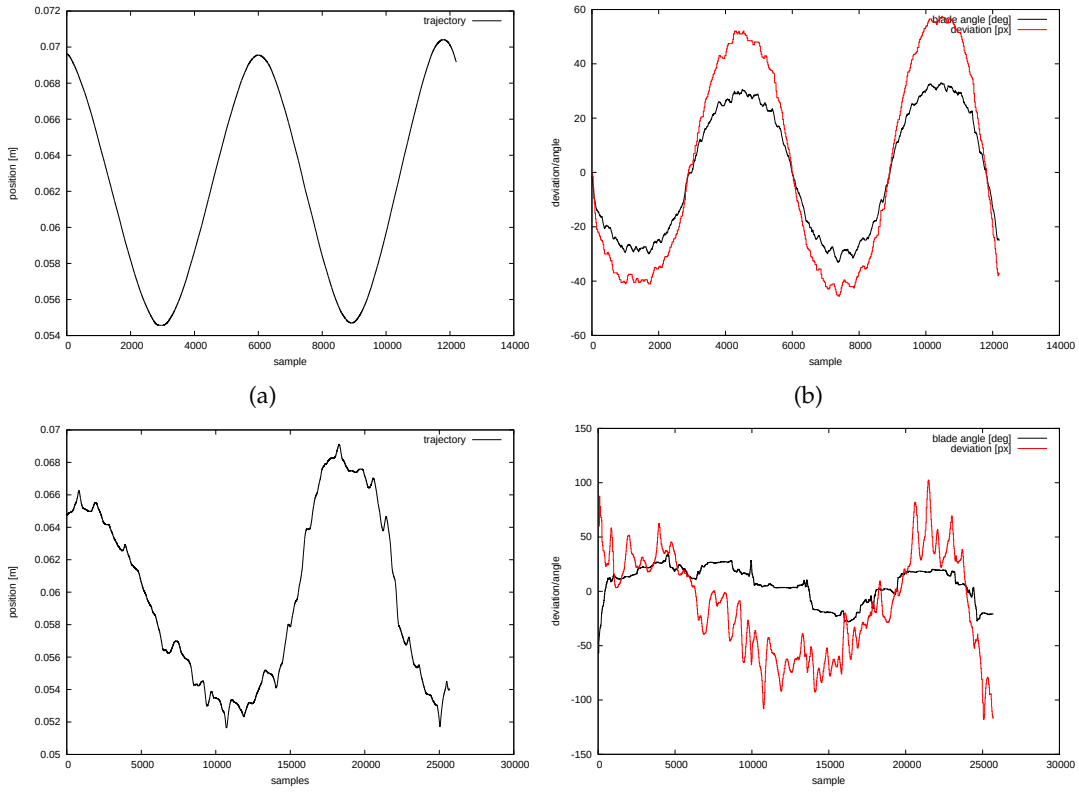


Figure 4.11: Instrument trajectory (left), deviation between blade and cut path, as well as blade orientation (right). The top row shows results for guided tissue dissection, the bottom row for manual control.

characterizes the smoothness of the cut and indicates tissue fissures. The plots of the trajectories (Fig. 4.11(a) and Fig. 4.11(c)) and the positioning errors (Fig. 4.11(b) and Fig. 4.11(d)) show a strong effect of assistance, whereas the amount of jitter is distinctive during manual control. Most important, it is difficult and exhausting to follow a path while simultaneously coping with manual blade alignment. The angle is constantly adjusted by the user, but the missing self-alignment torque makes it difficult to maintain the correct value. As mentioned above, a smooth movement is considered as a key factor for successful cutting. This fact becomes obvious, when comparing the blade angle with the actual movement direction of the instrument. In the ideal case, the blade angle follows the trajectory, as it happens during guided tissue dissection. Here, the lateral error would further decrease for a straight line, but the constant adaption to the sinusoidal curve prevents this. Note again, that the precise alignment becomes only possible with the camera-augmented instrument. As the trajectory plot shows, the curves amplitude is only about 7mm, whereas the width of the curve observed was 1.5mm. Regarding feedback generation, a more advanced haptic device, which provides torque force feedback, would support the angular orientation during manual control by actuating the stylus' self rotation. The calculated angular velocity therewith becomes a virtual self-alignment torque that can be fed back to the operator.

5 Reference Implementation

Although we have already introduced the concept of minimally invasive surgery at the beginning of this treatise, we have not yet considered the technical aspects. After discussing existing telesurgery systems, we proceed with the description of our own setup, which served as a testbed for the conducted experiments. Finally, we have tested our telemanipulator in an animal experiment.

5.1 Telesurgery

At this point it is important to highlight some of the successful laparoscopic robotic systems. Historically, so called robot assistants are the precursors of today's complex telesurgery setups. The use of these devices is rather task-specific. As the name implies, the manipulators mainly assist surgeons in holding and positioning laparoscopic instruments, such as the famous AESOP active camera holder [170], which was the first FDA approved system, LARS [190], Lapman [153], or LER (Light Endoscope Robot) [30]. The latter does not require an extended support frame, but fixes the camera directly on the abdominal wall of the patient. Based on these systems began development of telesurgery. Commercially speaking, there were two competitors on the market for surgical tele-manipulators. The Zeus™ system [158], developed by Computer Motion Inc. (CMI), and the daVinci™ robot [68], marketed by Intuitive Surgical Inc. The Zeus™ consists of modified 5 DoF arms of the aforementioned AESOP, which is also part of the system. When CMI was acquired by Intuitive the CMI products were discontinued and the daVinci™ started a remarkable success story. Today, the system can be seen as state of the art. After the initial use in cardiac surgery, it is now entering the application domains of urology and gynecology.

Beyond, the scientific community is contributing to this area with own systems. The MiroSurge™, developed at the German Aerospace Center (DLR), is often cited as one of the most sophisticated research systems. MIRO lightweight robots [71] carry the proprietary MICA instruments that come with a 7 DoF force/torque sensor, providing dexterous manipulation and haptic feedback from the operation site [193]. The impedance

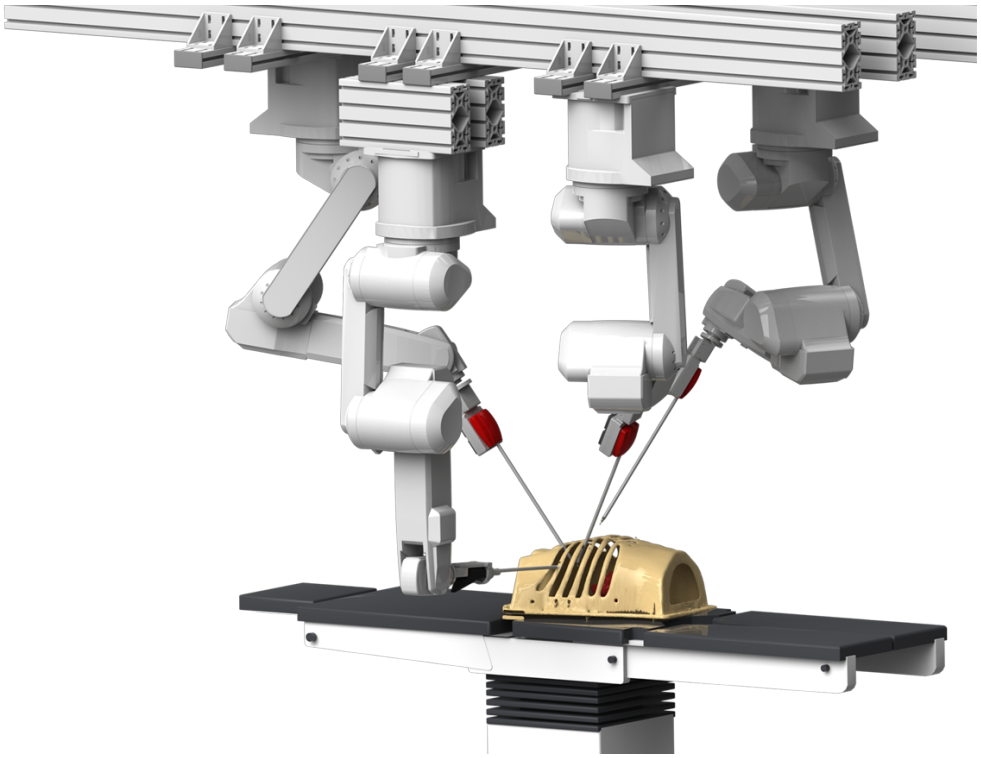


Figure 5.1: The ARAMIS research system for minimally invasive surgery: the slave-side of our telesurgery robot comprises four industrial manipulators that carry either surgical instruments or a stereoscopic laparoscope.

control mode of the robots permits gravity compensation, facilitating direct and safe interaction with humans according to the soft robotics paradigm. Keeping in mind the high cost factor in clinical practice, the design of the system was targeted to a versatile use. E.g., the replacement of pedicle screws was investigated [141]. The ARTEMIS project [175] of the University of Karlsruhe, Germany, started in 1991 and discontinued in the meanwhile, was one of the first research systems for minimally invasive telesurgery. Apart from a lack of force feedback, many aspects of today's systems, such as instruments with flexible tails, different input modalities and the integration of scene knowledge, have already been taken into account. Evaluations were carried out on animals. Madhani et al. introduced the Falcon manipulator [114]. The kinematics of the Raven telesurgical system [112], University of Washington Bionics Lab, was deduced from preliminary experiments, conducted with the "Red Dragon" [169]. This spherical mechanism allows measuring and recording the movements of surgical tools during an intervention. The devices were also used for training, to model the process of surgical interventions, and for surgical performance assessment [166]. The promise of short medical response time has attracted the military early, with the objective to perform remote surgery on the battlefield with physicians located safely distant [167]. Teleoperational capabilities were tested in transatlantic long-distance scenario, through communication via airborne wireless links of an unmanned aircraft [76] and during a NASA underwater operation. The Department of Engineering Syn-

thesis, University of Tokyo, tested their system in several realistic scenarios. After first experiments over a distance of 700km, a second experiment was conducted between Japan and Bankok [26]. The newly designed successor of their first system comprises three SCARA-type slave manipulators, holding forceps with force measuring capabilities, including grasping forces [123]. The "Second Generation Robotic Telesurgery Workstation" is a joint project between UC Berkeley and UC San Francisco, providing two pairs of modified "Millirobots" at the slave [42]. The microsurgical telerobot system [107], developed until 1999 at KAIST, Korea, made use of customized Steward-type parallel robots and was used for training purposes. A conventional laparoscopic tool can be included as handle at the force-reflecting master controller. The slave unit of the system comprises an industrial robot, with an additional wrist-mounted parallel kinematic. Beyond MIRS, an overview of other successful medical applications of robots is given in [74]. The general trend of miniaturization does not stop at medical technology and thus came the desire to advance the existing methods for even less invasive procedures. Currently, this progress is reflected by two techniques, both at an experimental stage, known as "single port access surgery" and "natural orifice transluminal endoscopic surgery" (NOTES) [157, 196]. Single port surgery is performed exclusively through a single access site to the patient, further reducing trauma and scarring. NOTES aims to realize scarless operations, which are performed though a natural orifice, such as mouth, anus or urethra. The access to the actual surgical site is reached by an inner body incision. These techniques are exacerbating the existing problems of MIRS and pose new technological challenges, in particular to the instrument design and control, which has to be flexible in order to accommodate the requirements of the complex access paths. As a logical continuation of our robotic platform, we participate in this development [4, 5].

5.2 The ARAMIS Research Platform

The testbed used and evolved throughout this treatise is based on the ARAMIS telesurgery system (Autonomous Robot Assisted Minimally Invasive Surgery System), originally presented to the research community in [118, 18]. Based on the experiences gained in several experiments, it has been constantly improved with respect to both software and hardware.

5.2.1 Telemanipulation System

The current design is a bimanual system with four 6 DoF industrial Mitsubishi MELFA 6SL™ robots. Each slave-manipulator weighs about 60kg and is ceiling-mounted on aluminum profiles to ensure good access to the operating table, as illustrated in Fig. 5.1. Each manipulator can either carry a surgical instrument or an endoscopic camera. The employed EndoWrist™ instruments are marketed as original equipment parts with the daVinci™ robot. The individual joints, dependent on the instrument type either three or four, are driven by bowden wires that connect to servo motors. The instruments have been augmented with strain-gauge force sensors. By means of haptic devices, depicted in Fig. 5.3, the operator can experience forces that occur at the instrument

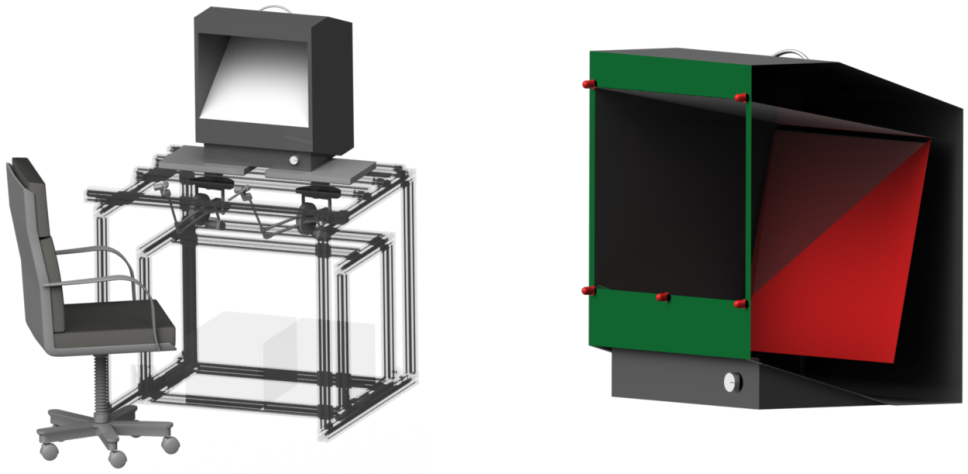


Figure 5.2: Master console with 3D screen and haptic input devices. Five infrared emitters (red) are attached to the screen for the eye-tracking system. The used eye tracker requires the transformation between the frontal plane (green) and the actual image plane (red).

shaft. A endoscopic stereo camera provides two separated optics, connected to two CCD sensors. Due to the design of the camera, only a circular section of the PAL resolution can be used. All tools are coupled via a magnetic mechanism to the robot flange to allow quick interchangeability. A hot-plug coupling made of spring contacts bridges the electrical signals as the instrument connects to the carrier system, illustrated in Fig. 5.4(a). In addition to the servo- and the strain-gauges signals, four pins are reserved for video transmission of our endoscopic micro camera. The adapter also comprises an automatic instrument identification. It reports a unique signature, which is realized by means of a resistor, to the system when voltage is applied. The kinematic structure, i.e. required during the proposed instrument localization approach and the associated calibration (cf. Sec. 5.2.3) is then retrieved from a data base, and applied to the system's configuration.

instrument identification

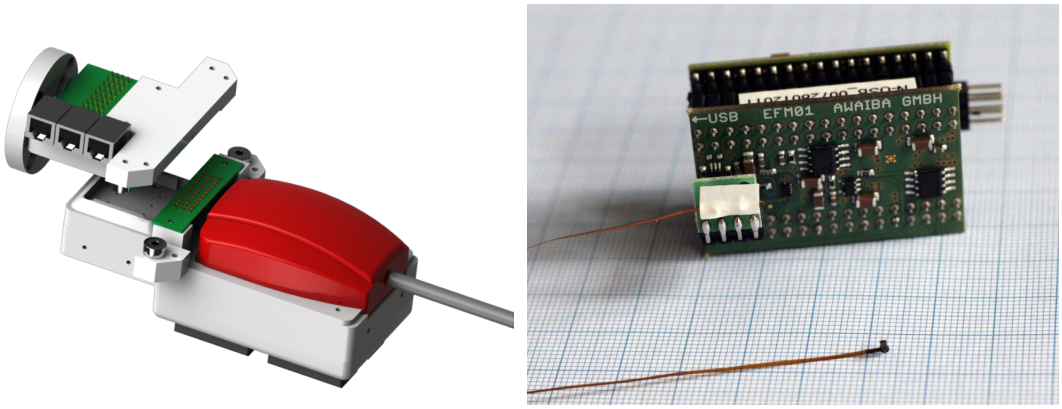
With the master console, depicted in Fig. 5.2, the goal is to restore manipulation and sensation capabilities of the surgeon. Immersion into the remote workspace is supported by visual and haptic feedback [2, 3]. Visual feedback is given by a monitor with 3D capabilities, taking images from the endoscopic stereo camera. The display is assembled of two orthogonally polarized screens, merging the left and the right camera image using a semi-transparent mirror. Stereopsis is created for the user by wearing polarized glasses, which assign the corresponding image to each eye. IR emitters are attached to the display housing to support gaze tracking using the approach presented in appendix A.3, specifically the bottom row consists of three markers and the top row of two markers. The surgical instruments are under direct control of the surgeon through teleoperation. As the operator acts with the two "Sensable Phantom Premium 1.5A™" haptic input devices, each providing six degrees of freedom, the slave manipulators follow that motion. The devices are configured overhead to ease handling and avoid singularities, i.e. gimbal lock positions. The three translational DoF of the

devices can be actuated actively to feed back the forces derived at the instrument tip. During previous evaluations [118], a shortcoming most subjects complained about was the digital closing of the gripper, where no intermediate steps were possible. With regard to the animal experiments planned at the German Heart Center Munich (cf. Sec. 5.3), we upgraded the devices with a custom handle design, which introduces a sev-

Figure 5.3: Continuos input handle (left) and mounted handle on Phantom™ device (right).

enth DoF. While interchangeable handles, such as snap-on thumb-pads or scissors, are commercially available for the more recent versions of the device family, the end effectors offered are not compatible with our employed device version. Moreover, these merely passive encoders come at a high price tag, but without force-feedback capabilities. Therefore, we designed a new handle that is similar in the handling and feel of a pair of tweezers. A secure grip is provided by two size-adjustable straps, kept by thumb and index finger (cf. Fig. 5.3). The movement of the forefinger actuates a rocker, which is connected to a small DC motor with integrated position encoder, specifically a Faulhaber™ 1724 with a resolution of 512 pulses per revolution. The corresponding

7 DoF Phantom™ extension



(a) (b)

Figure 5.4: (a) Magnetic hot-plug coupling with spring contacts that bridge electrical signals and instrument identification system; (b) Single NanEye™ sensor with base station.

motion controller offers a CAN-bus interface to send commands to the motor and to

read its state. We incorporated the controller into our existing CAN-network. Position signals are processed to obtain the actual opening angle of the rocker that is integrated into the kinematic calculation of the simulation software. Counterweights have been attached to the Phantoms™ to keep them in balance. The handle now also allows feeding back gripping and clamping forces, as suggested in [104]. Beyond, the medical workstation also offers four switches, which are places on the foot well of the console. They can be assigned their individual functionality.

5.2.2 Endoscopic Micro Camera

Our surgical instruments can be equipped with an innovative endoscopic micro stereo camera. The prototypic sensors were provided by our industrial partner, Awaiba GmbH, Nürnberg, Germany. According to the manufacturer, the NanEye™ is currently the world-smallest CMOS camera available. We provide a brief summary of the hardware, which is based on the original hardware specification [208]. The sensor features a total chip area of 0.34mm^2. The surface houses a pixel matrix of 62.500 pixels, yielding to a resolution of 250×250 pixels. The sensor is available as monochrome version or a RGB Bayer pattern reconstructs the color information. The integrated AD conversion and data transition are controlled by an on-chip ring oscillator and a readout machine, allowing the sensor to operate fully autonomous at a frame rate of approximately 40fps. A small lens with an opening angle of 90° is assembled together with a cover glass on the chip, resulting in a total package size of $1\times1\times1.5$mm. Two different apertures are available with $f/4.0$ and $f/2.7$. According to the specifications[1], the former has its best focus at 5mm, while the depth of focus is $3.5-30$mm. Images from the prototypic sensor available to us, are already becoming blurry at a distance of approximately 10mm. The specification of the latter lens is given with a best focus at 15mm and a depth of focus ranging between $8\text{mm}-\infty$. A four wire flat ribbon cable realizes the communication with an USB control circuit and power supply, shown in Fig. 5.4(b).

The driver software of the camera was written in C# at the time of the conducted experiments. An wrapper interface was created to integrate the managed C# code into our existing C++ software framework. More details are given in appendix A.4. Further, the driver software was extended to manage two cameras for the stereo setup.

5.2.3 System Calibration

We already mentioned the relatively poor absolute accuracy of the system, which primarily affects Cartesian control. We therefore introduced image-based control to improve results for delicate tasks. Although this closed-loop control method is less dependent on the quality of the system calibration, we need to determine the transformations between the various components involved, i.e. the individual robot bases and the mounted surgical tools. To replicate the global coordinate system of our simulation environment at the actual setup, the world reference frame is centered around a certain robot base, e.g. R_1. We assume perfect congruence of the underlying model

[1] www.awaiba.com

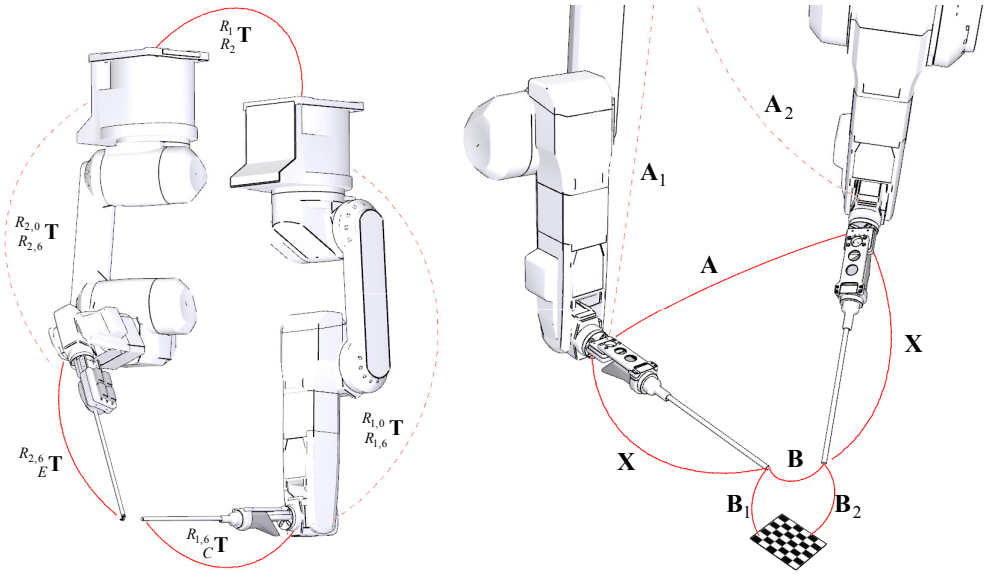


Figure 5.5: Calibration of individual system components: (a) overall transformation chain between surgical instrument and laparoscope; (b) hand-eye calibration of laparoscope.

and perform the calibration of all other manipulators $\mathrm{R}_2, \cdots, \mathrm{R}_4$ with respect to this robot base, as exemplified in Fig. 5.5(a). A common reference frame in global coordinates is established by positioning the tip of a hand-mounted trihedron on a number of calibration points. Now, the relationship between all manipulators can be described, since the location and orientation of the calibration frame is known in each of the local robot frames. The eccentricity of the EndoWrist's™ carbon fiber shaft is compensated following the method proposed in [118]. For the laparoscope we obtain the extrinsic parameter ${}^{\mathrm{R}_{1,6}}_{\mathrm{C}}\mathbf{T}$, with C being the camera frame by means of a hand-eye calibration. Therefore, a homogeneous matrix equation of the form $\mathbf{AX} = \mathbf{XB}$ is solved [179], i.e. $\mathbf{X} = {}^{\mathrm{R}_{1,6}}_{\mathrm{C}}\mathbf{T}$ the transform from the tool center position to the camera frame, $\mathbf{A} = \mathbf{A}_1^{-1}\mathbf{A}_2 = {}^{\mathrm{R}_{1,0}}_{\mathrm{R}_{1,6}}\mathbf{T}^{-1}\,{}^{\mathrm{R}_{1,0}}_{\mathrm{R}_{1,6}}\tilde{\mathbf{T}}$, and $\mathbf{B} = \mathbf{B}_1\mathbf{B}_2^{-1}$. Now, camera coordinates can be expressed in the base frame of the robot, i.e. to map sensor related measurements into the robot's workspace.

5.2.4 Software Architecture

The original software of the ARAMIS system has undergone a thoroughly refactoring [13]. The main focus was to replace the previous hierarchical structure [118], which was based on the model-view-controller paradigm, in favor of a more flexible and distributed architecture. In particular, common access to internal data, such as robot joint angles and calibration settings, should be given for external software components. Instead of a centralized framework, we now organize the software in small *modules*, each responsible for a certain task. A module is a software component of the system that can be deployed and run as a separate process on any machine, while sharing data

as well as functionality with other modules over the network. We utilize the open source cross-platform *cisst* libraries, developed by the Johns Hopkins University, Baltimore, USA. In appearance, modules are like individual applications, where they are coordinating by means of inter-process communication, as the interface connections in Fig. 5.6 suggest. Modules are realized using the *cisstMultiTask* library and therewith adopt a task-based framework. Specifically, each module is composed of one or more `mtsTasks`, each of which running on its own thread. Each task has a series of *provided interfaces* and *required interfaces*. The task exports its functions to its provided interfaces as *commands* to make them available to other tasks, and adds to its required interface functions that are needed from other tasks. Event handling can be implemented in a similar fashion to invoke remote functions based on occurring events. During its initialization, a module establishes run-time bindings between provided interfaces and required interfaces, independent of the involved modules are running locally or on different machines. For inter-process calls the *Internet Communication Engine* (ICE) serves as middleware, thereby hiding the complexity of network communication from tasks. The only difference is that, when setting up the bindings, process names must be specified in the latter case, together with the hostname or IP address of the computer where the `GlobalComponentManager` is running. This instance establishes connections between the modules, which are running on different computers without knowing each other's locations.

cisst libraries

The system's modules can be classified into three categories: the `Blackboard` module, the `MainModule`, and optional modules. The first two categories are mandatory, the latter are pluggable, meaning that they can be temporarily terminated when their functionality is not required by the current usage scenario of the system, and can be restarted at a later time. The `Blackboard` is at the center of the system architecture. It merely serves as a common place for data storage and exchange, where all the other modules can read from and write to, thus resembling a blackboard. Each of its tasks keeps some data in its internal storage, and offers getter and setter functions for those data via its provided interfaces. The provided interfaces can be connected to required interfaces of another module, thereby enabling that module to access the data. For example, it is usually desirable to create a task for each robot in order to share its joint angles. Each of the robot controller tasks in the `MainModule` can then call the setter function to keep the joint values up-to-date, whereas the getter function can be used by e.g., the visualization module to update the display of robot postures, or the instrument tracking module.

The `MainModule` implements several tasks. It provides a graphical user interface to alter the robot joints and implements key functionality of the system, includes kinematic control and communication with the robots as well as advanced application of the robotic system in various scenarios. The latter is realized by means of "surgical programs", each of which represents a certain system configuration or experiment. All these classes derive from an abstract `SurgicalProgram` class. The abstract class specifies common functions like `start()` and `stop()`. Every subclass needs to implement a state machine that dictates the workflow of the surgical program and holds calibration data, such as port locations, for the current scenario. The transition between the

distributed architecture

states is typically invoked by an event, e.g. the robots reaching their target positions, autonomous actions, a foot pedal being pressed, or other user triggered input commands.

Optional modules include device modules and application modules. A device module enables the integration of a new hardware device (e.g. an eye-tracker) into the system. Typically it wraps up the device driver and writes some device-specific data (e.g. the tracked position) to the `Blackboard`. Application modules are software components that supplement the functionality present in the `MainModule` and are usually employed in a specific `SurgicalProgram`. Many of them consume device specific data for further processing. Without enumerating all system components individually at this point, it is worth to mention the 3D visualization module, rendering of the endoscopic video stream with additional overlay information, the integration of sensor data, or the instrument tracking as representative examples.

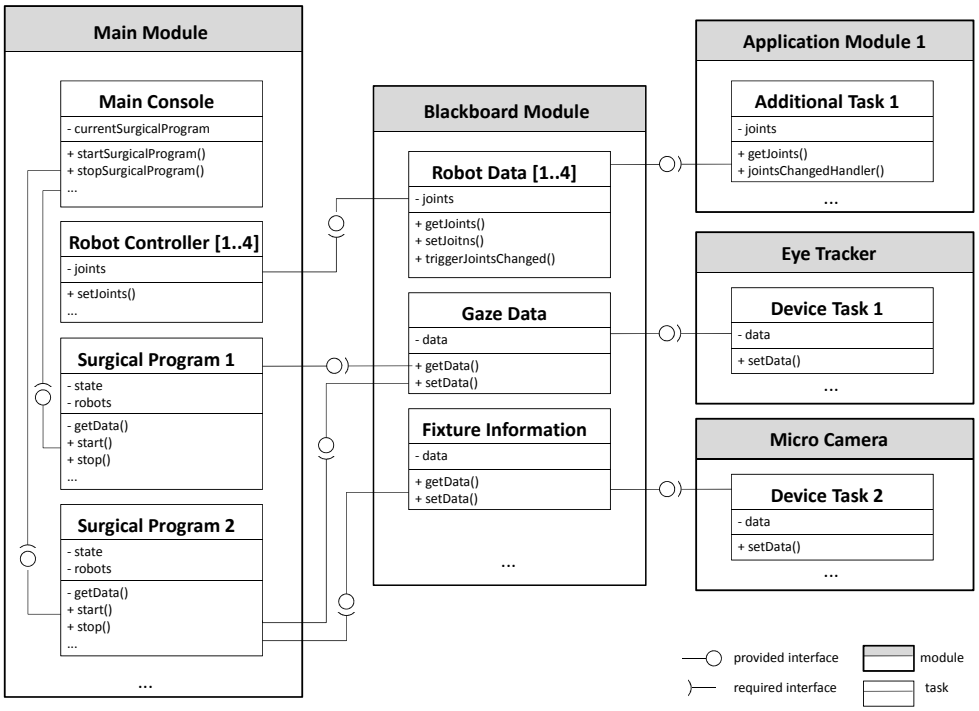
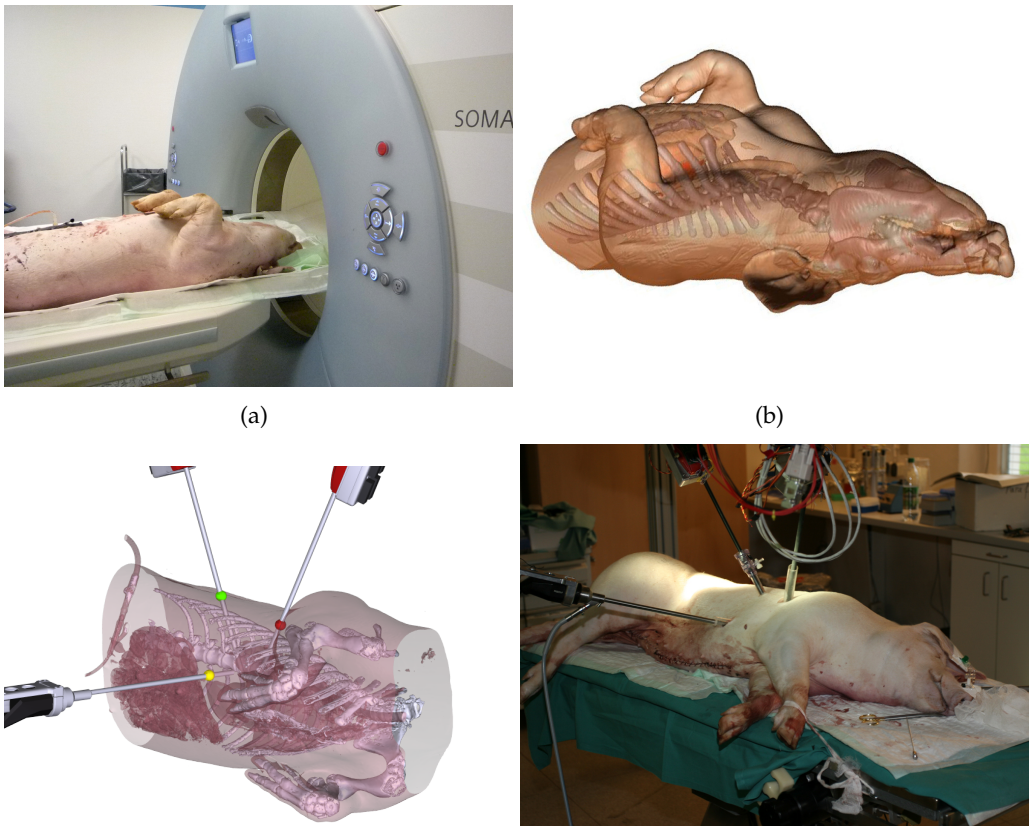


Figure 5.6: Illustration of the components and interfaces of the distributed architecture.

Beyond the previously mentioned libraries, which form the basis for distributed concurrent data processing, the *Robotics Library* [164] completes the core functionality of the system in terms of scene representation, based on the *OpenInventor* visualization software, and kinematic modeling. It is also noteworthy that currently a cisst package is under development that allows for a seamless integration into the popular *ROS* (Robot Operating System).



(a) (b)

(c) (d)

Figure 5.7: Preparation of the animal experiment: (a) pig during the CT scan; (b) segmentation of tissue types; (c) port planning in the simulation environment; (d) actual animal experiment

5.3 Animal Experiments

animal experiment

Although some studies on bovine and pig hearts were conducted in the past to investigate the effect of force feedback [119], no experiment regarding the actual application domain – minimally invasive surgery – has been performed with our system so far. Consequently, the next step was to stress the system in an animal experiment with closed torso. A pig carcass was selected, because of the similarity of the anatomical structures to the human body. To guarantee proper handling and to provide the essential medical equipment, the experiment was carried out at the German Heart Center Munich.

Materials and Methods

Before starting the experiment itself, the port location for inserting the surgical tools and the laparoscope needed to be specified. In conventional minimally invasive interventions, ports are selected according to anatomical landmarks. Acromastium and sternum, for instance, allow surgeons to assess the internal surgical sites, based on the patient's torso size. In robotic surgery, however, port selection also affects accessibility

and dexterity of the robots.

To minimize the risk of manipulator collisions, port locations were deduced from a preoperative CT scan (see Fig. 5.7). The scan provides data in Hounsfield units, represented as a gray image. It is derived from the radiation attenuation in tissue. Based on this linear scale, different tissue types can be differentiate and the bone structure was reconstructed geometrically for further processing in our simulation environment. Here, the reconstructed model allows to assess suitable port locations and an appropriate insertion depths of the instruments. Once all ports were selected, the resulting workspace of the manipulators was verified experimentally.

With the conductance of an animal experiment two major goals were pursued. First, the capability of the system in a realistic, minimally invasive scenario should be tested. Specifically, the handling of the surgical tools under trocar kinematics within a limited workspace and the handling of the newly designed continuously controllable input device were assessed. The second objective was to investigate the influence of the trocars on the force measurement and the haptic feedback. During previous experiments, the forces obtained were unbiased, because no interferences acted at the instrument shaft.

Results

The experiments conducted lasted about 4 hours. A significant amount of time had to be devoted for transferring the planned ports to the pig. Although the port positions could be approximated by means of anatomy, i.e. the rib structure of the thorax, the relative position of the animal to the system was unknown. Accordingly, the pig had

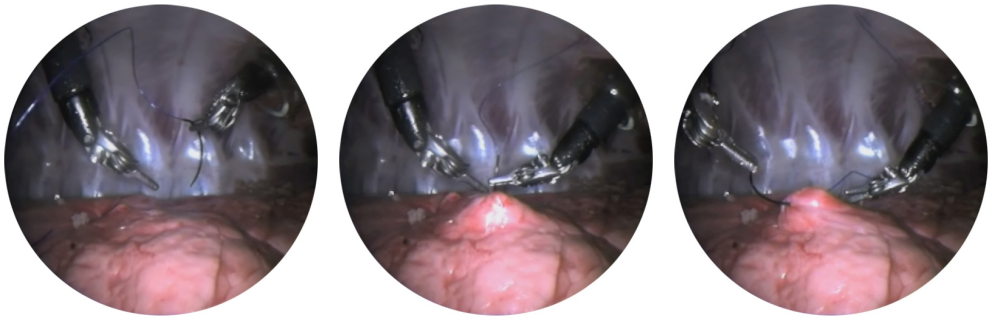


Figure 5.8: Suturing during the animal experiment from the laparoscopic camera perspective.

to be relocated several times. In particular, the size of the superstructural system parts restricted the available workspace. A laser registration system, as presented in [98], would reduce the setup time significantly. After successful positioning, various surgical tasks were accomplished. Tissue was penetrated and sutured repeatedly, as shown in Fig. 5.8, and stitches were secured with surgical knots.

The tasks mentioned involve complex movement sequences. Their execution was considerably facilitated by the depth information provided by the stereoscopic camera. In

comparison to the old digital stylus switch, the continuous input handle greatly improved the control of the forceps during grasping of suture material and tissue. The experiment also revealed that the trocars partially disturb force measurements. If instrument movements violated the remote center of motion, e.g. due to carcass movement, strong friction between instrument and tissue caused interfering forces. A direct integration of the force sensors into the functional part of the instrument could reduce this problem, but is technically challenging. Associated with this, also new questions about sterilization of the instruments would arise.

6 Conclusion

In this thesis, we have addressed challenges associated with robotic assistance and autonomy in minimally invasive robotic surgery. This last chapter summarizes the proposed solutions and contributions. While the individual methods have already been evaluated in the respective sections, we now draw conclusions with an overall assessment. Finally, perspectives for further improvements and future development of the topics discussed are provided.

6.1 Contributions

The current trend to trauma reduction changes surgical practice. To manage the increasing complexity of interventions, it is inevitable to automate (sub-) tasks and to provide technical assistance by the robotic system. Our concept of online surgery contributes to the development by acquiring task relevant data during the execution and therewith avoiding many error prone calibration steps.

Online Surgery

The proposed method of *online surgery*, which acquires all information necessary for autonomous and assisted task execution during the intervention itself, constitutes a promising approach to transit from the traditional "data acquisition $\rightarrow$ planning $\rightarrow$ execution" paradigm toward a more flexible and adaptive behavior. To dynamically generate task-related robot movements, environmental perception plays the major role, in which visual information is the driving force. A key concept in online surgery is to enhance the limited view of the conventional laparoscope with a task-specific view that provides sensor data always from the relevant perspective. To realize this concept, we have developed a camera-augmented instrument, which comprises a stereoscopic micro endoscope at the distal end. Therewith, knowledge required for trajectory generation can be acquired directly in situ. Since camera and instrument share one common coordinate frame, inherent calibration issues of the system, which usually impede the accurate alignment of preoperatively generated execution plans with the patient, are avoided and tasks are executed with machine precision. Since the camera-augmented

instrument can be placed freely, exploration of the surgical field from any viewing angle is possible. The procedure of online surgery was conceptually explored by the reference task of tissue dissection.

Localization of Surgical Instruments

We proposed a method to localize and track the position of the tip of a surgical tool in minimally invasive surgery. Without major modifications, the approach is applicable to a broad variety of instrument types typically used in MIRS. This is true for both, the appearance of the instrument's shaft as well as the shape of the functional part of the tool. Neither fiducial markers nor prolonged training of a model is required. Sensor readings, e.g. derived from joint readings or from external tracking modalities, provide a first estimate of the instrument's position in Euclidean space. The pose is then related to the camera image of the laparoscope. This projection is used as initial estimate and further refined by a local visual tracking step. The applied CCD tracking modality maximizes the difference of color statistics in the vicinity of the tool's shape with respect to the background, therewith being independent of the actual shaft coloration. The visually recovered position was used to compensate errors in the sensor prediction, which typically arise from system calibration uncertainties. The localized end of the shaft was complemented with the pose of the functional part of the instrument. The pose was deduced from the kinematics of the instrument connected, which can be identified by means of the newly introduced identification module. This component of our telesurgery system automatically detects instrument changes and reports the corresponding calibration data.

Simulation of Depth Perception by GPU-accelerated Ray Tracing

We devised a ray tracing emulation environment that is capable of simulating arbitrary scenarios of visual depth perception. We employed the Bouquet model to describe lens distortion, that is, intrinsic and extrinsic camera parameters can be set according to real-world calibration results. To mimic realistic settings, sensor noise, calibration uncertainties, and the camera's depth of focus were considered. A projector was implemented to enable the simulation of structured light reconstruction. Every permutation of a pattern mask can be projected. The GPU-acceleration of the framework allows real-time interaction with the scene, while offering realistic-looking images. Due to the working principle of ray tracing, zero-error ground truth depth can be obtained. In conjunction with the 3D reconstruction module of our framework, the generated range maps can be evaluated. Beyond, a closed loop assessment between a (simulated) camera setup and a specific reconstruction algorithm was used to optimize parameter settings.

Depth Reconstruction with Miniaturized Endoscopes supplemented with a Micro Projector

To reconstruct the surface of the surgical field from the task-relevant perspective, we proposed to augment surgical instruments with stereoscopic micro endoscopes. We

first paid attention to the arrangement of the two micro cameras used, each with a size of 1×1×1.5mm, to design a suitable stereo setup. The aligned camera chips were cut directly from silicon to precisely obtain a baseline of 1.2mm. The individual development steps, from simulation to the final camera system, are illustrated again in Fig. 6.1. Due to the close range between sensor and surface, the environment to be reconstructed appears poorly textured. To enhance texture, the stereo system was supplemented with a binary pattern projector. The projection mask was designed to be a globally unambiguous structure that encodes each pixel position, where a code word has a guaranteed minimum Hamming distance to all other code words. Decoding the indices allows reconstruction in the sense of structured light. The design of the pattern likewise supports window-based dissimilarity measures. Therefore, the ray tracing simulation was employed to rate the reconstruction quality while optimizing the pattern resolution. The final pattern design was transferred to an optical blackout mask and mounted behind the same type of lens that was already used for the micro cameras. Illumination was supplied via a fiber optic light guide, which is connected to a LED source. Assembled at the stereo camera pair, the final setup measures approximately 2×2×1.5mm. For depth perception, stereo was treated as a global energy minimization problem, formulated as Markov Random Field. The energy term integrates decoded pattern positions as a prior. To avoid the need for calibrating the projector with respect to the cameras, pattern decoding was performed in both stereo images, resulting in a sparse disparity map. The disparity value of decoded positions was then propagated in the neighborhood to interpolate a dense disparity map, while each pixel received a confidence measure. Pairwise costs are calculated during the correspondence search on the camera images by means of the Census transform. The framework currently supports energy minimization with graph cuts and belief propagation.

Gesture-Type Input Interface

The control of the range of different functionalities offered by surgical robots will likely get more complex with the increase of autonomy. A gesture-based user interface was implemented and evaluated as a time saving alternative to traditional menu input to trigger frequently demanded, automated or semi-automated surgical actions. The approach interprets movements of the haptic devices at the master console as a user command. The method allows for fast interaction times, customizable assignment of gestures and commands, and facilitates the indication of task-relevant in situ coordinates associated with the command. Surgeons can freely combine any personalized gesture with any system function. The most intuitive gestures have been identified during an explorative user study. Another experiment has been conducted to evaluate the efficiency, accuracy, and user experience of this input method compared to a traditional menu-type interface. The results could confirm the potential of gesture-type input, particularly in terms of time savings and enhanced user experience.

Instrument Control and Task Guidance

Based on the situs knowledge gained, we derived image-based control laws to assist the surgeon during instrument positioning and fine-manipulation tasks. Visually ser-

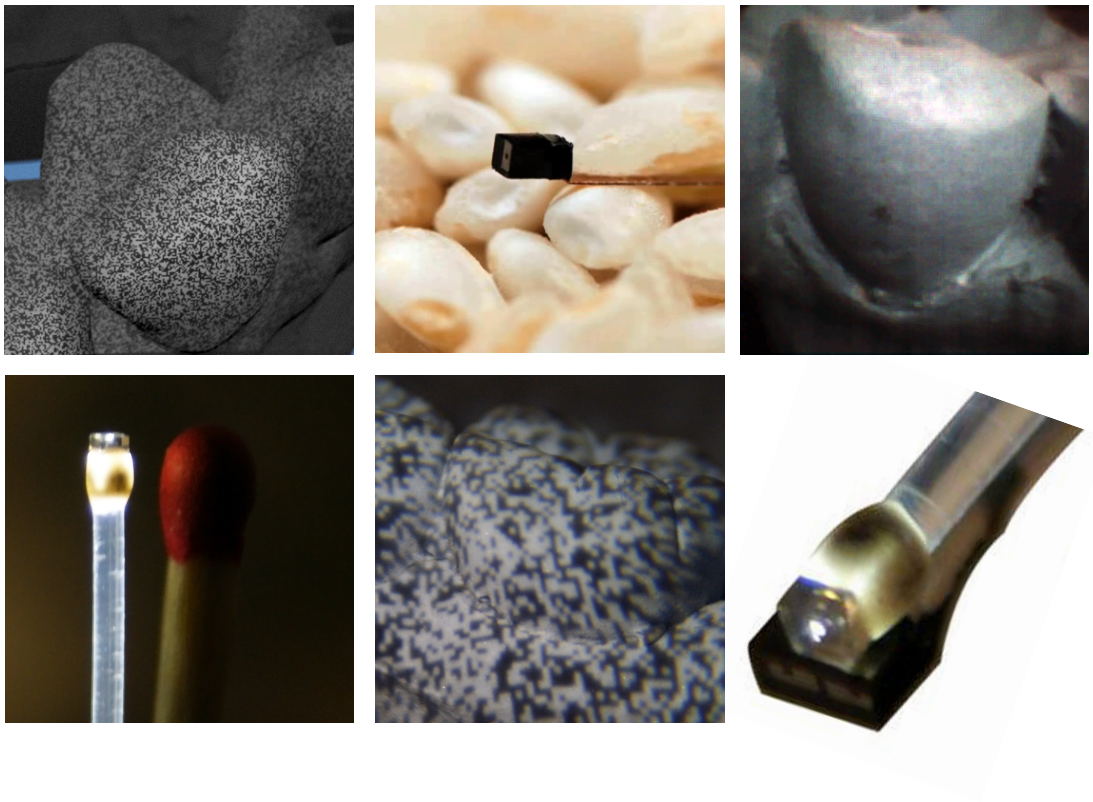


Figure 6.1: Development of miniaturized camera setup, from left to right: simulated camera image, NanEye™ micro camera [image courtesy of Awaiba GmbH], dental scene captured with micro camera, developed micro projector, pattern projected on dental scene, and final setup.

voed instruments are a promising approach in robot-assisted surgery to introduce autonomy and to overcome intrinsic system limitations, caused by calibration inaccuracies. We derived control laws to position both the laparoscopic camera and surgical instruments under the trocar constraint. Further, we introduced a method for remote surgical cutting by providing haptic guidance. The approach becomes independent from calibration uncertainties of typical telesurgery robots by augmenting surgical tools with the micro camera described above. A smooth cut path with corresponding scalpel orientation to guide the user toward the optimal trajectory was calculated and fed back to the operator using virtual fixtures.

Telesurgery Platform

To investigate the potential of the proposed methods, evaluations have been performed on a realistic setup for telesurgery. For the first time, the surgical capabilities of our system were stressed during an animal experiment. The original telemanipulation system has been constantly enhanced: on the hardware side, e.g. automatic instrument identification was introduced and the haptic devices were retrofitted to allow for gestures in seven degrees of freedom. A completely new distributed system software was devised,

which abstracts from the old monolithic structure and now organizes system tasks in separate software modules. Each module can be deployed over network and shares data as well as functionality with other modules.

6.2 Perspectives and Challenges

The step from simply replaying preplanned trajectories toward context-aware and situated behavior of surgical robots is challenging. The ultimate vision of task automation in cognitive surgery, which we outlined in [8], requires the interaction of many different technological aspects. The work of surgeons is facilitated by intuitive context-adaptive user interfaces and optimized visualization. Autonomy particularly requires comprehensive online data acquisition, analysis, and interpretation to associate knowledge about surgical tasks with the conditions found in situ and to provide the necessary adaptivity. In its current state, our approach complements data acquisition in traditional skill-transfer methodology (cf. Sec. 1.1.2). The information necessary to successfully generalize learned skills in new environments can be divided into two classes: low-level scene context and abstract task descriptions. We mainly focused on the former aspect by capturing the scene from the task-relevant perspective. The action of tissue dissection was performed without prior task knowledge and solely relied on sensor input. More complex actions, which might be defined by several cohesive subtasks, can probably only be inferred by using external knowledge or scaffolds. A future challenge will be to link the acquired dynamic scene knowledge in an appropriate manner with a priori knowledge about the surgical procedure to enable (near) complete autonomous operation. For this, interpretation of sensor data is necessary, which drives decisions that trigger the robot behavior. In our current scenario, we have assumed the cut path to be clearly identifiable in the camera images, therewith neglecting potentially interfering structures. In practice, the sensor information could be cross-checked with a surgical plan. The plan would then outline the approximate sequence, while the actual motion generation is performed online based on in situ data. While we restricted ourselves to cutting a plane surface, future work includes the extension of the task to the third dimension. Therefore, basic motion commands could be deduced from stereo reconstruction with the micro endoscope. The limited sampling rate of cameras might however be unsatisfactory for stable motion generation or virtual fixtures. Hence, the manipulator motions could additionally be constrained by combining visual information with force, measured at the instrument tip (see e.g. [91]).

A crucial factor of online surgery is adaptive user interaction. In the majority of cases the proposed automatic camera alignment according to the situational position of the instruments leads to the desired result. However, in some cases it might be necessary to manually affect the system's decision. Therefore, we also investigated gaze contingent camera control as an alternative input modality. Results for our particular setting are provided in appendix A.3. Comparing the gaze position on screen with the current instrument position allows to verify autonomous decisions. In case of inconsistency, manual control can be maintained. Also the pragmatic qualities of gesture-based input, such as its ability to integrate into the surgical workflow, need to be tested in long-term

user studies, particularly the possibility to intervene in the execution in case of misinterpreted commands.

The major limitation of the proposed miniaturized stereo setup is the above-mentioned resolution mismatch between camera and the prototypic projector, which currently impedes pattern decoding. Modifying the distance between projector and camera is not an option, since it yields distortions of the pattern imaged by the camera, therewith negating the advantage of using the same lens for both devices. Increasing the distance between camera and projector, as done for the stereo matching experiments, significantly reduces the light intensity of the projector. To provide better illumination, the currently used f/4.0 aperture should be replaced by a f/2.8 lens for the next generation of the micro projector. Further investigations are also necessary to find a suitable resolution that matches the employed camera. To meet the required alignment precision, camera and projector have to be assembled using an alignment device, which introduces similar precision to the transformation between camera and projector as demonstrated with the alignment of both cameras. A possibility worth looking at, is to learn parameter sets of the reconstruction algorithm within the ray tracing framework [147, 198, 172]. Since the micro cameras are designed for single use only, a simplified projector design would be advantageous. Specifically, placing the pattern mask at the light source instead of mounting it at the end of the fiber optic would strengthen a disposable design. Clearly, this goal is not achievable with the current binary mask, but concentric color patterns, which are placed at the light source, can be projected when a suitable optical fiber is selected. Beyond, interesting application scenarios of the miniaturized cameras are capsule robots [195], NOTES, and otolaryngology. Due to the small size of the cameras, a many-camera system that realizes panoramic views could be implemented.

Appendix

A.1 Inversion of the Bouguet Camera Model

A suitable model to describe the projective geometry of a lens is the Bouguet model [80]. Its wide use in camera calibration and the support in many open source vision libraries allows transferring real-world camera parameters to the ray tracing simulation framework introduced in Sec. 3.3.3. In contrast to the original Bouguet model, where all camera parameters are expressed in pixel units, we specify the parameters as a ratio with respect to the sensor size. This is necessary since measurements given in pixels correspond to a fixed sensor resolution, the simulation however supports user specified sensor resolutions.

In the sequel, we describe the inversion of the camera model, which is employed for raytracing. Instead of removing the distortion of an image as in image calibration, the Bouguet model is used to generate a distorted version of the image during ray tracing [79]. Due to the model's approximation of tangential and radial distortion with higher-order polynomials, the inversion is not straight forward. Starting from the image coordinates $\mathbf{x}_n = [x_{n_x}, x_{n_y}]^{\mathrm{T}}$, given by a regular pinhole camera model with camera matrix K, the distorted coordinates $\mathbf{x}_d = [x_{d_x}, x_{d_y}]^{\mathrm{T}}$ of $\mathbf{x}_n$ are

$$\mathbf{x}_d = \left(1 + k_1 r^2 + k_2 r^4 + k_5 r^6\right) \mathbf{x}_n + \mathbf{dx}, \tag{A.1}$$

with $\mathbf{dx}$ being the tangential distortion

$$\mathbf{dx} = \begin{bmatrix} 2k_3 x_{n_x} x_{n_y} + k_4 (r^2 + 2x_{n_x}^2) \\ k_3 (r^2 + 2x_{n_y}^2) + 2k_4 x_{n_x} x_{n_y} \end{bmatrix}. \tag{A.2}$$

The scalars $k_1, \cdots, k_5$ are the distortion coefficients, and $r^2 = x_{n_x}^2 + x_{n_y}^2$. The final projection is then expressed by

$$\mathbf{x}_p = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K \begin{bmatrix} x_{d_x} \\ x_{d_y} \\ 1 \end{bmatrix}. \tag{A.3}$$

To invert the model, first the effect of the linear camera calibration matrix is reverted by solving (A.3) for $\mathbf{x}_d$. The second step, the inversion of the terms expressing the radial and tangential distortion is more challenging. Since radial distortion is described by a polynomial of 6th grade, there is no analytical solution. First (A.1) is solved for $\mathbf{x}_n$

$$\mathbf{x}_n = \frac{\mathbf{x}_d - \mathbf{dx}}{1 + k_1 r^2 + k_2 r^4 + k_5 r^6}. \tag{A.4}$$

In the next step, an optimization is started with an initial guess $\mathbf{x}_p$ of $\mathbf{x}_n$ on the left side of the equation. On the right-hand term $\mathbf{x}_n$ is injected into the expression of $\mathbf{dx}$ and r. Evaluating (A.4) yields to a new $\hat{\mathbf{x}}_n$ which can then be used in the next iteration. The resulting ray direction is given as $x_{n_x} \cdot u + x_{n_y} \cdot v + w$, where (u, v, w) are homogeneous image plane coordinates.

A.2 Derivation of the Serret-Frenet Formulation

The instantaneous kinematics of the blade tip, used during hybrid control (cf. Sec. 4.4), is based on a solution presented in [180, 121]. The problem of approaching the optimal cut path while avoiding any lateral movement is treated in terms of a Serret-Frenet formulation, where the virtual target frame F moves tangential along $\mathbf{c}$. We consider a blade position $\mathbf{q}$ that deviates by $\mathbf{e}$ from the optimal curve point, as described in (4.31). The blade tip, expressed with respect to the inertial reference frame I is

$${}^{\mathrm{I}}\mathbf{q} = \begin{bmatrix} {}^{\mathrm{I}}q_x & {}^{\mathrm{I}}q_y & 0 \end{bmatrix}^{\mathrm{T}} \tag{A.5}$$

and

$${}^{\mathrm{F}}\mathbf{q} = \mathbf{e} = \begin{bmatrix} {}^{\mathrm{F}}q_{s_1} & {}^{\mathrm{F}}q_{y_1} & 0 \end{bmatrix}^{\mathrm{T}} \tag{A.6}$$

with respect to F. Equivalently, $\mathbf{p}$ is given in I as

$${}^{\mathrm{I}}\mathbf{p} = \begin{bmatrix} {}^{\mathrm{I}}p_x & {}^{\mathrm{I}}p_y & 0 \end{bmatrix}^{\mathrm{T}} \tag{A.7}$$

and in F always as

$${}^{\mathrm{F}}\mathbf{p} = \mathbf{0}_{[3\times 1]}. \tag{A.8}$$

Further, ${}^{\mathrm{I}}_{\mathrm{F}}\mathbf{R}(\theta_c)$ is the rotation matrix from the reference frame I to frame F, parameterized by angle θ_c, and ${}^{\mathrm{F}}_{\mathrm{I}}\mathbf{R} = {}^{\mathrm{I}}_{\mathrm{F}}\mathbf{R}^{-1}$ the reverse rotation respectively. The tangential velocity is denoted as $\dot{s}$ and the angular velocity of θ_c is defined by

$$\omega_c = \dot{\theta}_c = \kappa\big(\mathbf{c}(\hat{r}({}^{\mathrm{F}}\mathbf{q}))\big)\dot{s} \tag{A.9}$$

$$= \kappa({}^{\mathrm{F}}\mathbf{p})\dot{s}, \tag{A.10}$$

where $\kappa(\cdot)$ is the signed curvature [94] of path $\mathbf{c}$ at ${}^{\mathrm{F}}\mathbf{p}$, defined as

$$\kappa(\mathbf{p}) = \frac{p_y''}{(1 + p_y'^{\,2})^{3/2}}. \tag{A.11}$$

The problem of minimizing the angular error θ (cf. Fig. 4.9) is treated with respect to the Serret-Frenet frame F. The above introduced definitions allow expressing the

velocities of both the blade point $\mathbf{q}$ and the target $\mathbf{q}$ in both frames. The target point velocity with respect to frame F is

$$
\begin{aligned}
{}^{\mathrm{F}}\dot{\mathbf{p}} &= {}^{\mathrm{I}}_{\mathrm{F}}\mathbf{R}\,{}^{\mathrm{I}}\dot{\mathbf{p}} && \text{(A.12)}\\
&= \begin{bmatrix} \dot{s} & 0 & 0 \end{bmatrix}^{\mathrm{T}}. && \text{(A.13)}
\end{aligned}
$$

The velocity of the blade tip with respect to frame I can be expressed as

$$
\begin{aligned}
{}^{\mathrm{I}}\dot{\mathbf{q}} &= \begin{bmatrix} {}^{\mathrm{I}}\dot{q}_x & {}^{\mathrm{I}}\dot{q}_y & 0 \end{bmatrix}^{\mathrm{T}} && \text{(A.14)}\\
&= {}^{\mathrm{I}}\dot{\mathbf{p}} + {}^{\mathrm{F}}_{\mathrm{I}}\mathbf{R}\dot{\mathbf{e}} + {}^{\mathrm{F}}_{\mathrm{I}}\mathbf{R}(\omega_c \times \mathbf{e}), && \text{(A.15)}
\end{aligned}
$$

where $\mathbf{e}$ is the error vector between the blade and the target point on the trajectory. Multiplication of (A.15) with ${}^{\mathrm{I}}_{\mathrm{F}}\mathbf{R}$ yields to an expression of $\dot{\mathbf{q}}$ in F as

$$
\begin{aligned}
{}^{\mathrm{I}}\dot{\mathbf{p}} &= {}^{\mathrm{I}}_{\mathrm{F}}\mathbf{R}\,{}^{\mathrm{F}}\dot{\mathbf{q}} && \text{(A.16)}\\
&= {}^{\mathrm{F}}\dot{\mathbf{p}} + \dot{\mathbf{e}} + (\omega_c \times \mathbf{e}). && \text{(A.17)}
\end{aligned}
$$

Taking the cross product $(\omega_c \times \mathbf{e})$ yields to

$$
\begin{bmatrix} 0 \\ 0 \\ \kappa(s)\dot{s} \end{bmatrix} \times \begin{bmatrix} {}^{\mathrm{F}}q_{s_1} \\ {}^{\mathrm{F}}q_{y_1} \\ 0 \end{bmatrix} = \begin{bmatrix} -\kappa(s)\dot{s} \cdot {}^{\mathrm{F}}\, q_{y_1} \\ \kappa(s)\dot{s} \cdot {}^{\mathrm{F}}\, q_{s_1} \\ 0 \end{bmatrix}, \qquad \text{(A.18)}
$$

which is substituted in (A.17) and finally gives the blade velocity with respect to the target frame as

$$
{}^{\mathrm{F}}\dot{\mathbf{q}} = \begin{bmatrix} \dot{s}\big(1 - \kappa(s){}^{\mathrm{F}}q_{y_1}\big) + {}^{\mathrm{F}}\,\dot{q}_{s_1} \\ {}^{\mathrm{F}}q_{y_1} + \kappa(s)\dot{s} \cdot {}^{\mathrm{F}}\, q_{s_1} \\ 0 \end{bmatrix}. \qquad \text{(A.19)}
$$

After simple transformations, we obtain the axis velocity components ${}^{\mathrm{F}}\dot{q}_{s_1}$ and ${}^{\mathrm{F}}\dot{q}_{y_1}$:

$$
\begin{aligned}
{}^{\mathrm{F}}\dot{q}_{s_1} &= \begin{bmatrix} \cos\theta_c & \sin\theta_c \end{bmatrix} \begin{bmatrix} {}^{\mathrm{I}}\dot{q}_x \\ {}^{\mathrm{I}}\dot{q}_y \end{bmatrix} - \dot{s}\big(1 - \kappa(s) \cdot {}^{\mathrm{F}}\, q_{y_1}\big), && \text{(A.20)}\\
{}^{\mathrm{F}}\dot{q}_{y_1} &= \begin{bmatrix} -\sin\theta_c & \cos\theta_c \end{bmatrix} \begin{bmatrix} {}^{\mathrm{I}}\dot{q}_x \\ {}^{\mathrm{I}}\dot{q}_y \end{bmatrix} - \kappa(s)\dot{s}{}^{\mathrm{F}}q_{s_1}. && \text{(A.21)}
\end{aligned}
$$

Considering the current blade orientation θ_m, its forward velocity is given with

$$
\begin{bmatrix} {}^{\mathrm{I}}\dot{q}_x \\ {}^{\mathrm{I}}\dot{q}_y \end{bmatrix} = \upsilon \begin{bmatrix} \cos\theta_m \\ \sin\theta_m \end{bmatrix}. \qquad \text{(A.22)}
$$

Without loss of generality, υ is derived from the velocity applied by the user at the haptic device. Injecting (A.22) in (A.20) and (A.21) and applying the trigonometric difference formulae for $\theta = \theta_m - \theta_c$

$$
\begin{aligned}
\sin\theta &= \sin\theta_m \cos\theta_c + \cos\theta_m \sin\theta_c && \text{(A.23)}\\
\cos\theta &= \cos\theta_m \cos\theta_c + \sin\theta_m \sin\theta_c && \text{(A.24)}
\end{aligned}
$$

yields to the final instantaneous kinematic model of the blade tip, expressed in frame F, with

$$
\begin{aligned}
{}^{\mathrm{F}}\dot{q}_{s_1} &= \upsilon\cos\theta - \dot{s}\big(1 - \kappa(s)\cdot {}^{\mathrm{F}}q_{y_1}\big) && \text{(A.25)}\\
{}^{\mathrm{F}}\dot{q}_{y_1} &= \upsilon\sin\theta - \kappa(s)\dot{s}\cdot {}^{\mathrm{F}}q_{s_1} && \text{(A.26)}\\
\dot{\theta} &= \omega - \kappa(s)\dot{s}, && \text{(A.27)}
\end{aligned}
$$

where $\omega = \omega_m = \dot{\theta}_m$.

A.3 Gaze Contingent Control

In addition to the automated laparoscope control presented in Sec. 4.3, we also provide manual camera control by means of gaze contingent input [10]. The user can explicitly choose a target with which the camera is to be aligned by placing his gaze on screen. The polarized goggles required to perceive a depth impression with our 3D screen prevents however the use of remote eye trackers. The proposed solution combines polarized goggles with a head-worn eye tracker. The polarization foil sits between the viewer and the monitor, enabling a free field of view of the camera to the eye. Such kind of eye tracker yields the gaze direction in head coordinates, hence head tracking is required to determine the intersection of the line of sight with the display plane, while allowing the user a certain degree of mobility. In collaboration with the EyeSeeCam project we could adapt their monocular tracking glasses.

For a description of the tracking technique, we refer to [174, 95]. However, note that infrared markers are attached on the 3D screen to perform head tracking (cf. Fig. 5.2). Since the actual image plane of the monitor is offset to the rear by design (because of the semi-transparent mirror), the resulting transformation between the marker plane and the image plane must be considered.

Data Processing and Experimental Evaluation

The eye tracker samples 220 fixation positions each second, providing two-dimensional screen coordinates. The values are smoothed by means of a recursive exponential filter [186]. Therefore, the observation period may be chosen arbitrarily long without the need of storing previous data. A smoothed value x_{n+1} of observation $n+1$ can be written as linear combination of the filtered value x_n of observation n and the new data value z_{n+1} obtained in observation $n+1$:

$$x_{n+1} = x_n + k_{n+1}\left(z_{n+1} - x_n\right), \tag{A.28}$$

with gain factor k

$$k_{n+1} = \frac{k_n}{k_n + \exp\big(-\big(t_{n+1} - t_n\big)/\tau\big)}, \tag{A.29}$$

where τ governs the filter's time scale and t is a timestamp. When choosing a shorter time scale the filter adapts faster to new values, but attributes to high-frequency noise.

A longer time scales averages the gaze position in favor of past values.

The filter output is visualized on the stereoscopic screen and used to control the manipulator that holds the laparoscope, at what the error vector between the image center and the gaze point is minimized. Two different control modes were implemented: (1) continuous alignment of the camera with the situs area that is considered to be important, since it is observed by the surgeon over a longer time period, and (2) onetime alignment with the current gaze point. The first behavior is achieved by setting τ to large values. The endoscope then only adapts gradually to new positions, if the position is long enough in the user's focus of attention. The second behavior must be triggered explicitly by the surgeon. As long as a foot pedal is pressed, the endoscope follows the current gaze point. The filter's time scale is chosen small, therewith the alignment is performed rather quick. The influence of the time scale is illustrated in Fig. A.1(b).

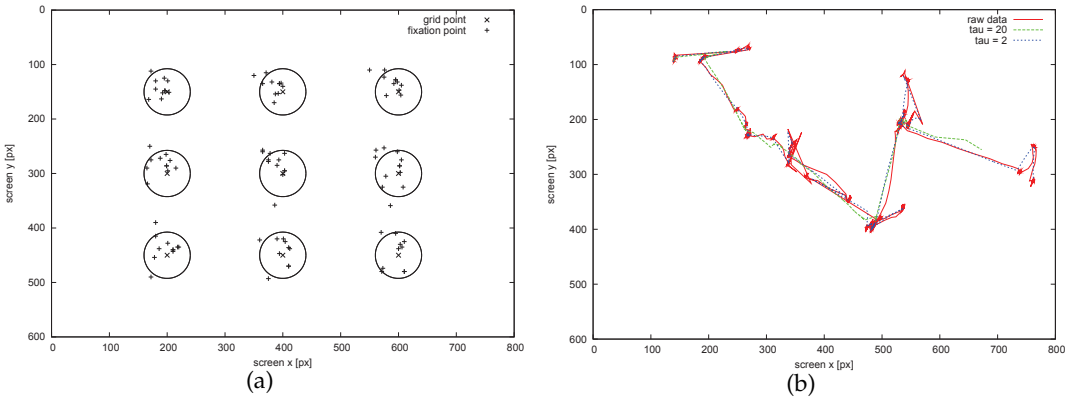


Figure A.1: (a) subject evaluation grid; (b) influence of the parameter τ on the trajectory generation.

We give details on the accuracy of the eye tracking system in our particular setup. For the conducted evaluation, 10 untrained subjects were instructed to fixate 9 regularly spaced points on a grid. The distance between a subject and the image plane was about 40cm, respectively 30cm to the frame of the screen. Note that the distance between user and image plane varies from the upper to the lower image area, since the image plane of the stereoscopic monitor is tilted by design. The stimulus points $(\times)$ were arranged using a screen resolution of 800×600px (cf. Fig. A.1(a)). The intersections between the image plane and the right eye's line of sight are indicated with the plus sign $(+)$. The subjects looked straight at the monitor. The average accuracy was 22.5px, which is sufficient for the employed closed-loop endoscope control. The error range is indicated by the circles with a radius of 40px. The better performance of the tracker in conjunction with conventional monitors [95] can be attributed to the additionally required transformation between LED markers and actual image plane of the stereo display in our setting.

A.4 Micro Camera Interface

The original NanEye™ driver required two major modifications in order to allow its integration into the existing framework. First, the driver could only handle one camera at a time. The issue could be solved by enumerating the connected devices and adapting the initialization routine of the USB communication board. Second, the camera's software is a managed C# .NET library, which usually can only be called from managed code. The driver had to be wrapped to be usable in combination with the existing (unmanaged) C++ code. While calling unmanaged code from managed code can be easily performed with "P/Invoke", the other way around is more complex. Often, using "COM Interop Assemblies" (Component Object Model) is a possible solution, but more difficult if third party libraries are involved. Making libraries accessible via COM requires them to be signed with a strong name. Strong naming is a concept in .NET similar to GUID's in traditional VS/C++ and is necessary to share the assembly in the global assembly cache that can be accessed by different applications.

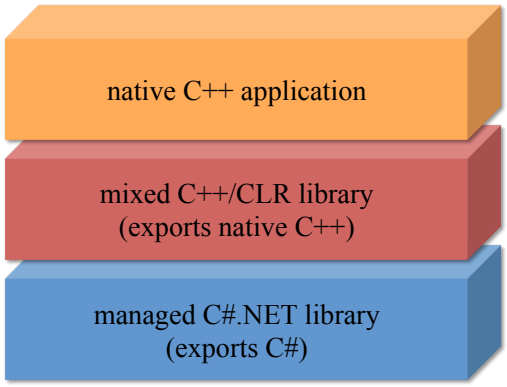


Figure A.2: Wrapper scheme

The Microsoft Visual Studio™(VS) development environment offers a tool chain that supports the process, including the applications `tlbexp.exe`, which exports a .NET assembly into a COM type-library: The tool `regasm.exe` then registers the assembly. Unfortunately, the driver comprises third-party code that could not be signed with strong names, and therefore the tool chain can not be used. In this case, a mixed-code wrapper library, consisting of managed and unmanaged code, is the only solution. The mixed native/-CLI module acts as a broker between native C++ and .NET and makes the conjunction with both worlds (cf. Fig. A.2). On one hand, it has the ability to call managed .NET code, on the other hand, it exports native interfaces via `NATIVEDLL_API __declspec(dllexport)` that can be loaded into the unmanaged process. However, it is not possible to store managed data types in unmanaged code. Fortunately, the *C++ Support Library*, includable via `<vcclr.h>`, provides a way to access managed types via the `System::Runtime::InteropServices::GCHandle` class, which is wrapped for convenience as the template `gcroot<T>` and allows to store reference pointers on managed classes. The "^" operator indicates a reference pointer to a managed object. Instantiating the class with `gcnew` means that we are allocating on the garbage collector protected environment. Having now access to the managed code, the last thing we need to take care about is data conversion of complex types, also known as marshalling. While simple data types don't need to be converted, the C++ Support Library offers methods to marshal e.g., arrays or strings via `InteropServices::Copy(...)` and `InteropServices::StringToHGlobalAuto(...)`.
In our case, the driver interface provides methods to access the image width and height as well as the image data, stored in a 4-channel byte array. First, the wrapper creates a

reference to the driver interface in the constructor:

```
NanEyeI^ fcns = gcnew NanEyeI();
gcroot<NanEyeI^> *interfacePtr = new gcroot<NanEyeI^>(fcns)
this->_pNanEyeFcnsClr = (void*)interfacePtr
```

Access to functions with non-complex return types, e.g., requesting the image width, is possible via:

```
gcroot<NanEyeInterface^> *pp =
   reinterpret_cast<gcroot<nanEyeI^>*>(_pNanEyeFcnsClr);
return ((nanEyeI^)*pp)->getImageWidth();
```

The image data itself can then either entirely be marshaled as introduced above, or, as required for our scenario, converted to a 3-channel char array by regular type-casting:

```
cli::array<Byte>^ imgData = ((NanEyeI^)*pp)->getImageData();
int width = this->getImageWidth();
int height = this->getImageHeight();
for (int i = 0; i < width * height; i++) {
   buffer[i*3] = (char)imgData[i*4];
   buffer[i*3+1] = (char)imgData[i*4+1];
   buffer[i*3+2] = (char)imgData[i*4+2];
}
```

Authored and Co-Authored Publications

[1] R. Bauernschmitt, E. Braun, M. Buss, F. Frohlich, S. Hirche, G. Hirzinger, J. Kammerl, A. Knoll, R. Konietschke, B. Kubler, R. Lange, H. Mayer, M. Rank, G. Schillhuber, C. Staub, E. Steinbach, A. Tobergte, H. Ulbrich, I. Vittorias, and C. Zhao, "On the role of multimodal communication in telesurgery systems," in *IEEE International Workshop on Multimedia Signal Processing*, Rio de Janeiro, Brazil, okt. 2009.

[2] E. U. Braun, C. Gärtner, H. Mayer, C. Staub, A. Knoll, R. Lange, and R. Bauernschmitt, "Features in telemanipulation for heart surgery: Haptic tasks," in *World Congress on Medical Physics and Biomedical Engineering*, ser. IFMBE Proceedings, vol. 2, no. 6, Munich, Germany, sep. 2009, pp. 6–7.

[3] E. U. Braun, C. Gärtner, C. Staub, A. Knoll, R. Lange, and R. Bauernschmitt, "Force feedback: Plus factor in telemanipulation for heart surgeons," in *Proceedings of the Russian-Bavarian Conference on Biomedical Engineering*, Munich, Germany, jul. 2009.

[4] S. Can, B. Jensen, E. Dean-Leon, C. Staub, A. Knoll, A. Fiolka, A. Schneider, A. Meining, and H. Feussner, "Kinematics, control and workspace analysis of a bowden wire actuated manipulator for minimally invasive single-port surgery," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, Guangzhou, China, dec. 2012.

[5] S. Can, C. Staub, A. Knoll, A. Fiolka, A. Schneider, and H. Feussner, "Design, development and evaluation of a highly versatile robot platform for minimally invasive single-port surgery," in *Proceedings of the IEEE/RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics*, jun. 2012, pp. 817–822.

[6] M. Kaiser, B. Kwolek, C. Staub, and G. Rigoll, "Registration of 3d facial surfaces using covariance matrix pyramids," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, USA, may 2010, pp. 1002–1007.

[7] A. Knoll, H. Mayer, C. Staub, and R. Bauernschmitt, "Selective automation and skill transfer in medical robotics: a demonstration on surgical knot-tying," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 8, no. 4, pp. 384–97, dec. 2012.

[8] M. Kranzfelder, C. Staub, A. Fiolka, A. Schneider, S. Gillen, D. Wilhelm, H. Friess, A. Knoll, and H. Feussner, "Toward increased autonomy in the surgical or: needs, requests, and expectations," *Surgical Endoscopy*, pp. 1–8, 2012.

[9] T. Osa, C. Staub, and A. Knoll, "Framework of automatic robot surgery system using visual servoing," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, oct. 2010, pp. 1837–1842.

[10] C. Staub, S. Can, B. Jensen, A. Knoll, and S. Kohlbecher, "Human-computer interfaces for interaction with surgical tools in robotic surgery," in *Proceedings of the IEEE/RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics*, Rome, Italy, jun. 2012, pp. 81–86.

[11] C. Staub, A. Heider, M. Grimm, and A. Knoll, "On the generation of ground truth data for depth reconstruction," *Workshop of the International Conference of Medical Image Computing and Computer Assisted Intervention (MIDAS Journal)*, oct. 2012.

[12] C. Staub, C. Lenz, B. Jensen, S. Can, A. Knoll, and R. Bauernschmitt, "Micro camera augmented endoscopic instruments: Towards superhuman performance in remote surgical cutting," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Portugal, oct. 2012, pp. 2000–2006.

[13] C. Staub, Y. Ning, S. Can, and A. Knoll, "A distributed software framework for robotic surgery," *Workshop of the International Conference of Medical Image Computing and Computer Assisted Intervention (MIDAS Journal)*, aug. 2011.

[14] C. Staub, S. Can, and A. Knoll, "Haptic gesturing as human-machine interface in minimally invasive robotic surgery," in *Proceedings of the Hamlyn Symposium on Medical Robotics*, London, UK, jul. 2011, pp. 57–58.

[15] C. Staub, S. Can, A. Knoll, V. Nitsch, I. Karl, and B. Färber, "Implementation and evaluation of a gesture-based input method in robotic surgery," in *Proceedings of the IEEE International Symposium on Haptic Audio–Visual Environments and Games*, Qinhuangdao, China, oct. 2011, pp. 1–7.

[16] C. Staub, A. Knoll, T. Osa, and R. Bauernschmitt, "Autonomous high precision positioning of surgical instruments in robot-assisted minimally invasive surgery under visual guidance," in *Proceedings of the IEEE International Conference on Autonomic and Autonomous Systems*, Cancun, Mexico, mar. 2010, pp. 64–69.

[17] C. Staub, C. Lenz, G. Panin, A. Knoll, and R. Bauernschmitt, "Contour-based surgical instrument tracking supported by kinematic prediction," in *Proceedings of the IEEE/RAS International Conference on Biomedical Robotics and Biomechatronics*, Tokyo, Japan, sep. 2010, pp. 746–752.

[18] C. Staub, H. Mayer, T. Osa, E. U. Braun, A. Knoll, and R. Bauernschmitt, "Setup of a scientific research platform for robot-assisted minimally invasive heart surgery scenarios," in *World Congress on Medical Physics and Biomedical Engineering*, ser. IFMBE Proceedings, vol. 25, no. 6, Munich, Germany, sep. 2009, pp. 259–262.

[19] C. Staub, K. Ono, H. Mayer, A. Knoll, H. Ulbrich, and R. Bauernschmitt, "Remote minimally invasive surgery – haptic feedback and selective automation in medical robotics," *Applied Bionics and Biomechanics*, vol. 8, no. 2, pp. 221–236, aug. 2011.

[20] C. Staub, T. Osa, A. Knoll, and R. Bauernschmitt, "Automation of tissue piercing using circular needles and vision guidance for computer aided laparoscopic surgery," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, USA, may 2010, pp. 4585–4590.

[21] C. Staub, G. Panin, and A. Knoll, "Visual instrument guidance in minimally invasive robot surgery," *International Journal on Advances in Life Sciences*, vol. 2, no. 3&4, pp. 103–104, 2010.

References

[22] D. Aarno, S. Ekvall, and D. Kragic, "Adaptive virtual fixtures for machine-assisted teleoperation tasks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, April 2005, pp. 897 – 903.

[23] J. Abbott, P. Marayong, and A. Okamura, "Haptic virtual fixtures for robot-assisted manipulation," in *Robotics Research*, ser. Springer Tracts in Advanced Robotics, S. Thrun, R. Brooks, and H. Durrant-Whyte, Eds. Springer Berlin Heidelberg, 2007, vol. 28, pp. 49–64.

[24] M. Abu Gazala, N. Shussman, S. Abu Gazala, A. Schlager, R. Elazary, O. Ponomernco, A. Khalaila, A. Rivkind, and Y. Mintz, "Miniature camera for enhanced visualization for single-port surgery and notes." *Journal of Laparoendoscopic & Advanced Surgical Techniques*, vol. 22, no. 10, pp. 984–8, December 2012.

[25] I. Albitar, P. Graebling, and C. Doignon, "Robust structured light coding for 3d reconstruction," in *Proceedings of the IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, oct. 2007, pp. 1 –6.

[26] J. Arata, H. Takahashi, P. Pitakwatchara, S. Warisawa, K. Tanoue, K. Konishi, S. Ieiri, S. Shimizu, N. Nakashima, K. Okamura, Y. Fujino, Y. Ueda, P. Chotiwan, M. Mitsuishi, and M. Hashizume, "A remote surgery experiment between japan and thailand over internet using a low latency codec system," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Rome, Italy, apr. 2007, pp. 953 –959.

[27] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *Proceedings of the ACM-SIAM symposium on Discrete algorithms*, Philadelphia, PA, USA, 2007, pp. 1027–1035.

[28] E. Bauzano, V. Munoz, and I. Garcia-Morales, "Auto-guided movements on minimally invasive surgery for surgeon assistance," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, oct. 2010, pp. 1843 –1848.

[29] R. Beasley and R. Howe, "Increasing accuracy in image-guided robotic surgery through tip tracking and model-based flexion correction," *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 292 –302, apr. 2009.

[30] P. Berkelman, P. Cinquin, J. Troccaz, J. Ayoubi, C. Letoublon, and F. Bouchard, "A compact, compliant laparoscopic endoscope manipulator," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, Washington, DC, USA, may 2002, pp. 1870 – 1875.

[31] A. Bettini, P. Marayong, S. Lang, A. M. Okamura, and G. D. Hager, "Vision-assisted control for manipulation using virtual fixtures," *IEEE Transactions on Robotics*, vol. 20, no. 6, pp. 953 – 966, dec. 2004.

[32] A. Bigdelou, R. Stauder, T. Benz, A. Okur, T. Blum, R. Ghotbi, and N. Navab, "Hci design in the or: A gesturing case-study," in *MICCAI Workshop on Modeling and Monitoring of Computer Assisted Interventions*, Nice, France, oct. 2012.

[33] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., 2006.

[34] F. Blais, "Review of 20 years of range sensor development." *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 231–243, jan. 2004.

[35] A. F. Bobick and S. S. Intille, "Large occlusion stereo," *International Journal of Computer Vision*, vol. 33, no. 3, pp. 181–200, sep. 1999.

[36] P. Breeveld, "Observation, manipulation, and hand-eye coordination problems in minimally invasive surgery," in *Proceedings of the European Annual Conference on Human Decision Making and Manual Control*, Kassel, Germany, dec. 1997, pp. 219–231.

[37] M. Brown, D. Burschka, and G. Hager, "Advances in computational stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 993 – 1008, aug. 2003.

[38] D. Burschka, J. J. Corso, M. Dewan, W. W. Lau, M. Li, H. C. Lin, P. Marayong, N. A. Ramey, G. D. Hager, B. Hoffman, D. Larkin, and C. J. Hasser, "Navigating inner space: 3-d assistance for minimally invasive surgery," *Robotics and Autonomous Systems*, vol. 52, no. 1, pp. 5–26, jul. 2005.

[39] L. H. C. Guan and D. L. Lau, "Composite structured light pattern for three-dimensional video," *Optics Express*, vol. 11, no. 5, pp. 406–417, mar. 2003.

[40] S. Can, "A highly versatile single-port system for minimally invasive surgery," Ph.D. dissertation, Technische Universität München, München, Germany, 2012.

[41] A. Casals, J. Amat, and E. Laporte, "Automatic guidance of an assistant robot in laparoscopic surgery," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Minneapolis, Minnesota, USA, apr. 1996, pp. 895–900.

[42] M. Çavaşoğlu, W. Williams, F. Tendick, and S. Sastry, "Robotics for telesurgery: Second generation Berkeley/UCSF laparoscopic telesurgical workstation and looking towards the future applications," *Industrial Robot, Special Issues on Medical Robotics*, vol. 30, no. 1, pp. 22–29, 2003.

[43] F. Chaumette and S. Hutchinson, "Visual servo control. basic approaches." *Robotics & Automation Magazine, IEEE*, vol. 13, no. 4, pp. 82–90, dec. 2006.

[44] ——, "Visual servo control. advanced approaches." *IEEE Robotics & Automation Magazine*, vol. 14, no. 1, pp. 109–118, mar. 2007.

[45] J. Climent and P. Mares, "Automatic instrument localization in laparoscopic surgery," *Electronic Letters on Computer Vision and Image Analysis*, vol. 4, no. 1, pp. 21–31, nov. 2004.

[46] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, 2nd ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1989.

[47] P. Dario, E. Guglielmelli, B. Allotta, and M. Carrozza, "Robotics for medical applications," *IEEE Robotics Automation Magazine*, vol. 3, no. 3, pp. 44 –56, sep. 1996.

[48] J. P. Desai and N. Ayache, "Special issue on medical robotics," *The International Journal of Robotics Research*, vol. 28, no. 9, jul. 2009.

[49] ——, "Special issue on medical robotics," *The International Journal of Robotics Research*, vol. 28, no. 10, oct. 2009.

[50] C. Doignon, P. Graebling, and M. de Mathelin, "Real-time segmentation of surgical instruments inside the abdominal cavity using a joint hue saturation color feature," *Real-Time Imaging*, vol. 11, no. 5-6, pp. 429–442, oct. 2005.

[51] P. Dutkiewicz, M. Kielczewski, and M. Kowalski, "Visual tracking of surgical tools for laparoscopic surgery," in *Proceedings of the Fourth International Workshop on Robot Motion and Control*, Puszczykowo, Poland, jun. 2004, pp. 23 – 28.

[52] P. H. C. Eilers and B. D. Marx, "Flexible smoothing with b-splines and penalties," *Statistical Science*, vol. 11, no. 2, pp. 89–102, 1996.

[53] R. Elble and W. Koller, *Tremor*. Baltimore, USA: John Hopkins University Press, 1990.

[54] B. Espiau, "Effect of camera calibration errors on visual servoing in robotics," in *Proceedings of the International Symposium on Experimental Robotics*, 1994, pp. 182–192.

[55] P. Felzenszwalb and D. Huttenlocher, "Efficient belief propagation for early vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, Washington, DC, USA, jun./jul. 2004, pp. 261–268.

[56] G. Fichtinger, P. Kazanzides, A. Okamura, G. Hager, L. Whitcomb, and R. Taylor, "Surgical and interventional robotics," *IEEE Robotics Automation Magazine*, vol. 15, no. 3, pp. 94 –102, sep. 2008.

[57] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, jun. 1981.

[58] F. A. Fröhlich, G. Passig, A. Vazquez, and G. Hirzinger*, "Robot assisted internal mammary artery detection for coronary revascularisation surgery," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, oct. 2010, pp. 1849 –1855.

[59] D. Gallup, J.-M. Frahm, P. Mordohai, and M. Pollefeys, "Variable baseline/resolution stereo," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, Alaska, USA, jun. 2008, pp. 1 –8.

[60] A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Proceedings of the IEEE Asian Conference on Computer Vision*, Queenstown, New Zealand, nov. 2011, pp. 25–38.

[61] J. Geng, "Structured-light 3d surface imaging: a tutorial," *Adv. Opt. Photon.*, vol. 3, no. 2, pp. 128–160, jun. 2011.

[62] T. L. Gibo, L. N. Verner, D. D. Yuh, and A. M. Okamura, "Design considerations and human-machine performance of moving virtual fixtures," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Kobe, Japan, may 2009, pp. 671–676.

[63] J.-B. Gómez, A. Ceballos, F. Prieto, and T. Redarce, "Mouth gesture and voice command based robot command interface," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Kobe, Japan, may 2009, pp. 333–338.

[64] P. M. Griffin, L. S. Narasimhan, and S. R. Yee, "Generation of uniquely encoded light patterns for range data acquisition," *Pattern Recognition*, vol. 25, no. 6, pp. 609 – 616, jun. 1992.

[65] K. Guerin, B. Vagvolgyi, A. Deguet, C. Chen, D. Yuh, and R. Kumar, "Reachin: A modular vision based interface for teleoperation," in *Workshop on Systems and Architectures for Computer Assisted Interventions*, aug. 2010.

[66] S. Günter and H. Bunkey, "Hmm-based handwritten word recognition: on the optimization of the number of states, training iterations and gaussian components," *Pattern Recognition*, vol. 37, no. 10, pp. 2069–2079, jan. 2004.

[67] W. Guo-Qing, K. Arbter, and G. Hirzinger, "Real-time visual servoing for laparoscopic surgery. controlling robot motion with color image segmentation," *IEEE Engineering in Medicine and Biology Magazine*, vol. 16, no. 1, pp. 40–45, jan./feb. 1997.

[68] G. Guthart and J. Salisbury, "The intuitive™ telesurgery system: overview and application," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, San Francisco, California, USA, apr. 2000, pp. 618–621.

[69] I. Guyon, P. Albrecht, Y. L. Cun, J. Denker, and W. Hubbard, "Design of a neural network character recognizer for a touch terminal," *Pattern Recognition*, vol. 24, no. 2, pp. 105 – 119, 1991.

[70] G. Hager, A. Okamura, P. Kazanzides, L. Whitcomb, G. Fichtinger, and R. Taylor, "Surgical and interventional robotics," *IEEE Robotics Automation Magazine*, vol. 15, no. 4, pp. 84 –93, dec. 2008.

[71] U. Hagn, M. Nickl, S. Jörg, G. Passig, T. Bahls, A. Nothhelfer, F. Hacker, L. Le-Tien, A. Albu-Schäffer, R. Konietschke, M. Grebenstein, R. Warpup, R. Haslinger, M. Frommberger, and G. Hirzinger, "Dlr miro: A versatile lightweight robot for surgical applications," *Industrial Robot*, vol. 35, no. 4, pp. 324–336, 2008.

[72] R. Hanek, T. Schmitt, S. Buck, and M. Beetz, "Fast image-based object localization in natural scenes," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, Lausanne, Switzerland, sep. 2002, pp. 116–122.

[73] R. Hanek and M. Beetz, "The contracting curve density algorithm: Fitting parametric curve models to images using local self-adapting separation criteria," *International Journal of Computer Vision*, vol. 59, no. 3, pp. 233–258, 2004.

[74] B. Hannaford, *Surgical Robotics: Systems Applications and Visions*, J. Rosen and R. Satava, Eds. Springer, Berlin Heidelberg, 2011.

[75] N. Hansen, "The cma evolution strategy: A comparing review," *Advances on estimation of distribution algorithms*, pp. 1769–1776, 2006.

[76] B. M. Harnett, C. R. Doarn, J. Rosen, B. Hannaford, and T. J. Broderick, "Evaluation of unmanned airborne vehicles and mobile robotic telesurgery in an extreme environment," *Telemedicine and e-Health*, vol. 14, no. 6, pp. 534–544, jul./aug. 2008.

[77] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. New York, NY, USA: Cambridge University Press, 2003.

[78] M. Hassenzahl, M. Burmester, and F. Koller, "Attrakdiff: Ein fragebogen zur messung wahrgenommener hedonischer und pragmatischer qualität," in *Mensch & Computer 2003: Interaktion in Bewegung*, G. Szwillus and J. Ziegler, Eds., Stuttgart, Germany, 2003, pp. 187–196.

[79] A. Heider, "Simulation of pmd sensors using ray tracing," Bachelor Thesis, Technische Universität München, 2011.

[80] J. Heikkila, "Accurate camera calibration and feature based 3-d reconstruction," 1997.

[81] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328 –341, feb. 2008.

[82] S. Hutchinson, G. Hager, and P. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651–670, oct. 1996.

[83] P. Hynes, G. Dodds, and A. Wilkinson, "Uncalibrated visual-servoing of a dual-arm robot for surgical tasks," in *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Espoo, Finland, jun. 2005, pp. 151 – 156.

[84] K. Höller, "Novel techniques for spatial orientation in natural orifice translumenal endoscopic surgery (notes)," Ph.D. dissertation, Friedrich-Alexander-University Erlangen-Nürnberg, 2010.

[85] M. Isard and A. Blake, "Condensation - conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[86] I. Ishii, K. Yamamoto, K. Doi, and T. Tsuji, "High-speed 3d image acquisition using coded structured light projection," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, California, USA, oct. 2007, pp. 925–930.

[87] S. Jäger, S. Manke, J. Reichert, and A. Waibel, "Online handwriting recognition: the npen++ recognizer," *International Journal on Document Analysis and Recognition*, vol. 3, no. 3, pp. 169–180, 2001.

[88] A. Kapoor and R. Taylor, "A constrained optimization approach to virtual fixtures for multi-handed tasks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Pasadena, California, USA, may 2008, pp. 3401–3406.

[89] J. Kast, J. Neuhaus, F. Nickel, H. Kenngott, M. Engel, E. Short, M. Reiter, H.-P. Meinzer, and L. Maier-Hein, "Der telemanipulator davinci als mechanisches trackingsystem," in *Bildverarbeitung für die Medizin*, ser. Informatik aktuell, H.-P. Meinzer, T. M. Deserno, H. Handels, and T. Tolxdorff, Eds. Springer Berlin Heidelberg, 2009, pp. 92–96.

[90] P. Kazanzides, G. Fichtinger, G. Hager, A. Okamura, L. Whitcomb, and R. Taylor, "Surgical and interventional robotics - core concepts, technology, and design," *IEEE Robotics Automation Magazine*, vol. 15, no. 2, pp. 122 –130, jun. 2008.

[91] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal Robotics and Automation*, vol. 3, no. 1, pp. 43–53, feb. 1987.

[92] M.-S. Kim, J.-S. Heo, and J.-J. Lee, "Visual tracking algorithm for laparoscopic robot surgery," in *Fuzzy Systems and Knowledge Discovery*, ser. Lecture Notes in Computer Science, L. Wang and Y. Jin, Eds. Springer Berlin Heidelberg, 2005, vol. 3614, pp. 491–491.

[93] A. Knoll and R. Sasse, "An active stereometric triangulation technique using a continuous colour pattern," in *Graphics and Robotics*. Springer, 1995, pp. 191–206.

[94] S. Kobayashi and K. Nomizu, *Foundations of differential geometry*, 2nd ed. John Wiley & Sons Australia, 1969.

[95] S. Kohlbecher, K. Bartl, S. Bardins, and E. Schneider, "Low-latency combined eye and head tracking system for teleoperating a robotic head in real-time," in *ACM Symposium on Eye Tracking Research and Applications*, Austin, Texas, USA, mar. 2010, pp. 117–120.

[96] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 65–81, feb. 2004.

[97] N. Komodakis and G. Tziritas, "Approximate labeling via graph cuts based on linear programming," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 8, pp. 1436–1453, aug. 2007.

[98] R. Konietschke, A. Busam, T. Bodenmüller, T. Ortmaier, M. Suppa, J. Wiechnik, T. Welzel, G. Eggers, G. Hirzinger, and R. Marmulla, "Accuracy identification of markerless registration with the dlr handheld 3d-modeller in medical applications," in *Tagsungsband der 6. Jahrestagung der Deutschen Gesellschaft für Computergestüte Chirurgie*, Karlsruhe, Germany, oct. 2007.

[99] K. Konolige, "Small vision systems: Hardware and implementation," in *Robotics Research*, Y. Shirai and S. Hirose, Eds. Springer London, 1998, pp. 203–212.

[100] ——, "Projected texture stereo," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, USA, may 2010, pp. 148–155.

[101] A. Krupa, C. Doignon, J. Gangloff, and M. de Mathelin, "Combined image-based and depth visual servoing applied to robotized laparoscopic surgery," in *Proceedings of the IEEE/RSJ International Conference Intelligent Robots and System*, vol. 1, Lausanne, Switzerland, oct. 2002, pp. 323–329.

[102] A. Krupa, J. Gangloff, C. Doignon, M. de Mathelin, G. Morel, J. Leroy, L. Soler, and J. Marescaux, "Autonomous 3-d positioning of surgical instruments in robotized laparoscopic surgery using visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 5, pp. 842–853, oct. 2003.

[103] B. Kübler, "A new approach to establish tactility in minimally invasive robotic surgery - development, design, and first evaluation of a haptic-tactile feedback system for improved localization of arteries during surgery such as closed-chest revascularization," Ph.D. dissertation, Universität Stuttgart, 2010.

[104] B. Kuebler, R. Gruber, C. Joppek, J. Port, G. Passig, J. H. Nagel, and G. Hirzinger, "Tactile feedback for artery detection in minimally invasive robotic surgery: Preliminary results of a new approach," in *Proceedings of the World Congress on Medical Physics and Biomedical Engineering*, vol. 25/6, Munich, Germany, sep. 2009, pp. 299–302.

[105] D. Kwartowitz, S. Herrell, and R. Galloway, "Toward image-guided robotic surgery: determining intrinsic accuracy of the da vinci robot," *International Journal of Computer Assisted Radiology and Surgery*, vol. 1, no. 3, pp. 157–165, 2006.

[106] D. Kwartowitz, M. Miga, S. Herrell, and R. Galloway, "Towards image guided robotic surgery: multi-arm tracking through hybrid localization," *International Journal of Computer Assisted Radiology and Surgery*, vol. 4, no. 3, pp. 281–286, 2009.

[107] 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 International Conference on Intelligent Robots and Systems*, vol. 2, Victoria, Canada, oct. 1998, pp. 945 –950.

[108] M. Li, M. Ishii, and R. Taylor, "Spatial motion constraints using virtual fixtures generated by anatomy," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 4–19, feb. 2004.

[109] J. Lim, "Optimized projection pattern supplementing stereo systems," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Kobe, Japan, may 2009, pp. 2823 –2829.

[110] H. C. Lin, I. Shafran, D. Yuh, and G. D. Hager, "Towards automatic skill evaluation: Detection and segmentation of robot-assisted surgical motions," *Computer Aided Surgery*, vol. 11, no. 5, pp. 220–230, sep. 2006.

[111] B. Lo, A. Darzi, and G.-Z. Yang, "Episode classification for the analysis of tissue/instrument interaction with multiple visual cues," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, Montréal, Canada, nov. 2003, pp. 230–237.

[112] M. J. H. Lum, D. C. W. Friedman, G. Sankaranarayanan, H. King, K. Fodero, R. Leuschke, B. Hannaford, J. Rosen, and M. N. Sinanan, "The raven: Design and validation of a telesurgery system," *International Journal of Robotics Research*, vol. 28, no. 9, pp. 1183–1197, sep. 2009.

[113] T. Lüth and G. Stauß, "Diskussion unterschiedlicher assistenzmethoden für die endoskopie aus technischer sicht," *Endo heute*, vol. 23, no. 1, pp. 53–58, 2010.

[114] A. Madhani, G. Niemeyer, and J. Salisbury, "The black falcon: a teleoperated surgical instrument for minimally invasive surgery," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, Victoria, Canada, oct. 1998, pp. 936–944.

[115] P. Marayong, M. Li, A. Okamura, and G. Hager, "Spatial motion constraints: theory and demonstrations for robot guidance using virtual fixtures," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, Taipei, Taiwan, sep. 2003, pp. 1954–1959.

[116] X. Maurice, C. Albitar, C. Doignon, and M. de Mathelin, "A structured light-based laparoscope with real-time organs' surface reconstruction for minimally invasive surgery," in *Proceedings of the IEEE International Conference on Engineering in Medicine and Biology Society*, San Diego, California, USA, sep. 2012, pp. 5769 –5772.

[117] H. Mayer, I. Nagy, A. Knoll, E. Braun, R. Lange, and R. Bauernschmitt, "Adaptive control for human-robot skilltransfer: Trajectory planning based on fluid dynamics," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Roma, Italy, apr. 2007, pp. 1800–1807.

[118] H. Mayer, "Human-machine skill transfer in robot assisted, minimally invasive surgery," Ph.D. dissertation, Technische Universität München, München, Germany, 2008.

[119] H. Mayer, I. Nagy, A. Knoll, E. U. Braun, R. Bauernschmitt, and R. Lange, "Haptic feedback in a telepresence system for endoscopic heart surgery," *Presence: Teleoper. Virtual Environ.*, vol. 16, no. 5, pp. 459–470, oct. 2007.

[120] S. J. McKenna, C. H. Nait, and T. Frank, "Towards video understanding of laparoscopic surgery: Instrument tracking," in *Proceedings of the International Conference on Image and Vision Computing*, Dunedin, New Zealand, nov. 2005.

[121] A. Micaelli, A. Micaelli, C. Samson, C. Samson, and P. Icare, "Trajectory tracking for unicycle-type and two-steering-wheels mobile robots," INRIA, Research Report RR-2097, 1993.

[122] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 3, pp. 311 –324, may. 2007.

[123] M. Mitsuishi, J. Arata, K. Tanaka, M. Miyamoto, T. Yoshidome, S. Iwata, M. Hashizume, and S. Warisawa, "Development of a remote minimally-invasive surgical system with operational environment transmission capability," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, Taipei, Taiwan, sep. 2003, pp. 2663 – 2670.

[124] R. Morano, C. Ozturk, R. Conn, S. Dubin, S. Zietz, and J. Nissano, "Structured light using pseudorandom codes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 322 –327, mar. 1998.

[125] J. A. Morgan, J. C. Peacock, T. Kohmoto, M. J. Garrido, B. M. Schanzer, A. R. Kherani, D. W. Vigilance, F. H. Cheema, S. Kaplan, C. R. Smith, M. C. Oz, and M. Argenziano, "Robotic techniques improve quality of life in patients undergoing atrial septal defect repair," *The Annals of Thoracic Surgery*, vol. 77, no. 4, pp. 1328 – 1333, apr. 2004.

[126] P. Morguet, "Stochastische modellierung von bildsequenzen zur segmentierung und erkennung dynamischer gesten," Ph.D. dissertation, Technische Universität München, München, Germany, 2000.

[127] G. P. Moustris, S. C. Hiridis, K. M. Deliparaschos, and K. M. Konstantinidis, "Evolution of autonomous and semi-autonomous robotic surgical systems: a review of the literature," *International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 7, no. 4, pp. 375–392, aug. 2011.

[128] G. Mylonas, A. Darzi, and G.-Z. Yang, "Gaze contingent depth recovery and motion stabilisation for minimally invasive robotic surgery," in *Proceedings of the International Workshop on Medical Imaging and Augmented Reality*, Beijing, China, aug. 2004, pp. 311–319.

[129] G. Mylonas, K.-W. Kwok, A. Darzi, and G.-Z. Yang, "Gaze contingent motor channelling and haptic constraints for minimally invasive robotic surgery," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, New York, New York, USA, sep. 2008, pp. 676–683.

[130] F. Nageotte, P. Zanne, C. Doignon, and M. de Mathelin, "Visual servoing-based endoscopic path following for robot-assisted laparoscopic surgery," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, oct. 2006, pp. 2364–2369.

[131] F. Nageotte, C. Doignon, M. de Mathelin, P. Zanne, and L. Soler, "Circular needle and needleholder localization for computer-aided suturing in laparoscopic surgery," *Medical Imaging 2005: Visualization, Image-Guided Procedures, and Display*, vol. 5744, no. 1, pp. 87–98, apr. 2005.

[132] F. Nageotte, P. Zanne, C. Doignon, and M. de Mathelin, "Stitching planning in laparoscopic surgery : Towards robot-assisted suturing," *International Journal of Robotics Research*, vol. 28, no. 10, pp. 1303–1321, oct. 2009.

[133] I. Nagy, "3d situs reconstruction in minimally invasive surgery." Ph.D. dissertation, Technische Universität München, München, Germany, 2009.

[134] T. Neumuth, P. Jannin, G. Strauß, J. Meixensberger, and O. Burgert, "Validation of knowledge acquisition for surgical process models," *Journal of the American Medical Informatics Association*, vol. 16, no. 1, pp. 72–80, jan.–feb. 2009.

[135] F. Nickel, I. Wegner, H. Kenngott, J. Neuhaus, B. P. Müller-Stich, H.-P. Meinzer, and C. N. Gutt, "Magnetisches tracking für die navigation mit dem da vinci surgical system," in *Bildverarbeitung für die Medizin*, ser. Informatik aktuell, T. Tolxdorff, J. Braun, T. M. Deserno, A. Horsch, H. Handels, and H.-P. Meinzer, Eds. Springer Berlin Heidelberg, 2008, pp. 148–152.

[136] A. Nishikawa, T. Hosoi, K. Koara, D. Negoro, A. Hikita, S. Asano, H. Kakutani, F. Miyazaki, M. Sekimoto, M. Yasui, Y. Miyake, S. Takiguchi, and M. Monden, "Face mouse: A novel human-machine interface for controlling the position of a laparoscope," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 5, pp. 825 – 841, oct. 2003.

[137] D. Noonan, G. Mylonas, A. Darzi, and G.-Z. Yang, "Gaze contingent articulated robot control for robot assisted minimally invasive surgery," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, France, sep. 2008, pp. 1186 –1191.

[138] D. Noonan, G. Mylonas, J. Shang, C. Payne, A. Darzi, and Y. G., "Gaze contingent control for an articulated mechatronic laparoscope," in *Proceedings of the IEEE/RAS International Conference on Biomedical Robotics and Biomechatronics*, Tokyo, Japan, sep. 2010, pp. 759–764.

[139] Y. Normandin, "Optimal splitting of hmm gaussian mixture components with mmie training," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, Detroit, Michigan, USA, may 1995, pp. 449 –452.

[140] H. Nyquist, "Certain topics in telegraph transmission theory," *American Institute of Electrical Engineers, Transactions of the*, vol. 47, no. 2, pp. 617–644, 1928.

[141] T. Ortmaier, H. Weiss, U. Hagn, M. Grebenstein, M. Nickl, A. Albu-Schaffer, C. Ott, S. Jorg, R. Konietschke, L. Le-Tien, and G. Hirzinger, "A hands-on-robot for accurate placement of pedicle screws," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Orlando, Florida, USA, may 2006, pp. 4179 –4186.

[142] T. Ortmaier, "Motion compensation in minimally invasive robotic surgery," Ph.D. dissertation, Technische Universität München, München, Germany, 2002.

[143] N. Padoy and G. Hager, "Human-machine collaborative surgery using learned models," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Shanghai, China, may 2011, pp. 5285 –5292.

[144] N. Padoy, T. Blum, S.-A. Ahmadi, H. Feussner, M.-O. Berger, and N. Navab, "Statistical modeling and recognition of surgical workflow." *Medical Image Analysis*, vol. 16, no. 3, pp. 632–641, april 2012.

[145] N. Padoy and G. D. Hager, "3d thread tracking for robotic assistance in tele-surgery," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, California, USA, sep. 2011, pp. 2102 –2107.

[146] J. Pages, J. Salvi, R. Garcia, and C. Matabosch, "Overview of coded light projection techniques for automatic 3d profiling," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 1, Taipei, Taiwan, sep. 2003, pp. 133 – 138.

[147] C. Pal, J. Weinman, L. Tran, and D. Scharstein, "On learning conditional random fields for stereo," *International Journal of Computer Vision*, vol. 99, pp. 319–337, 2012.

[148] G. Panin, *Model-based Visual Tracking: The OpenTL Framework*. Wiley, 2011.

[149] S. Park, R. D. Howe, and D. F. Torchiana, "Virtual fixtures for robotic cardiac surgery," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, Utrecht, The Netherlands, oct. 2001, pp. 1419–1420.

[150] S. G. Parker, J. Bigler, A. Dietrich, H. Friedrich, J. Hoberock, D. Luebke, D. McAllister, M. McGuire, K. Morley, A. Robison, and M. Stich, "Optix: a general purpose ray tracing engine," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 66:1–66:13, jul. 2010.

[151] C. Passenberg, R. Groten, A. Peer, and M. Buss, "Towards real-time haptic assistance adaptation optimizing task performance and human effort," in *Proceedings of the IEEE World Haptics Conference*, Istanbul, Turkey, jun. 2011, pp. 155–160.

[152] Z. Pezzementi, S. Voros, and G. D. Hager, "Articulated object tracking by rendering consistent appearance parts," in *Proceedings of the IEEE international Conference on Robotics and Automation*, Kobe, Japan, may 2009, pp. 1225–1232.

[153] R. Polet and J. Donnez, "Using a laparoscope manipulator (lapman)," *Laparoscopic Gynecological Surgery*, vol. 17, pp. 187–191, 2008.

[154] P. Pook and D. Ballard, "Deictic teleassistance," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, Munich, Germany, sep. 1994, pp. 245 –252.

[155] R. Prada and S. Payandeh, "A study on design and analysis of virtual fixtures for cutting in training environments," in *Proceedings of the Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Pisa, Italy, mar. 2005, pp. 375–380.

[156] L. R. Rabiner, *Readings in speech recognition*, A. Waibel and K.-F. Lee, Eds. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1990.

[157] D. Rattner and A. Kalloo, "Asge/sages working group on natural orifice translumenal endoscopic surgery," *Surgical Endoscopy*, vol. 20, no. 2, pp. 329–333, feb. 2006.

[158] H. Reichenspurner, R. Damiano, M. Mack, D. Boehm, H. Gulbins, C. Detter, B. Meiser, R. Ellgass, and B. Reichart, "Use of the voice-controlled and computer-assisted surgical system zeus for endoscopic coronary artery bypass grafting." *J Thorac Cardiovasc Surg*, vol. 118, no. 1, pp. 11–6, 1999.

[159] C. E. Reiley and G. D. Hager, "Task versus subtask surgical skill evaluation of robotic minimally invasive surgery," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, London, UK, sep. 2009, pp. 435–442.

[160] C. Reiley, H. Lin, D. Yuh, and G. Hager, "Review of methods for objective surgical skill evaluation," *Surgical Endoscopy*, vol. 25, no. 2, pp. 356–366, jul. 2011.

[161] A. Reiter and P. Allen, "Marker-less articulated surgical tool detection," in *Proceedings of the International Conference on Computer Assisted Radiology and Surgery*, Pisa, Italy, jun. 2012.

[162] A. Reiter, P. Allen, and T. Zhao, "Learning features on robotic surgical tools," in *IEEE Workshop on Computer Vision and Pattern Recognition*, Providence, Rhode Island, USA, jun. 2012, pp. 38–43.

[163] J. Ren, R. Patel, K. McIsaac, G. Guiraudon, and T. Peters, "Dynamic 3-d virtual fixtures for minimally invasive beating heart procedures," *IEEE Transactions on Medical Imaging*, vol. 27, no. 8, pp. 1061 –1070, aug. 2008.

[164] M. Rickert, "Efficient motion planning for intuitive task execution in modular manipulation systems," Ph.D. dissertation, Technische Universität München, München, Germany, 2011.

[165] S. Roman, *Coding and Information Theory*. Springer, 1992.

[166] J. Rosen, J. Brown, L. Chang, M. Sinanan, and B. Hannaford, "Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete markov model," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 3, pp. 399 –413, mar. 2006.

[167] J. Rosen and B. Hannaford, "Doc at a distance," *IEEE Spectrum*, pp. 34–39, oct. 2006.

[168] L. Rosenberg, "Virtual fixtures: Perceptual tools for telerobotic manipulation," in *IEEE Virtual Reality Annual International Symposium*, Seattle, Washington, USA, sep. 1993, pp. 76–82.

[169] G. S., J. Rosen, B. Hannaford, and M. Sinanan, "The red dragon: A multi-modality system for simulation and training in minimally invasive surgery," in *Proceedings of Medicine Meets Virtual Reality*, Long Beach, Californa, USA, feb. 2007, pp. 149–154.

[170] J. Sackier and Y. Wang, "Robotically assisted laparoscopic surgery," *Surgical Endoscopy*, vol. 8, pp. 63–66, 1994.

[171] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado, "A state of the art in structured light patterns for surface profilometry," *Pattern Recognition*, vol. 43, no. 8, pp. 2666 – 2680, aug. 2010.

[172] A. Saxena, S. H. Chung, and A. Y. Ng, "3-d depth reconstruction from a single still image," *International Journal of Computer Vision*, vol. 76, no. 1, pp. 53–69, jan. 2008.

[173] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, apr. 2001.

[174] E. Schneider, T. Villgrattner, J. Vockeroth, K. Bartl, S. Kohlbecher, S. Bardins, H. Ulbrich, and T. Brandt, "Eyeseecam: An eye movement–driven head camera for the examination of natural visual exploration," *Annals of the New York Academy of Sciences*, vol. 1164, no. 1, pp. 461–467, may 2009.

[175] M. O. Schurr, G. Buess, B. Neisius, and U. Voges, "Robotics and telemanipulation technologies for endoscopic surgery," *Surgical Endoscopy*, vol. 14, no. 4, pp. 375–381, apr. 2000.

[176] L. Schwarz, A. Bigdelou, and N. Navab, "Learning gestures for customizable human-computer interaction in the operating room," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, Toronto, Canada, sep. 2011.

[177] T. Seisen, S. J. Drouin, V. Phé, J. Parra, P. Mozer, M.-O. Bitker, O. Cussenot, and M. Rouprêt, "Current role of image-guided robotic radiosurgery (cyberknife®) for prostate cancer treatment," *BJU International*, pp. n/a–n/a, 2013.

[178] K. Seong-Young and K. Dong-Soo, "A surgical knowledge based interaction method for a laparoscopic assistant robot," in *Proceedings of the IEEE International Workshop on Robot and Human Interactive Communication*, sep. 2004, pp. 313 – 318.

[179] Y. Shiu and S. Ahmad, "Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form ax=xb," *Robotics and Automation, IEEE Transactions on*, vol. 5, no. 1, pp. 16 –29, feb. 1989.

[180] D. Soetanto, L. Lapierre, and A. Pascoal, "Adaptive, non-singular path-following control of dynamic wheeled robots," in *Proceedings of the IEEE Conference on Decision and Control*, vol. 2, Maui, Hawaii, USA, dec. 2003, pp. 1765–1770.

[181] S. Speidel, M. Delles, C. Gutt, and R. Dillmann, "Tracking of instruments in minimally invasive surgery for surgical skill analysis," in *Medical Imaging and Augmented Reality*, ser. Lecture Notes in Computer Science, G.-Z. Yang, T. Jiang, D. Shen, L. Gu, and J. Yang, Eds. Springer Berlin / Heidelberg, 2006, vol. 4091, pp. 148–155.

[182] T. Stewart, "Usability or user experience - what's the difference?" in *System Concepts*, 2008.

[183] D. Stoyanov, "Surgical vision," *Annals of Biomedical Engineering*, vol. 40, no. 2, pp. 332–345, jan. 2012.

[184] D. Stoyanov, G. Mylonas, and G.-Z. Yang, "Gaze contingent 3d control for focused energy ablation in robotic assisted surgery," in *Medical Image Computing and Computer-Assisted Intervention*, D. Metaxas, L. Axel, G. Fichtinger, and G. Székely, Eds. Springer Berlin / Heidelberg, 2008, pp. 347–355.

[185] D. Stoyanov, M. Scarzanella, P. Pratt, and G.-Z. Yang, "Real-time stereo reconstruction in robotically assisted minimally invasive surgery," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, Beijing, China, sep. 2010, pp. 275–282.

[186] P. D. Stroud, "A recursive exponential filter for time-sensitive data," Los Alamos National Laboratory, Tech. Rep. LAUR-99-5573, oct. 1999.

[187] X. Sun, X. Mei, S. Jiao, M. Zhou, and H. Wang, "Stereo matching with reliable disparity propagation," in *Proceedings of the IEEE International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, Washington, DC, USA, may 2011, pp. 132–139.

[188] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, "A comparative study of energy minimization methods for markov random fields with smoothness-based priors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 6, pp. 1068 –1080, jun. 2008.

[189] R. Szeliski, *Computer Vision: Algorithms and Applications*, 1st ed. New York, NY, USA: Springer-Verlag New York, Inc., 2010.

[190] R. Taylor, J. Funda, B. Eldridge, S. Gomory, K. Gruben, D. LaRose, M. Talamini, L. Kavoussi, and J. Anderson, "A telerobotic assistant for laparoscopic surgery," *IEEE Engineering in Medicine and Biology Magazine*, vol. 14, no. 3, pp. 279 –288, may/jun. 1995.

[191] R. Taylor, B. Mittelstadt, H. Paul, W. Hanson, P. Kazanzides, J. Zuhars, B. Williamson, B. Musits, E. Glassman, and W. Bargar, "An image-directed robotic system for precise orthopaedic surgery," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 3, pp. 261 –275, jun. 1994.

[192] R. Taylor and D. Stoianovici, "Medical robotics in computer-integrated surgery," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 5, pp. 765 – 781, oct. 2003.

[193] S. Thielmann, U. Seibold, R. Haslinger, G. Passig, T. Bahls, S. Jörg, M. Nickl, A. Nothhelfer, U. Hagn, and G. Hirzinger, "Mica - a new generation of versatile instruments in robotic surgery," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, oct. 2010, pp. 871 –878.

[194] A. Tobergte, G. Passig, B. Kuebler, U. Seibold, U. A. Hagn, F. A. Fröhlich, R. Konietschke, S. Jörg, M. Nickl, S. Thielmann, R. Haslinger, M. Groeger, A. Nothhelfer, L. Le-Tien, R. Gruber, A. Albu-Schäffer, and G. Hirzinger, "Mirosurge-advanced user interaction modalities in minimally invasive robotic surgery," *Presence: Teleoperators and Virtual Environments*, vol. 19, no. 5, pp. 400–414, oct. 2010.

[195] S. Tognarelli, V. Castelli, G. Ciuti, C. Natali, E. Sinibaldi, P. Dario, and A. Menciassi, "Magnetic propulsion and ultrasound tracking of endovascular devices," *Journal of Robotic Surgery*, vol. 6, no. 1, pp. 5–12, mar. 2012.

[196] M. Tomikawa, H. Xu, and M. Hashizume, "Current status and prerequisites for natural orifice translumenal endoscopic surgery (notes)," *Surgery Today*, vol. 40, no. 10, pp. 909–916, oct. 2010.

[197] O. Tonet, R. U. Thoranaghatte, G. Megali, and P. Dario, "Tracking endoscopic instruments without a localizer: A shape-analysis-based approach," *Computer Aided Surgery*, vol. 12, no. 1, pp. 35–42, jan. 2007.

[198] I. Tosic, B. A. Olshausen, and B. J. Culpepper, "Learning sparse representations of depth," *Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 941–952, 2010.

[199] S. Tully, A. Bajo, G. Kantor, H. Choset, and N. Simaan, "Constrained filtering with contact detection data for the localization and registration of continuum robots in flexible environments," in *Proceedings of the IEEE International Conference on Robotics and Automation*, St. Paul, Minnesota, USA, may 2012, pp. 3388 –3394.

[200] T. Tuytelaars and K. Mikolajczyk, *Local Invariant Feature Detectors: A Survey*. Hanover, MA, USA: Now Publishers Inc., 2008.

[201] D. R. Uecker, C. Lee, Y. F. Wang, and Y. Wang, "Automated instrument tracking in robotically-assisted laparoscopic surgery," *Journal of Image Guided Surgery*, vol. 1, no. 6, pp. 308–325, 1998.

[202] A. Uneri, M. A. Balicki, J. Handa, P. Gehlbach, R. H. Taylor, and I. Iordachita, "New steady-hand eye robot with micro-force sensing for vitreoretinal surgery," in *Proceedings of the IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, Tokyo, Japan, sep. 2010, pp. 814–819.

[203] J. van den Berg, S. Miller, D. Duckworth, H. Hu, A. Wan, X.-Y. Fu, K. Goldberg, and P. Abbeel, "Superhuman performance of surgical tasks by robots using iterative learning from human-guided demonstrations," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, USA, may 2010, pp. 2074 –2081.

[204] M. Visentini-Scarzanella, G. Mylonas, D. Stoyanov, and G.-Z. Yang, "i-brush: A gaze-contingent virtual paintbrush for dense 3d reconstruction in robotic assisted surgery," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, London, UK, sep. 2009, pp. 353–360.

[205] S. Voros, J.-A. Long, and P. Cinquin, "Automatic detection of instruments in laparoscopic images: A first step towards high-level command of robotic endoscopic holders," *The International Journal of Robotics Research*, vol. 26, no. 11-12, pp. 1173–1190, nov. 2007.

[206] P. Vuylsteke and A. Oosterlinck, "Range image acquisition with a single binary-encoded light pattern," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 2, pp. 148 –164, feb. 1990.

[207] L. Wang and R. Yang, "Global stereo matching leveraged by sparse ground control points," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, Colorado Springs, Colorado, USA, jun. 2011, pp. 3033–3040.

[208] M. Wäny, S. Voltz, F. Gaspar, and L. Chen, "Minimal form factor digital-image sensor for endoscopic applications," in *Proceedings of the SPIE: Sensors, Cameras, and Systems for Industrial/Scientific Applications*, vol. 7249, San Jose, USA, 2009.

[209] O. Weede, H. Monnich, B. Muller, and H. Worn, "An intelligent and autonomous endoscopic guidance system for minimally invasive surgery," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Shanghai, China, may 2011, pp. 5762 –5768.

[210] O. Weede, F. Dittrich, H. Wörn, B. Jensen, A. Knoll, D. Wilhelm, M. Kranzfelder, A. Schneider, and H. Feussner, "Workflow analysis and surgical phase recognition in minimally invasive surgery," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, Guangzhou, China, dec. 2012.

[211] Y. Wei and L. Quan, "Region-based progressive stereo matching," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 1, Washington, DC, USA, jun. 2004, pp. 106–113.

[212] G. Welch and G. Bishop, "An introduction to the kalman filter," Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, Tech. Rep., 1995.

[213] L. Windisch, F. Cheriet, and G. Grimard, "Bayesian differentiation of multi-scale line-structures for model-free instrument segmentation in thoracoscopic images," in *Image Analysis and Recognition*, ser. Lecture Notes in Computer Science, M. Kamel and A. Campilho, Eds. Springer Berlin / Heidelberg, 2005, vol. 3656, pp. 938–948.

[214] Z. Yaniv and K. Cleary, "Image-guided procedures: A review," Georgetown University, Imaging Science and Information Systems Center, Washington, DC, Technical Report CAIMR TR-2006-3, apr. 2006.

[215] R. Zabih and J. Woodfill, "A non-parametric approach to visual correspondence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1994.

[216] ——, "Non-parametric local transforms for computing visual correspondence," in *Proceedings of the European Conference on Computer Vision*, J.-O. Eklundh, Ed., vol. 801. Stockholm, Sweden: Springer Berlin Heidelberg, may 1994, pp. 151–158.

[217] L. Zhang, B. Curless, and S. M. Seitz, "Rapid shape acquisition using color structured light and multi-pass dynamic programming," in *The 1st IEEE International Symposium on 3D Data Processing, Visualization, and Transmission*, Chapel Hill, North Carolina, USA, jun. 2002, pp. 24–36.

[218] X. Zhang and S. Payandeh, "Application of visual tracking for robot-assisted laparoscopic surgery," *Journal of Robotic Systems*, vol. 19, no. 7, pp. 315–328, apr. 2002.

[219] J. Zhu, L. Wang, R. Yang, J. Davis, and Z. Pan, "Reliability fusion of time-of-flight depth and stereo geometry for high quality depth maps," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 7, pp. 1400–1414, jul. 2011.